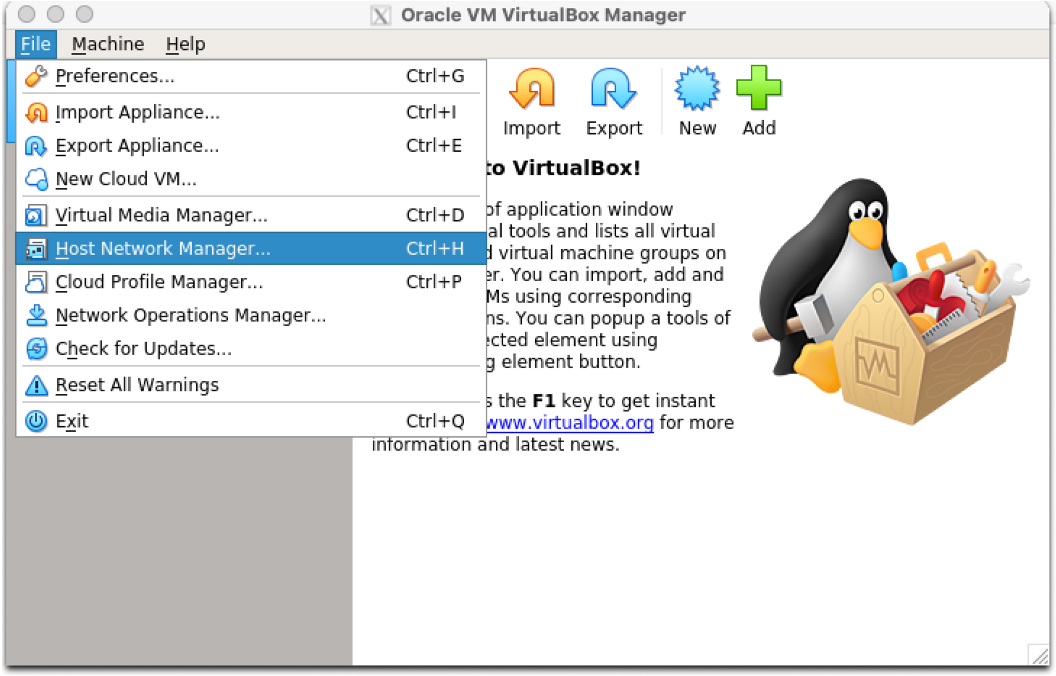
Building The Real Network

Part A: Configuring a Local Area Network (LAN) to host our small-business network

Step 1: Adding a new virtual network adaptor [which will be used to support a host-only LAN]

- Select "File” -> “Host Network Manager"

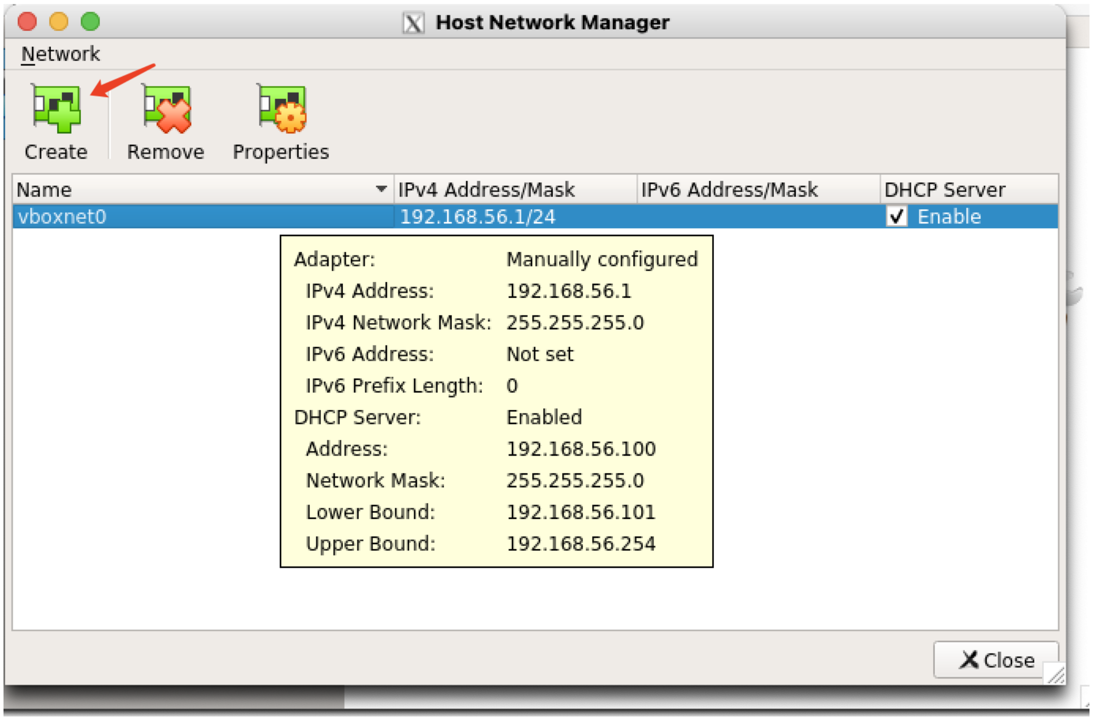

Oracle VM VirtualBox Manager
File
Machine
Help
Preferences... Ctrl+G
Import Appliance... Ctrl+I
Export Appliance... Ctrl+E
New Cloud VM...
Virtual Media Manager... Ctrl+D
Host Network Manager... Ctrl+H
Cloud Profile Manager... Ctrl+P
Network Operations Manager...
Check for Updates...
Reset All Warnings
Exit Ctrl+Q
Import
Export
New
Add

to VirtualBox!

of application window al tools and lists all virtual d virtual machine groups on er. You can import, add and Ms using corresponding ns. You can popup a tools of ected element using g element button.

s the **F1** key to get instant www.virtualbox.org for more information and latest news.

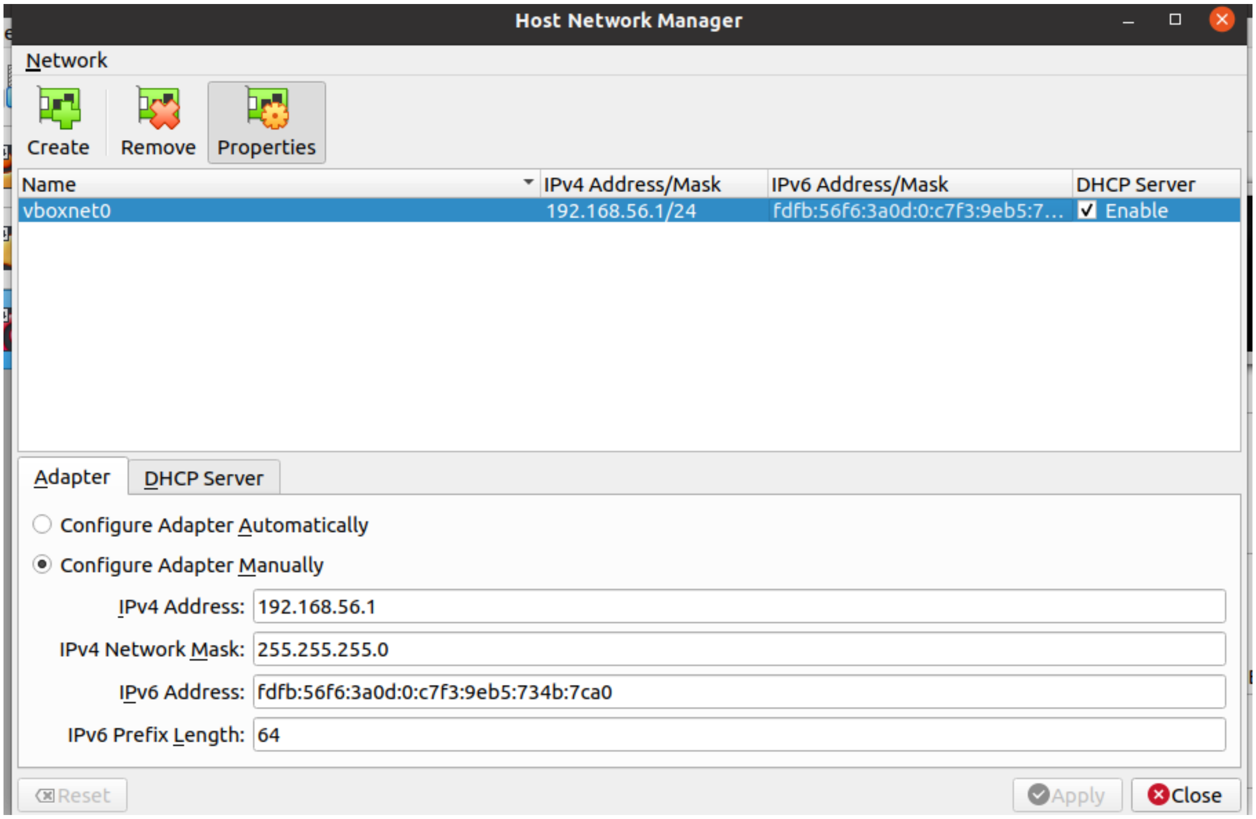
- Create a new virtual network adapter



Name	IPv4 Address/Mask	IPv6 Address/Mask	DHCP Server
vboxnet0	192.168.56.1/24		☑ Enable

Adapter:	Manually configured
IPv4 Address:	192.168.56.1
IPv4 Network Mask:	255.255.255.0
IPv6 Address:	Not set
IPv6 Prefix Length:	0
DHCP Server:	Enabled
Address:	192.168.56.100
Network Mask:	255.255.255.0
Lower Bound:	192.168.56.101
Upper Bound:	192.168.56.254

- Make sure you “Enable” DHCP
- Make sure you configure “Adapter” as follows



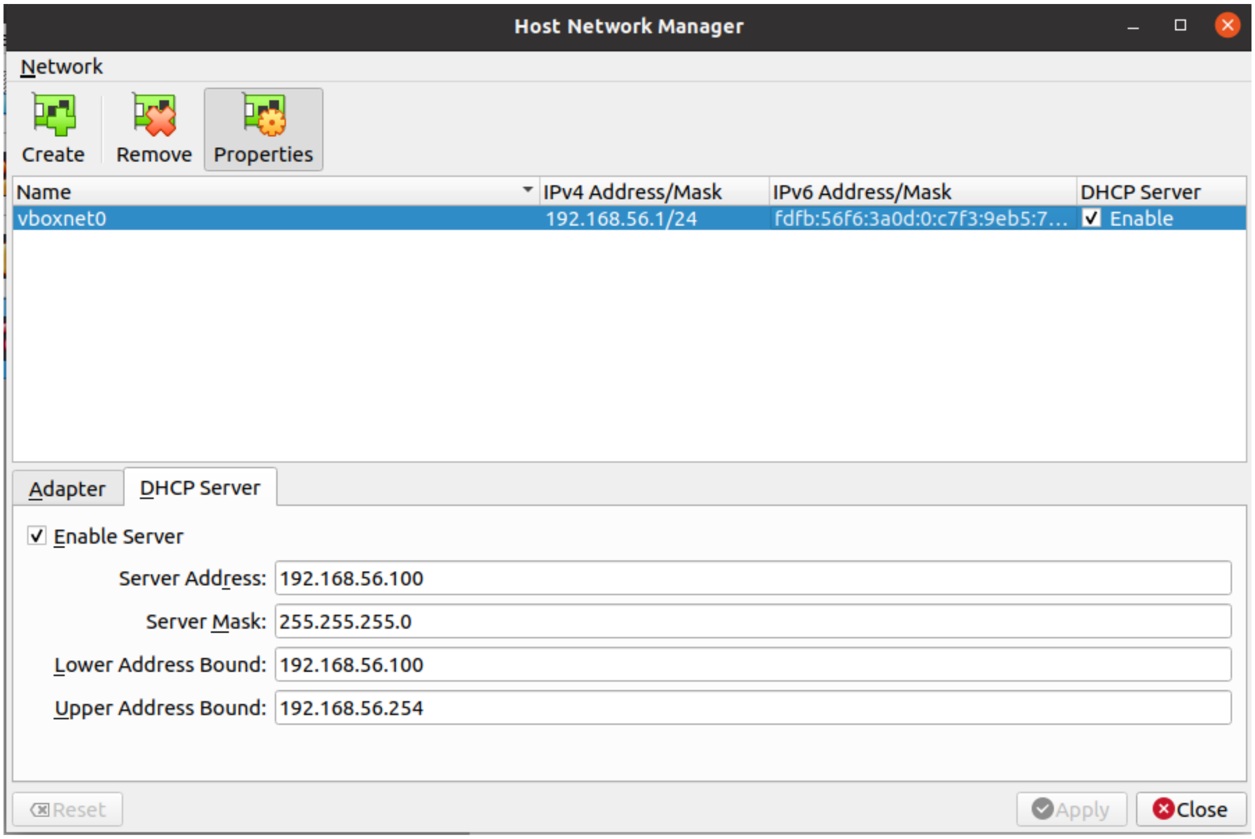
Name	IPv4 Address/Mask	IPv6 Address/Mask	DHCP Server
vboxnet0	192.168.56.1/24	fdfb:56f6:3a0d:0:c7f3:9eb5:7…	☑ Enable

○ Configure Adapter Automatically

● Configure Adapter Manually

IPv4 Address:	192.168.56.1
IPv4 Network Mask:	255.255.255.0
IPv6 Address:	fdfb:56f6:3a0d:0:c7f3:9eb5:734b:7ca0
IPv6 Prefix Length:	64

- Make sure you configure “DHCP” as follows



Name	IPv4 Address/Mask	IPv6 Address/Mask	DHCP Server
vboxnet0	192.168.56.1/24	fdfb:56f6:3a0d:0:c7f3:9eb5:7…	☑ Enable

☑ Enable Server

Server Address: 192.168.56.100

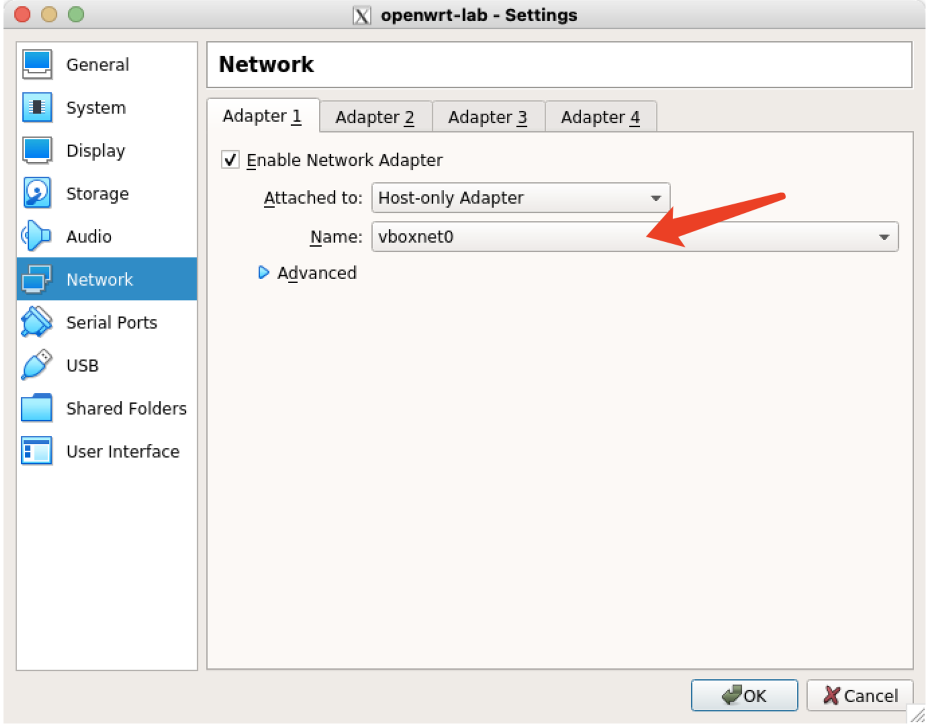
Server Mask: 255.255.255.0

Lower Address Bound: 192.168.56.100

Upper Address Bound: 192.168.56.254

Step 2: Add an OpenWrt-based router for our LAN

- Import the OpenWrt OVA file from: https://drive.google.com/file/d/1V6xFiMGwSi9N5Y1VH9ZMmXCQeYpxN5bs/view?usp=sharing
- After importing, choose the virtual adapter we just created as Adapter 1. Attach it to “Host-only Adapter”



Network

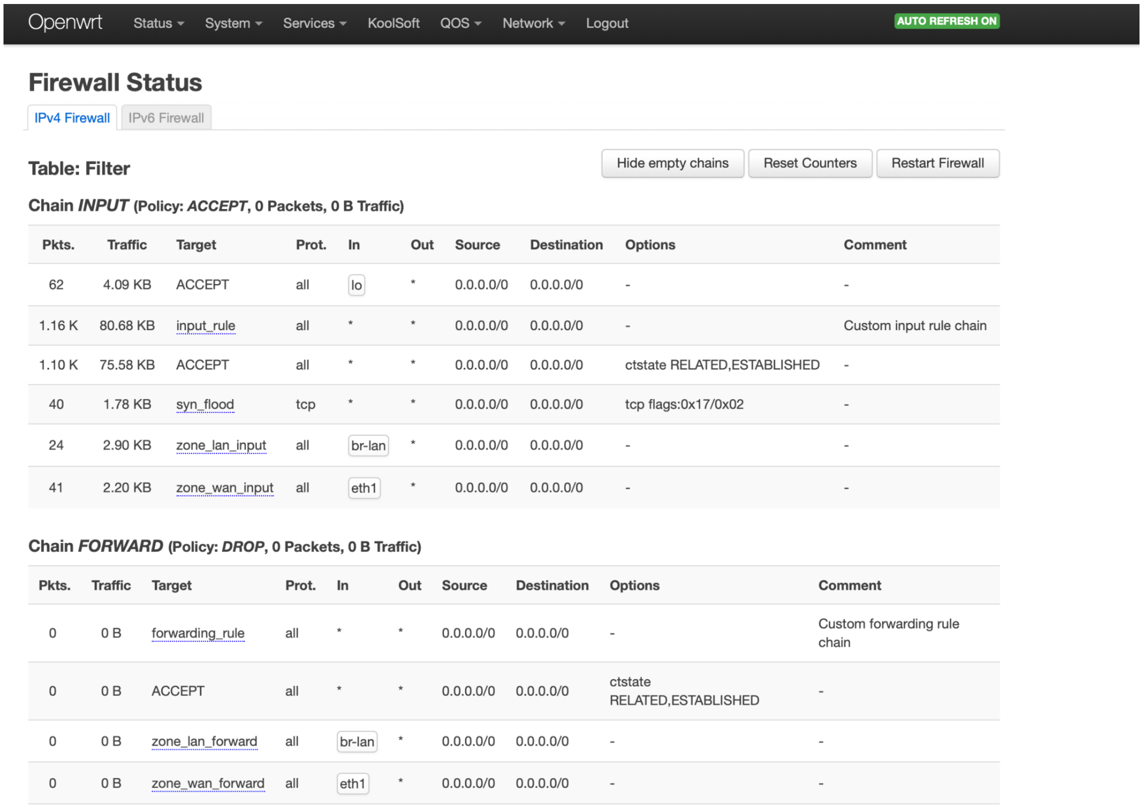
☑ Enable Network Adapter

Attached to: Host-only Adapter

Name: vboxnet0

Advanced

- **Test 1:** start the VM and test if it works as expected: visit http://192.168.56.10 on the browser of your host machine. (password: ut@Hut@H). If everything works fine, you should be able to see something like below



Firewall Status

Table: Filter

Chain *INPUT* (Policy: *ACCEPT*, 0 Packets, 0 B Traffic)

Pkts.	Traffic	Target	Prot.	In	Out	Source	Destination	Options	Comment
62	4.09 KB	ACCEPT	all	lo	*	0.0.0.0/0	0.0.0.0/0	-	-
1.16 K	80.68 KB	input_rule	all	*	*	0.0.0.0/0	0.0.0.0/0	-	Custom input rule chain
1.10 K	75.58 KB	ACCEPT	all	*	*	0.0.0.0/0	0.0.0.0/0	ctstate RELATED,ESTABLISHED	-
40	1.78 KB	syn_flood	tcp	*	*	0.0.0.0/0	0.0.0.0/0	tcp flags:0x17/0x02	-
24	2.90 KB	zone_lan_input	all	br-lan	*	0.0.0.0/0	0.0.0.0/0	-	-
41	2.20 KB	zone_wan_input	all	eth1	*	0.0.0.0/0	0.0.0.0/0	-	-

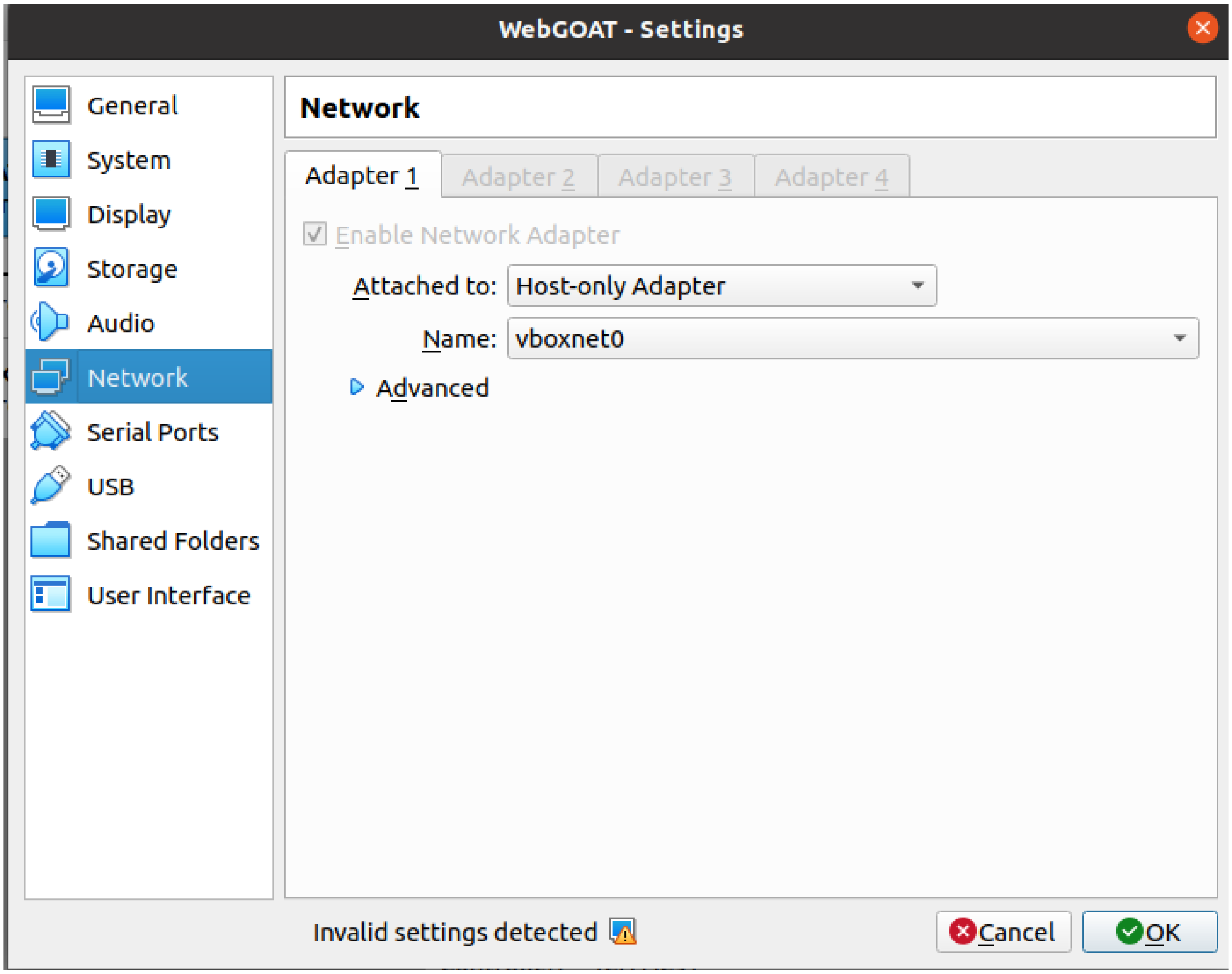
Chain *FORWARD* (Policy: *DROP*, 0 Packets, 0 B Traffic)

Pkts.	Traffic	Target	Prot.	In	Out	Source	Destination	Options	Comment
0	0 B	forwarding_rule	all	*	*	0.0.0.0/0	0.0.0.0/0	-	Custom forwarding rule chain
0	0 B	ACCEPT	all	*	*	0.0.0.0/0	0.0.0.0/0	ctstate RELATED,ESTABLISHED	-
0	0 B	zone_lan_forward	all	br-lan	*	0.0.0.0/0	0.0.0.0/0	-	-
0	0 B	zone_wan_forward	all	eth1	*	0.0.0.0/0	0.0.0.0/0	-	-

Step 3: Add the WebGoat web server to the LAN

Note: link to download WebGoat: https://download.vulnhub.com/webgoat/WebGOAT.ova

- Go to “Settings” → “Network”, change “Adapter 1” to attach to the network adapter we just created



Network

☑ Enable Network Adapter

Attached to: Host-only Adapter

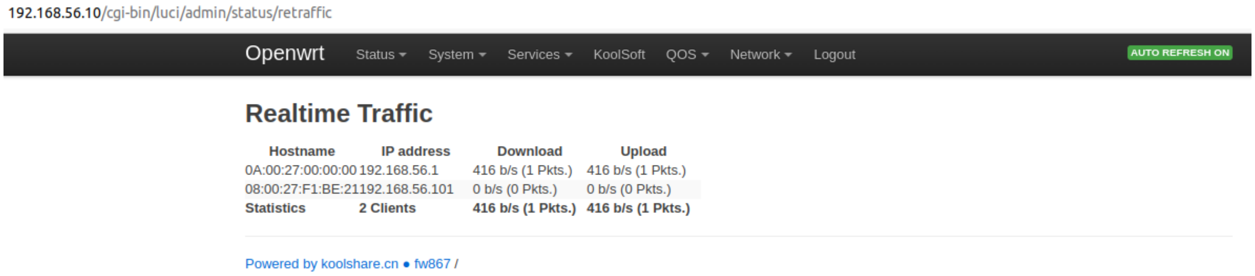
Name: vboxnet0

Invalid settings detected

- Start the Web Server instance, and set the default routing rule as follows so as to use the openwrt as the router in this LAN (**NOTE: This must be done every time you restart the Webserver VM**):

```
sudo ip route add default via 192.168.56.10
```

- **Test 2**:
  - Start the VM for the OpenWrt router;
  - Visit “192.168.56.10” in the web browser of your host machine (trust me, host machine is OK);
  - Log into the router portal using password “ut@Hut@H”, go to “Status” → “Realtime Traffic“, the webpage should show something like following:



Realtime Traffic

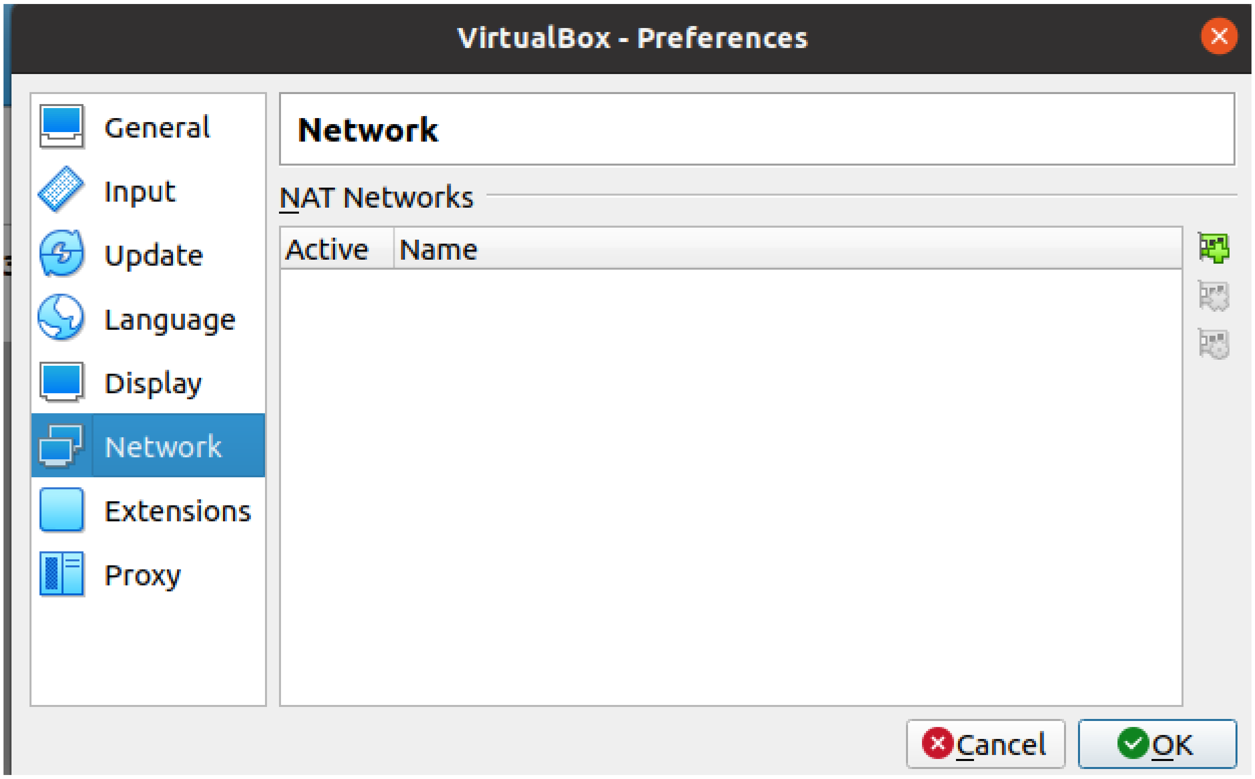
Hostname	IP address	Download	Upload
0A:00:27:00:00:00	192.168.56.1	416 b/s (1 Pkts.)	416 b/s (1 Pkts.)
08:00:27:F1:BE:21	192.168.56.101	0 b/s (0 Pkts.)	0 b/s (0 Pkts.)
Statistics	**2 Clients**	**416 b/s (1 Pkts.)**	**416 b/s (1 Pkts.)**

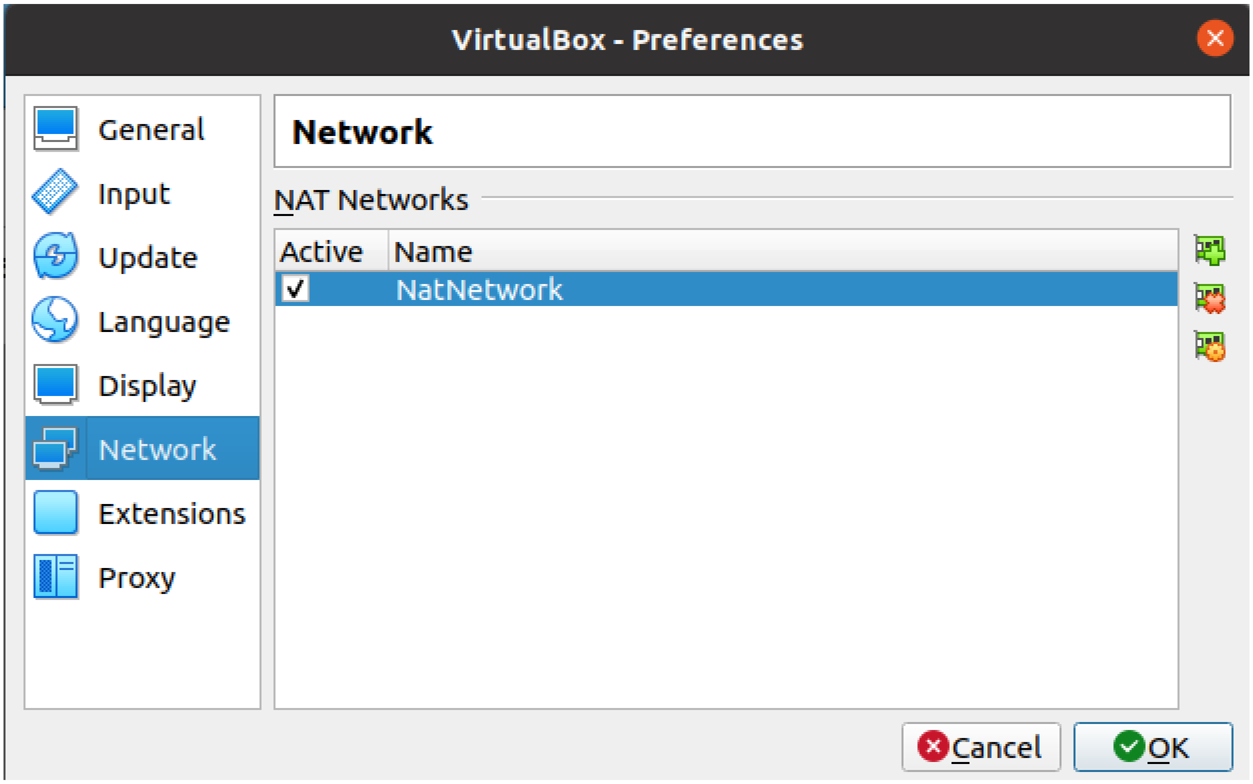
Powered by koolshare.cn ● fw867 /

Part B: Configuring a Wide Area Network (WAN) to simulate the Internet

Step 1: Adding a new virtual network adaptor [which will be used to support a nat-network WAN]

- **Go to “File” → “Preferences” → “Network” → “+”**





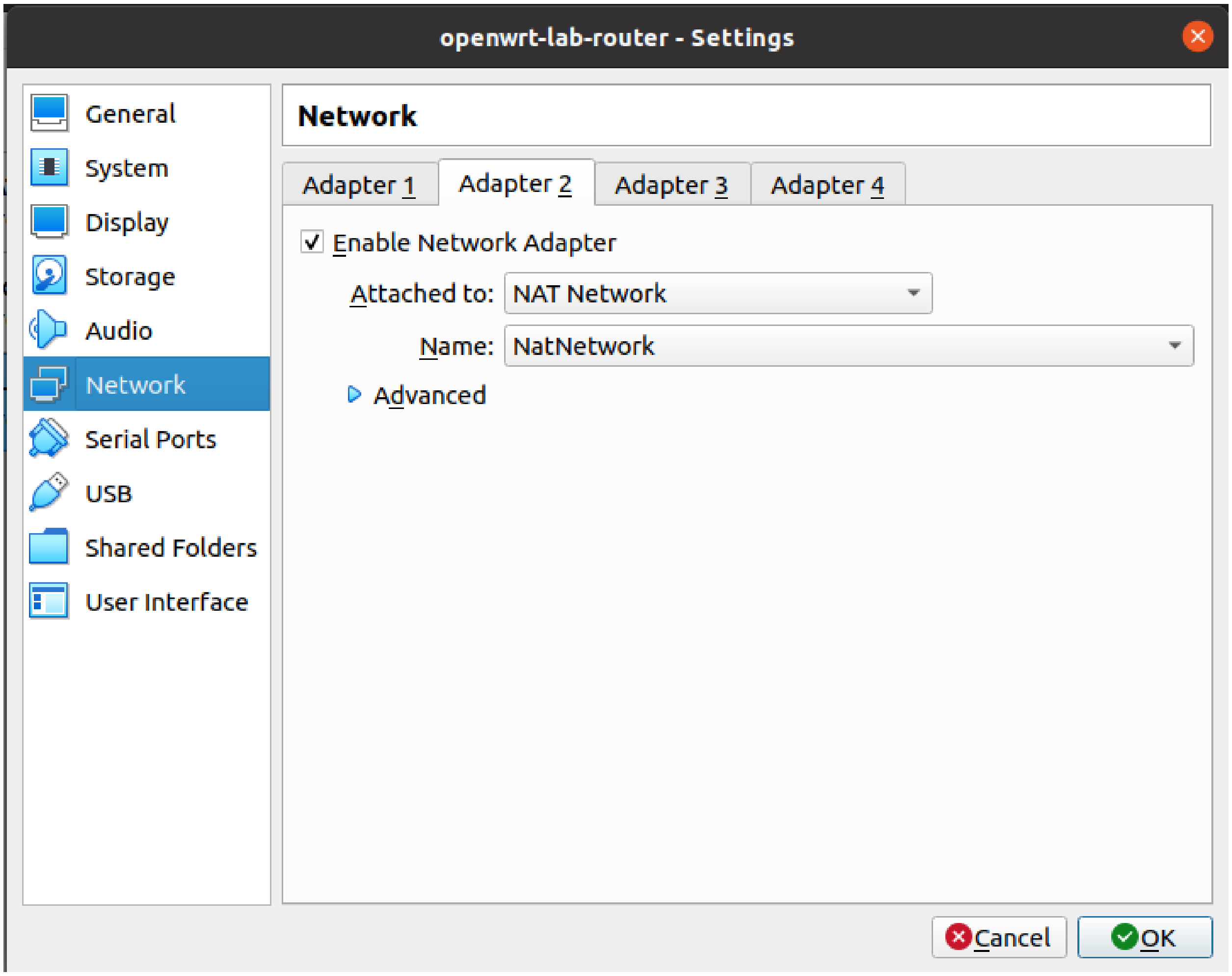
Network

NAT Networks

Active	Name
☑	NatNetwork

Step 2: Connect our OpenWrt-based router to the WAN

- Select your OpenWrt VM in VirtualBox→ “Settings” → “Network” → “Adapter 2”, configure it as follows



openwrt-lab-router - Settings

Network

☑ Enable Network Adapter

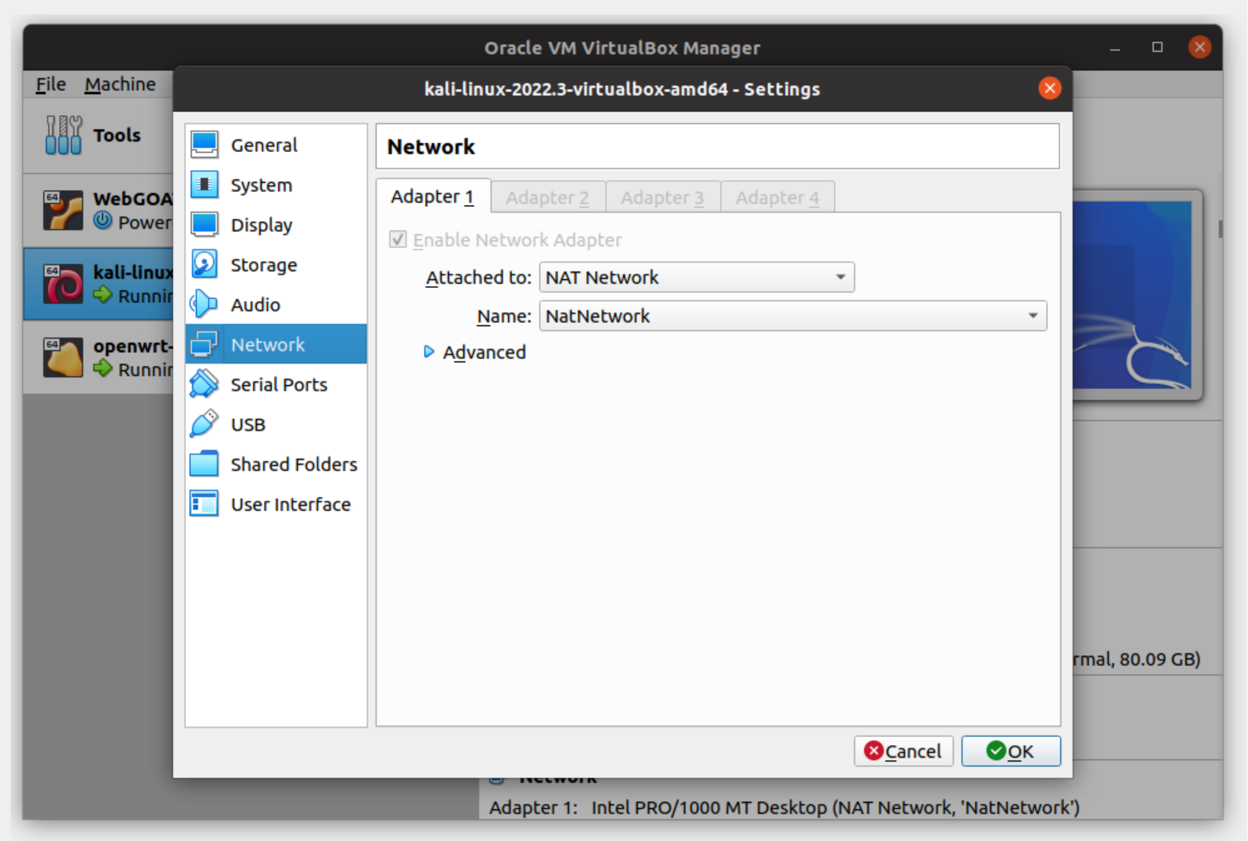
Attached to: NAT Network

Name: NatNetwork

- Test 3: start the Router VM → Type "Enter" in the terminal → Run "ip -a" to check if the VM has an IP address "10.0.2.4" (you can do "shift + fn + ↑" if you need to scroll up the terminal)

Step 3: adding another machine to the WAN

- Downloading the VM image: https://kali.download/virtual-images/kali-2022.3/kali-linux-2022.3-virtualbox-amd64.7z
- Unpacking the image and import "kali-linux-2022.3-virtualbox-amd64.vdi" to VirtualBox
- Select the newly imported VM in VirtualBox → "Settings" → "Network", and configure "Adapter 1" as follows:



Network

Adapter 1

☑ Enable Network Adapter

Attached to: NAT Network

Name: NatNetwork

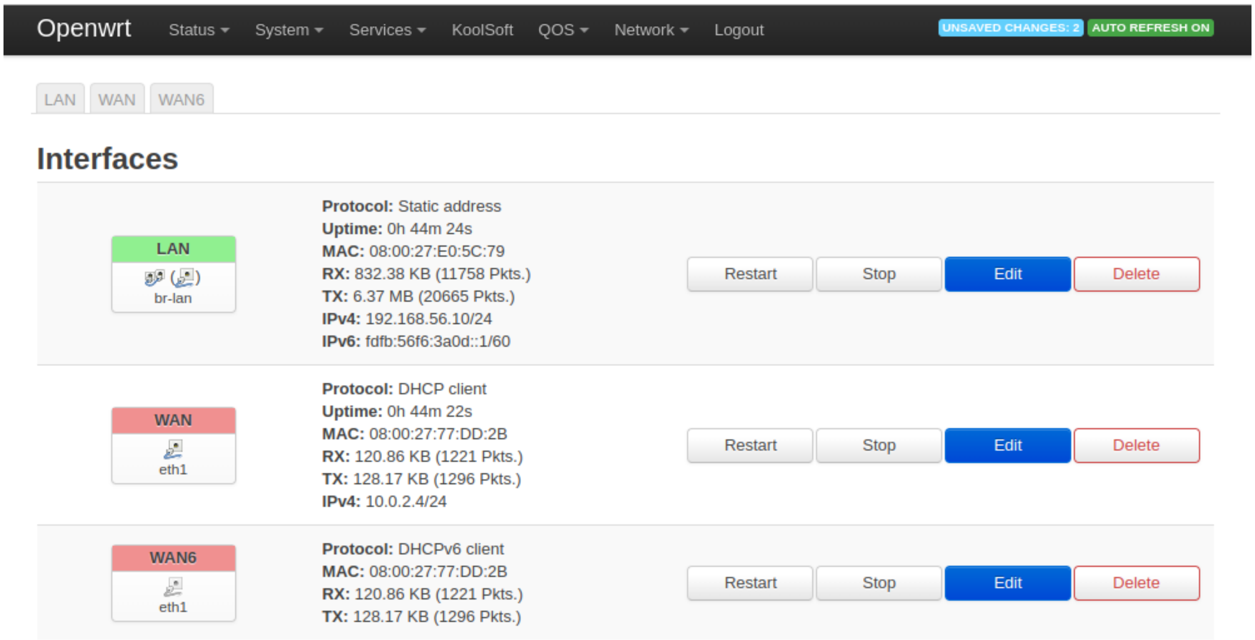
Adapter 1: Intel PRO/1000 MT Desktop (NAT Network, 'NatNetwork')

- **Test 4:** launch the Kali VM and log into it with username “kali” and password “kali”; Open terminal and type command “ip -a”; Check if you get an IP address like “10.0.2.5”

Part C: configuring port forwarding for the webserver on the OpenWrt router, so that you can access the webserver from the WAN

Step 1: check the zoning of LAN and WAN in the router

- Start the VM for the OpenWrt router
- Visit “192.168.56.10” in the web browser of your host machine (trust me, host machine is OK)
- Log into the router portal using password “ut@Hut@H”, go to “Network” → “Interfaces“, the webpage should show something like following:



Interfaces

Protocol: Static address
Uptime: 0h 44m 24s
MAC: 08:00:27:E0:5C:79
RX: 832.38 KB (11758 Pkts.)
TX: 6.37 MB (20665 Pkts.)
IPv4: 192.168.56.10/24
IPv6: fdfb:56f6:3a0d::1/60

Protocol: DHCP client
Uptime: 0h 44m 22s
MAC: 08:00:27:77:DD:2B
RX: 120.86 KB (1221 Pkts.)
TX: 128.17 KB (1296 Pkts.)
IPv4: 10.0.2.4/24

Protocol: DHCPv6 client
MAC: 08:00:27:77:DD:2B
RX: 120.86 KB (1221 Pkts.)
TX: 128.17 KB (1296 Pkts.)

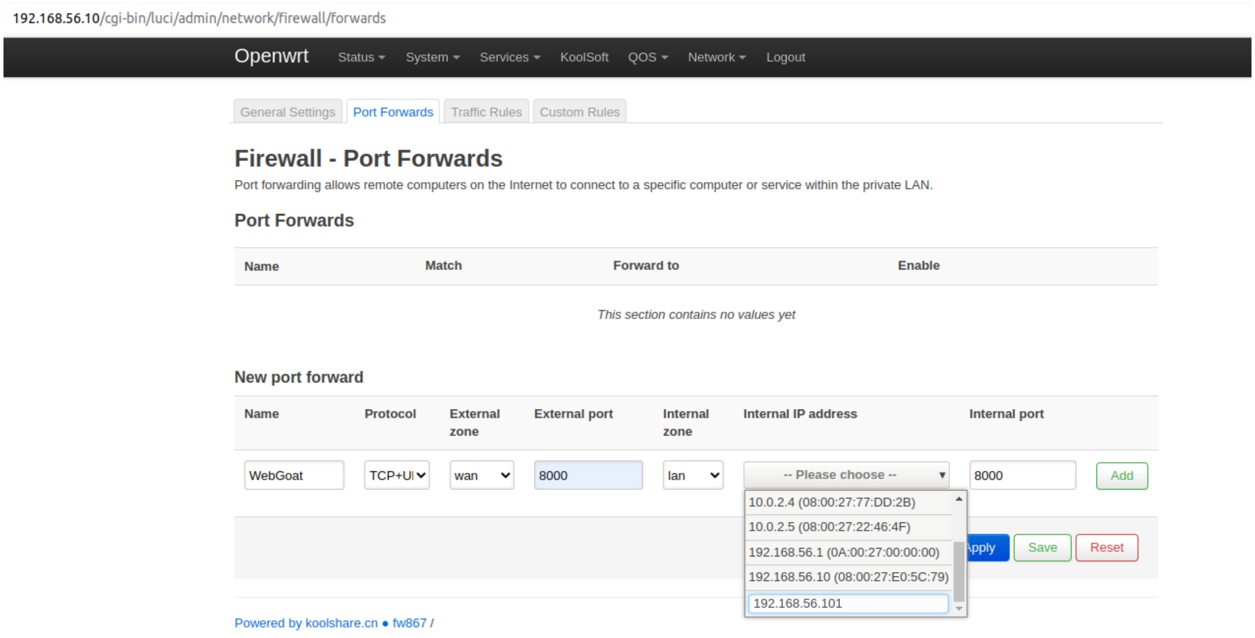
Step 2: Start the Web Server VM in VirtualBox

- In the terminal, run (this command needs to be run every time you restart the web server VM)

```
sudo ip route add default via 192.168.56.10
```

Step 3: Configure HTTP port forwarding in the Router VM

- Visit "192.168.56.10" in the web browser of your host machine (trust me, host machine is OK)
- Log into the router portal using password "ut@Hut@H", go to "Network" → "Firewall" → " Port Forwards", add a rule like the following:



Firewall - Port Forwards

Port forwarding allows remote computers on the Internet to connect to a specific computer or service within the private LAN.

Port Forwards

Name	Match	Forward to	Enable
This section contains no values yet			

New port forward

Name	Protocol	External zone	External port	Internal zone	Internal IP address	Internal port
WebGoat	TCP+UI	wan	8000	lan	-- Please choose --	8000

10.0.2.4 (08:00:27:77:DD:2B)
10.0.2.5 (08:00:27:22:46:4F)
192.168.56.1 (0A:00:27:00:00:00)
192.168.56.10 (08:00:27:E0:5C:79)
192.168.56.101

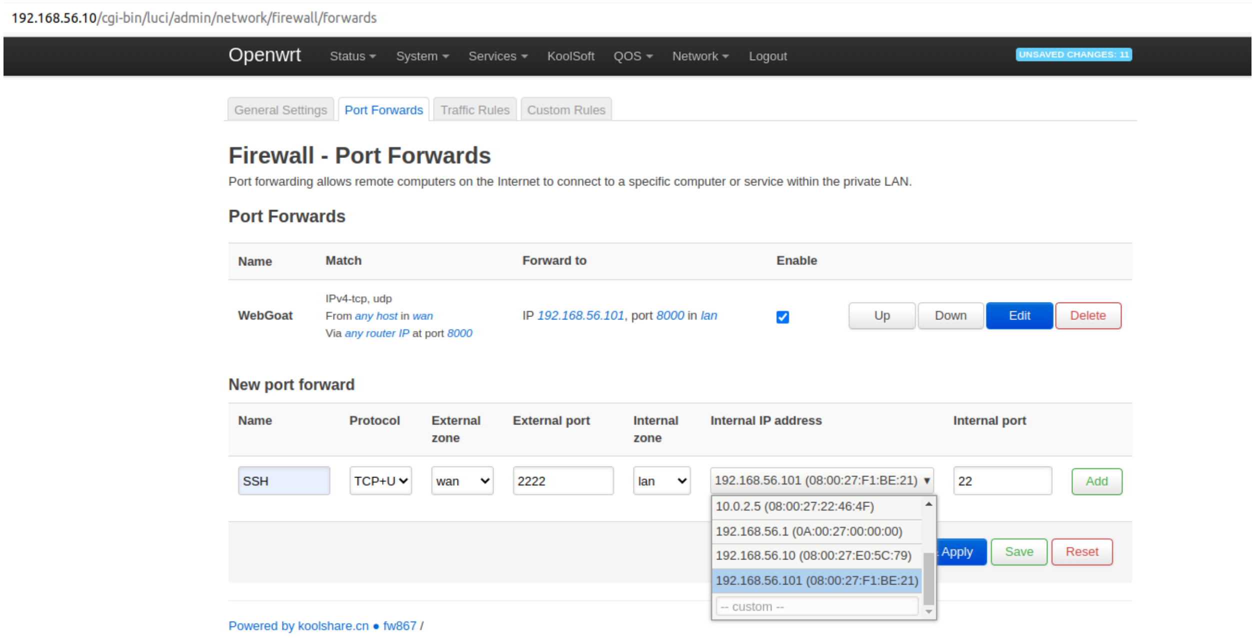
Make sure you type the correct IP address of your web server for the “Internal IP address”. It may not show up as an option, but you can type as you would like.

After you set up the parameters as above, click “Add” and then click “Save and Apply”. Both clicks must be done to make sure the forwarding rule can work!!!

- **Test 5:** go to the kali VM, open the web browser to visit: 10.0.2.4:8000/WebGoat (replace “10.0.2.4” with the “10.0.2.*” style IP of your Router VM if it is not 10.0.2.4); See if you can see the home page of the webserver running inside the LAN

Step 4: Configure SSH port forwarding in the Router VM

- Repeating Step 3 with the following configurations (do not forget to “Add” and “Save & Apply”)



192.168.56.10/cgi-bin/luci/admin/network/firewall/forwards

Firewall - Port Forwards

Port forwarding allows remote computers on the Internet to connect to a specific computer or service within the private LAN.

Port Forwards

Name	Match	Forward to	Enable
WebGoat	IPv4-tcp, udp From *any host* in *wan* Via *any router IP* at port *8000*	IP *192.168.56.101*, port *8000* in *lan*	☑

New port forward

Name	Protocol	External zone	External port	Internal zone	Internal IP address	Internal port
SSH	TCP+U	wan	2222	lan	192.168.56.101 (08:00:27:F1:BE:21)	22

Internal IP address options:

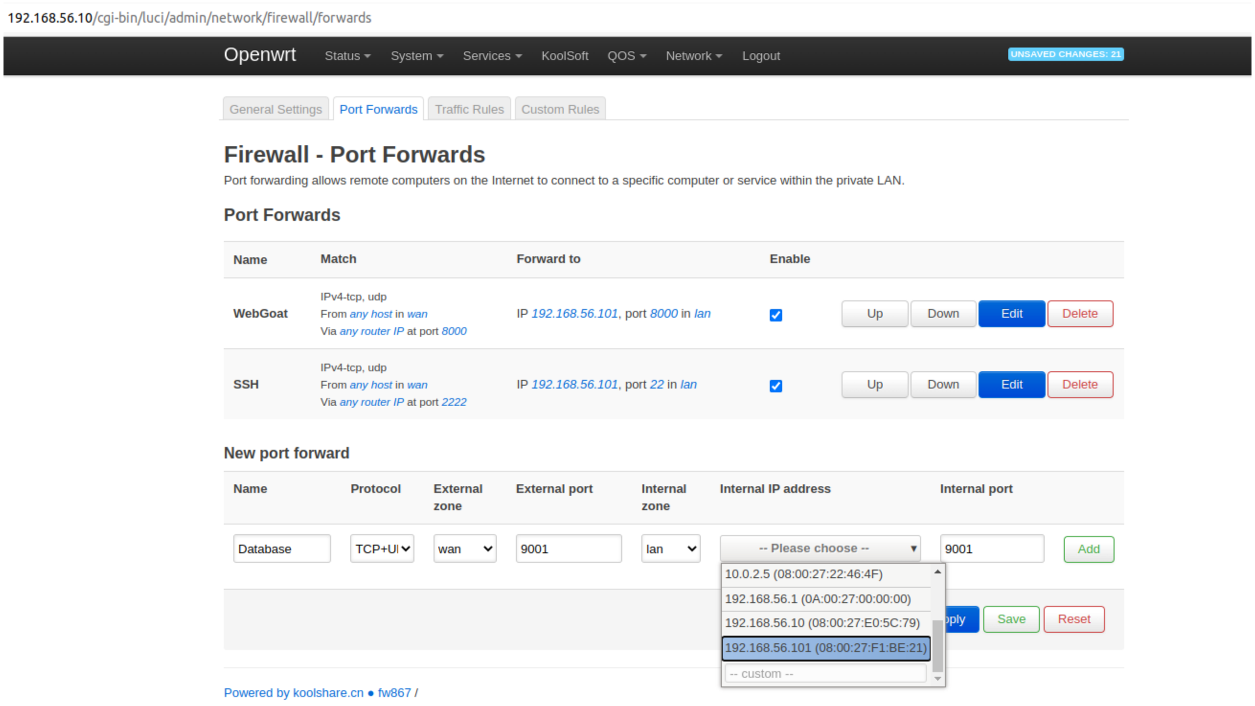
- 10.0.2.5 (08:00:27:22:46:4F)
- 192.168.56.1 (0A:00:27:00:00:00)
- 192.168.56.10 (08:00:27:E0:5C:79)
- 192.168.56.101 (08:00:27:F1:BE:21)
- -- custom --

Powered by koolshare.cn ● fw867 /

- **Test 6:** go to the kali VM, open the terminal and type “ssh webgoat@10.0.2.4 -p 2222” (replace “10.0.2.4” with the “10.0.2.*” style IP of your Router VM if it is not 10.0.2.4); See if you can connect to SSH in the web server

Step 4: Configure Database port forwarding in the Router VM

- Repeating Step 3 with the following configurations (do not forget to “Add” and “Save & Apply”)



192.168.56.10/cgi-bin/luci/admin/network/firewall/forwards

Firewall - Port Forwards

Port forwarding allows remote computers on the Internet to connect to a specific computer or service within the private LAN.

Port Forwards

Name	Match	Forward to	Enable	
WebGoat	IPv4-tcp, udp From *any host* in *wan* Via *any router IP* at port *8000*	IP *192.168.56.101*, port *8000* in *lan*	☑	Up Down Edit Delete
SSH	IPv4-tcp, udp From *any host* in *wan* Via *any router IP* at port *2222*	IP *192.168.56.101*, port *22* in *lan*	☑	Up Down Edit Delete

New port forward

Name	Protocol	External zone	External port	Internal zone	Internal IP address	Internal port	
Database	TCP+UI	wan	9001	lan	-- Please choose --	9001	Add

10.0.2.5 (08:00:27:22:46:4F)
192.168.56.1 (0A:00:27:00:00:00)
192.168.56.10 (08:00:27:E0:5C:79)
192.168.56.101 (08:00:27:F1:BE:21)
-- custom --

Powered by koolshare.cn ● fw867 /