Introduction to Kali Linux and Network Simulation

- ★ **Be Careful with Kali Linux:**
- ★ Kali Linux is a powerful tool for penetration testing, and it's essential to use it responsibly. I cannot emphasize enough the importance of ensuring that you are disconnected from the internet when practicing your penetration testing skills. This precaution is necessary to avoid unintended consequences or potential legal issues.
- ★ **Course Content Overlap:**
- ★ Some of the hands-on session content overlaps with CSCI6706, a graduate course taught by Professor Nur Zincir-Heywood, who specializes in cybersecurity topics. Professor Zincir-Heywood, my supervisor, is an excellent source of knowledge. I have had the honor of working with her as a Research Assistant (RA) and Teaching Assistant (TA) for four years. Her insights and expertise have significantly influenced this course material.
- ★ If you are not comfortable with installing all these packages and Gns3 and so on, you can jump to the DOWNLOAD THE PREPARED KALI VERSION step.

In this practical session, we'll introduce Kali Linux and set up essential tools like GNS3, Docker, and Virtualbox for network simulation.

1. Kali Linux

Kali Linux, version 2024.1, is a Debian-derived distribution tailored for digital forensics and penetration testing. It comes pre-installed with various penetration-testing programs.

Download the preinstall Kali Linux

- Choose the VMware you want to work in and download the right version accordingly.
- https://cdimage.kali.org/kali-2024.1/kali-linux-2024.1-vmware-amd64.7z
- https://cdimage.kali.org/kali-2024.1/kali-linux-2024.1-virtualbox-amd64.7z

- How to install Kali Linux:
  - Extract the VMware image
  - Open a Virtual Machine
  - Navigate to the location where our VM is downloaded and find the .vmx file.
  - Continue forward.

If you're new to Linux, acquaint yourself with basic commands and terminal usage through this article: Basic Linux Commands for Beginners: https://maker.pro/linux/tutorial/basic-linux-commands-for-beginners
For those unfamiliar with Debian-based distros, refer to this guide for basic apt package management commands: **https://www.digitalocean.com/community/tutorials/how-to-manage-packages-in-ubuntu-and-debian-with-apt-get-apt-cache**
The default user account in Preinstalled Kali is Kali with password Kali.

***** If you prefer not to repeatedly use 'sudo' for administrative commands, you can switch to the root user by accessing the root directory. Simply enter 'sudo -i' to switch to the root user, allowing you to execute subsequent commands without the need for 'sudo'.**

1. Add a new user with your chosen name (e.g., `user_name`):
   sudo useradd -m user_name

1. Set a password for the new user:

sudo passwd user_name

2. Add the user to the sudoers group for administrative tasks:

```
sudo usermod -aG sudo user_name
```

3. Set bash as the default shell:

```
sudo chsh -s /bin/bash user_name
```

Remember to change the root user's password as well:

```
sudo passwd root
```

2. Setting up GNS3, Docker, and VirtualBox

2.1. GNS3

GNS3 is a network emulator that allows the combination of virtual and real devices to simulate complex networks.

4. Begin by updating the system and installed packages:

```
sudo apt update
sudo apt upgrade -y
```

5. Install necessary dependencies:

```
sudo apt install python3 python3-pip pipx python3-pyqt5 python3-pyqt5.qtwebsockets python3-pyqt5.qtsvg qemu-kvm qemu-utils libvirt-clients libvirt-daemon-system virtinst dynamips software-properties-common ca-certificates curl gnupg2
```

6. Install GNS3:

```
pipx install gns3-server
pipx install gns3-gui
pipx ensurepath
pipx completions
pipx inject gns3-gui gns3-server PyQt5
```

7. Test GNS3 by choosing the local server in the pop-up (open a new CMD).

2.2. Docker

Docker simplifies the creation, deployment, and running of applications through containers.

- **Docker** is a tool that helps create, deploy, and run applications in containers.
- **Containers** are lightweight and portable units that package an application and its dependencies but share the host system's kernel.
- **Docker Images** are blueprints for containers, defining what goes inside.
- **Docker Containers** are instances of Docker images, running isolated processes.

8. Install Docker:

```
sudo apt update
sudo apt install -y docker.io
sudo systemctl enable docker.service && sudo systemctl start docker.service
sudo usermod -aG docker $USER
```

echo "deb [arch=amd64 signed-by=/etc/apt/keyrings/docker.gpg] https://download.docker.com/linux/debian bookworm stable" | sudo tee /etc/apt/sources.list.d/docker.list
curl -fsSL https://download.docker.com/linux/debian/gpg | sudo gpg --dearmor -o /etc/apt/keyrings/docker.gpg
sudo apt update
sudo apt install -y docker-ce docker-ce-cli containerd.io

9. Add the GNS3 repository and install additional packages:

sudo tee /etc/apt/sources.list.d/gns3.list <<EOF
deb http://ppa.launchpad.net/gns3/ppa/ubuntu bionic main
deb-src http://ppa.launchpad.net/gns3/ppa/ubuntu bionic main
EOF
**Check the file to make sure these lines are added:
cat /etc/apt/sources.list.d/gns3.list

sudo apt-key adv --keyserver keyserver.ubuntu.com --recv-keys F88F6D313016330404F710FC9A2FD067A2E3EF7B
sudo apt update
sudo apt install dynamips ubridge

10. Add your user to the required groups:

sudo usermod -aG kvm,libvirt,docker,ubridge,wireshark $USER
Log Out and Log Back In: Logging out and back in ensures that your session recognizes the new group membership.

11. Test Docker:

sudo docker run hello-world

VirtualBox

12. Install VirtualBox:

sudo apt-get install -y virtualbox

2.3. Additional Configuration

Resolve errors and install necessary packages:

13. Install libpcap-dev:

sudo apt install libpcap-dev

14. Clone and install ubridge:

git clone https://github.com/GNS3/ubridge.git
cd ubridge
make
sudo make install

15. Install vpcs:

sudo apt install vpcs
cd ~/Downloads
wget https://sourceforge.net/projects/vpcs/files/0.8/vpcs_0.8b_amd64.deb
sudo dpkg -i vpcs_0.8b_amd64.deb
sudo apt-get install xterm

16. install busybox:

```
sudo apt install busybox-static
```

With these setups, you're ready to explore Kali Linux and conduct network simulations effectively. Feel free to reach out with any questions or concerns.

3. Network Device Configuration

3.1. Cisco Router Configuration

We will use the Cisco command line interface (CLI) for configuring network devices. This tutorial also applies to open-source routing solutions like FRR and Quagga.

- **Download the router IOS image:**
  - https://yaser-rahmati.gitbook.io/gns3/cisco-ios-images-for-dynamips
  - Download the IOS image for the router (e.g., C7200).
- **Add the IOS image to GNS3:**
  - Go to GNS3 Preferences > Dynamips > IOS router > New.
  - Select Run on this computer, then add the downloaded image.
  - The router will appear as C7200 in the device list.

3.2. Host Configuration

Linux host configuration can be managed using ifconfig, route, and host commands in the terminal. In GNS3, you can configure network settings through the device menu:

1. **Edit the device configuration:**
   - Right-click on the device, select Edit config, and save the changes.

For more on Linux host configuration, refer to: Basic Linux Commands for Beginners

3.2.1. Import an Ubuntu Docker Container

- **Pull the Ubuntu Image from Docker Hub:**
- Docker Hub is a repository of Docker images that hosts a wide range of pre-built images, including Ubuntu. By running the command docker pull ubuntu, you instruct Docker to download the latest Ubuntu image from Docker Hub to your local machine. Here's the command:

```
docker pull ubuntu
```

- **In GNS3, Add a Docker Container:**
- GNS3 allows you to integrate Docker containers into your network topologies, enabling you to simulate complex network environments. To add a Docker container in GNS3:
  - Open the GNS3 application on your computer.
  - From the toolbar, click on the "Docker Container" icon or navigate to Edit > Preferences > Docker containers.
  - Click on "New" to create a new Docker container.
  - In the configuration window, provide a name for the container and select the "ubuntu" image from the dropdown menu. You may also specify additional settings such as RAM, CPU, and network adapters as needed.
  - Click on "Apply" or "OK" to save the configuration.
- **Selecting the Ubuntu Image:**
- When adding a Docker container in GNS3, you are prompted to select the Docker image you want to use for the container. Since you have pulled the Ubuntu image from Docker Hub in the previous step, you can simply choose the "ubuntu" image from the list of available images.

3.3. Adding an FTP Server(optional)

- **Pull the vsftpd Docker image:**

docker pull fauria/vsftpd

- **Run the FTP server container:**

docker run -d -p 21:21 -p 21100-21110:21100-21110 -v /home/ftp:/home/vsftpd --name=ftp_server fauria/vsftpd

- **Verify the FTP server is running:**

The command docker ps is used to list all the running Docker containers on your system. When you run this command, it provides information about the containers such as their container ID, names, status, ports, and images.

docker ps

- **Connect the FTP server to GNS3:**
  - In GNS3, add a Docker container and select the ftp_server container.

4. Connecting GNS3 to the Internet

sudo apt update
sudo apt install uml-utilities
sudo ip tuntap add dev tap0 mode tap
sudo ip link set tap0 up
sudo ip addr add 192.168.111.1/24 dev tap0
sudo nano /etc/network/if-up.d/tap0

- Add following to the file which is opened in your terminal:

#!/bin/bash

ip tuntap add dev tap0 mode tap
ip link set tap0 up
ip addr add 192.168.111.1/24 dev tap0
sudo iptables -t nat -A POSTROUTING -o eth0 -j MASQUERADE
sudo iptables -A FORWARD -i eth0 -o tap0 -m state --state RELATED,ESTABLISHED -j ACCEPT
sudo iptables -A FORWARD -i tap0 -o eth0 -j ACCEPT

- Press CTRL+x to save and exit

sudo nano /etc/sysctl.conf

- Ensure this line is present and uncommented:

net.ipv4.ip_forward=1

- Press CTRL+x to save and exit

sudo sysctl -p

- Check the ip address for the interfaces with:

ip addr show

**Use the tap0 interface in the cloud device within GNS3 to connect to the Internet.

4.1. Cloud Configuration Steps in GNS3

1. **Start GNS3** and open your project.
2. **Drag a "Cloud" node** onto the workspace.
3. **Right-click on the "Cloud" node**, select "Configure."
4. **Go to the "NIO Ethernet" tab**, click "Add."
5. **Select `tap0`** from the list and click "Add."

6. **Click "OK"** to save the configuration.
7. **Connect the Cloud node** to other devices in your GNS3 project.
8. **Start the devices** and test the network connectivity.

Let's dive in!

Practice 1:

Please refer to Figure 1 and attempt to construct the same network. Take special care to observe the subnet configurations: nodes should be able to communicate within their respective subnetwork, but not with nodes in other subnetworks.

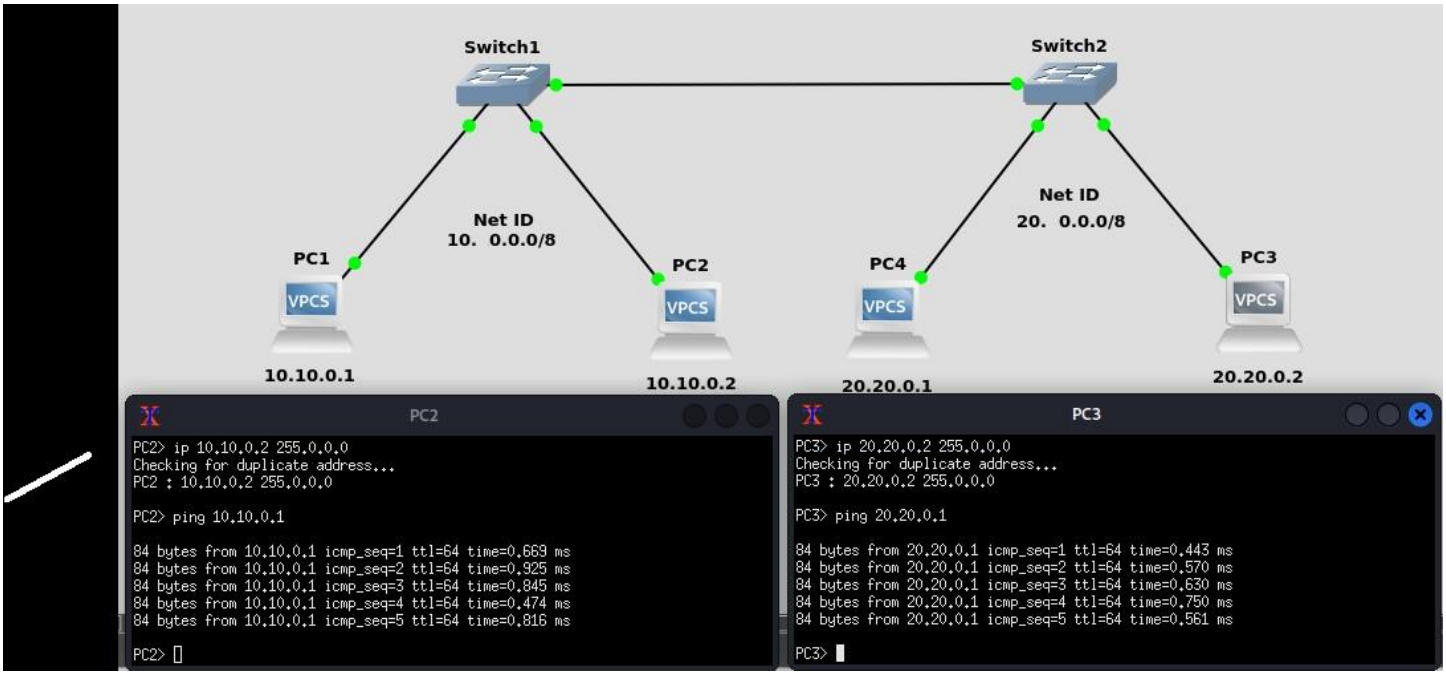

Figure1- Network framework

Resources:

[1] CSCI6706 Hands-On sessions

[2] Linux man-pages project. tc - show / manipulate traffic control settings. http://man7. org/linux/man-pages/man8/tc.8.html.

[3] Techopedia. Network configuration. https://www.techopedia.com/definition/ 25766/network-configuration.

[4] The Linux Documentation Project. Traffic Control HOWTO. https://tldp.org/ HOWTO/Traffic-Control-HOWTO/index.html.

[5] https://gns3.com

[6] https://www.kali.org