Lab 1 - Bootstrap & and Network Lab Setup

This assignment focuses on bootstrapping and configuring the network environment for the practical classes of this course. Each student/group should instantiate/provision 4 Virtual Machines (VMs), 1) Linux Kali, 2) Fedora Workstation, 3) Fedora Server, and 4) Windows 10. These must be configured in a similarly way as depicted in Figure 1. Each VM has 2 Network Interface Cards (NICs), one will be connected to the `192.168.0.0/25` NAT network, and another one that needs to be configured.

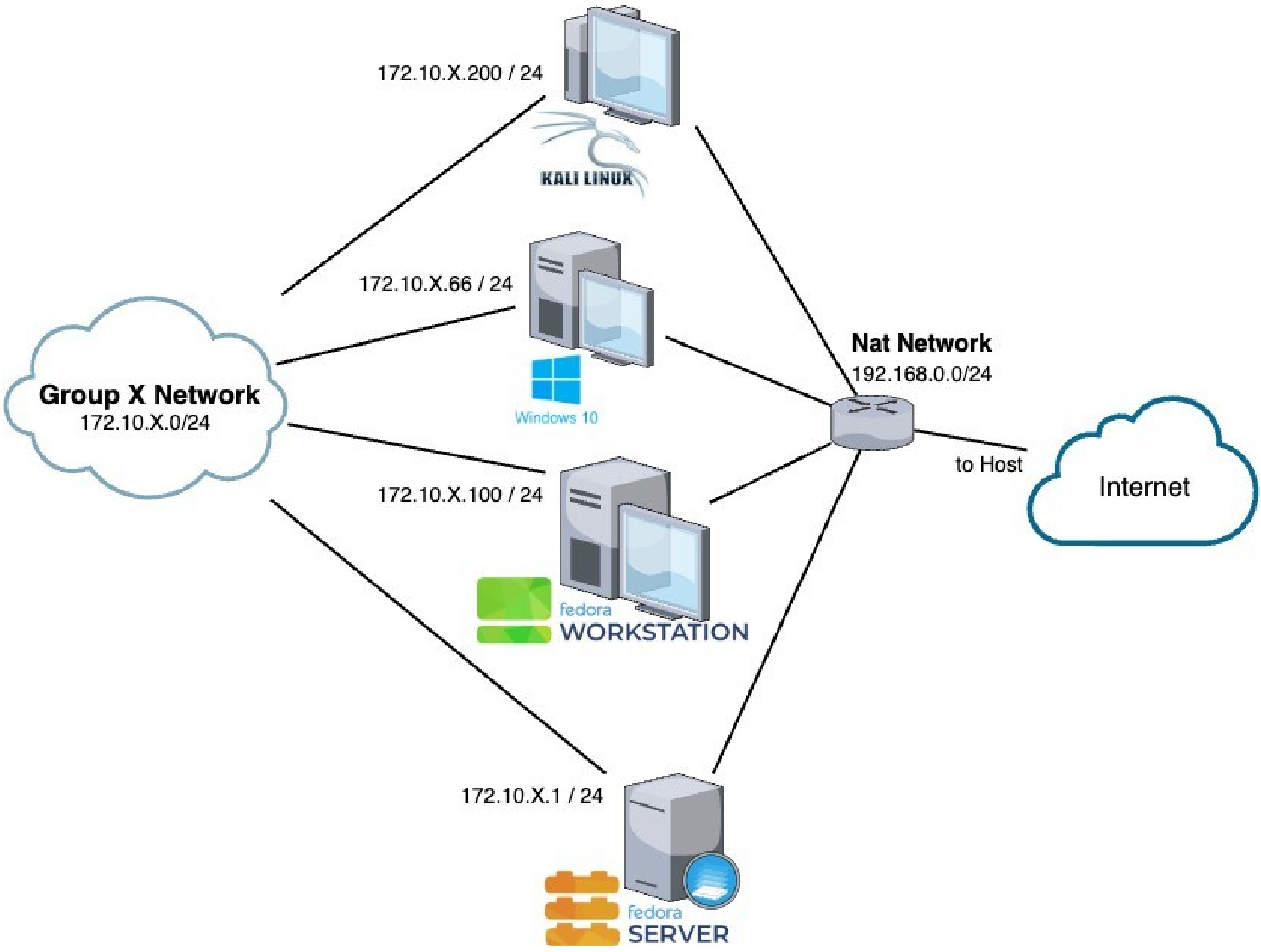
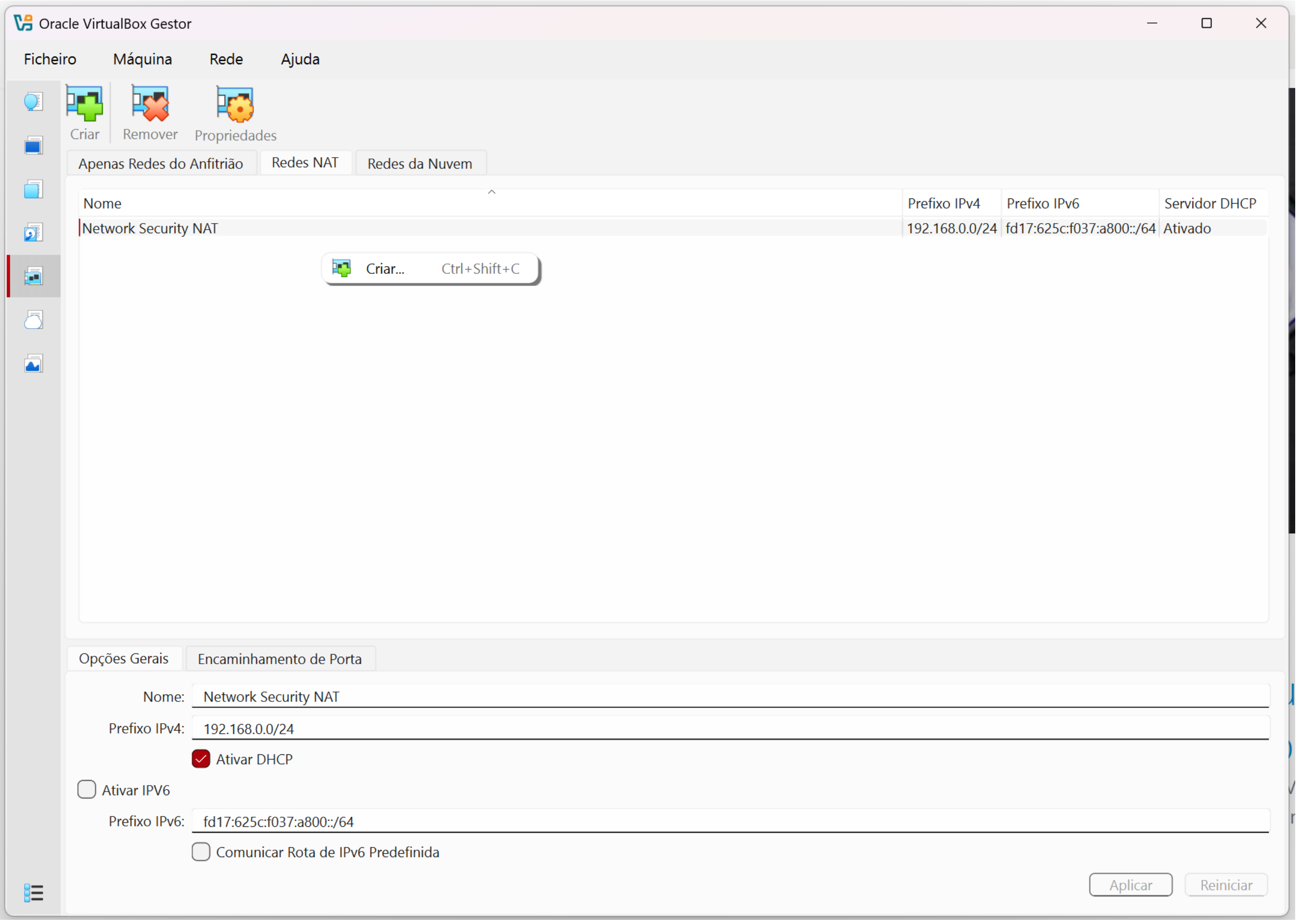


Figure 1 - Network Organization

Configure VMs

For running the VMs you need to install a Virtual Machine Monitor (VMM). We will be using VirtualBox, so you need to install it on your machine (called Host). You can download it from here.



Nome	Prefixo IPv4	Prefixo IPv6	Servidor DHCP
Network Security NAT	192.168.0.0/24	fd17:625c:f037:a800::/64	Ativado

Nome: Network Security NAT

Prefixo IPv4: 192.168.0.0/24

☑ Ativar DHCP

□ Ativar IPV6

Prefixo IPv6: fd17:625c:f037:a800::/64

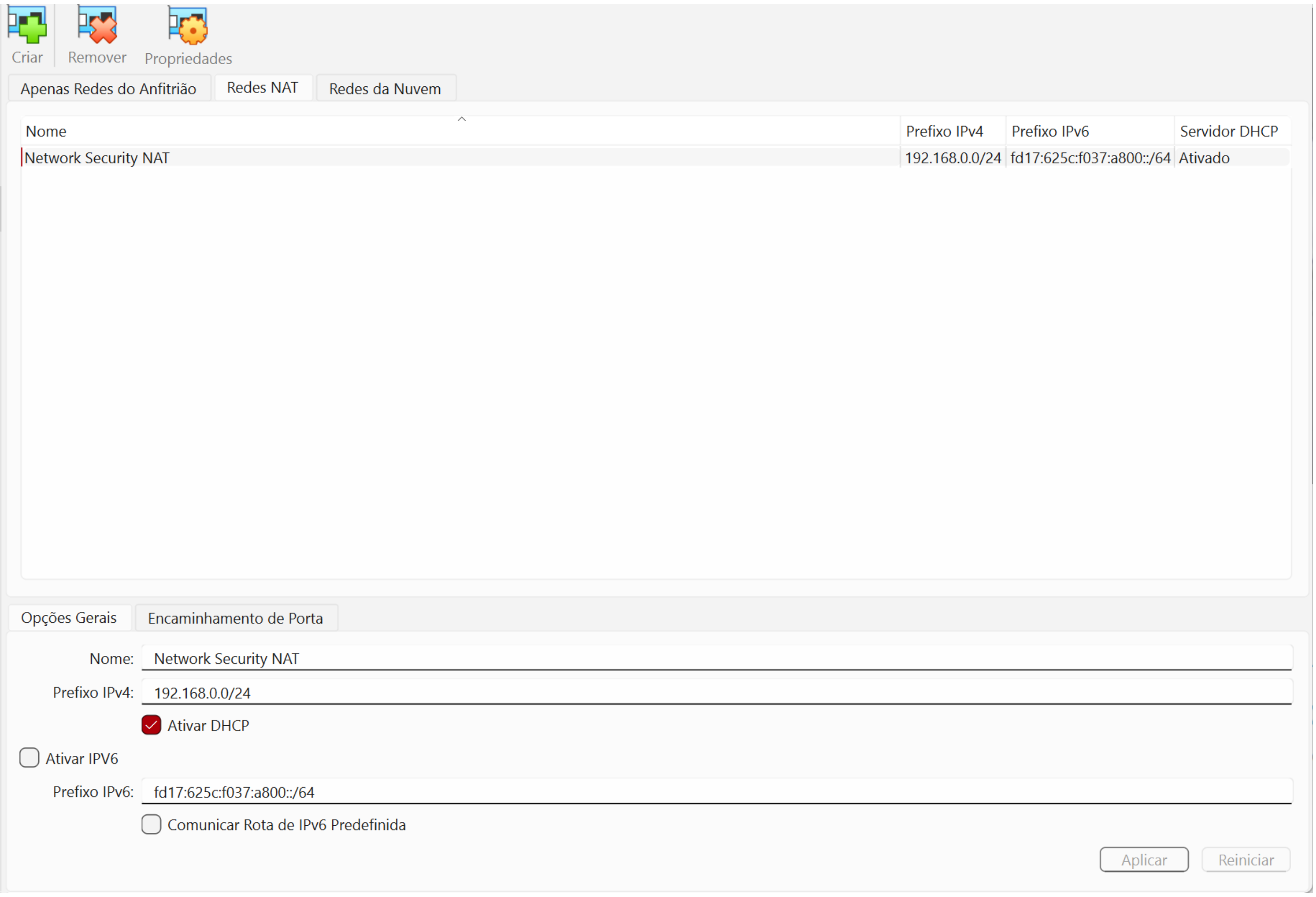
□ Comunicar Rota de IPv6 Predefinida

Figure 2 - Create NAT Network on VirtualBox

Creating a NAT Network

Before creating or importing VMs into VirtualBox you need to create a new NAT Network. So, in VirtualBox, on the panel on the left, click on the Network icon and select the “NAT Networks” tab. Then use the mouse right button and then press the Create button, as illustraed in Figure 2. This will create a new NAT Network that you will then configure with the following information (as presented in Figure 3):

- Network Name: Network Security
- NAT IPV4 Prefix: 192.168.0.0/24
- Enable DHCP: Check



Nome	Prefixo IPv4	Prefixo IPv6	Servidor DHCP
Network Security NAT	192.168.0.0/24	fd17:625c:f037:a800::/64	Ativado

Nome: Network Security NAT

Prefixo IPv4: 192.168.0.0/24

☑ Ativar DHCP

☐ Ativar IPV6

Prefixo IPv6: fd17:625c:f037:a800::/64

☐ Comunicar Rota de IPv6 Predefinida

Figure 3 - NAT Network configuration

Create/Import VMs

We are now going to create the 4 VMs. While you can manually create them and download, install and configure each VMs Operating System, you can go to osboxes.org and download the Linux images you need, already ready to be run on VBox. For the Windows machine, you can directly import into your VirtualBox the file in OVA that you can download from windows.ova.

Attention: These files are very large (several Gigabytes) and can take a long time to download. While you wait you can familiarize yourself with the different types of networks that are supported by VBox.

Configure VMs to connect to `Network Security NAT`

After downloading each one of the bootable Linux disk images, create a VirtualBox machine for each one of them (with at least 4Gig of Ram for the client machines and 1GIG RAM for the server) and use the dowloaded disk images as the virtual bootable disks for those virtual machines. In the case of the windows machine, since it is in OVA format, it already has all the virtual machine parameters defined and you just have to import it directly to VBox, using the Import menu under the file TAB to create the windows virtual machine.

You need to ensure that each VM is connected to the previously created NAT Network. So, for each VM:

- Right click the VM entry;
- Select into `Settings`

- On the `Settings` menu
  - Select `Network` tab
  - Select `Adapter 1` and configure it as following:
    - Enable Network Adapter - check
    - Attached to: select `NAT Network` (not only NAT)
    - Name: select `Network Security NAT` (i.e., the name of the NAT Network you created and configured previously)
  - Select `Adapter 2` and configure it as following:
    - Enable Network Adapter - check
    - Attached to: `Internal Network`
    - Name: `SR`

Now you can boot each VM. Test if the NAT network is working by performing a software upgrade for each one of the VMs.

Please be aware that in **VirtualBox**, "Internal Network" is the most isolated networking mode:

- Only VMs attached to the same ***internal network name*** can communicate.
- By design, **VirtualBox does *not* provide DHCP or IP management** for Internal Networks.
- You must configure addressing yourself (static IPs, or run your own DHCP server inside one of the VMs attached to that "Internal Network).

Assignment 0 - Users and Credentials

Boot each Linux VM and create a new user.

`auser` - with password `horseCACAnow`

Ensure `auser` has `sudo` or administration privileges, configure it otherwise. See this link for additional information on how to give sudo privileges to users in Linux.

In the Windows host use the following credentials

login: `kevin`
password: `H@ckM3!fYouCan`

You can then change them to something more to your liking.

Assignment 1 - SSH Access

All machines need to be configured to be accessible by SSH from the Host. However the SSH daemon is not started on any machine. You need to **install**, **configure**, **enable** and **start** the daemon so that users can log in using SSH.

Before configuring SSH, you need to ensure that the Host machine can SSH into the VMs. Follow the steps described here (hint: you neet to configure port forwarding in the NAT network):

SSH into VirtualBox machine

Configure each machine so that only `auser` can have SSH access. `root` or `admin` users cannot have access through SSH, and all other users should only have access using RSA keys. You can find additional information regarding SSH installation and configuration using the following links:

- SSH on Linux
- SSH Key Pair for User Authentication
- sshd – SSH server process

Assignment 2 - Private Network

All machines should be directly connected using a private LAN on the second NIC (identified by network 172.10.X.0/24 in Figure 1).

. So, you need to configure a static IP address on all machines for the NIC that is not bridged with the host, and create all routes needed so that each machine is able to contact and communicate with every other one.

- Configure Static IP Address
- Configure a static IP address on Fedora
- Route Add Command in Linux

For Windows use the graphical interface. Check that all machines have proper connectivity through the private LAN (172.10.X.0/24), e.g. using `ping`.

. Test if you can connect to each machine using both networks.

. Find out how you can restrict SSH access from different networks.
