



Universiteit Antwerpen
| Faculteit Toegepaste
Ingenieurswetenschappen

Lab of 3-Network Architecture

Ruben Nietvelt, Nabeel Nisar Bhat

2024-2025

Scheduled labs for PR01

Session	Date	Subject	Evaluation	Deadline (23:59)
1	01/10/2024	Introduction to the Linux Operating System	N/A	N/A
2	08/10/2024	Using the shell & exploring the filesystem	Report	14/10/2024
3	15/10/2024	Working with text files, managing running processes and writing shell scripts	Report	22/10/2024
4	23/10/2024	Learning system administration, getting & managing software	Report	28/10/2024
5	29/10/2024	Wireshark introduction	Report	05/11/2024
6	06/11/2024	Protocols in action: TCP and UDP	Report	11/11/2024
7	12/11/2024	Ethernet and ARP	Report	19/11/2024
8	20/11/2024	Setting up a DHCP server	Report	25/11/2024
9	26/11/2024	Setting up a DNS server	Report	03/12/2024
10	04/12/2024	Network Address Translation	Report	09/12/2024
11	10/12/2024	Remote Access & Firewalls (1)		N/A
12	18/12/2024	Remote Access & Firewalls (2)	Blackboard test	

Scheduled labs for PR02

Session	Date	Subject	Evaluation	Deadline (23:59)
1	02/10/2024	Introduction to the Linux Operating System	N/A	N/A
2	09/10/2024	Using the shell & exploring the filesystem	Report	15/10/2024
3	16/10/2024	Working with text files, managing running processes and writing shell scripts	Report	22/10/2024
4	23/10/2024	Learning system administration, getting & managing software	Report	29/10/2024
5	30/10/2024	Wireshark introduction	Report	05/11/2024
6	06/11/2024	Protocols in action: TCP and UDP	Report	12/11/2024
7	13/11/2024	Ethernet and ARP	Report	19/11/2024
8	20/11/2024	Setting up a DHCP server	Report	26/11/2024
9	27/11/2024	Setting up a DNS server	Report	03/12/2024
10	04/12/2024	Network Address Translation	Report	10/12/2024
11	11/12/2024	Remote Access & Firewalls (1)		N/A
12	18/12/2024	Remote Access & Firewalls (2)	Blackboard test	

Session 8

Setting up a DHCP Server

Introduction

Dynamic Host Configuration Protocol (DHCP) is a protocol that can dynamically assign network parameters to devices within its network.

Which parameters?

- IP address
- subnet mask
- default gateway ..

The benefit lies in efficiency and reduction of manual work.

Conceptually, your ISP (Telenet, Proximus,...) and the university's network all deploy DHCP servers to automatically assign your devices the aforementioned parameters.

When all goes well, the assignment follows the "DORA" process:



Objective

Create the following network diagram using **3 VMs**.

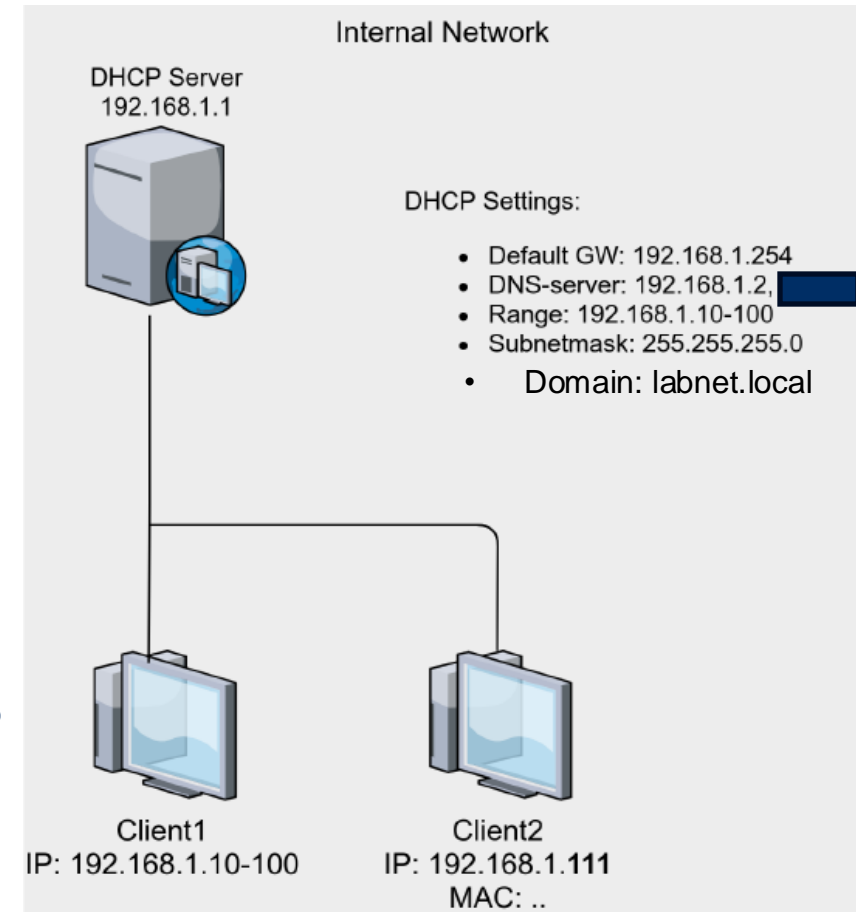
- Set up your own DHCP server using **kea-dhcp4-server**
 - Install kea: **apt-get install kea-dhcp4-server**
- Configure the following:
 - **IP Address range**, fixed IP address for client2 (by MAC address)
 - **subnet** mask
 - default **gateway**
 - **DNS** server
 - **Domain**
 - **Log level: debug**

Test the working of your DHCP service by confirming the expected network parameter assignments on your client VMs.

At the same time, examine the DHCP service **status** and **logs** to confirm the correct **process** has been followed.

```
systemctl status kea-dhcp4-server
```

Reflect on the general 4 (4. excluded) steps of setting up a server in your reports.



Introduction to networking with VMs

- Types of interfaces
 - Bridged: Host and VM on same network
 - Internal network: Network separated from host.
 - NAT: Communicate through host.
 - NAT network: Create separated network, while being to communicate through host.

IP address

The **ip** command. An example output of
`$ ip -c a` (equal to `$ ip -c address`)

The internal network adapter
name

MAC address

IP address

```
student@dhcpserver:~$ ip -c a
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
   link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
   inet 127.0.0.1/8 scope host lo
       valid_lft forever preferred_lft forever
   inet6 ::1/128 scope host
       valid_lft forever preferred_lft forever
2: enp0s3: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default qlen 1000
   link/ether 08:00:27:92:b8:f5 brd ff:ff:ff:ff:ff:ff
   inet 10.0.2.15/24 metric 100 brd 10.0.2.255 scope global dynamic enp0s3
       valid_lft 68448sec preferred_lft 68448sec
   inet6 fe80::a00:27ff:fe92:b8f5/64 scope link
       valid_lft forever preferred_lft forever
3: enp0s8: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default qlen 1000
   link/ether 08:00:27:86:d8:9b brd ff:ff:ff:ff:ff:ff
   inet 192.168.1.1/24 brd 192.168.1.255 scope global enp0s8
       valid_lft forever preferred_lft forever
   inet6 fe80::a00:27ff:fe86:d89b/64 scope link
       valid_lft forever preferred_lft forever
```


Network adapter address assignment

Assigning an IP address to a certain network adapter.

Temporarily: use the **ip** command.

```
dhcp-server@dhcp:~$ sudo ip a add 192.168.1.1/24 dev enp0s8
```

MODIFYING ADDRESS AND LINK PROPERTIES

SUBCOMMAND	DESCRIPTIONS AND TASKS
------------	------------------------

addr add	Add an address ip addr add 192.168.1.1/24 dev em1 Add address 192.168.1.1 with netmask 24 to device em1
addr del	Delete an address ip addr del 192.168.1.1/24 dev em1 Remove address 192.168.1.1/24 from device em1

From the “ip command cheat sheet for Red Hat enterprise Linux”

Network adapter address assignment

Assigning an IP address to a certain network adapter.

Permanently: configure the interfaces config files (`/etc/network/interfaces`)

```
dhcp-server@dhcp:~$ sudo nano /etc/network/interfaces
```

```
# This file describes the network interfaces available on your system
# and how to activate them. For more information, see interfaces(5).

source /etc/network/interfaces.d/*

# The loopback network interface
auto lo
iface lo inet loopback

# The primary network interface
allow-hotplug enp1s0
iface enp1s0 inet static
    address 192.168.1.1/24
    gateway 192.168.1.254
    # dns-* options are implemented by the resolvconf package, if installed
    dns-nameservers 192.168.1.2
    dns-search labnet.local
```

Setup of a DHCP Server

Command list overview – Virtual Machine

Command	Explanation
<code>sudo virt-install</code>	Installation of virtual machine. Extra parameters needed.
<code>sudo virsh list --all</code>	List all VMs.
<code>sudo virsh start <my_vm></code>	Start a VM. Replace <my_vm> with the name of your VM.
<code>sudo virsh shutdown <my_vm></code>	Shutdown a VM. Replace <my_vm> with the name of your VM.
<code>sudo virsh destroy <my_vm></code>	Forcefully shutdown a VM. Replace <my_vm> with the name of your VM.
<code>sudo virsh undefine <my_vm></code>	Delete a VM. Replace <my_vm> with the name of your VM.
<code>sudo virsh console <my_vm></code>	Connect to VM console. Replace <my_vm> with the name of your VM.
Qwerty: “Ctrl + J” Azerty: “Ctrl + \$”	Exit the console to go back to host.

Command list overview – Networks

Command	Explanation
<code>sudo virsh net-define <network.xml></code>	Network definition. Replace <network.xml> with the correct filename.
<code>sudo virsh net-start <name></code>	Start the network. Replace <name> with the network name defined in your config.
<code>sudo virsh net-autostart <name></code>	Automatic startup of a virtual network. Replace <name> with the name of your network name.
<code>sudo virsh domiflist <vm_name></code>	List all attached interfaces. Replace <vm_name> with your VM name.
<code>sudo virsh attach-interface --type network --source <name> --model virtio <vm_name> --persistent</code>	Attach a network interface to a VM. Replace <vm_name> with the name of the VM, and <name> with your network name.
<code>sudo virsh detach-interface <vm_name> network <mac_address></code>	Detach a network interface from your VM. Replace <vm_name> with the name of the VM, and <mac_address> with the mac address retrieved using domiflist.

Setup of a service

General steps for setting up a “server”:

1. Install the service
2. Configure the server
3. Start the server
4. Secure the server (*This step is reserved for future sessions*)
5. Monitor the server (service status, log files, ...)

1. Install the service

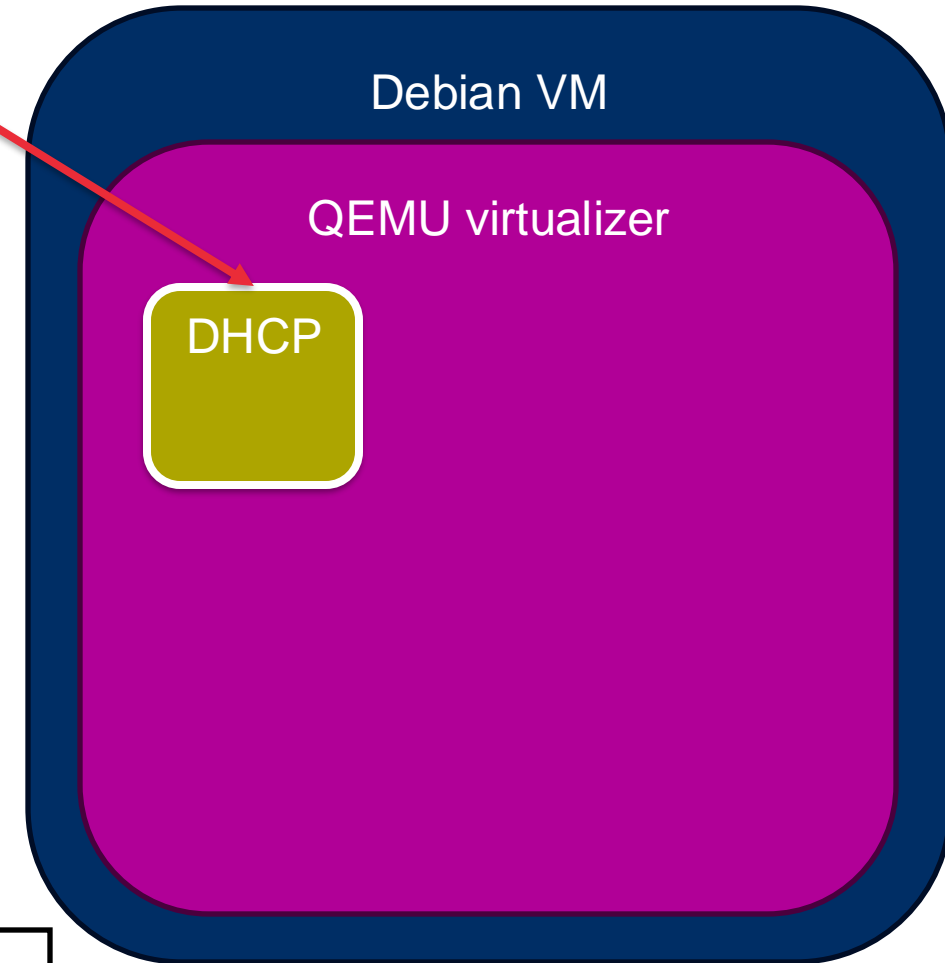


- Install a virtual machine
 - Go to `/var/lib/libvirt/images`
 - Download [Debian image](#) in this folder using **wget**
 - Configure a virtual hard drive

```
sudo qemu-img create -f qcow2 /var/lib/libvirt/images/dhcp.qcow2 8G
```

- Setup a NAT network: `nat-network.xml`

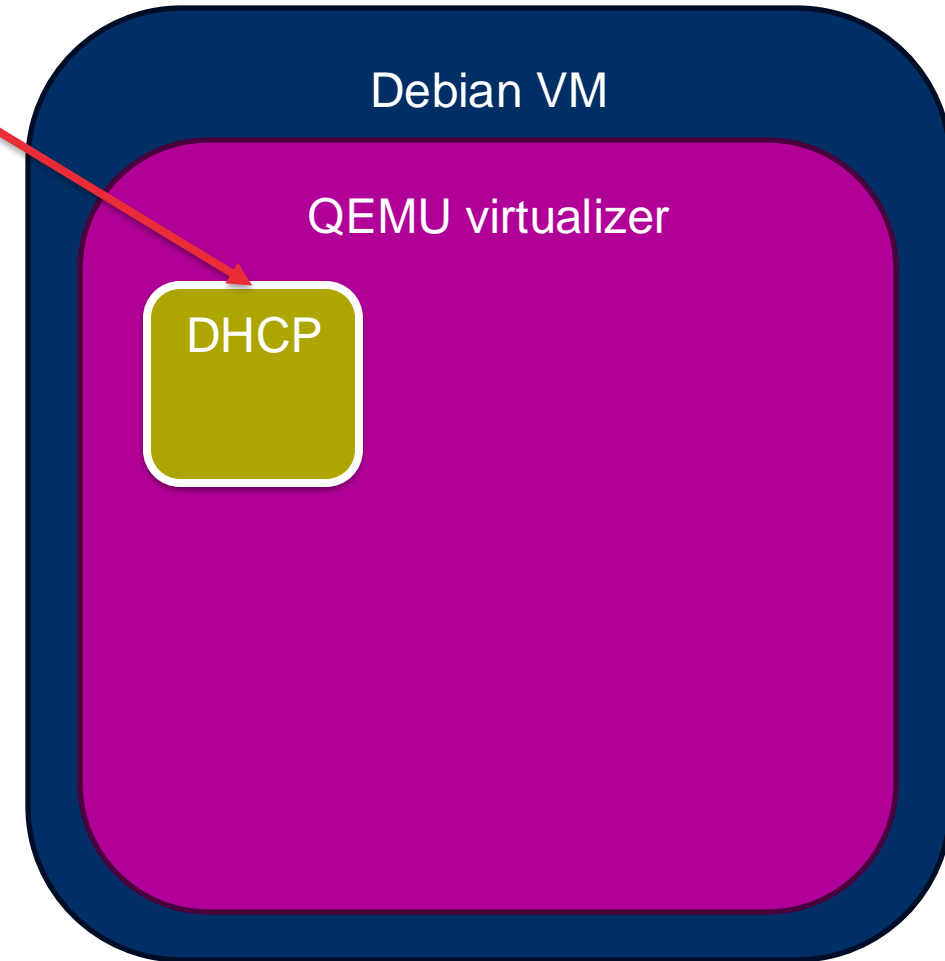
```
<network>
  <name>nat-network</name>
  <forward mode='nat'/>
  <bridge name='virbr2' stp='on' delay='0'/>
  <ip address='192.168.200.1' netmask='255.255.255.0'/>
</network>
```



= virtual machine

1. Install the service

- Setup a NAT network
 - Define network
 - Start network (and enable autostart)
- Install VM: follow guide on Blackboard



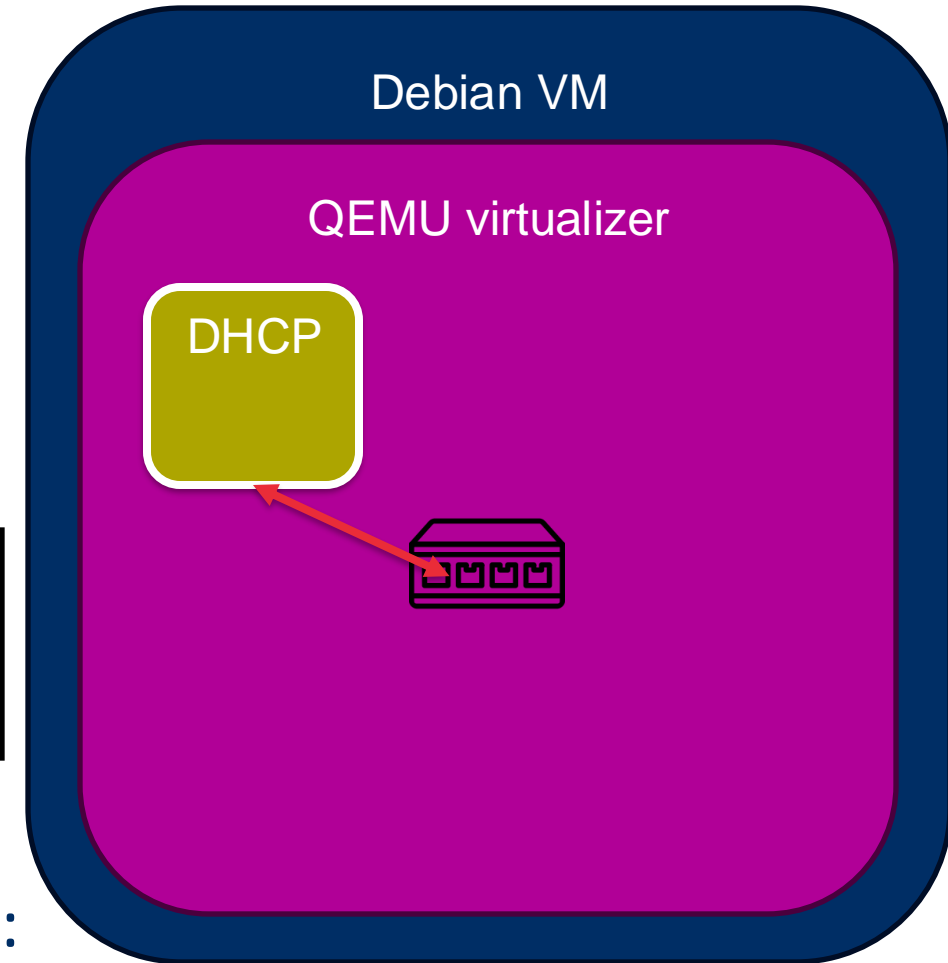
```
sudo virt-install --name dhcp --ram 1024 --vcpus 1 --disk  
path=/var/lib/libvirt/images/dhcp.qcow2,format=qcow2 --os-variant debian11 --network network=nat-network --  
graphics none --extra-args="console=ttyS0" --location /var/lib/libvirt/images/<downloaded_image>.iso
```


2. Configure the server

- Install the required packages
- Disconnect the NAT network from the DHCP
- Create an internal network:
internal-network.xml

```
<network>  
  <name>internal-network</name>  
  <bridge name='virbr3' stp='on' delay='0' />  
</network>
```

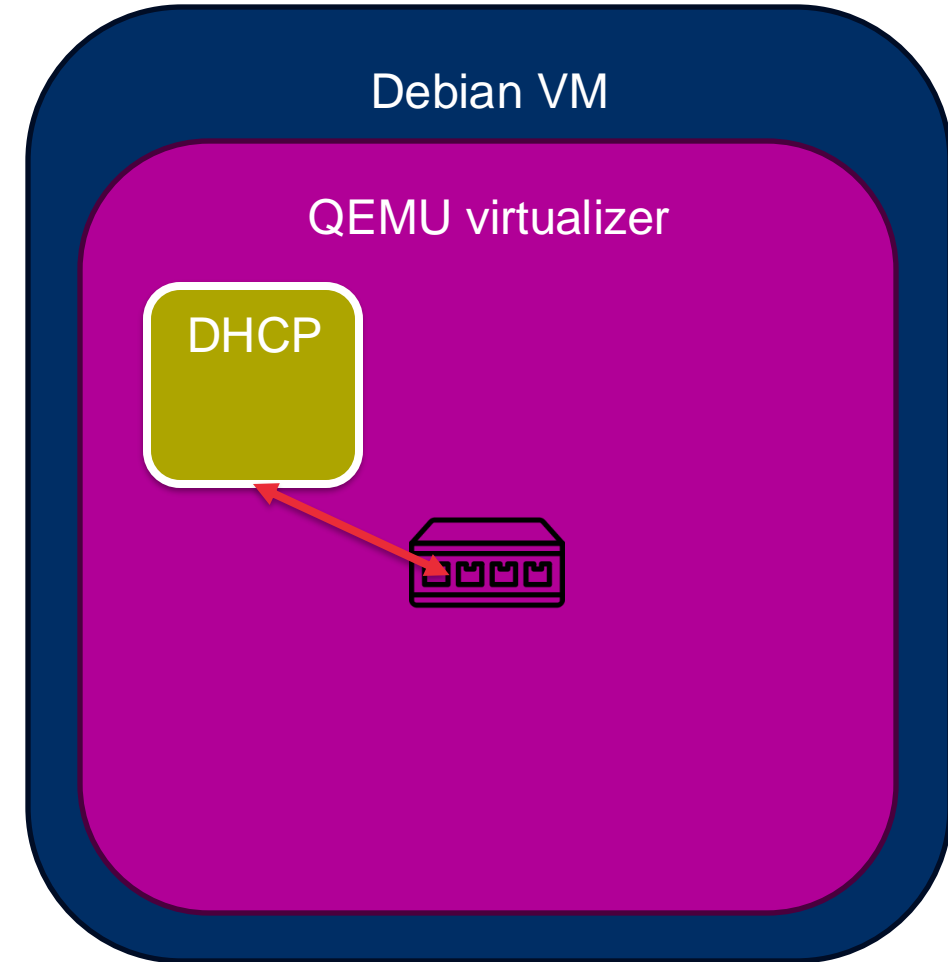
- Connect it to the DHCP
- Make a backup of the current configuration:
/etc/kea/kea-dhcp4.conf
- Create/edit a new configuration, adhering to
to the setup shown on slide 7



3. Start the server

- To (re)load the config, you should (re)start the service

```
systemctl restart kea-dhcp4-server
```



5. Monitor the server

- Create two extra VMs
- Connect both clients to the internal network
- Check logs of the server

```
systemctl status kea-dhcp4-server
```

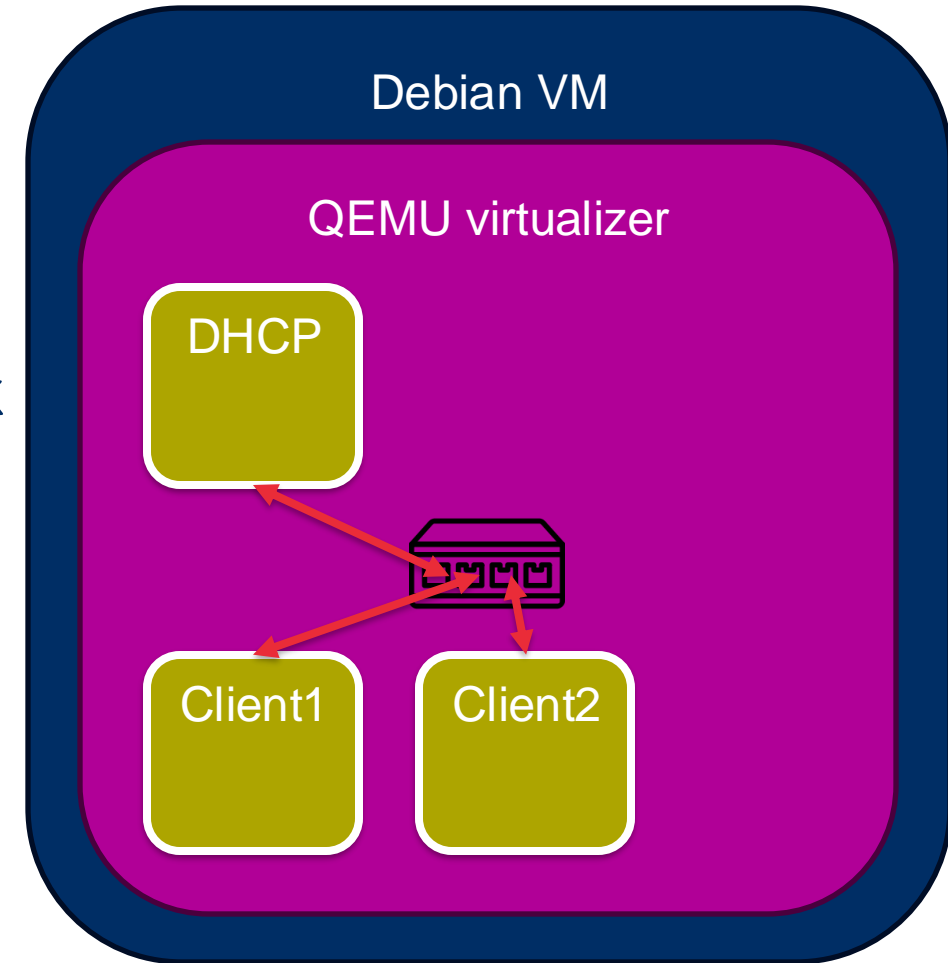
```
journalctl -u kea-dhcp4-server
```

- Check leases:

```
/var/lib/kea/kea-leases4.csv
```

- Renew DHCP lease on clients:

```
dhclient -r  
dhclient
```



Resources

- <https://ubuntu.com/server/docs/how-to-install-and-configure-isc-kea>
- <https://kea.readthedocs.io/en/kea-2.0.1/arm/config.html>

