Lab 1 - Bootstrap & 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.

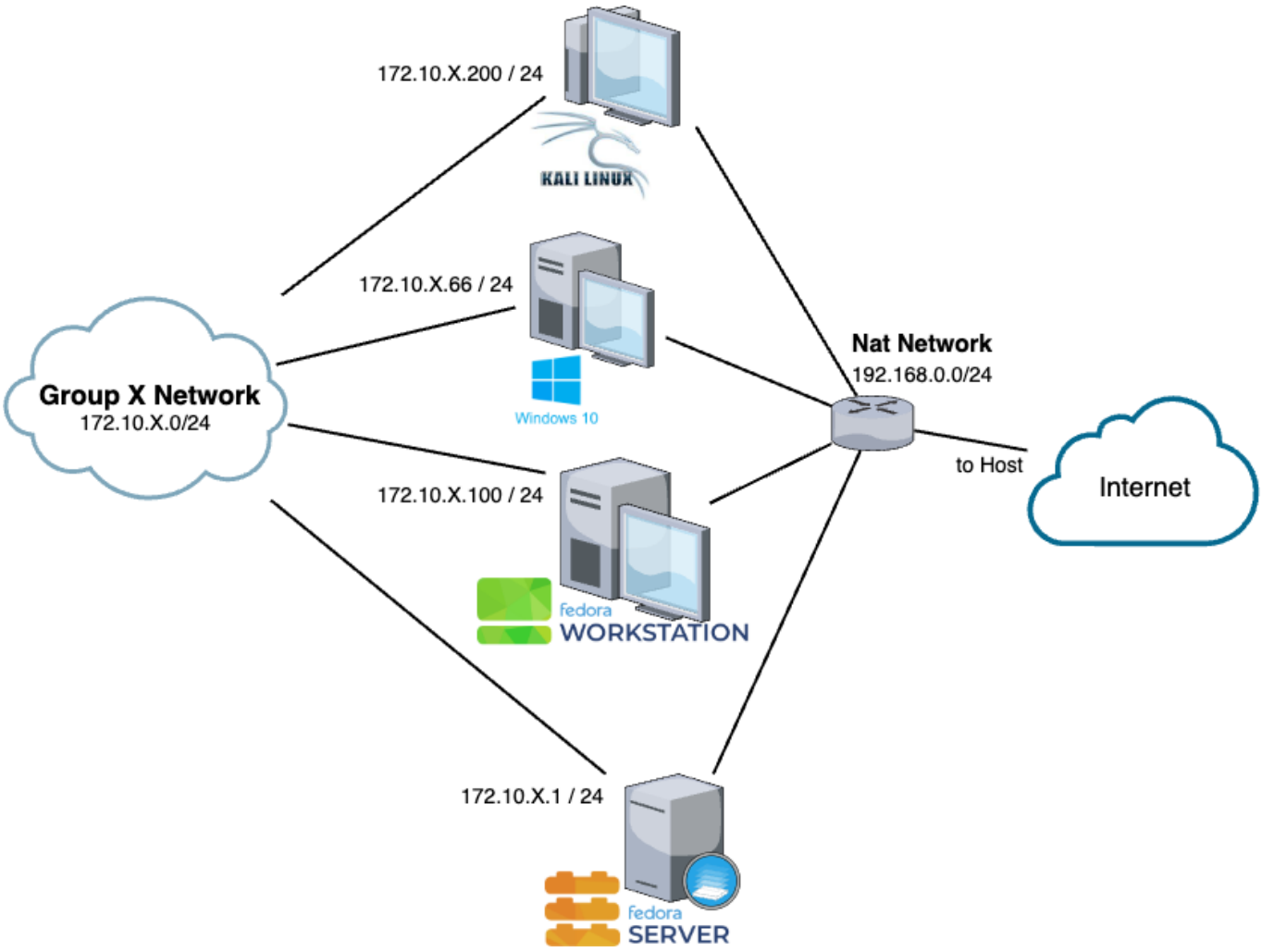

Figure 1 - Network Organization

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.

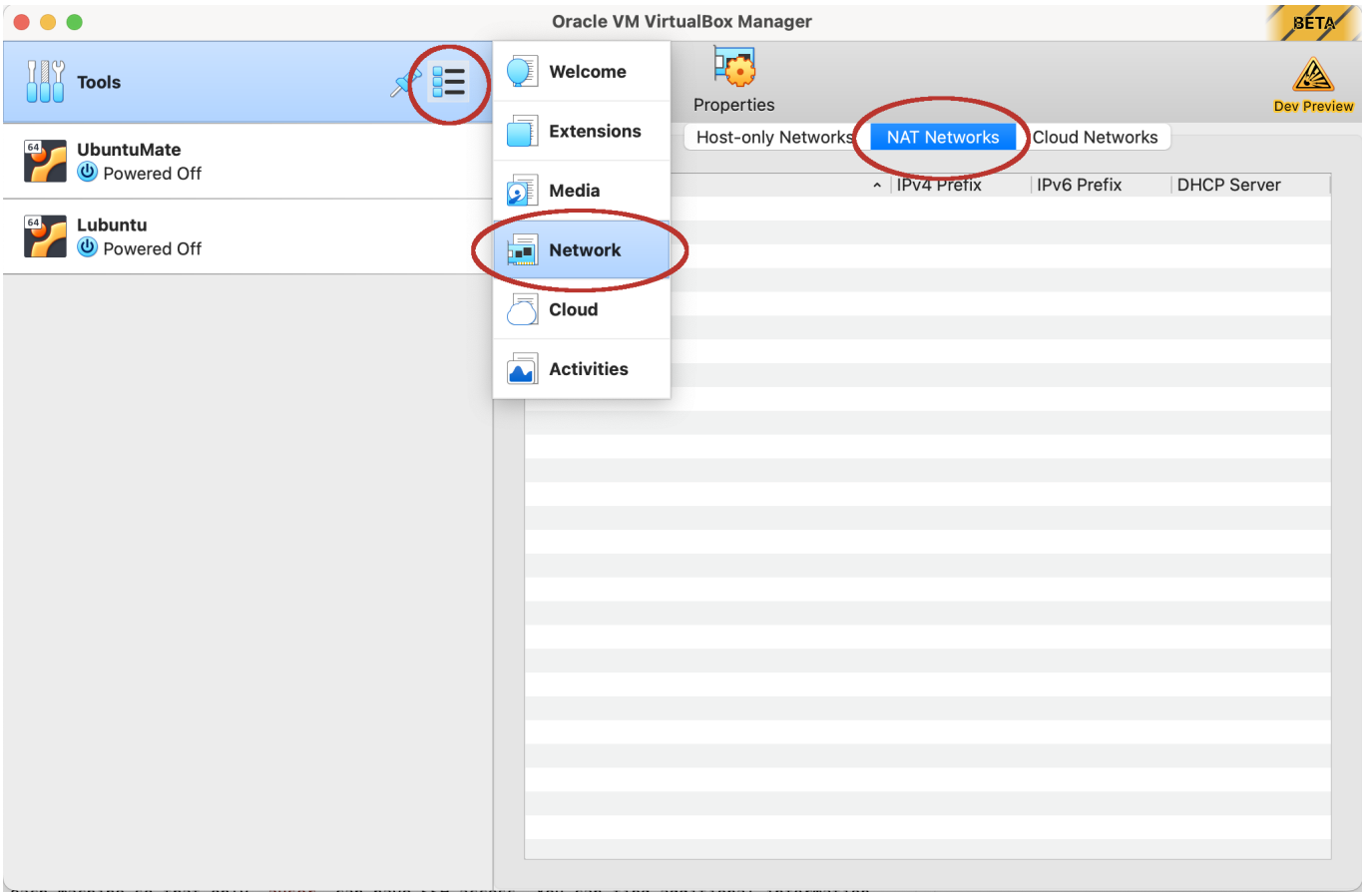

Figure 2 - Create NAT Network on VirtualBox

Creating a NAT Network

Before creating or importing VMs into VirtualBox you need to create a new `NAT Network`. So, in VirtualBox, under `Tools`, select `Network`. Select `NAT Networks`, as presented in Figure 2. Then click the `Create` button. This will create a new NAT Network that you will 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

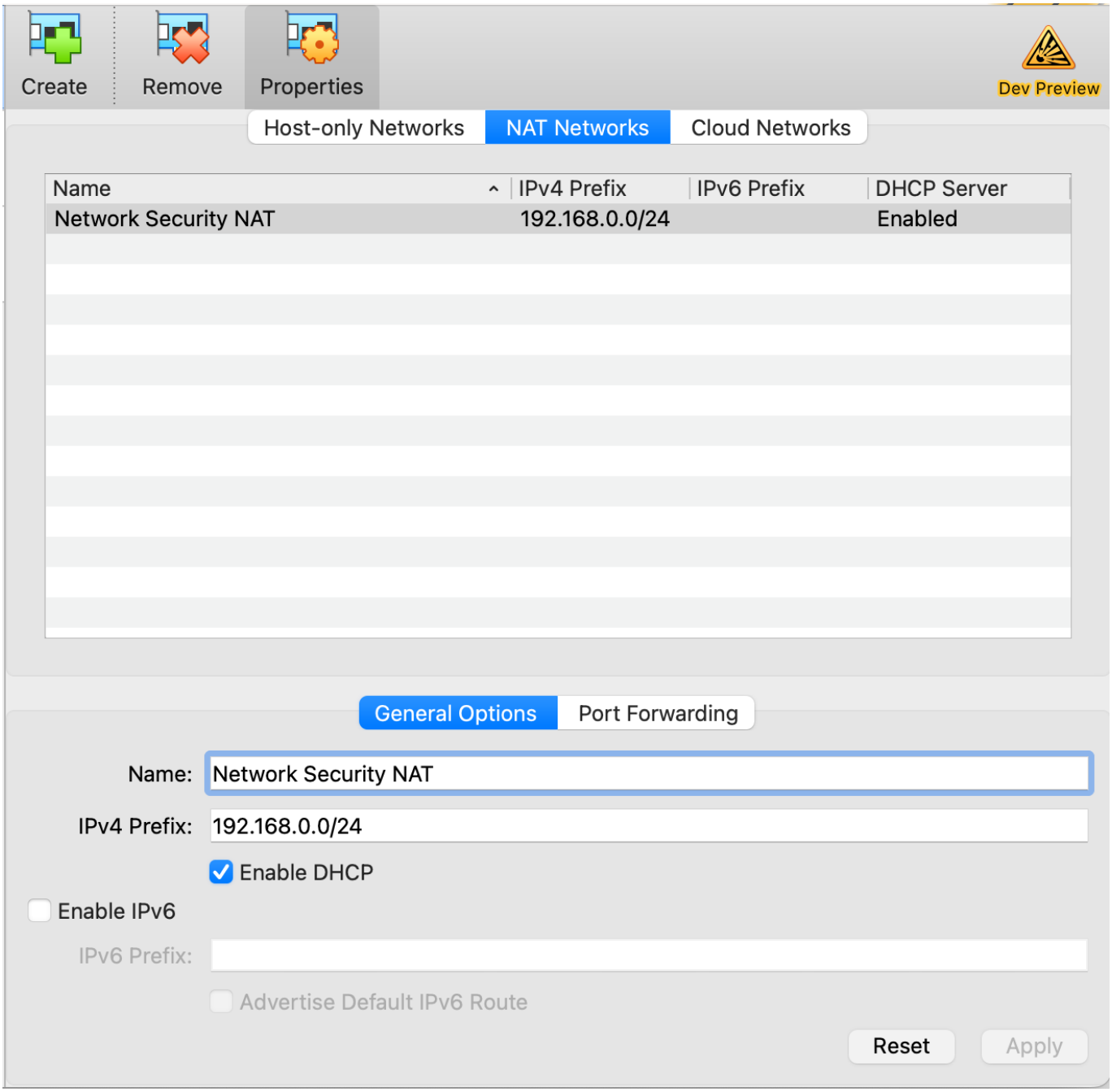

Figure 3 - NAT Network configuration

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, we provide you with the images/configurations for each of the VMs so you can import them directly into VirtualBox. These can be downloaded from here.

Configure VMs to connect to `Network Security NAT`

After downloading each VM configuration file, import them into VirtualBox (see `Import Appliance` option under `File` menu).

You need to ensure that each VM is connected to the previoulsy 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)

Now you can boot each VM.

Assignment 0 - Users and Credentials

1. Boot each VM and ensure you can access them using the following users:

- `root` (or `admin` on the Windows machine)
  - with password `ruteRULA`
- `auser`
  - with password `horseCACAnow`

2. 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.

Update 21-Sept-2023

- On the Windows host use the following credetials
  - login: `kevin`
  - password: `H@ckM3!fYouCan`

Assignment 1 - SSH Access

All machine need to be confiigured 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.

1. Before configuring SSH, you need to ensure that the **Host** machine can SSH into the VMs. Follow the steps described here:

- SSH into VirtualBox machine

2. 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
- SSH Key authentication

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).

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`.

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

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