DHCP AND VAGRANT PROJECT

INSTRUCTIONS AND DOCUMENTATION

DavidHlanz / **DHCP-Proyecto**

ÍNDICE

🧰 Requirements

1- VirtualBox

Vagrant uses VirtualBox as its default hypervisor.

2- Vagrant

Virtualized development environments

3- Git (optional)

Download the code with a command; you can also download it as a compressed file.

⚙️Tutorial to use

1- Clone the repository

```
git clone https://github.com/DavidHlanz/DHCP-Proyecto.git
```

2- Open the project folder

```
cd ~/Documentos/DHCP-Proyecto
```

3- Create a host-only network on 192.168.57.1/24

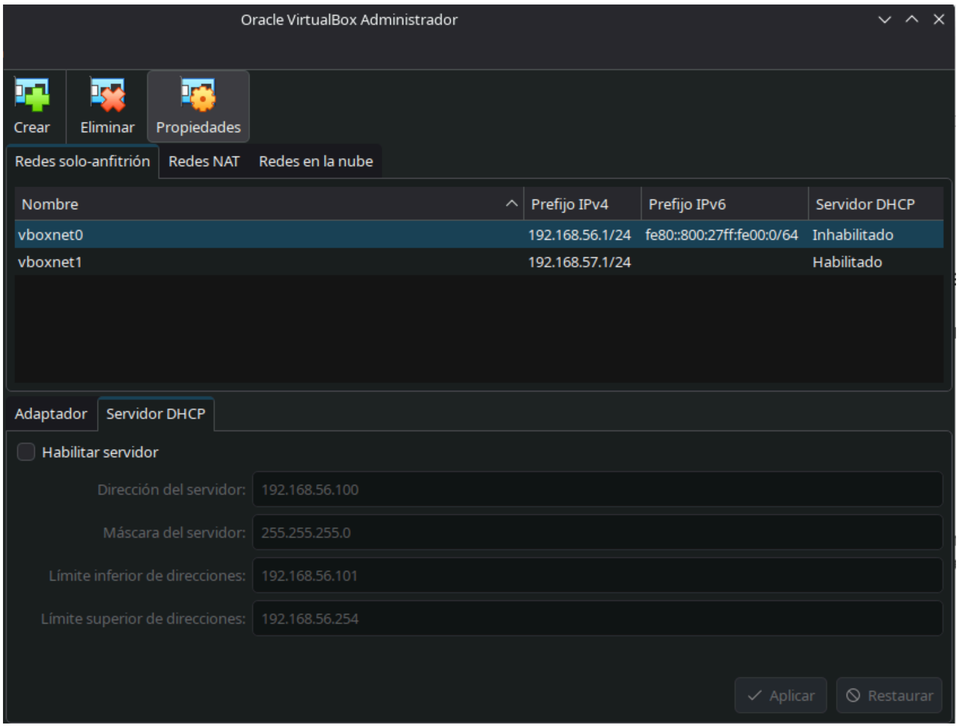
Open VirtualBox → Preferences → Network → Host-only networks.

Create a new network with the following settings:

IPv4: 192.168.57.1

Subnet mask: 255.255.255.0

DHCP server: Disabled



Nombre	Prefijo IPv4	Prefijo IPv6	Servidor DHCP
vboxnet0	192.168.56.1/24	fe80::800:27ff:fe00:0/64	Inhabilitado
vboxnet1	192.168.57.1/24		Habilitado

□ Habilitar servidor

Dirección del servidor: 192.168.56.100

Máscara del servidor: 255.255.255.0

Límite inferior de direcciones: 192.168.56.101

Límite superior de direcciones: 192.168.56.254

4- Raise the machines

```
vagrant up
```

Vagrant will download the box (this may take a while the first time), create the VMs, and apply your network configuration and provisioning automatically. If it does not, use: vagrant provision server.

⚒️Test commands

- How to see machines

```
vagrant status
```

- How to go inside server machine

```
vagrant ssh servidor
```

- How to go inside client machines

```
vagrant ssh c1 / vagrant ssh c2
```

- How to reload the server

```
vagrant reload servidor
```

- How to test network

```
ping 192.168.57.10

ip  -a

cat /etc/resolv.conf
```

- How to stop the machines

```
vagrant halt
```

- How to destroy the machines

```
vagrant destroy -f
```

Configuration

Exercise requirements:

Three machines: server (DHCP server), c1 (client requesting dynamic IP), c2 (client that must always receive a fixed IP according to its MAC).

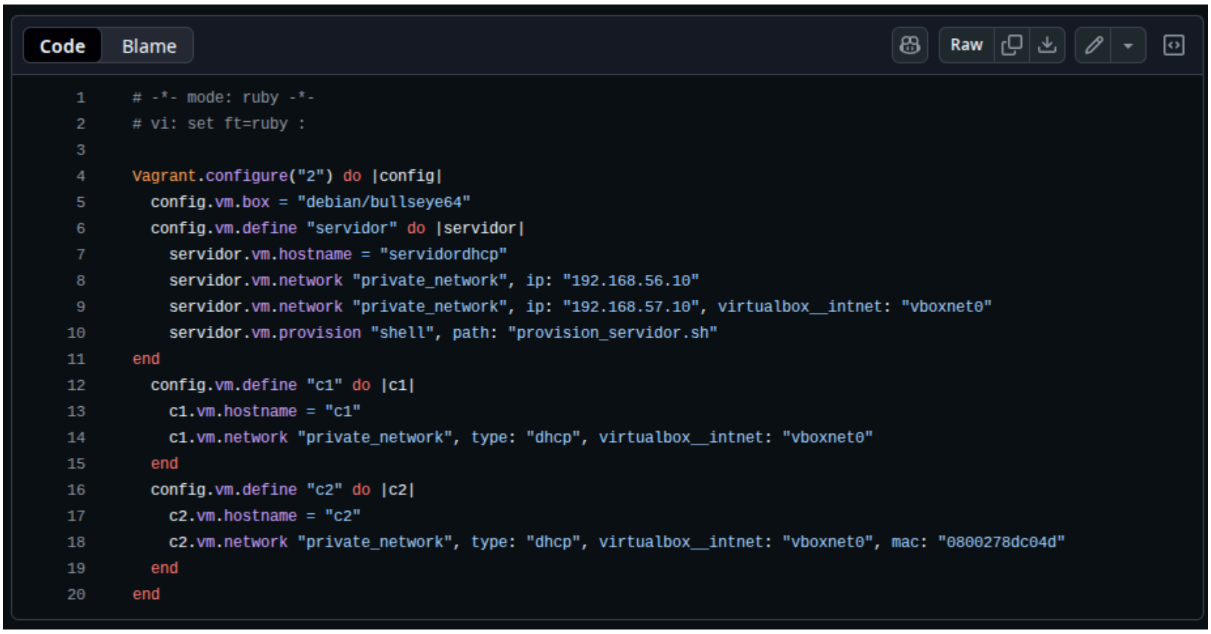
Internal network used for DHCP: 192.168.57.0/24. The server will have the IP 192.168.57.10 on that network.

Range that the server will assign: 192.168.57.25 → 192.168.57.50.

c2 must receive a fixed IP address of 192.168.57.4 according to its MAC address.

1- Vagrantfile configuration

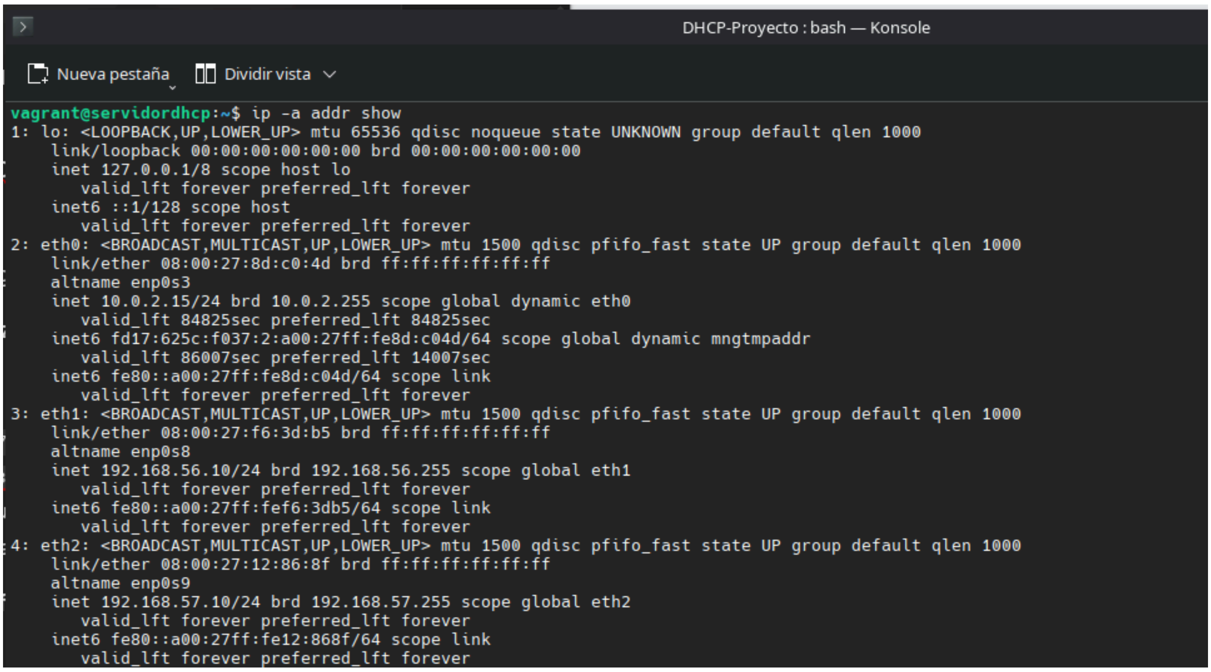
Here we can see the server and the two clients configured, and then proceed to launch them with “vagrant up client/server.”



```ruby
# -*- mode: ruby -*-
# vi: set ft=ruby :

Vagrant.configure("2") do |config|
  config.vm.box = "debian/bullseye64"
  config.vm.define "servidor" do |servidor|
    servidor.vm.hostname = "servidordhcp"
    servidor.vm.network "private_network", ip: "192.168.56.10"
    servidor.vm.network "private_network", ip: "192.168.57.10", virtualbox__intnet: "vboxnet0"
    servidor.vm.provision "shell", path: "provision_servidor.sh"
end
  config.vm.define "c1" do |c1|
    c1.vm.hostname = "c1"
    c1.vm.network "private_network", type: "dhcp", virtualbox__intnet: "vboxnet0"
  end
  config.vm.define "c2" do |c2|
    c2.vm.hostname = "c2"
    c2.vm.network "private_network", type: "dhcp", virtualbox__intnet: "vboxnet0", mac: "0800278dc04d"
  end
end
```

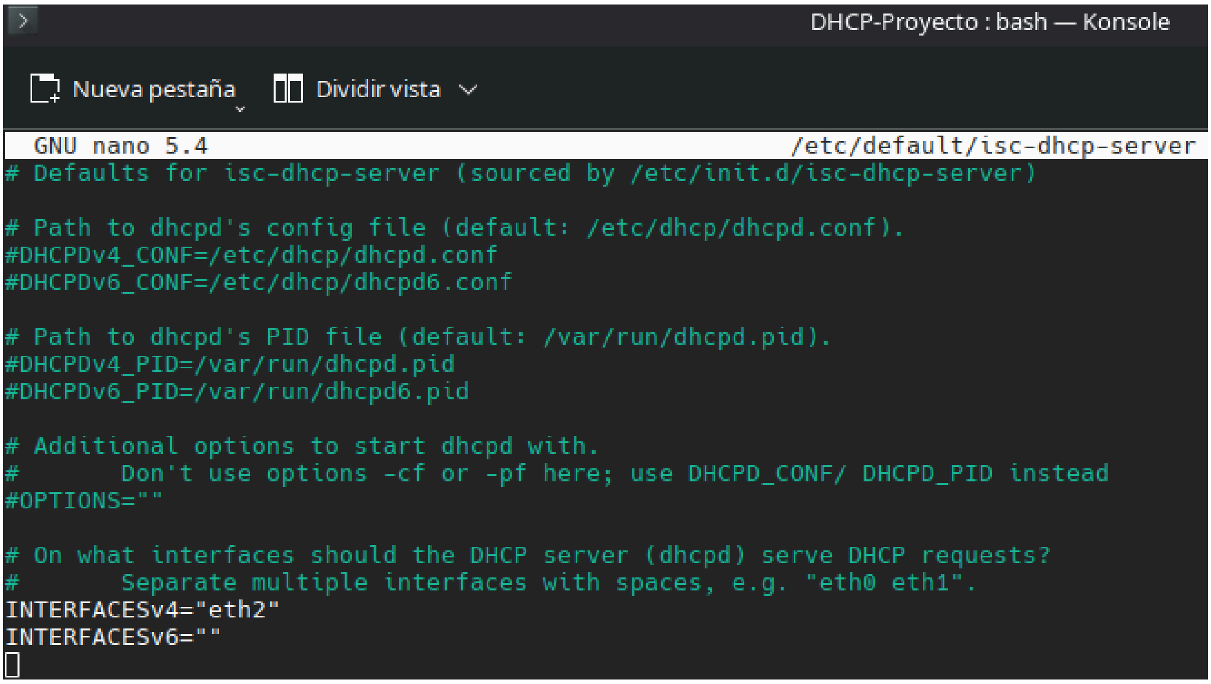
2- Locate the interface

Command → ip -a



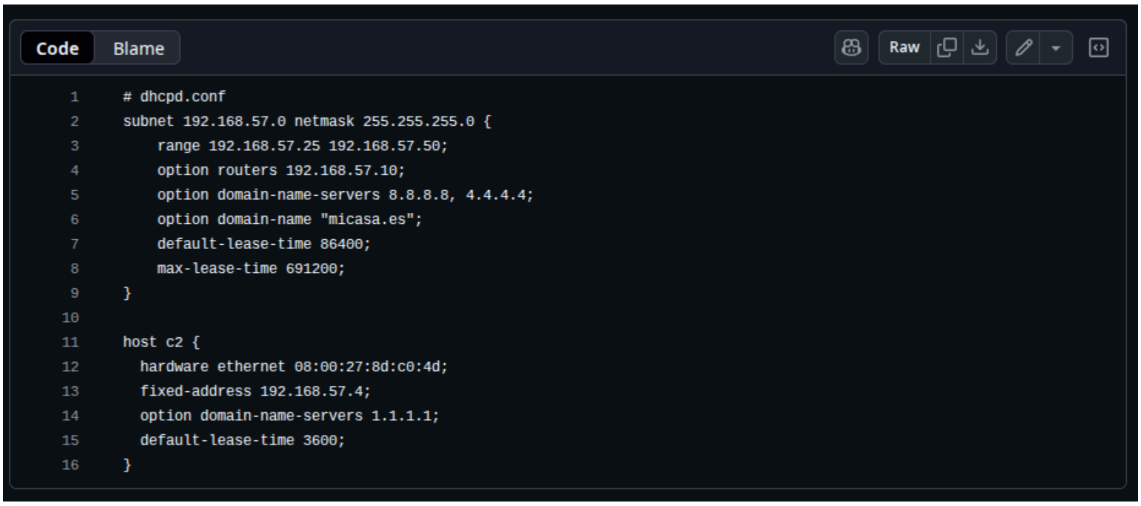
```
vagrant@servidordhcp:~$ ip -a addr show
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
       valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host
       valid_lft forever preferred_lft forever
2: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast state UP group default qlen 1000
    link/ether 08:00:27:8d:c0:4d brd ff:ff:ff:ff:ff:ff
    altname enp0s3
    inet 10.0.2.15/24 brd 10.0.2.255 scope global dynamic eth0
       valid_lft 84825sec preferred_lft 84825sec
    inet6 fd17:625c:f037:2:a00:27ff:fe8d:c04d/64 scope global dynamic mngtmpaddr
       valid_lft 86007sec preferred_lft 14007sec
    inet6 fe80::a00:27ff:fe8d:c04d/64 scope link
       valid_lft forever preferred_lft forever
3: eth1: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast state UP group default qlen 1000
    link/ether 08:00:27:f6:3d:b5 brd ff:ff:ff:ff:ff:ff
    altname enp0s8
    inet 192.168.56.10/24 brd 192.168.56.255 scope global eth1
       valid_lft forever preferred_lft forever
    inet6 fe80::a00:27ff:fef6:3db5/64 scope link
       valid_lft forever preferred_lft forever
4: eth2: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast state UP group default qlen 1000
    link/ether 08:00:27:12:86:8f brd ff:ff:ff:ff:ff:ff
    altname enp0s9
    inet 192.168.57.10/24 brd 192.168.57.255 scope global eth2
       valid_lft forever preferred_lft forever
    inet6 fe80::a00:27ff:fe12:868f/64 scope link
       valid_lft forever preferred_lft forever
```

Command → nano /etc/default/isc-dhcpd-server



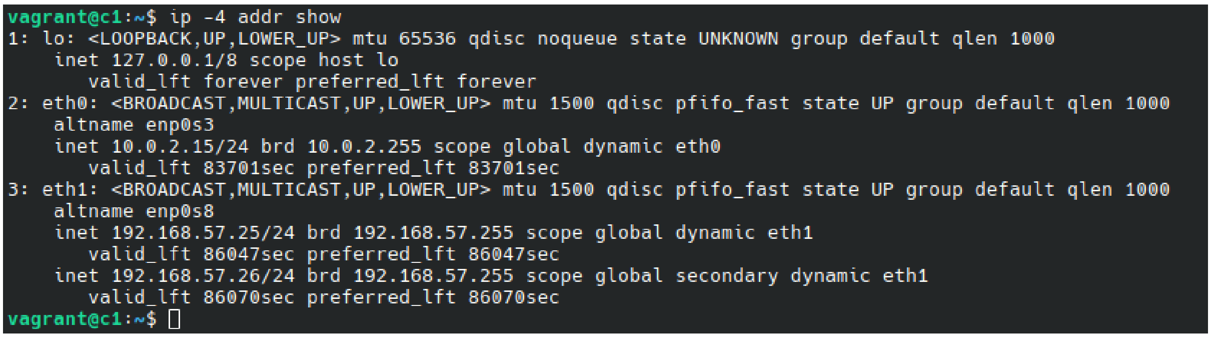
```
 GNU nano 5.4                                /etc/default/isc-dhcp-server
# Defaults for isc-dhcp-server (sourced by /etc/init.d/isc-dhcp-server)

# Path to dhcpd's config file (default: /etc/dhcp/dhcpd.conf).
#DHCPDv4_CONF=/etc/dhcp/dhcpd.conf
#DHCPDv6_CONF=/etc/dhcp/dhcpd6.conf

# Path to dhcpd's PID file (default: /var/run/dhcpd.pid).
#DHCPDv4_PID=/var/run/dhcpd.pid
#DHCPDv6_PID=/var/run/dhcpd6.pid

# Additional options to start dhcpd with.
#       Don't use options -cf or -pf here; use DHCPD_CONF/ DHCPD_PID instead
#OPTIONS=""

# On what interfaces should the DHCP server (dhcpd) serve DHCP requests?
#       Separate multiple interfaces with spaces, e.g. "eth0 eth1".
INTERFACESv4="eth2"
INTERFACESv6=""
```

3- dhcpd.conf configuration

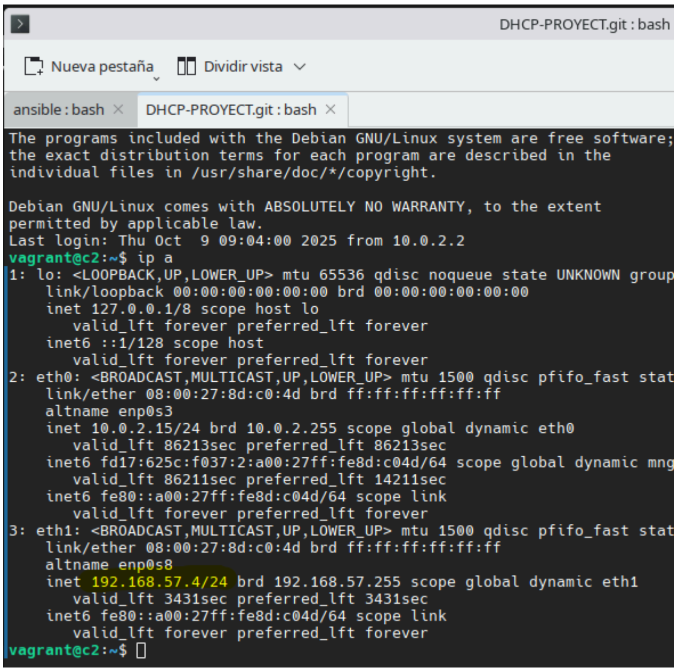


```
# dhcpd.conf
subnet 192.168.57.0 netmask 255.255.255.0 {
    range 192.168.57.25 192.168.57.50;
    option routers 192.168.57.10;
    option domain-name-servers 8.8.8.8, 4.4.4.4;
    option domain-name "micasa.es";
    default-lease-time 86400;
    max-lease-time 691200;
}

host c2 {
  hardware ethernet 08:00:27:8d:c0:4d;
  fixed-address 192.168.57.4;
  option domain-name-servers 1.1.1.1;
  default-lease-time 3600;
}
```

4- Test IP clients

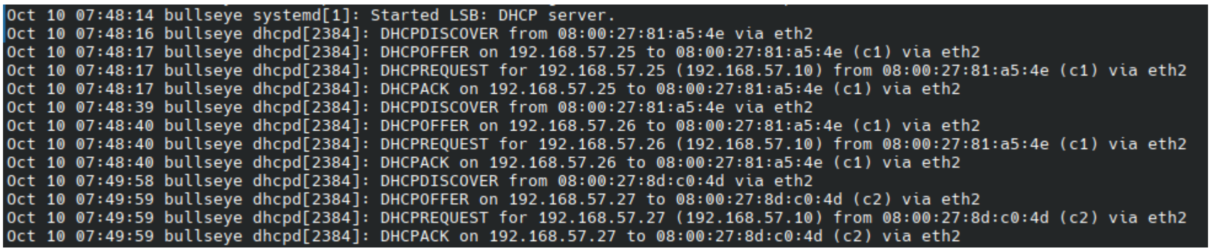


```
vagrant@c1:~$ ip -4 addr show
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
    inet 127.0.0.1/8 scope host lo
       valid_lft forever preferred_lft forever
2: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast state UP group default qlen 1000
    altname enp0s3
    inet 10.0.2.15/24 brd 10.0.2.255 scope global dynamic eth0
       valid_lft 83701sec preferred_lft 83701sec
3: eth1: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast state UP group default qlen 1000
    altname enp0s8
    inet 192.168.57.25/24 brd 192.168.57.255 scope global dynamic eth1
       valid_lft 86047sec preferred_lft 86047sec
    inet 192.168.57.26/24 brd 192.168.57.255 scope global secondary dynamic eth1
       valid_lft 86070sec preferred_lft 86070sec
vagrant@c1:~$
```



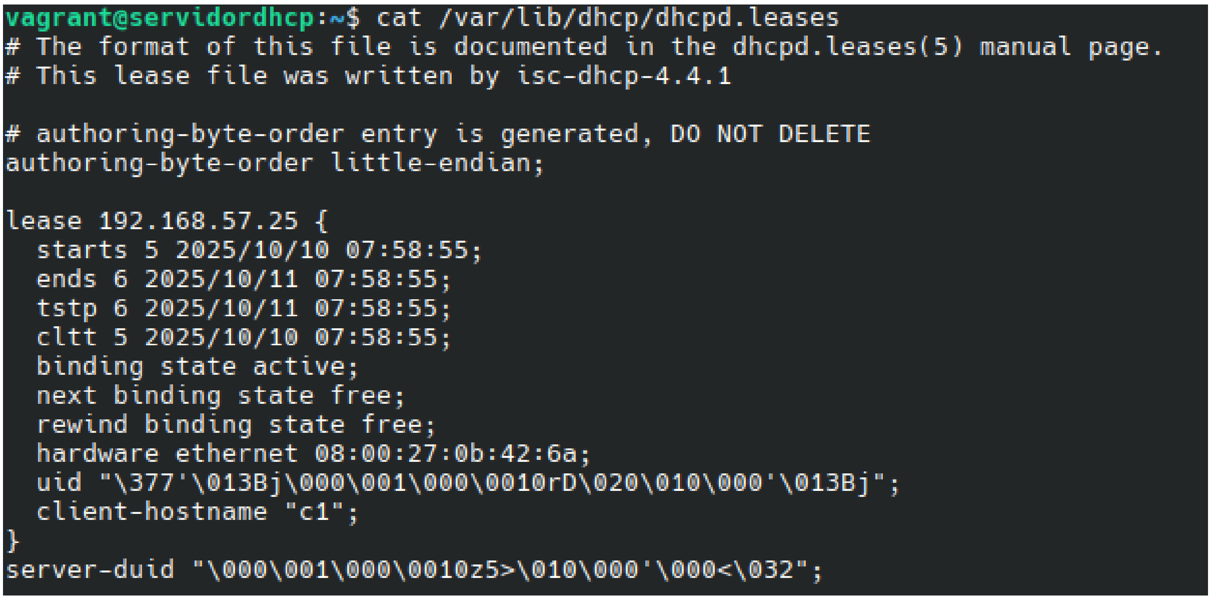
```
The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
Last login: Thu Oct  9 09:04:00 2025 from 10.0.2.2
vagrant@c2:~$ ip a
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group
    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: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast stat
    link/ether 08:00:27:8d:c0:4d brd ff:ff:ff:ff:ff:ff
    altname enp0s3
    inet 10.0.2.15/24 brd 10.0.2.255 scope global dynamic eth0
       valid_lft 86213sec preferred_lft 86213sec
    inet6 fd17:625c:f037:2:a00:27ff:fe8d:c04d/64 scope global dynamic mng
       valid_lft 86211sec preferred_lft 14211sec
    inet6 fe80::a00:27ff:fe8d:c04d/64 scope link
       valid_lft forever preferred_lft forever
3: eth1: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast stat
    link/ether 08:00:27:8d:c0:4d brd ff:ff:ff:ff:ff:ff
    altname enp0s8
    inet 192.168.57.4/24 brd 192.168.57.255 scope global dynamic eth1
       valid_lft 3431sec preferred_lft 3431sec
    inet6 fe80::a00:27ff:fe8d:c04d/64 scope link
       valid_lft forever preferred_lft forever
vagrant@c2:~$
```

5- Review Logs



```
Oct 10 07:48:14 bullseye systemd[1]: Started LSB: DHCP server.
Oct 10 07:48:16 bullseye dhcpd[2384]: DHCPDISCOVER from 08:00:27:81:a5:4e via eth2
Oct 10 07:48:17 bullseye dhcpd[2384]: DHCPOFFER on 192.168.57.25 to 08:00:27:81:a5:4e (c1) via eth2
Oct 10 07:48:17 bullseye dhcpd[2384]: DHCPREQUEST for 192.168.57.25 (192.168.57.10) from 08:00:27:81:a5:4e (c1) via eth2
Oct 10 07:48:17 bullseye dhcpd[2384]: DHCPACK on 192.168.57.25 to 08:00:27:81:a5:4e (c1) via eth2
Oct 10 07:48:39 bullseye dhcpd[2384]: DHCPDISCOVER from 08:00:27:81:a5:4e via eth2
Oct 10 07:48:40 bullseye dhcpd[2384]: DHCPOFFER on 192.168.57.26 to 08:00:27:81:a5:4e (c1) via eth2
Oct 10 07:48:40 bullseye dhcpd[2384]: DHCPREQUEST for 192.168.57.26 (192.168.57.10) from 08:00:27:81:a5:4e (c1) via eth2
Oct 10 07:48:40 bullseye dhcpd[2384]: DHCPACK on 192.168.57.26 to 08:00:27:81:a5:4e (c1) via eth2
Oct 10 07:49:58 bullseye dhcpd[2384]: DHCPDISCOVER from 08:00:27:8d:c0:4d via eth2
Oct 10 07:49:59 bullseye dhcpd[2384]: DHCPOFFER on 192.168.57.27 to 08:00:27:8d:c0:4d (c2) via eth2
Oct 10 07:49:59 bullseye dhcpd[2384]: DHCPREQUEST for 192.168.57.27 (192.168.57.10) from 08:00:27:8d:c0:4d (c2) via eth2
Oct 10 07:49:59 bullseye dhcpd[2384]: DHCPACK on 192.168.57.27 to 08:00:27:8d:c0:4d (c2) via eth2
```

6- View Leases



```
vagrant@servidordhcp:~$ cat /var/lib/dhcp/dhcpd.leases
# The format of this file is documented in the dhcpd.leases(5) manual page.
# This lease file was written by isc-dhcp-4.4.1

# authoring-byte-order entry is generated, DO NOT DELETE
authoring-byte-order little-endian;

lease 192.168.57.25 {
  starts 5 2025/10/10 07:58:55;
  ends 6 2025/10/11 07:58:55;
  tstp 6 2025/10/11 07:58:55;
  cltt 5 2025/10/10 07:58:55;
  binding state active;
  next binding state free;
  rewind binding state free;
  hardware ethernet 08:00:27:0b:42:6a;
  uid "\377'\013Bj\000\001\000\0010rD\020\010\000'\013Bj";
  client-hostname "c1";
}
server-duid "\000\001\000\0010z5>\010\000'\000<\032";
```