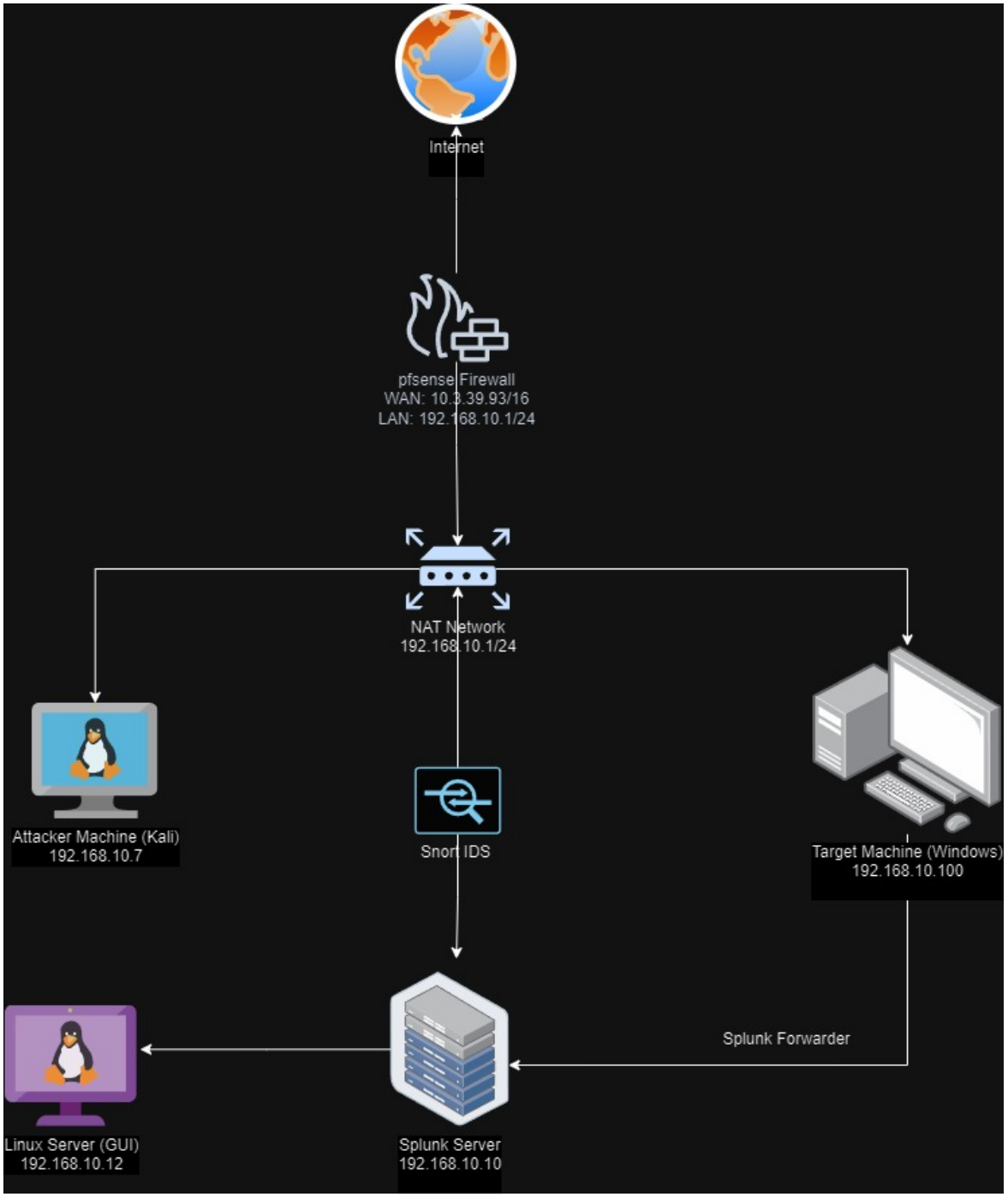
IDS and Firewall Setup

The firewall, implemented using pfSense, serves as the first line of defense, controlling incoming and outgoing network traffic based on predetermined security rules. Alongside, Snort, a widely used IDS, monitors network traffic in real-time, identifying and alerting administrators to any suspicious activities that could indicate a security breach.

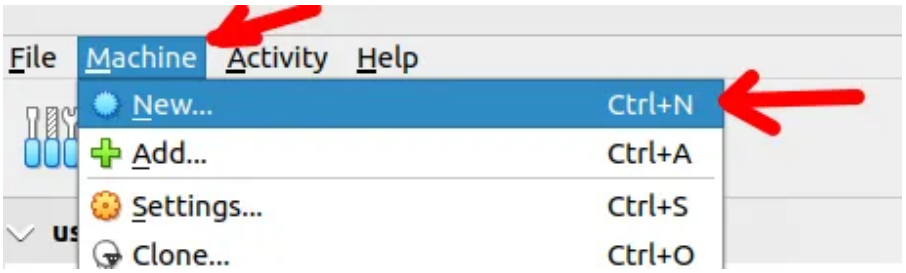


Acquiring the pfSense Image

To download pfSense, we will access the pfSense website and then go to the Downloads section.

Creating the Virtual Machine in VirtualBox

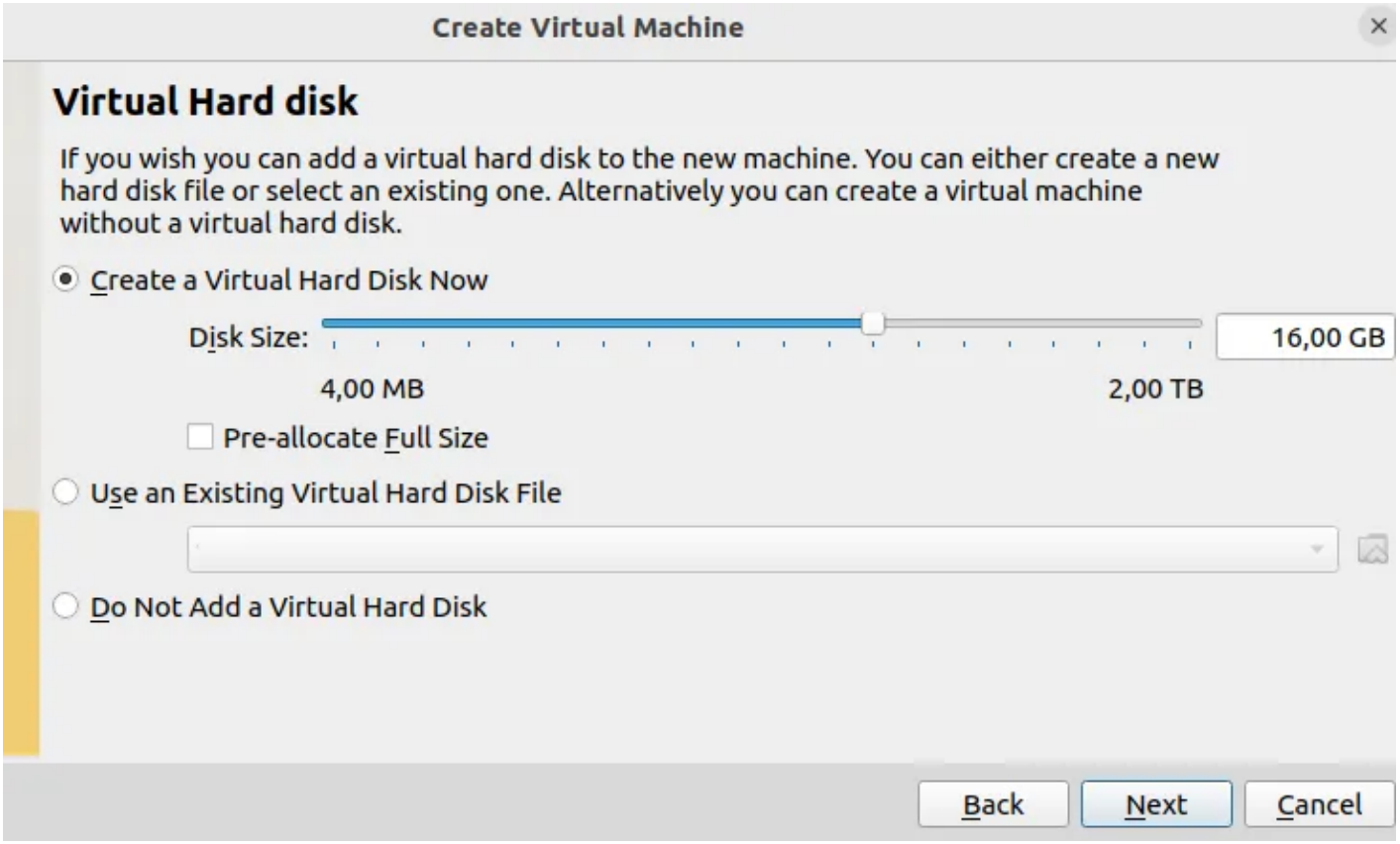
Now, let's open VirtualBox, click on the "Machine" tab, and select the "New" option to create a new virtual machine.



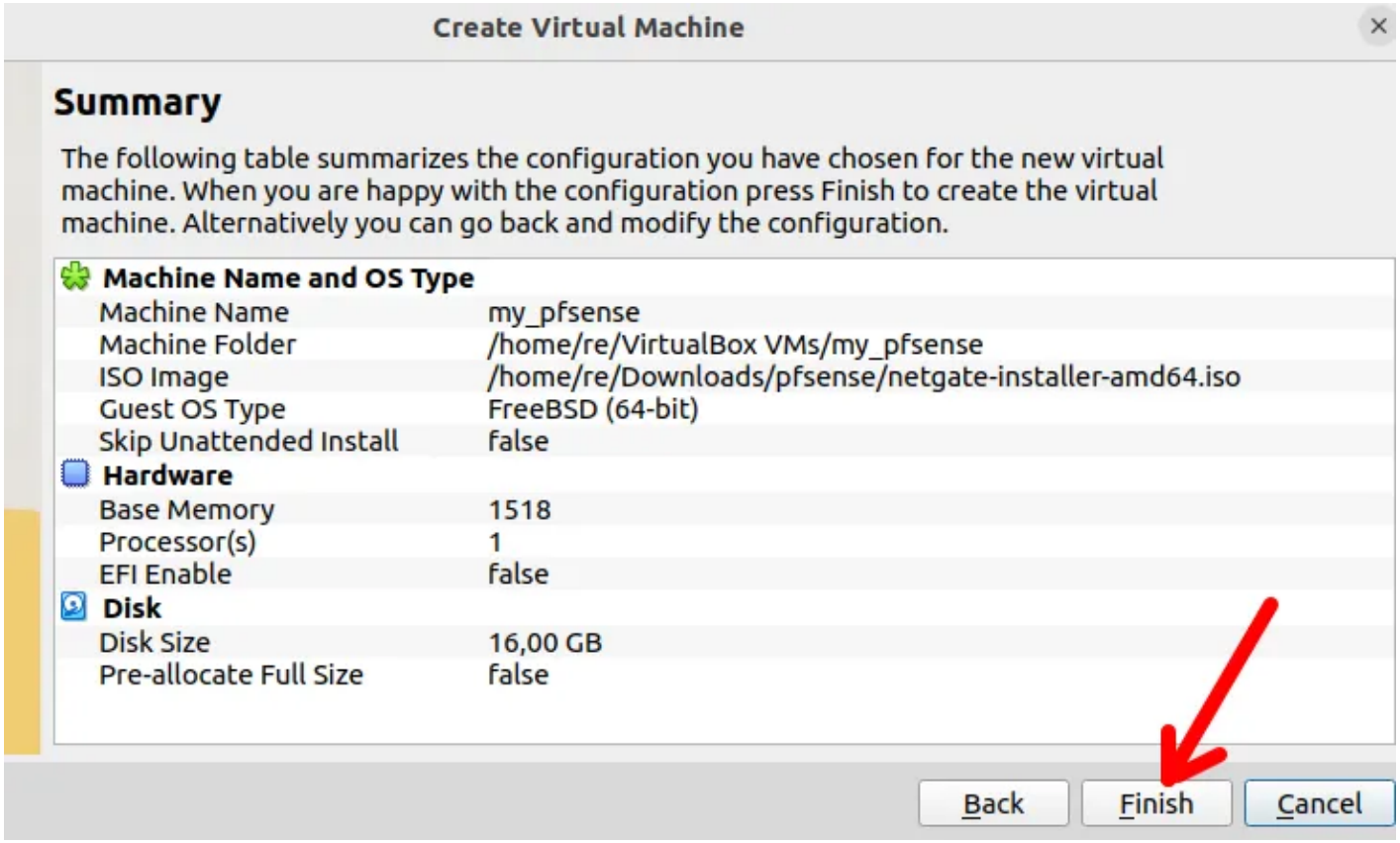
Then, let's name our virtual machine. In this case, we are naming it "my_pfsense".

In the "ISO Image" section, we will select the image we unzipped "iso". After that, we will select "Type = BSD" and "Version = FreeBSD 64 bit".

- Name: Name of your virtual machine. In our example, we are using the name "my_pfsense".
- Folder: The directory where we will store the pfSense virtual machine. ISO Image: We will select the pfSense ".
- ISO" image file that we downloaded and unzipped.
- Type: This will be the type of operating system. In our case, it is BSD.
- Version: Linux version, in this case we are using FreeBSD (64 bit).
- Base Memory: The minimum amount of RAM that will be allocated to the virtual machine. In this case, "1500 MB" is enough for our purposes. Remember that you can increase it according to your needs.
- Processors: The number of virtual processor cores allocated to our pfSense virtual machine. In this case, we will only use "1" core, but you can increase it according to your needs and the capacity of the host hardware.

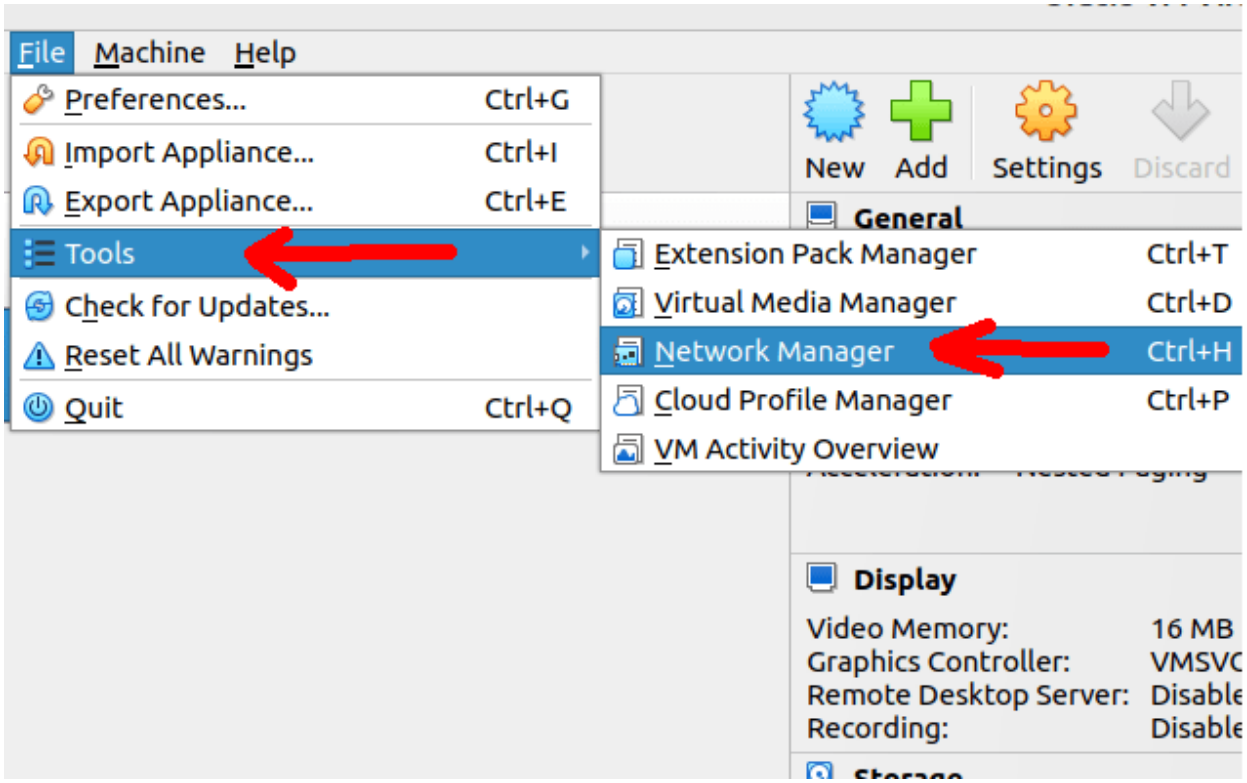


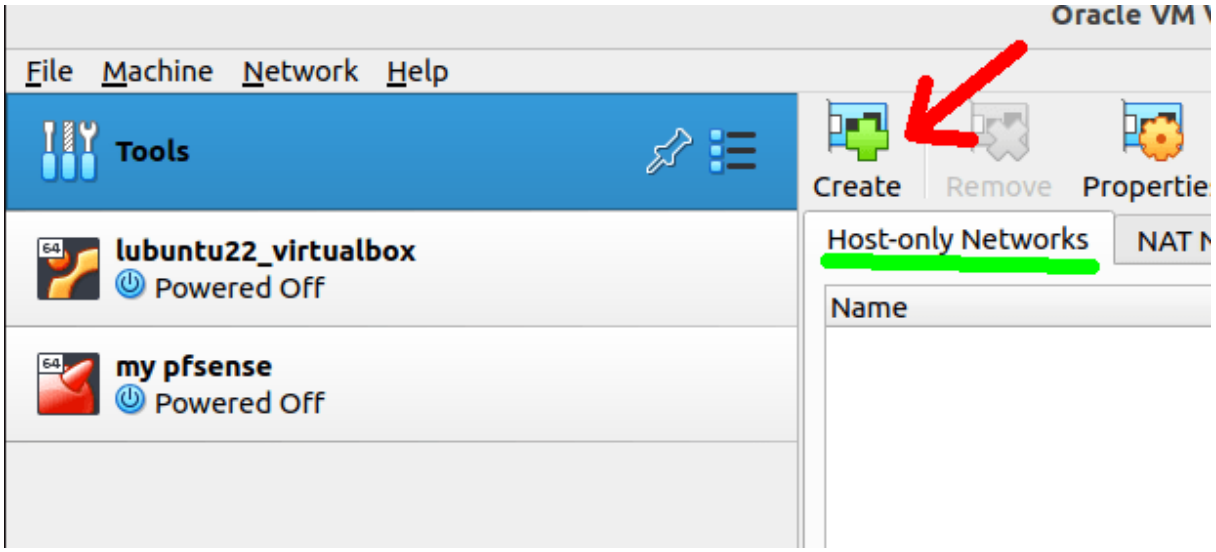
Done by clicking on Finish.



Creating a host-only interface in VirtualBox

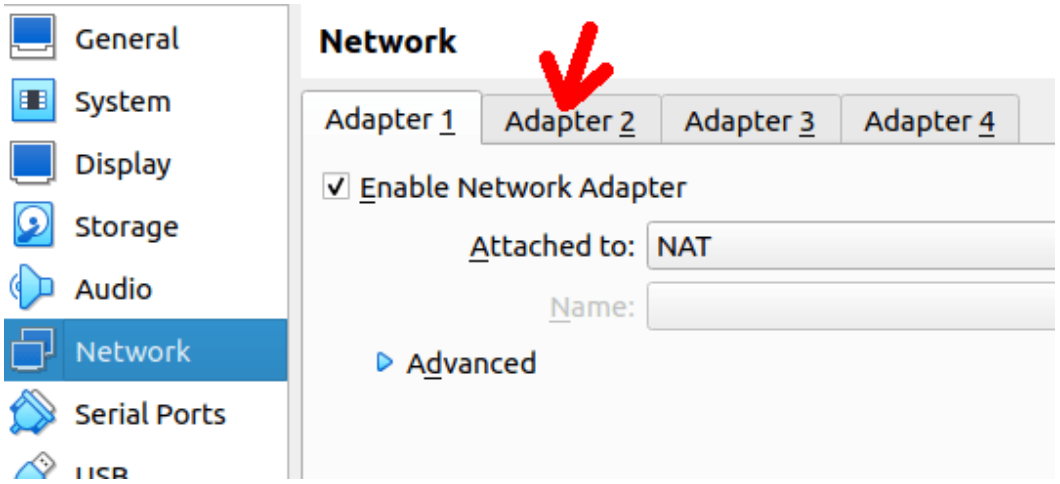
We will create a “host-only” network in VirtualBox. The reason for creating this network is to allow your real machine to access the pfSense LAN interface.



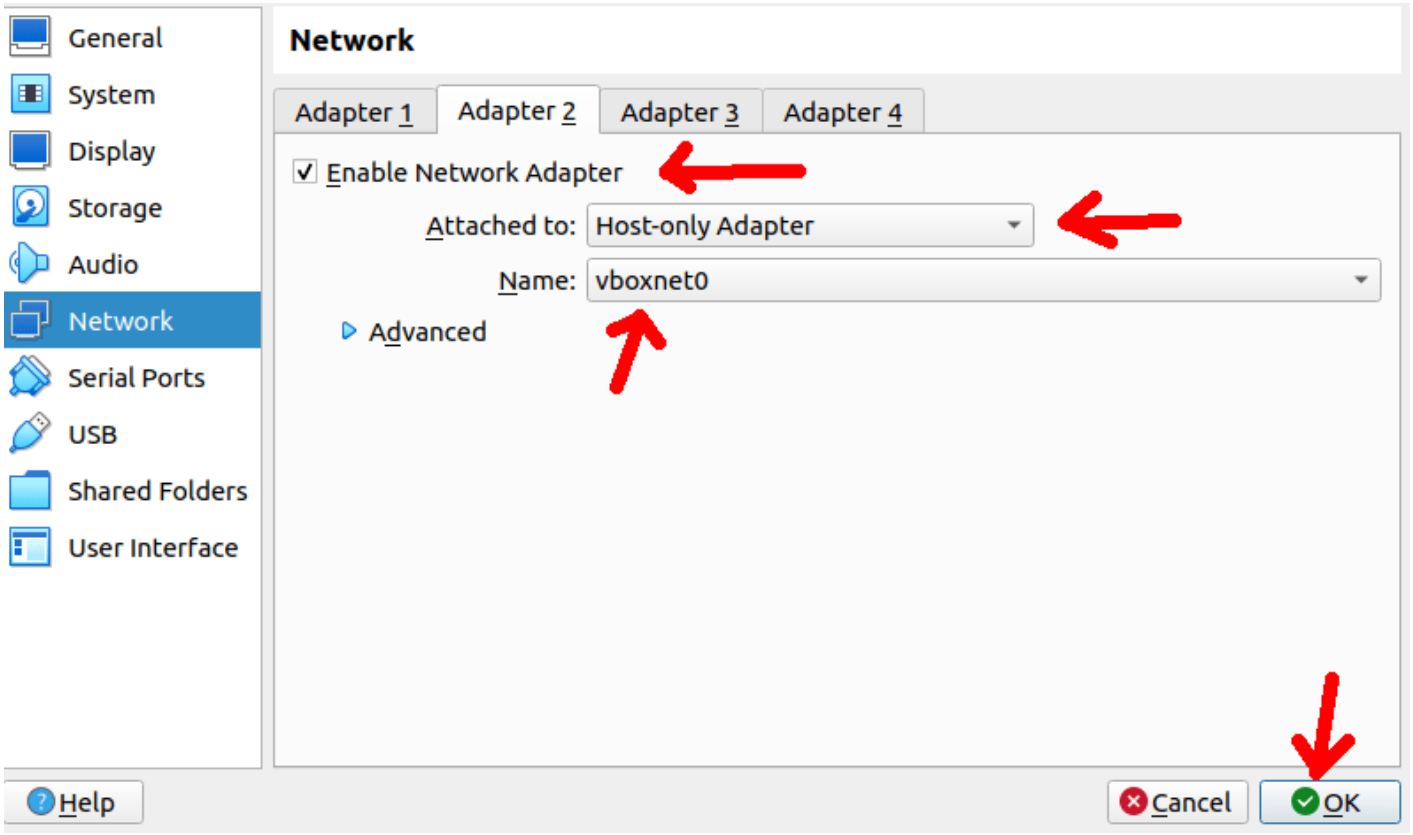


Adding pfSense networks in VirtualBox

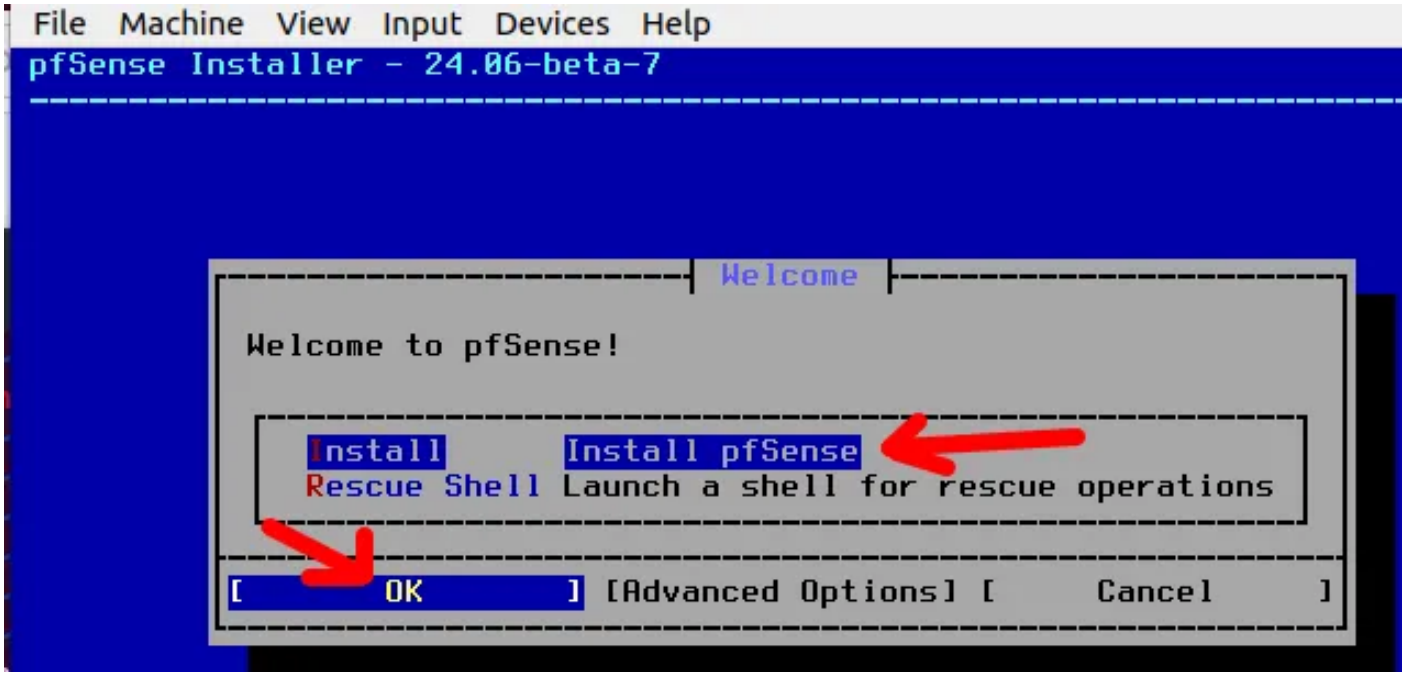
Now, we can see that a virtual machine has been created with the name we gave earlier “my pfsense”



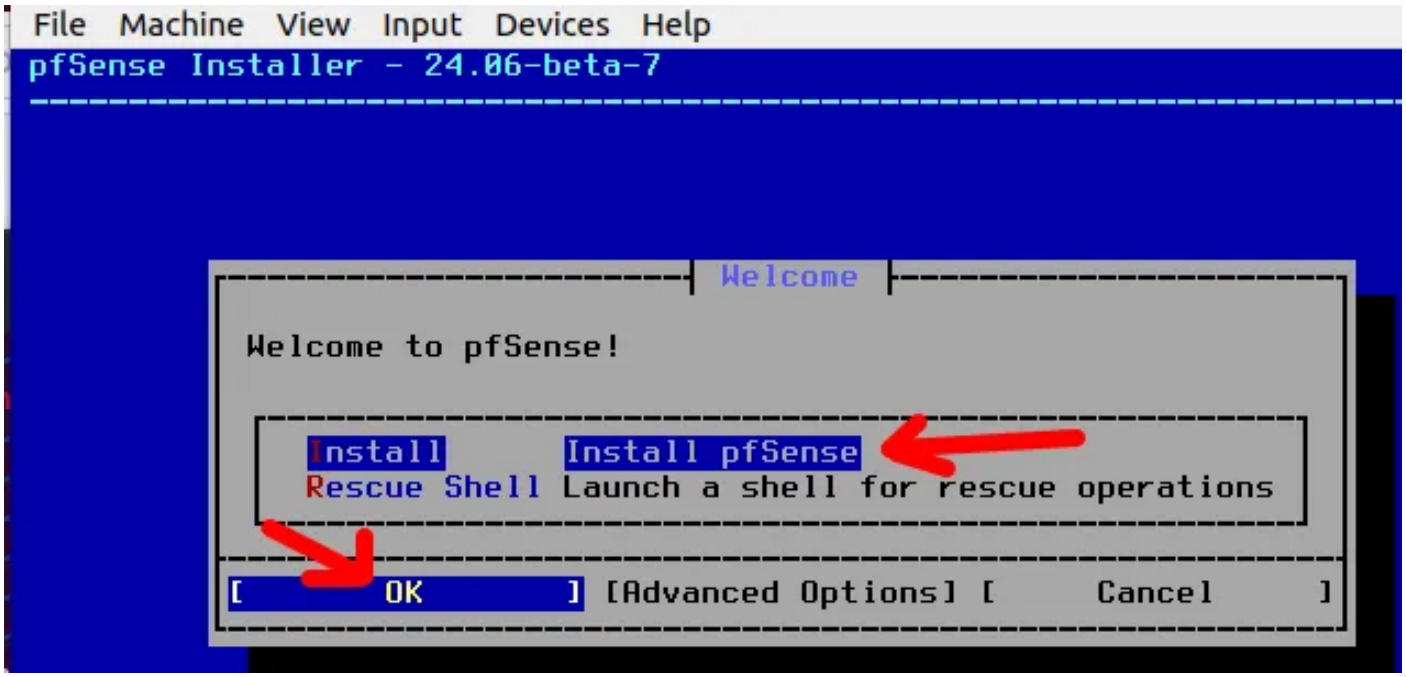
In “Adapter 2” we will select “Enable Network Adapter” and in “Attached to” we will select the “Host-only Adapter” option with the “Name” containing the name of our host-only network (vboxnet0)



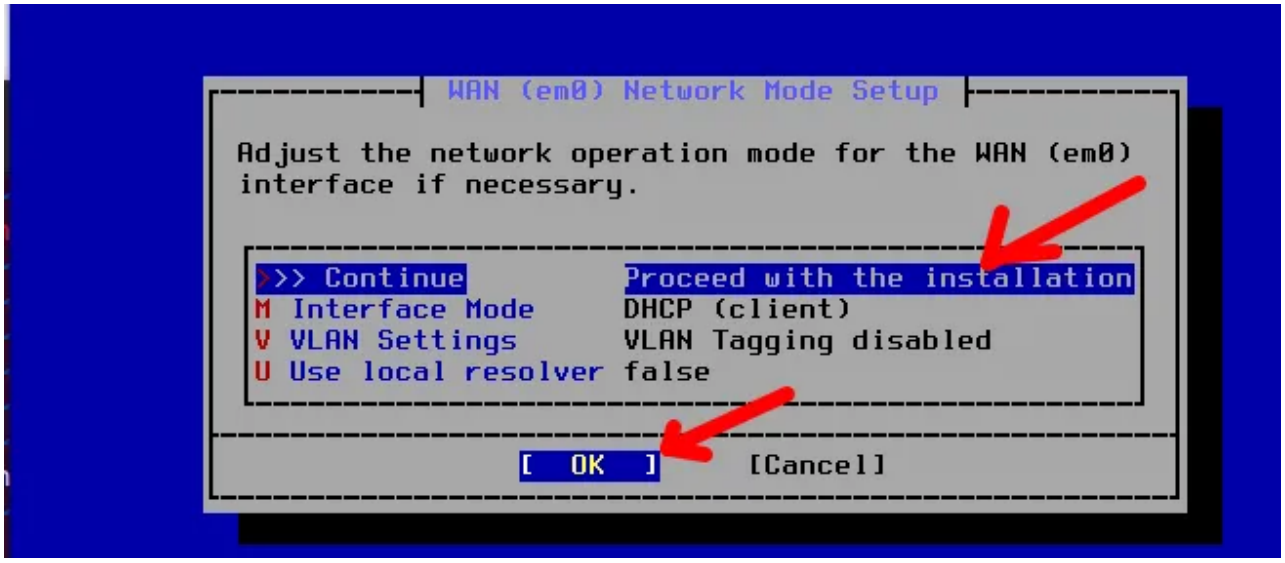
After agreeing to the terms, we will have the screen below where we will select “Install”.



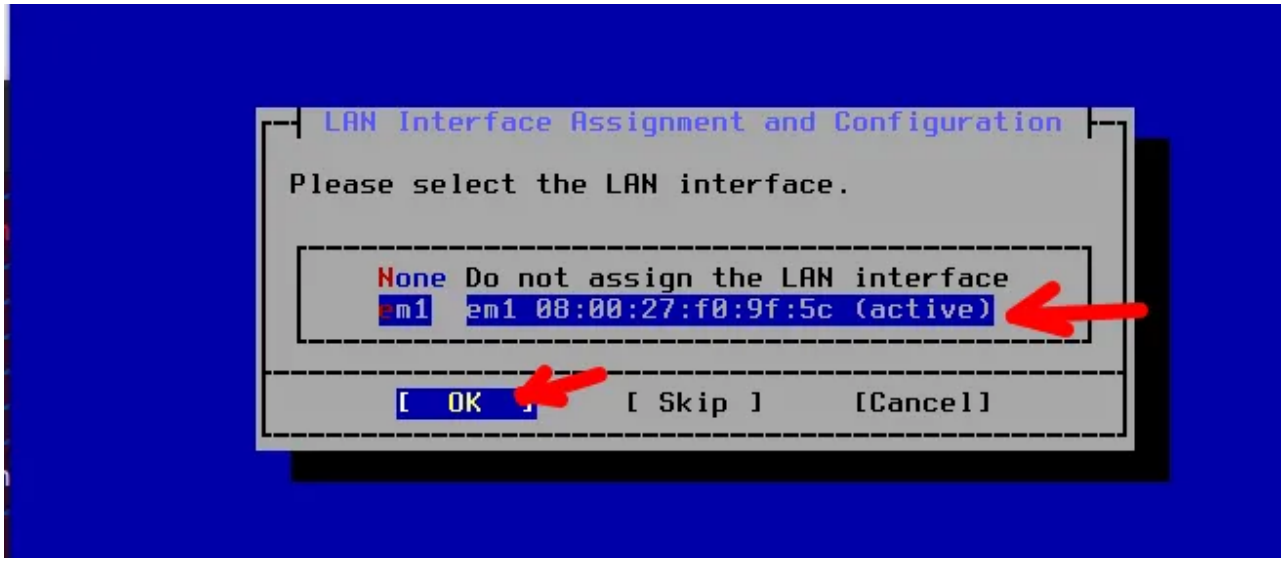
Now we will choose the pfSense WAN interface. In this case, we will choose the first “**em0**” as this will be the interface we have already configured as **NAT** in our VirtualBox.



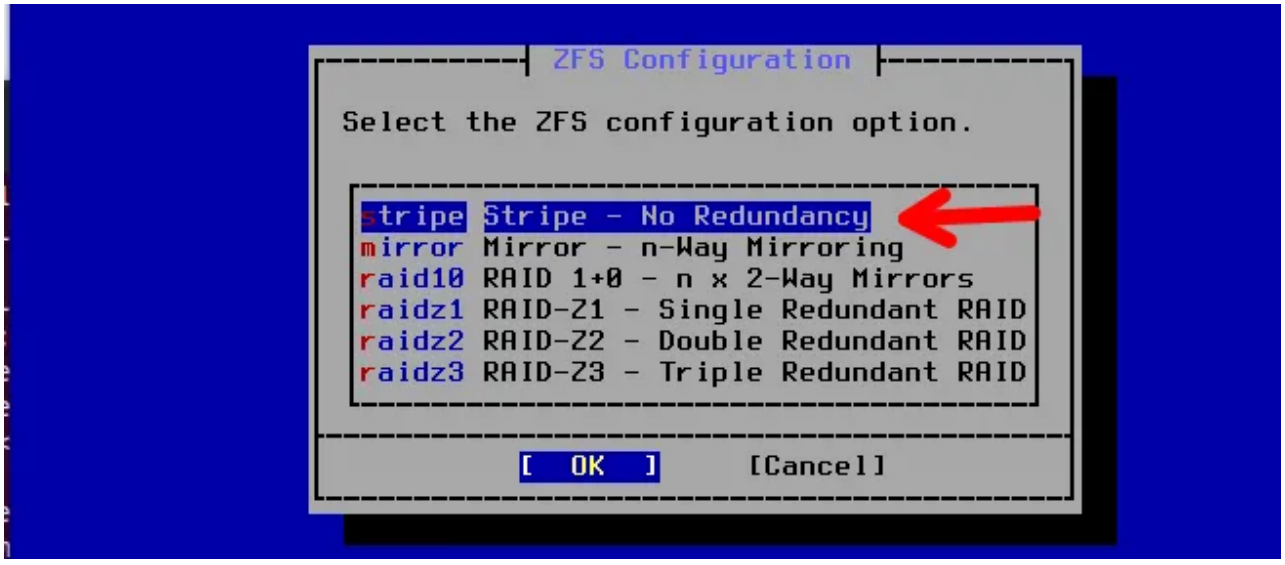
Next, we will select “Continue” and press “ENTER” to proceed with the installation.



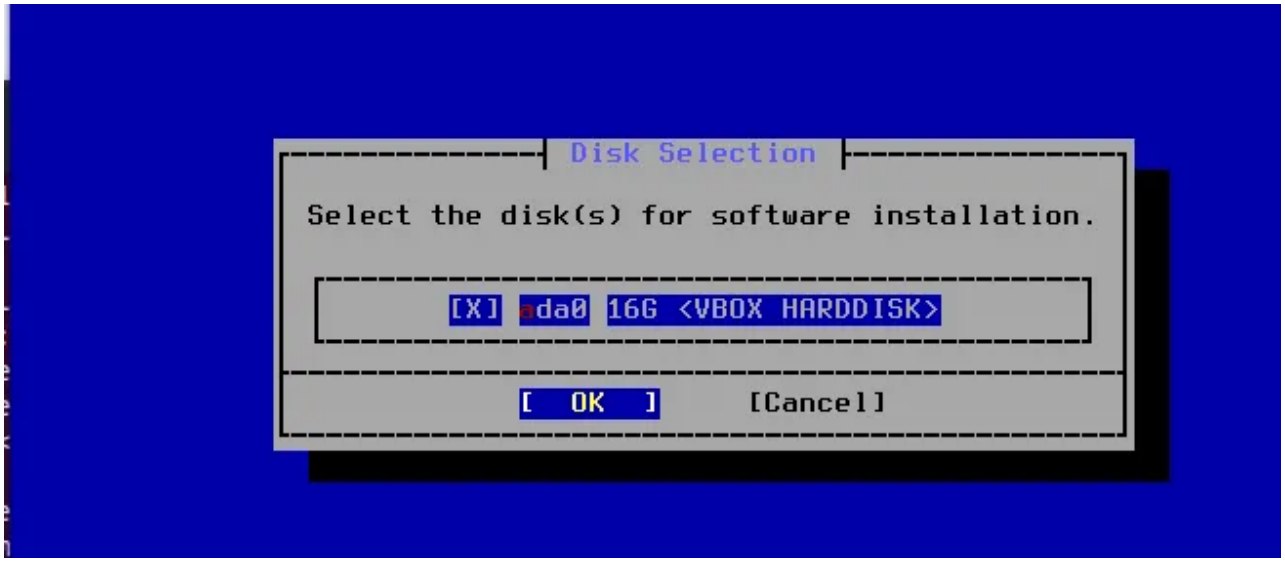
Now the system will ask you which interface will be used for the LAN. In this case, we will use **em1** which is set to “**host-only**” in our VirtualBox.



Now we will enter the ZFS configuration. In this case, we are not using redundancy, so we will select the first option "**Stripe – No Redundancy**"

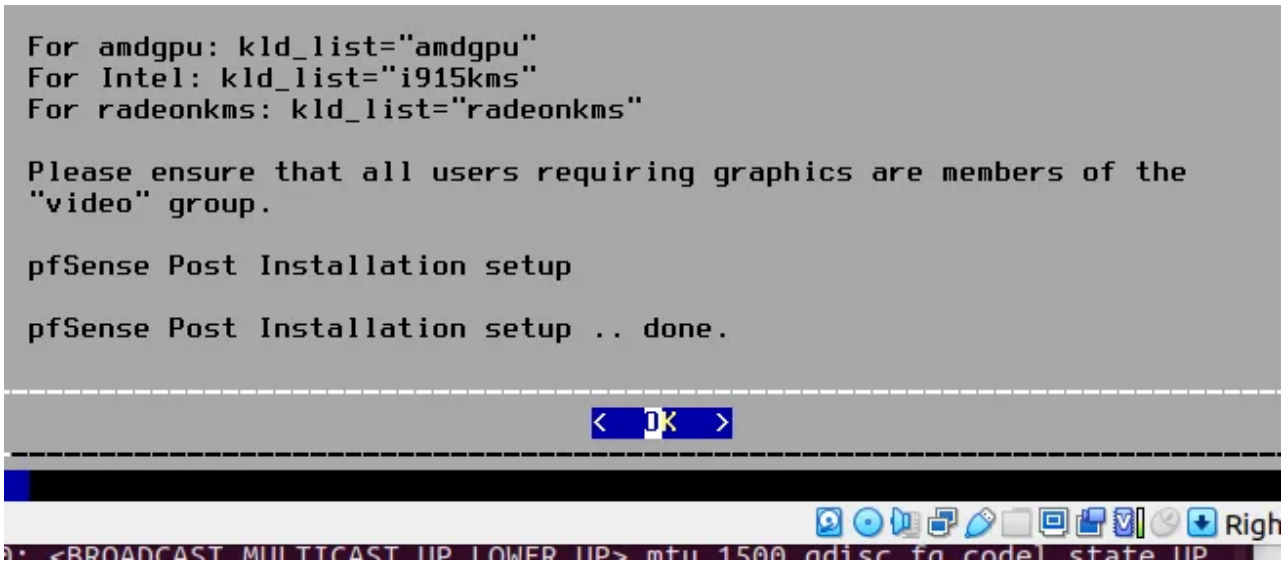


Next, we will select the disk that will be used for the installation. Remember that this disk is the virtual disk we will use in VirtualBox.

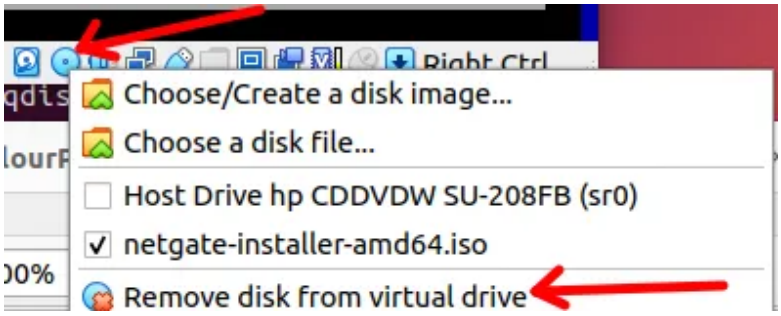


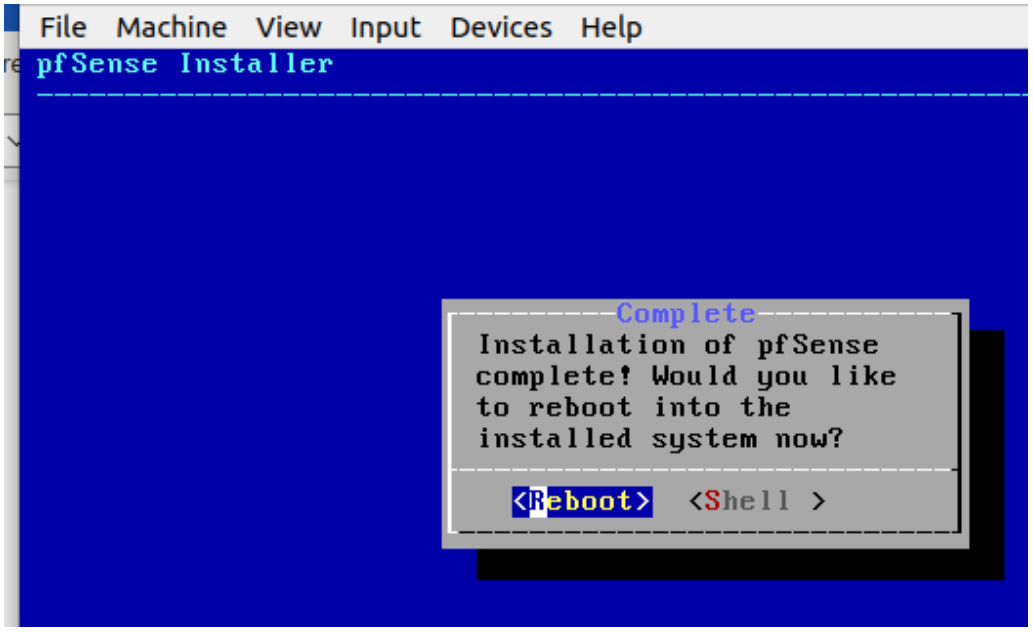
Now comes the question of whether we want to destroy the data on disk **ada0**. In this configuration, I don't need this disk, so I can destroy it.

After the installations are done, we will press “ENTER”



Before rebooting, let’s remove the pfSense installation media. To do this, go to the disc icon shown in the figure below and then click on “**Remove disk from virtual drive**“.





Changing the LAN IP of pfSense

We will change the LAN IP to be compatible with the **host-only** IP of **VirtualBox**.

In this way, we will be able to use the VirtualBox host-only network to access the WEB interface of our pfSense.

To change the LAN IP, we will type the number **2** + “ENTER”. Option **2** allows you to change the IPs of the interfaces “**Set interfaces IP address**“.

Then, we will type 2 again and press “ENTER” to select the LAN interface.

Next, for the question about obtaining IP on the LAN interface using DHCP, we will answer no = **n**.

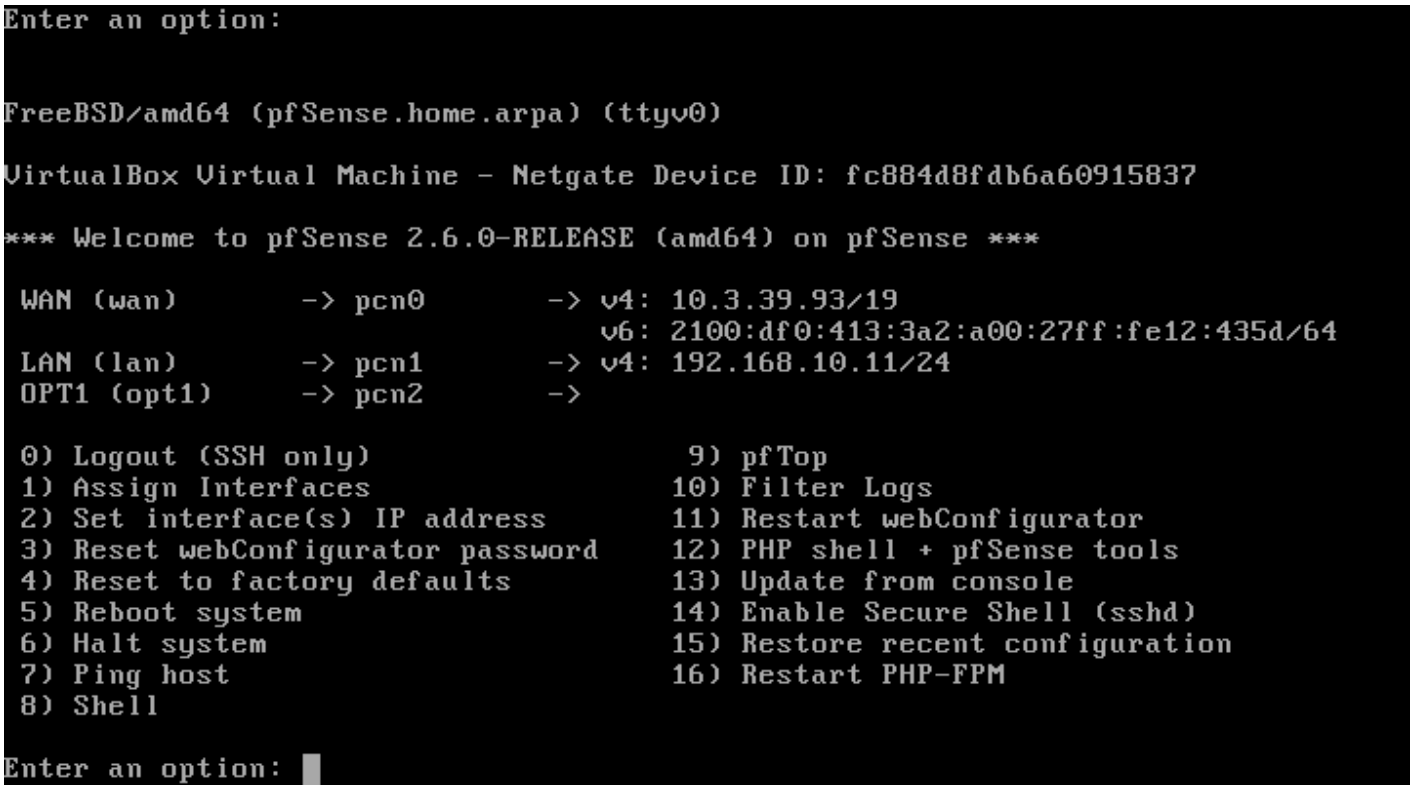
Now we will assign an IP to the LAN interface within the IP range of our host-only network.

In this case, we are entering the IP 192.168.10.11 for the LAN interface and the mask 24.

Now press Enter for the question about the LAN IPv4 upstream gateway. This is because we will use the default configuration.

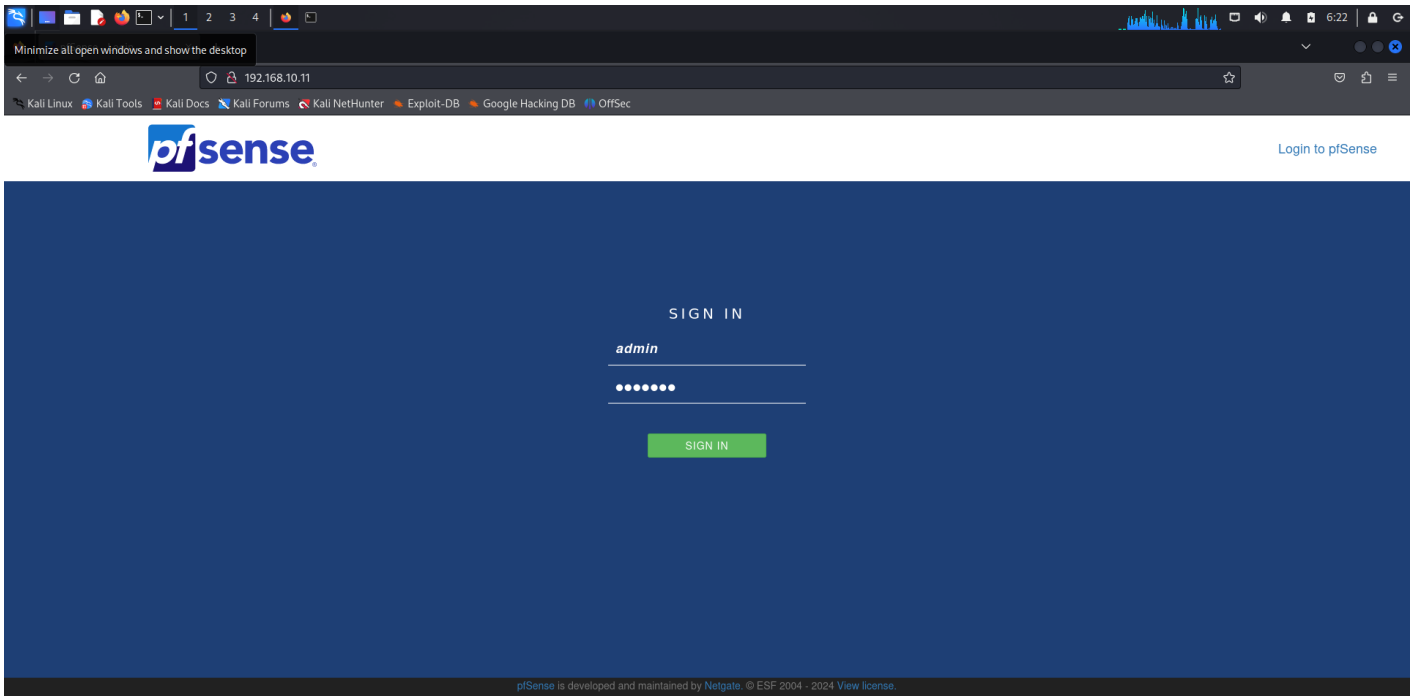
Next, we will decide if we want to use IPv6 on the LAN interface. In this case, we don’t want to, so we will type **“n**” and then press **ENTER** when asked what the IPv6 address would be.

After that, the question comes up whether we want to use DHCP on the LAN. In this case, we do, so we will answer with **“y**” and then enter the start of the network given by DHCP (192.168.10.1) and then the maximum (192.168.56.101).

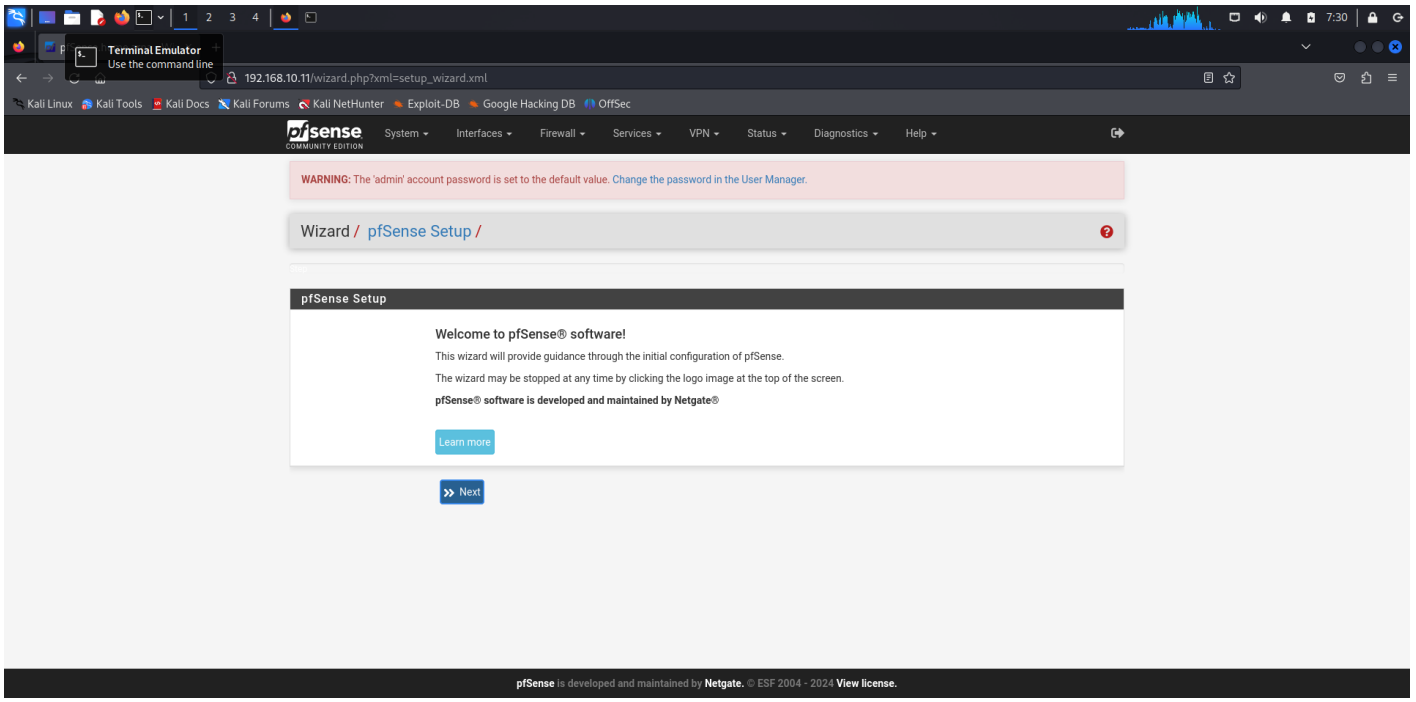


Accessing the pfSense WEB Interface

Now we will use the IP of our LAN to access the pfSense WEB interface. To do this, we will open a browser and type 192.168.10.11:80

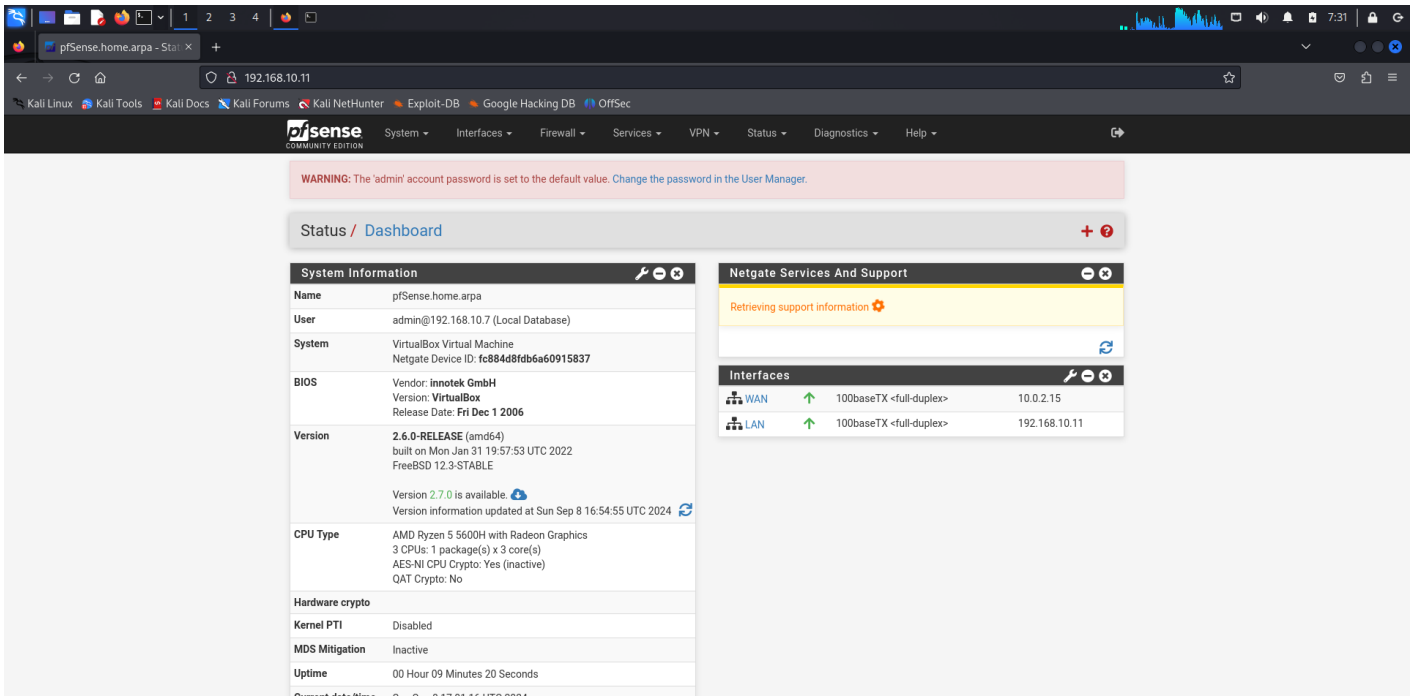


The website can be accessed with default credentials of username = admin & password = pfsense.

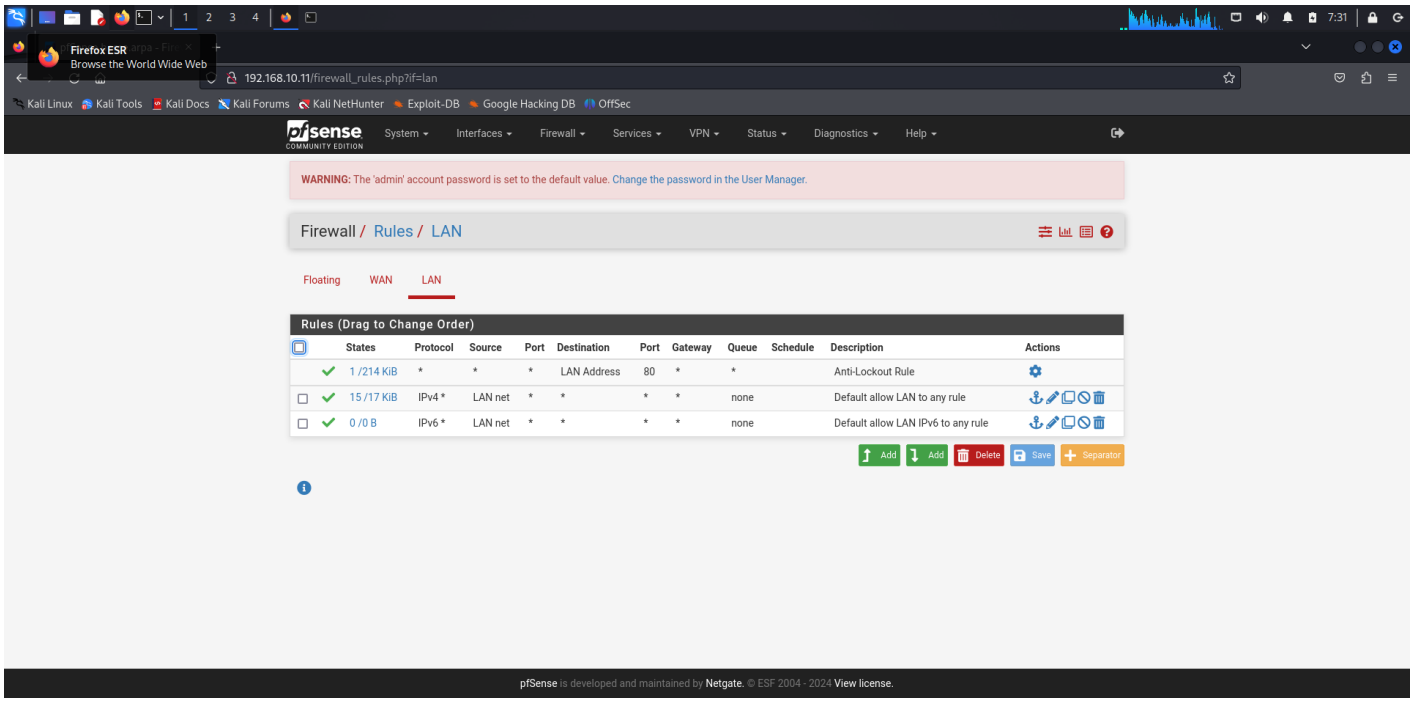


Click on next to move to start setting up the firewall rules.

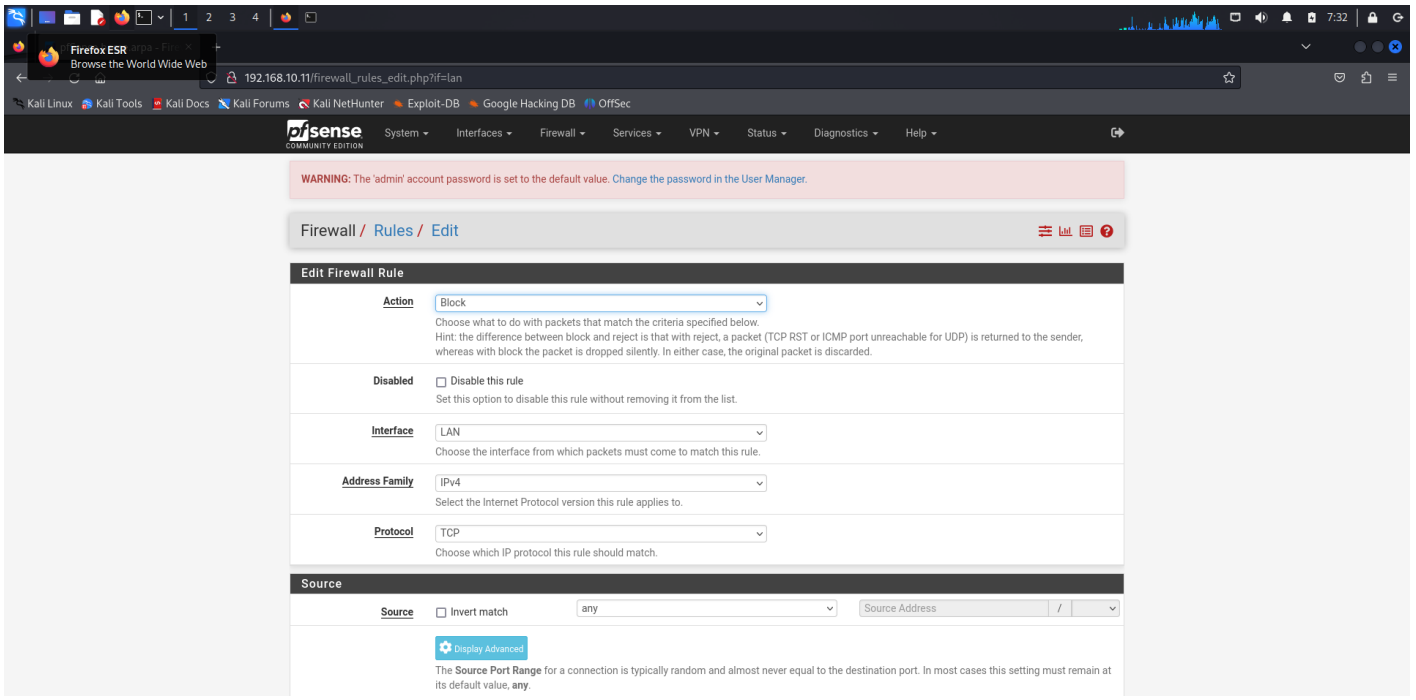
The following is the dashboard of pfsense, you can set rule by going in the Firewall>Rules tab.



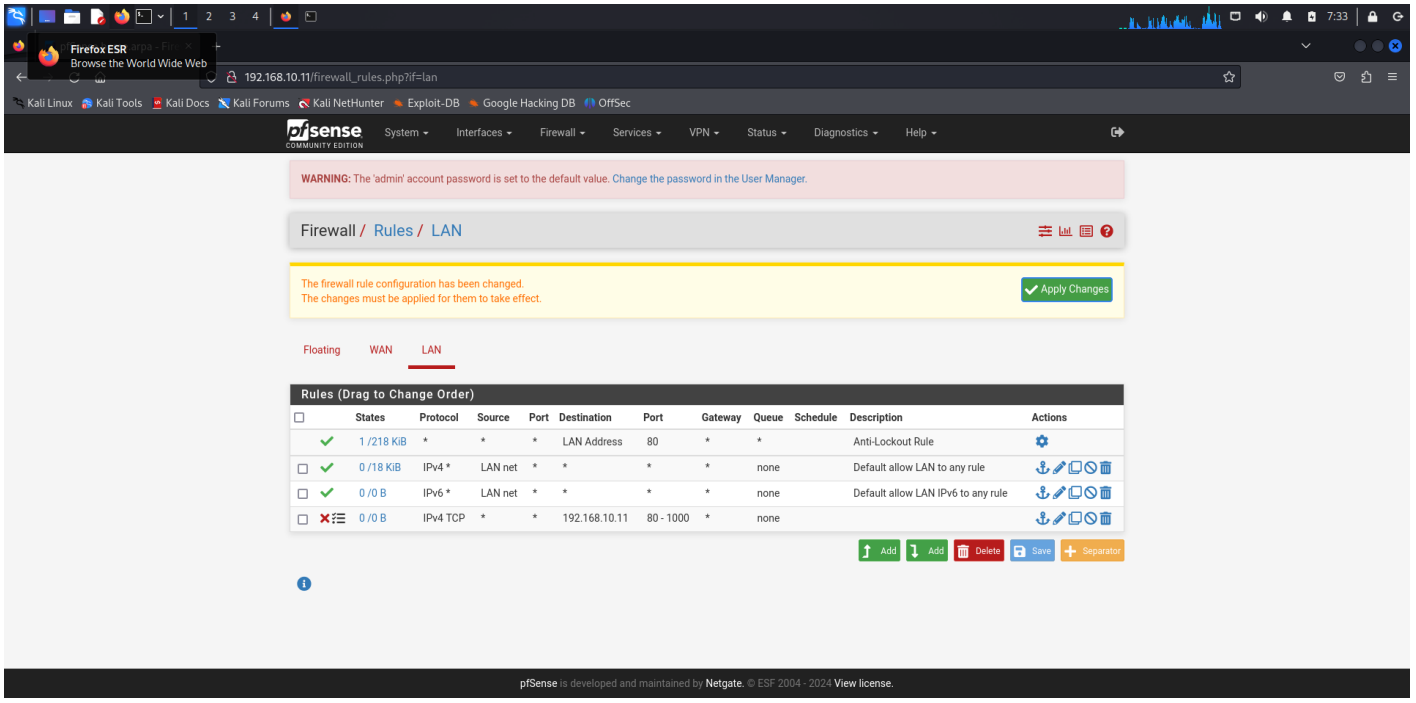
We will be Setting the rules for LAN .



Acourding to rule all the request coming to to 192.168.11.10 on the port range of 80-1000 will be Blocked by the Firewall.



After creating a the rule we can activate the rule by clicking "Apply Changes".



What Is Snort?

Snort analyzes network traffic in real-time and flags up any suspicious activity. In particular, it looks for anything that might indicate unauthorized access attempts and other attacks on the network. A comprehensive set of rules define what counts as "suspicious" and what Snort should do if a rule is triggered.

The Snort Rules

There are three sets of rules:

- **Community Rules:** These are freely available rule sets, created by the Snort user community.
- **Registered Rules:** These rule sets are provided by Talos. They are freely available also, but you must register to obtain them. Registration is free and only takes a moment. You'll receive a personal oinkcode that you need to include in the

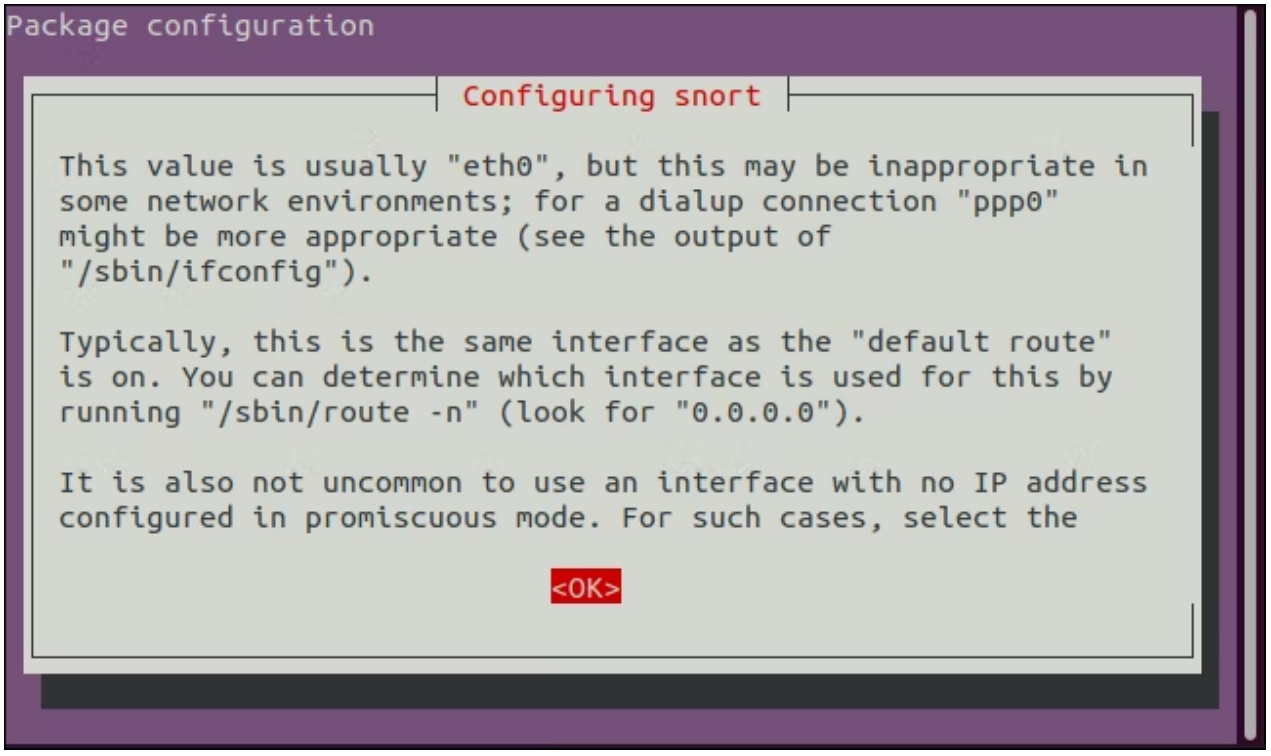
download request.

- **Subscription Rules:** These are the same rules as the registered rules. However, subscribers receive the rules about a month before they're released as free rule sets for registered users. At the time of writing, 12-month subscriptions start at USD $29 for personal use and USD $399 for business use.

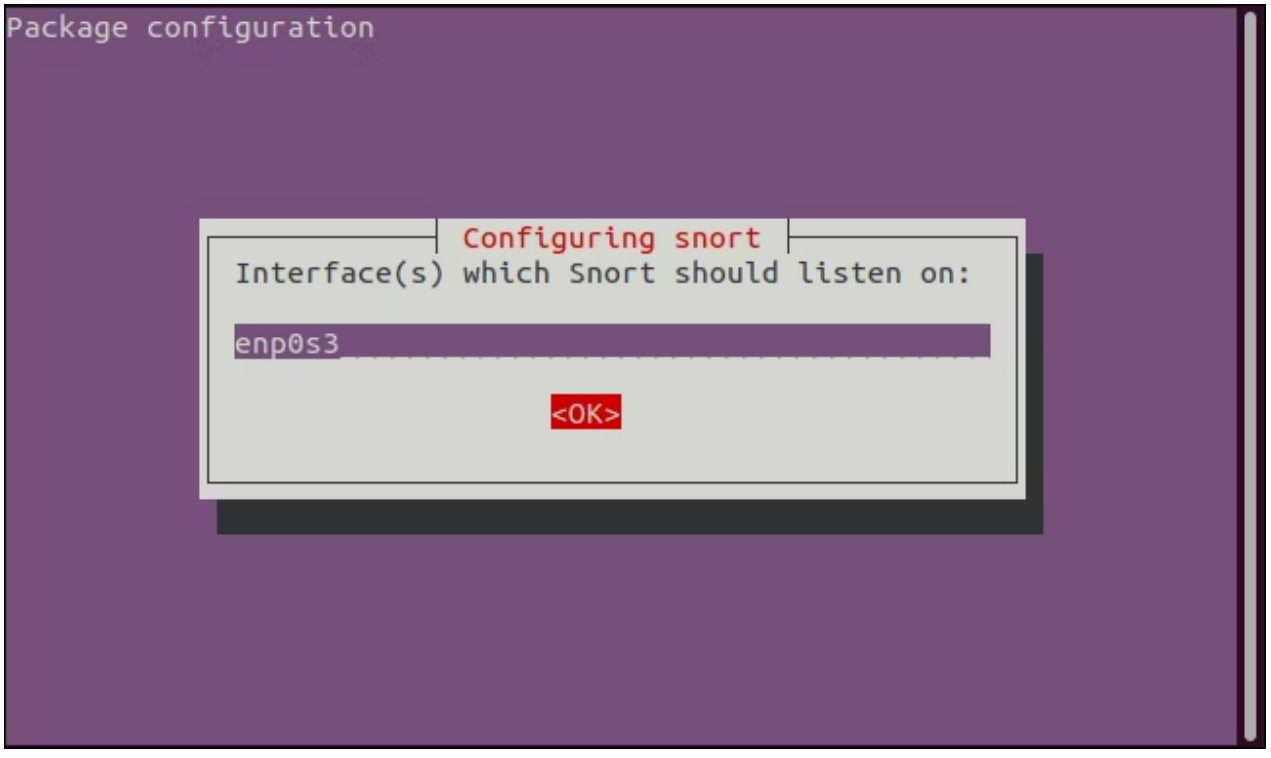
To install Snort on Ubuntu, use this command:

```
sudo apt-get install snort
```

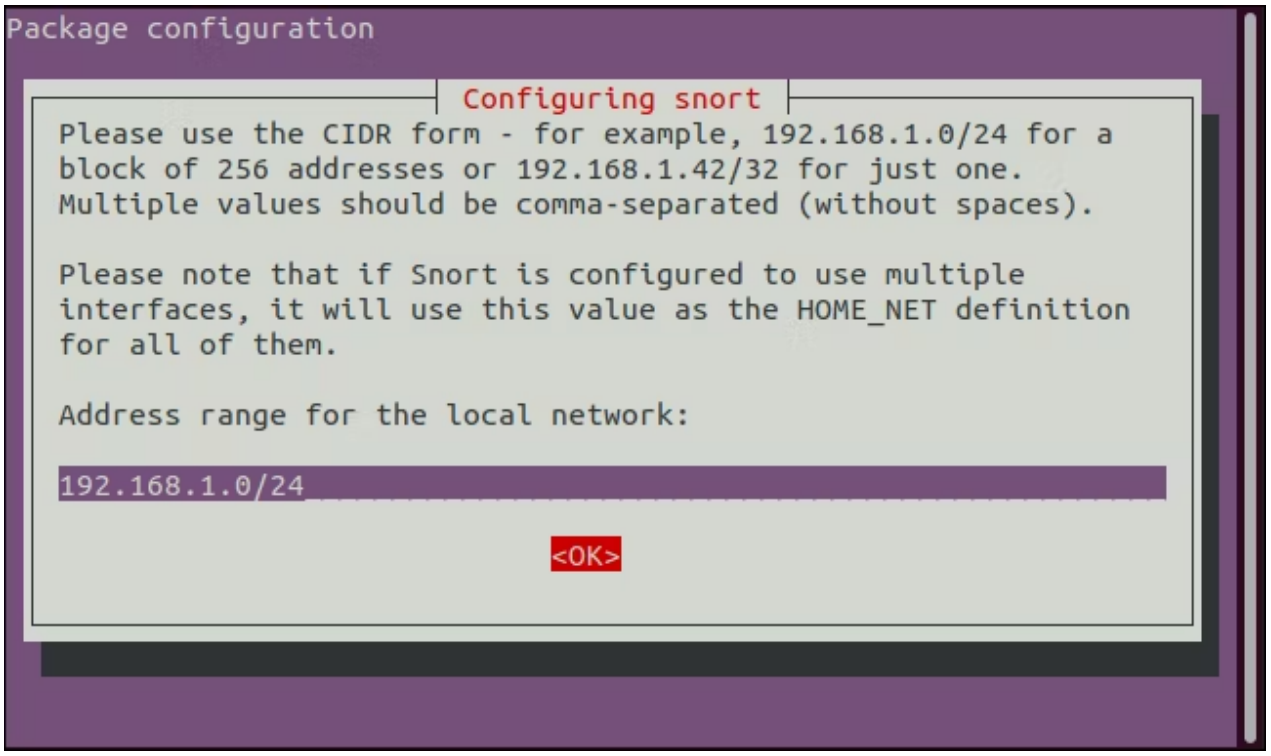
Press "Tab" to highlight the "OK" button, and press "Enter."



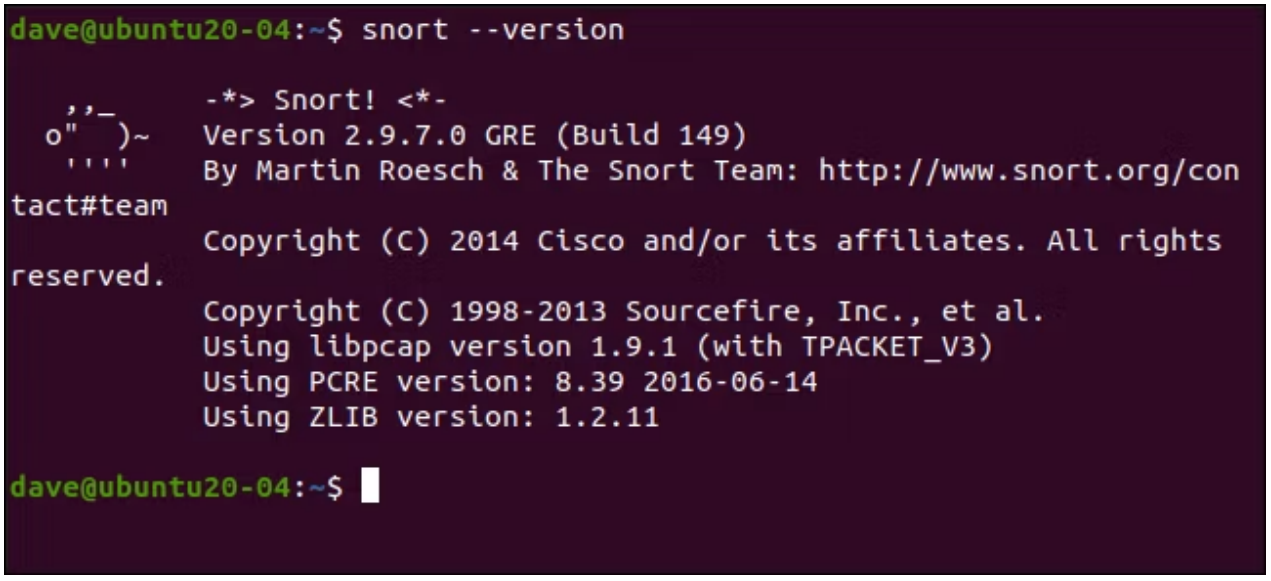
Type the name of the network interface name and press "Tab" to highlight the "OK" button, and press "Enter."



Type the network address range in CIDR format, press "Tab" to highlight the "OK" button, and press "Enter."




```
snort --version
```



Configuring Snort

There are a few steps to complete before we can run Snort. We need to edit the "snort.conf" file.

```
sudo nano /etc/snort/snort.conf
```

```
# Setup the network addresses you are protecting
#
# Note to Debian users: this value is overriden when starting
# up the Snort daemon through the init.d script by the
# value of DEBIAN_SNORT_HOME_NET s defined in the
# /etc/snort/snort.debian.conf configuration file
#
ipvar HOME_NET 192.168.1.0/24

# Set up the external network addresses. Leave as "any" in most situ
ipvar EXTERNAL_NET any
# If HOME_NET is defined as something other than "any", alternative,
# use this definition if you do not want to detect attacks from your
# IP addresses:
#ipvar EXTERNAL_NET !$HOME_NET

```

```
sudo systemctl start snort.service
```

To Access the Logs of the Snort:

```
sudo cat /var/log/snort/snort.log.fast
```

```
      valid_lft forever preferred_lft forever
splunk@splunk:~$ sudo cat /var/log/snort/
[sudo] password for splunk:
sudo: a password is required
splunk@splunk:~$ sudo cd  /var/log/snort/
[sudo] password for splunk:
sudo: cd: command not found
sudo: "cd" is a shell built-in command, it cannot be run directly.
sudo: the -s option may be used to run a privileged shell.
sudo: the -D option may be used to run a command in a specific directory.
splunk@splunk:~$ cd  /var/log/snort/
splunk@splunk:/var/log/snort$ ls
snort.alert  snort.alert.fast  snort.log
splunk@splunk:/var/log/snort$ sudo cat snort.alert.fast
09/08-10:33:00.618768  [**] [1:1421:11] SNMP AgentX/tcp request [**] [Classification: Attempted Information Leak] [Priority: 2] {TCP} 192.168.10.12:38008 -> 192
.168.10.10:705
09/08-10:33:00.762051  [**] [1:1418:11] SNMP request tcp [**] [Classification: Attempted Information Leak] [Priority: 2] {TCP} 192.168.10.12:57864 -> 192.168.10
.10:161
09/08-10:33:47.219273  [**] [1:1421:11] SNMP AgentX/tcp request [**] [Classification: Attempted Information Leak] [Priority: 2] {TCP} 192.168.10.12:40746 -> 192
.168.10.10:705
09/08-10:33:47.898568  [**] [1:1420:11] SNMP trap tcp [**] [Classification: Attempted Information Leak] [Priority: 2] {TCP} 192.168.10.12:49372 -> 192.168.10.10
:162
09/08-10:33:51.704448  [**] [1:1418:11] SNMP request tcp [**] [Classification: Attempted Information Leak] [Priority: 2] {TCP} 192.168.10.12:33234 -> 192.168.10
.10:161
09/08-10:34:02.297598  [**] [1:249:8] DDOS mstream client to handler [**] [Classification: Attempted Denial of Service] [Priority: 2] {TCP} 192.168.10.12:40018
-> 192.168.10.10:15104
09/08-10:35:12.305985  [**] [1:1421:11] SNMP AgentX/tcp request [**] [Classification: Attempted Information Leak] [Priority: 2] {TCP} 192.168.10.12:57018 -> 192
.168.10.10:705
09/08-10:35:14.123347  [**] [1:249:8] DDOS mstream client to handler [**] [Classification: Attempted Denial of Service] [Priority: 2] {TCP} 192.168.10.12:57018
-> 192.168.10.10:15104
09/08-10:35:16.562670  [**] [1:1420:11] SNMP trap tcp [**] [Classification: Attempted Information Leak] [Priority: 2] {TCP} 192.168.10.12:57018 -> 192.168.10.10
:162
09/08-10:35:21.768343  [**] [1:1418:11] SNMP request tcp [**] [Classification: Attempted Information Leak] [Priority: 2] {TCP} 192.168.10.12:57018 -> 192.168.10
.10:161
09/08-10:36:37.331892  [**] [1:249:8] DDOS mstream client to handler [**] [Classification: Attempted Denial of Service] [Priority: 2] {TCP} 192.168.10.12:52515
-> 192.168.10.10:15104
09/08-10:36:39.645915  [**] [1:1421:11] SNMP AgentX/tcp request [**] [Classification: Attempted Information Leak] [Priority: 2] {TCP} 192.168.10.12:52515 -> 192
.168.10.10:705
09/08-10:36:48.336853  [**] [1:1418:11] SNMP request tcp [**] [Classification: Attempted Information Leak] [Priority: 2] {TCP} 192.168.10.12:52515 -> 192.168.10
.10:161
09/08-10:36:49.934162  [**] [1:1420:11] SNMP trap tcp [**] [Classification: Attempted Information Leak] [Priority: 2] {TCP} 192.168.10.12:52515 -> 192.168.10.10
:162
09/08-11:26:08.294791  [**] [1:527:8] BAD-TRAFFIC same SRC/DST [**] [Classification: Potentially Bad Traffic] [Priority: 2] {UDP} 0.0.0.0:68 -> 255.255.255.255:
67
09/08-12:52:19.090040  [**] [1:527:8] BAD-TRAFFIC same SRC/DST [**] [Classification: Potentially Bad Traffic] [Priority: 2] {UDP} 0.0.0.0:68 -> 255.255.255.255:
67
09/08-12:52:19.105977  [**] [1:527:8] BAD-TRAFFIC same SRC/DST [**] [Classification: Potentially Bad Traffic] [Priority: 2] {IPV6-ICMP} :: -> ff02::16
09/08-12:52:19.623368  [**] [1:527:8] BAD-TRAFFIC same SRC/DST [**] [Classification: Potentially Bad Traffic] [Priority: 2] {IPV6-ICMP} :: -> ff02::16
09/08-12:52:19.844044  [**] [1:527:8] BAD-TRAFFIC same SRC/DST [**] [Classification: Potentially Bad Traffic] [Priority: 2] {IPV6-ICMP} :: -> ff02::1:ff76:2726
splunk@splunk:/var/log/snort$
```