Cybersecurity Home Lab Setup using Proxmox VE

Overview

This document outlines the setup and configuration of a home cybersecurity lab using Proxmox VE, hosted on an old HP Pavilion laptop. The purpose of this lab is to provide a safe and isolated environment for cybersecurity testing, training, and virtual machine management.

Host System Specifications

- Laptop: HP Pavilion (circa 2017)
- RAM: 16 GB DDR4
- Storage: 1 TB SSD
- OS: Proxmox VE 8 (installed on bare metal)

Initial Setup

The setup began by downloading the Proxmox VE ISO and creating a bootable USB using Rufus. The installer was accessed by booting from the USB, and Proxmox was installed on the internal SSD using the graphical interface. A static IP was configured during installation.

Networking Issue and Resolution

After installation, the Proxmox machine showed no Ethernet connectivity. The onboard Ethernet card was detected via `lspci`, but no interface appeared in `ip a` because the Realtek driver was missing.

To resolve this, a USB-to-Ethernet adapter was purchased and plugged into the Proxmox machine. The interface `enx...` appeared and a static IP was temporarily assigned using:

```
ip addr add 192.168.8.100/24 dev enxXXXXXXX
ip link set enxXXXXXXX up
```

Private LAN with Travel Router

To avoid using the shared student residence network, a GL.iNet GL-SFT1200 (Opal) travel router was used to create a secure LAN. The Proxmox machine (via the USB Ethernet adapter) was connected to a LAN port on the router. The management laptop connected to the router via Wi-Fi.

The travel router used the default IP range `192.168.8.0/24`, assigning an IP to the laptop and allowing direct access to the Proxmox Web UI from the browser at:

https://192.168.8.100:8006

Persistent Network Configuration

To make the USB Ethernet interface persistent and bridge it with Proxmox's virtual network bridge `vmbr0`, the following configuration was added to `/etc/network/interfaces`:

```
auto lo
iface lo inet loopback

auto enxXXXXXXX
iface enxXXXXXXX inet manual

auto vmbr0
iface vmbr0 inet static
  address 192.168.8.100
  netmask 255.255.255.0
  gateway 192.168.8.1
  bridge_ports enxXXXXXXX
  bridge_stp off
  bridge_fd 0
```

Virtual Machine Setup (Kali Linux)

The Kali Linux installer ISO was downloaded from the official site. In the Proxmox Web UI, it was uploaded via:

Datacenter > <node> > local > ISO Images > Upload

To create the VM:

1. Click 'Create VM'
2. Name the VM and select the Kali ISO
3. Use defaults for System and BIOS
4. Allocate CPU, RAM, and Disk
5. Use bridge `vmbr0` for networking
6. Finish and start the VM

Installation was completed through the Console tab.

Virtual Machine Setup (Ubuntu Server)

> *An Ubuntu Server virtual machine was also created to simulate common server environments. The ISO was downloaded from the official Ubuntu site and uploaded to Proxmox via:*
>
> *Datacenter > <node> > local > ISO Images > Upload*
>
> *The VM was created with the following steps:*
>
> *1. Click 'Create VM'*
> *2. Set a name and select the Ubuntu ISO*
> *3. Choose system defaults (BIOS/UEFI)*
> *4. Allocate appropriate CPU cores, RAM, and disk space*
> *5. Ensure network interface is bridged via `vmbr0`*
> *6. Complete the wizard and install Ubuntu using the Console*

Conclusion

This home lab demonstrates the ability to configure bare-metal virtualization, troubleshoot hardware-level networking issues, and create a secure environment for cybersecurity learning and testing. Future plans include adding Windows VMs, vulnerable apps, and log aggregation tools like ELK or Wazuh.