

Linux Ağ Yönetimi 2020 Bahar Final Projesi

Abdulhamit Kumru 170202020

Kullanılan yazılımların versiyonları Ubuntu Desktop / Server 18, Virtual Box 6.1

Makinelerin Klonlanması

Base olarak kullanacağımız makineyi oluşturduktan sonra klon makineyi menüden sağ tıklayıp istediğimiz şekilde oluşturabiliyoruz. Biz daha az yer kullanmak ve ortak networklere bağlanacağımız için **Linked Clone** ve **Generate new MAC addresses for all network adapters** seçeneğini kullanacağız.

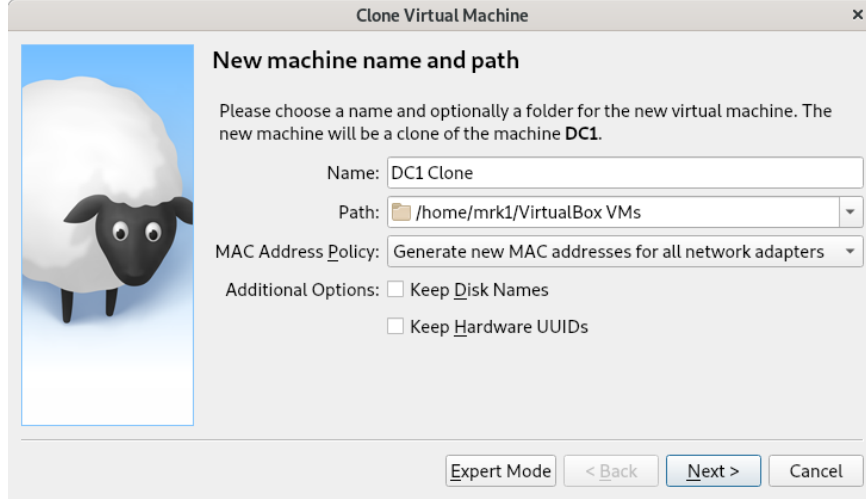


Figure 1: Klonlama

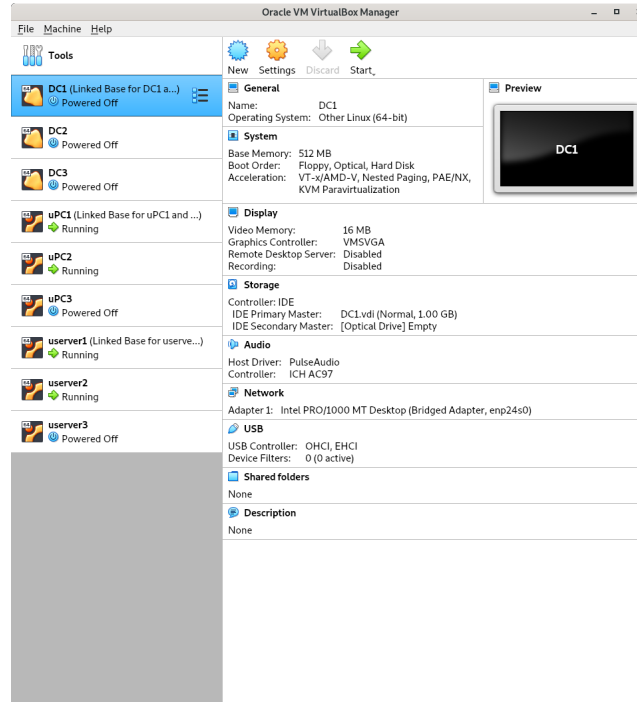


Figure 2: Klonlanmış Makineler

Senaryo 1

NAT ile Host-Guest Ubuntu PC Bağlantısı

Başlangıç olarak guest makinenin Network Adapterini NAT olarak seçiyoruz

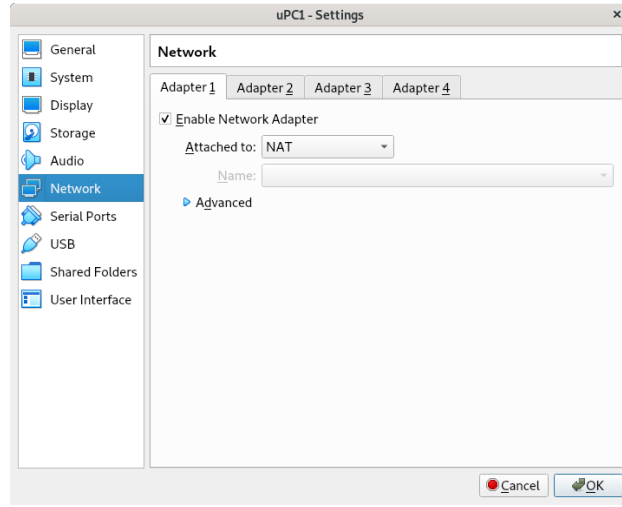


Figure 3: NAT adapter ayarı

SSH

host makineye NAT bağlantı ile erişebilmek için öncelikle gerekli gerekli portu yönlendiriyoruz daha sonra gueste ssh ile bağlanabilmek için open-ssh serveri apt ile yüklüyoruz.

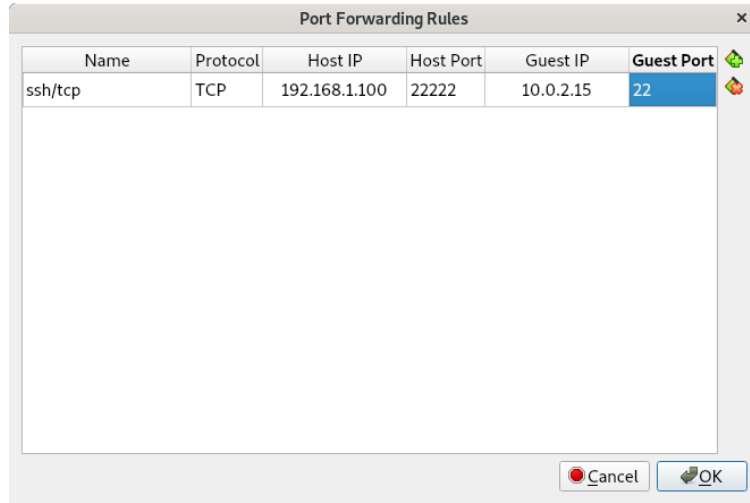


Figure 4: Port forwarding

```
$ apt install opessh-server
```

ssh server servisinin ayarlarını dosyasından 22 numaralı portu ve parolayı kabul edecek şekilde yapıyoruz.

```
$ echo "Port 22" >> /etc/ssh/sshd_config
```

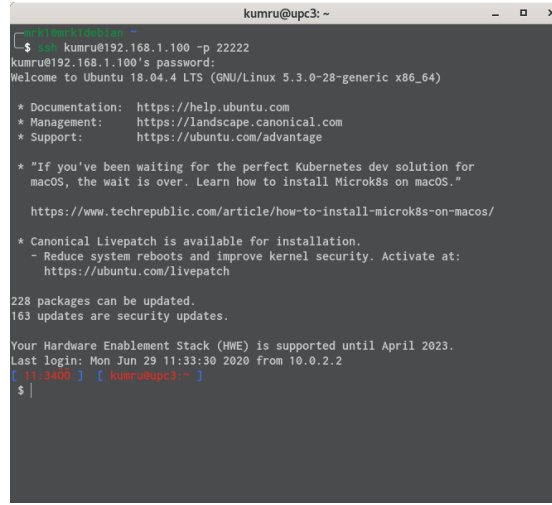
```
$ echo "PasswordAuthentication yes" >> /etc/ssh/sshd_config
```

Ayarların aktif olması için ssh servisimizi yeniden başlatıyoruz.

```
$ sudo service ssh restart
```

Forward ettiğimiz port üzerinden guest'e bağlanabiliriz.

```
$ ssh kumru@192.168.1.100 -p 22222 # hostun ip adresi
```

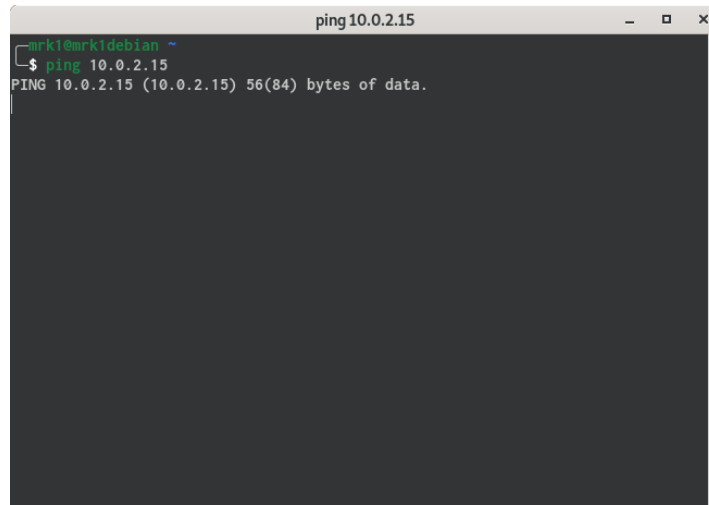


```
kumru@upc3: ~  
$ ssh kumru@192.168.1.100 -p 22222  
kumru@192.168.1.100's password:  
Welcome to Ubuntu 18.04.4 LTS (GNU/Linux 5.3.0-28-generic x86_64)  
  
 * Documentation:  https://help.ubuntu.com  
 * Management:    https://landscape.canonical.com  
 * Support:        https://ubuntu.com/advantage  
  
 * "If you've been waiting for the perfect Kubernetes dev solution for  
  macOS, the wait is over. Learn how to install Microk8s on macOS."  
  https://www.techrepublic.com/article/how-to-install-microk8s-on-macos/  
  
 * Canonical Livepatch is available for installation.  
   - Reduce system reboots and improve kernel security. Activate at:  
     https://ubuntu.com/livepatch  
  
228 packages can be updated.  
163 updates are security updates.  
  
Your Hardware Enablement Stack (HWE) is supported until April 2023.  
Last login: Mon Jun 29 11:33:30 2020 from 10.0.2.2  
[ 11:3400 ] [ kumru@upc3:~ ]  
$ |
```

Figure 5: Gueste ssh ile baglanma

Ping

Eğer ICMP port ile çalışsaydı aynı şekilde onunda portunu yönlendirip gueste ping atabilirdik fakat ICMP TCP/UDP portları üzerinden çalışmıyor.



```
ping 10.0.2.15  
mrkl@mrkldebian ~  
$ ping 10.0.2.15  
PING 10.0.2.15 (10.0.2.15) 56(84) bytes of data.
```

Figure 6: Gueste ping gönderilemiyor

Misafir Eklentileri

Misafir eklentilerini guest makineye sanal disk takarak yükleyebiliyoruz bu sayade hosttan gueste veya guestten hosta kopyala yapıştır ve dosya sürükleyip bırak yapabiliyoruz. Eklentiği yüklemek zor değil sanal makine penceresinden **Devices > Insert Guest Additions CD...** seçeneğini seçince bize `autorun.sh`'ı çalıştırır mı diye soruyor evet diyip parolamızı girince eklentiler yükleniyor

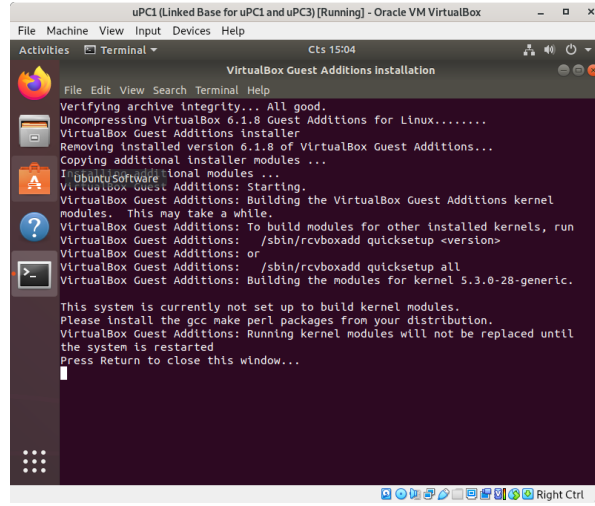


Figure 7: autorun.sh çıktısı

Sanal makinenin penceresinden **Devices > Drag and Drop**, **Devices > Shared Clipboard** seçeneklerinden detaylı ayar yapılabilir.

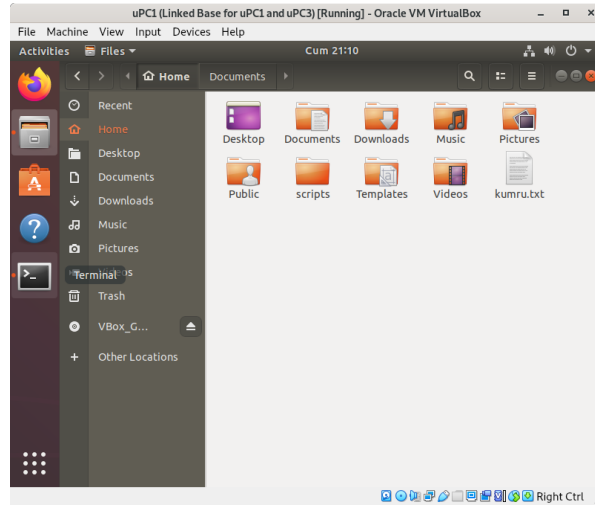


Figure 8: Gueste eklentileri ile sürükleyip bırak

gif linki (github)

Network Bridge ile Host, TinyCore Guest Bağlantısı

Guestin network adapter ayarını Bridge Adapter olarak seçiyoruz

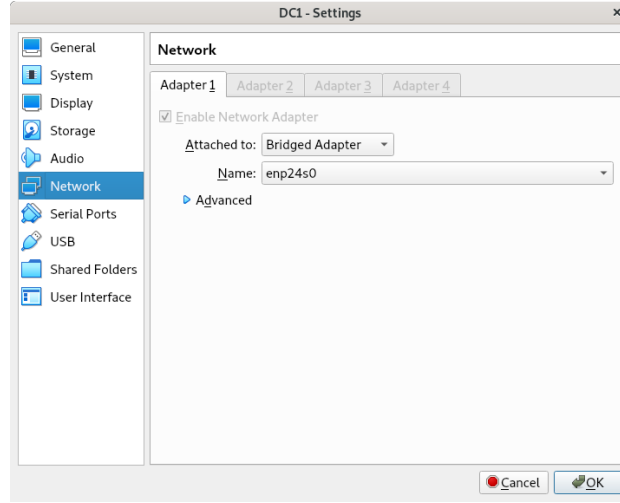


Figure 9: Network Bridge Ayarı

TinyCoreda IP'mizi ifconfig komutu ile kontrol edebiliyoruz.

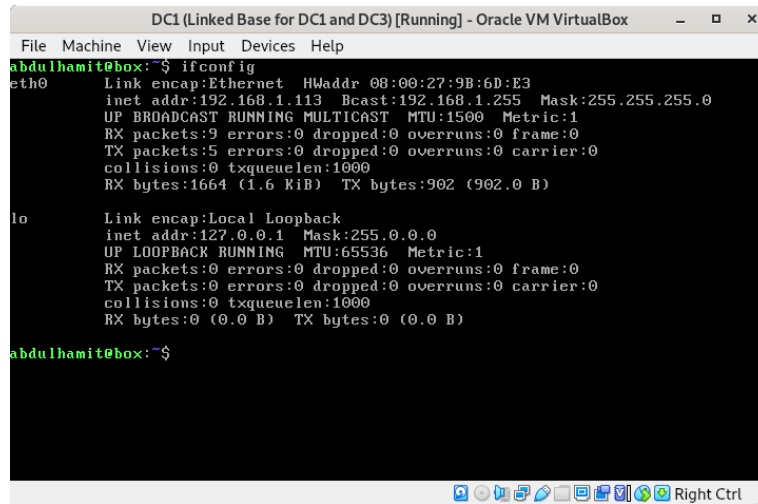


Figure 10: ifconfig çıktısı

Network Bridge ile Guest LANdaki makine gibi kullanılabilir.

SSH

tce-load programı ile TinyCoreda program yükleyebiliyoruz.

```
# openssh client/server paketini yükleme
tce-load -w -i openssh.tcz
# ssh/sshd ayarları
cp /usr/local/etc/ssh/ssh_config.orig /usr/local/etc/ssh/ssh_config
cp /usr/local/etc/ssh/sshd_config.orig /usr/local/etc/ssh/sshd_config
echo "Port 22" >> /usr/local/etc/ssh/sshd_config
echo "PasswordAuthentication yes" >> /usr/local/etc/ssh/sshd_config
# servisi calistirma
/usr/local/etc/init.d/openssh start
```

```
ssh abdulhamit@192.168.1.113
mrkl@mrkldebian ~
$ uname -a
Linux mrkldebian 4.19.0-9-amd64 #1 SMP Debian 4.19.118-2+deb10u1 (2020-06-07) x86_64 GNU/Linux
mrkl@mrkldebian ~
$ ssh abdulhamit@192.168.1.113
abdulhamit@192.168.1.113's password:
( '>')
/) TC (\   Core is distributed with ABSOLUTELY NO WARRANTY.
(/_--_\)   www.tinycorelinux.net

abdulhamit@box:~$ uname -a
Linux box 5.4.3-tinycore #2020 SMP Tue Dec 17 17:00:50 UTC 2019 i686 GNU/Linux
abdulhamit@box:~$
```

Figure 11: Gueste ssh bağlantısı

Ping

```
ping 192.168.1.113
mrkl@mrkldebian ~
$ ping 192.168.1.113
PING 192.168.1.113 (192.168.1.113) 56(84) bytes of data.
64 bytes from 192.168.1.113: icmp_seq=1 ttl=64 time=0.244 ms
64 bytes from 192.168.1.113: icmp_seq=2 ttl=64 time=0.237 ms
```

Figure 12: Gueste ping gönderebiliyoruz

Host only Adapter ile Host, Ubuntu Server Guest Bağlantısı

Host only Adapter kullanabilmek için öncelikle Host Network oluşturmak gerekli. Ana Menüden File > Host Network Manager...i seçip create tıklarız ve Network Oluşuyor.

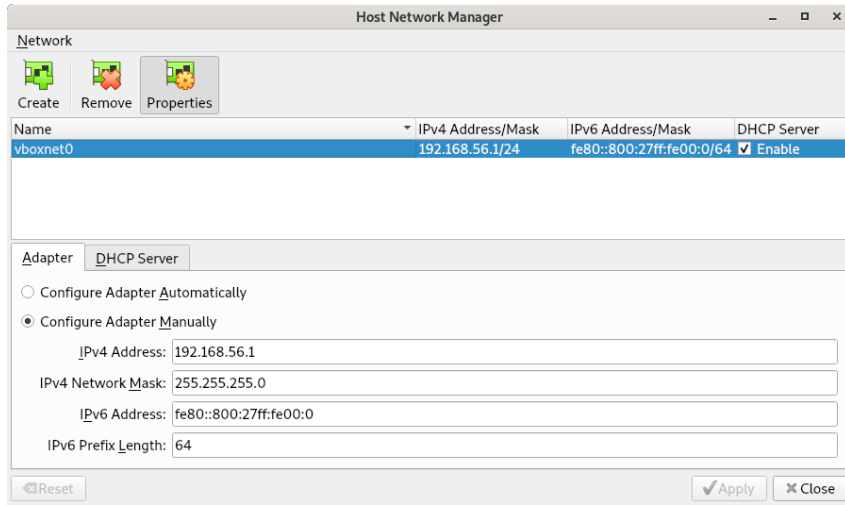


Figure 13: Host Network

Guestimizin Network adapterini Host only Adapter seçip alt seçenekten oluşturduğumuz Host Network'ü seçiyoruz.

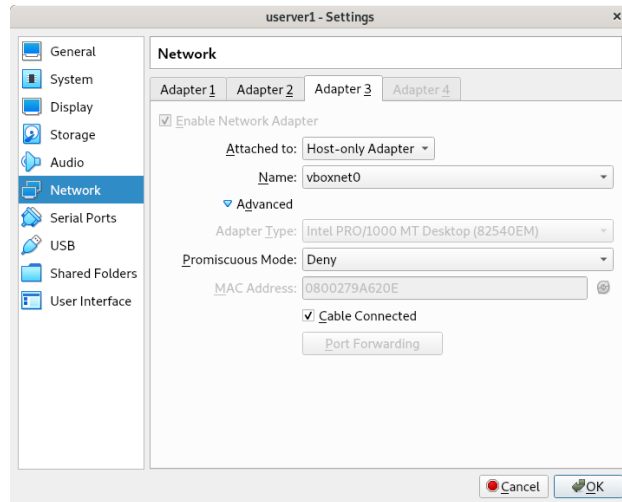


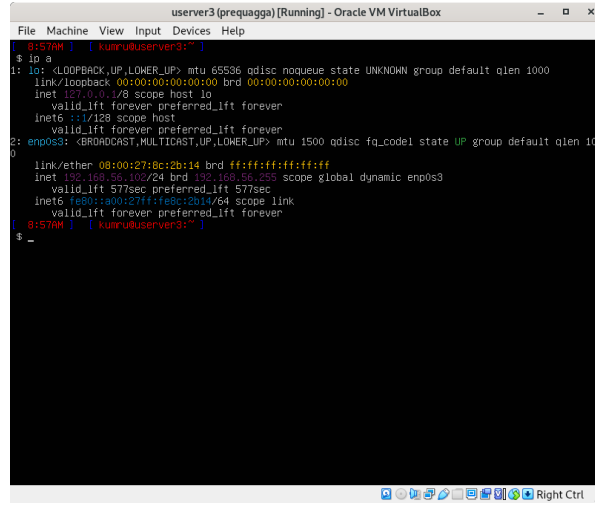
Figure 14: Host Only Adapter Ayarı

yaml formatındaki /etc/netplan/50-cloud-init.yaml dosyasını Network Interfacemizi DHCP ile yönetilmesi için ayarlıyoruz

```
# /etc/netplan/50-cloud-init.yaml
network:
  version: 2
  renderer: networkd
  ethernets:
    enp0s9:
      dhcp4: yes
```

Yeni ayarların kullanması için netplan komutunu çalıştırıyoruz.

```
$ sudo netplan --debug apply
```

```
userver3 (prequagga) [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
8:57AM [ kumru@userver3:~ ]
$ ip 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 08:00:27:8c:2b:14 brd ff:ff:ff:ff:ff:ff
    inet 192.168.56.102/24 brd 192.168.56.255 scope global dynamic enp0s3
        valid_lft 577sec preferred_lft 577sec
    inet6 fe80::a00:27ff:fe8c:2b14/64 scope link
        valid_lft forever preferred_lft forever
8:57AM [ kumru@userver3:~ ]
$ _
```

Figure 15: Host-only Adapterin Host Networkten aldığı IP.

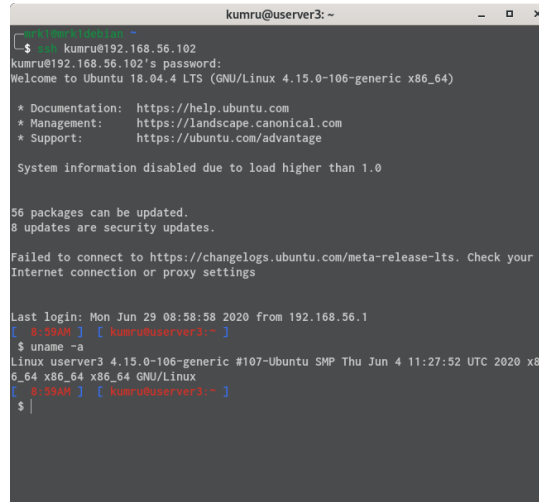
SSH

NAT ile Host, Guest Ubuntu PC Bağlantısı bölümünde yaptığımız komutları burada tekrar ediyoruz.

```
$ apt install opessh-server
$ echo "Port 22" >> /etc/ssh/sshd_config
$ echo "PasswordAuthentication yes" >> /etc/ssh/sshd_config
$ sudo service ssh restart
```

userver1'e SSH ile bağlanıyoruz.

```
$ ssh kumru@192.168.56.101
```



```
kumru@userver3: ~
$ ssh kumru@192.168.56.102
kumru@192.168.56.102's password:
Welcome to Ubuntu 18.04.4 LTS (GNU/Linux 4.15.0-106-generic x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:    https://landscape.canonical.com
 * Support:       https://ubuntu.com/advantage

System information disabled due to load higher than 1.0

56 packages can be updated.
8 updates are security updates.

Failed to connect to https://changelogs.ubuntu.com/meta-release-lts. Check your
Internet connection or proxy settings

Last login: Mon Jun 29 08:58:58 2020 from 192.168.56.1
[ 8:59AM ] [ kumru@userver3:~ ]
$ uname -a
Linux userver3 4.15.0-106-generic #107-Ubuntu SMP Thu Jun 4 11:27:52 UTC 2020 x86_64 x86_64 x86_64 GNU/Linux
[ 8:59AM ] [ kumru@userver3:~ ]
$
```

Figure 16: Gueste ssh ile bağlanma

Ping

```
mrk1@mrk1debian: ~  
└─$ ping 192  
└─$ ping 192.168.56.102  
PING 192.168.56.102 (192.168.56.102) 56(84) bytes of data.  
64 bytes from 192.168.56.102: icmp_seq=1 ttl=64 time=0.267 ms  
64 bytes from 192.168.56.102: icmp_seq=2 ttl=64 time=0.595 ms  
64 bytes from 192.168.56.102: icmp_seq=3 ttl=64 time=0.360 ms  
64 bytes from 192.168.56.102: icmp_seq=4 ttl=64 time=0.350 ms  
^C  
--- 192.168.56.102 ping statistics ---  
4 packets transmitted, 4 received, 0% packet loss, time 75ms  
rtt min/avg/max/mdev = 0.267/0.393/0.595/0.122 ms  
└─$ |
```

Figure 17: Guest Ping gönderme

Senaryo 2

Internal Network ile Host-Guest, Guest-Guest Bağlatısı

Internal Network Ayarı

Senaryo gereği ip Ubuntu Serverların ipleri

host: 192.168.0.1 userver1: 192.168.0.2 userver2: 192.168.0.3 userver3: 192.168.0.4

Internal network için Network Adapterin modunu seçmek yeterli.

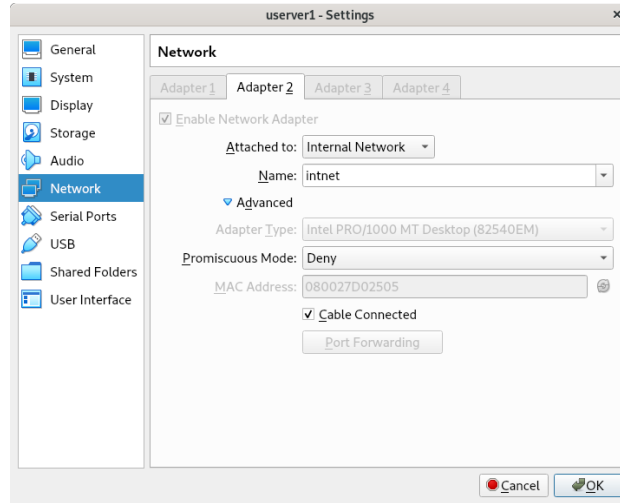


Figure 18: Internal Network Ayarları

Network

Adapter 1: Intel PRO/1000 MT Desktop (Internal Network, 'intnet')

Figure 19: userver1

Network

Adapter 1: Intel PRO/1000 MT Desktop (Internal Network, 'intnet')

Figure 20: userver2

Network

Adapter 1: Intel PRO/1000 MT Desktop (Internal Network, 'intnet')

Figure 21: userver3

Netplan Ayarları

netplan ayarlarını Senaryo 1 de yaptığımız gibi yapıyoruz. Farklı olarak dhcp olmadan statik bir şekilde IP alıyoruz

- userver1

```
# /etc/netplan/50-cloud-init.yaml
# S:0 R:2 userver1
network:
  version: 2
  renderer: networkd
  ethernets:
    enp0s8:
      dhcp4: no
      addresses:
        - 192.168.0.2/24
      gateway4: 192.168.0.1
```

- userver2

```
# /etc/netplan/50-cloud-init.yaml
# S:0 R:2 userver2
network:
  version: 2
  renderer: networkd
  ethernets:
    enp0s8:
      dhcp4: no
      addresses:
        - 192.168.0.3/24
      gateway4: 192.168.0.1
```

- userver3

```
# /etc/netplan/50-cloud-init.yaml
# S:0 R:2 userver3
network:
  version: 2
  renderer: networkd
  ethernets:
    enp0s8:
      dhcp4: no
      addresses:
        - 192.168.0.4/24
      gateway4: 192.168.0.1
```

SSH

SSH server yüklediğimiz için ve diğer makineler klon olduğu için bir daha SSH yükleme ve ayarlamaya gerek yok. Guestler arasında SSH bağlantısı yapılabiliyor.

```
userver2 [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
6:34PM | kumru@userver2:~$ ssh 192.168.0.2
The authenticity of host '192.168.0.2 (192.168.0.2)' can't be established.
ECDSA key fingerprint is SHA256:fEn1Q2McENj2KX05Fukg3Y1rByRk8X4MAYQ3bnZc0Y.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added '192.168.0.2' (ECDSA) to the list of known hosts.
kumru@192.168.0.2's password:
Welcome to Ubuntu 18.04.4 LTS (GNU/Linux 4.15.0-106-generic x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:    https://landscape.canonical.com
 * Support:       https://ubuntu.com/advantage

System information as of Sun Jun 21 18:34:51 UTC 2020

System load: 0.0          Processes:           93
Usage of /:  43.9% of 9.7GB Users logged in:       1
Memory usage: 6%         IP address for enp0s8: 192.168.0.2
Swap usage:  0%

48 packages can be updated.
0 updates are security updates.

Failed to connect to https://changelogs.ubuntu.com/meta-release-lts. Check your Internet connection or proxy settings

Last login: Sun Jun 21 18:28:54 2020
zsh: corrupt history file /home/kumru/.zsh_history
6:34PM | kumru@userver1:~$
```

Figure 22: Guestler Arasında SSH

```
userver1 (Linked Base for userver1 and userver3) [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
6:40PM | kumru@userver1:~$ ssh 192.168.0.4
The authenticity of host '192.168.0.4 (192.168.0.4)' can't be established.
ECDSA key fingerprint is SHA256:fEn1Q2McENj2KX05Fukg3Y1rByRk8X4MAYQ3bnZc0Y.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added '192.168.0.4' (ECDSA) to the list of known hosts.
kumru@192.168.0.4's password:
Welcome to Ubuntu 18.04.4 LTS (GNU/Linux 4.15.0-106-generic x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:    https://landscape.canonical.com
 * Support:       https://ubuntu.com/advantage

System information as of Sun Jun 21 18:41:55 UTC 2020

System load: 0.08         Processes:           88
Usage of /:  43.9% of 9.7GB Users logged in:       1
Memory usage: 7%         IP address for enp0s8: 192.168.0.4
Swap usage:  0%

48 packages can be updated.
0 updates are security updates.

Failed to connect to https://changelogs.ubuntu.com/meta-release-lts. Check your Internet connection or proxy settings

Last login: Sun Jun 21 18:30:25 2020
zsh: corrupt history file /home/kumru/.zsh_history
6:41PM | kumru@userver3:~$
```

Figure 23: Guestler Arasında SSH

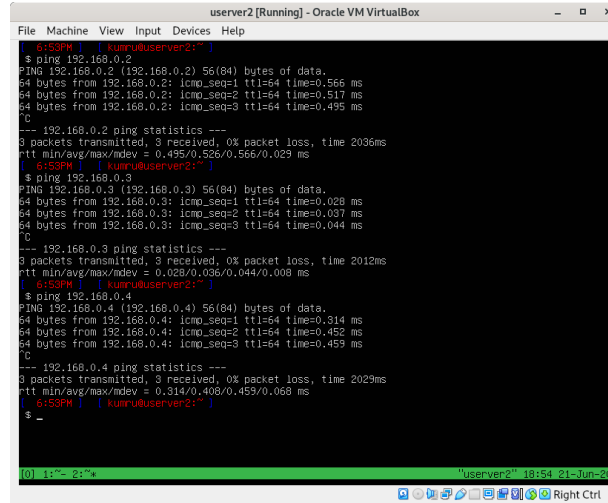
Internal Networkte hostdan guestlere ulaşmak mümkün değil.

```
mrk1@mrk1debian: ~
$ ssh kumru@192.168.0.3
ssh: connect to host 192.168.0.3 port 22: No route to host
mrk1@mrk1debian: ~
$
```

Figure 24: Hosttan Geste SSH denemesi

Ping

Guestler birbirlerine ulaşabildiği için birbirlerine ping göndermeleri mümkün.



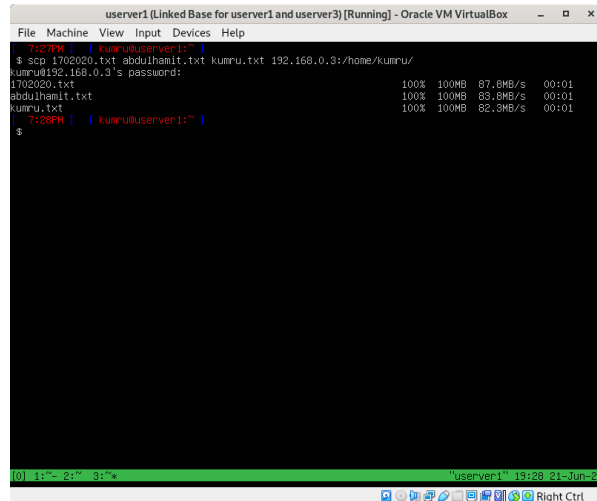
```
userver2 [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
$ ping 192.168.0.2
PING 192.168.0.2 (192.168.0.2) 56(84) bytes of data:
64 bytes from 192.168.0.2: icmp_seq=1 ttl=64 time=0.566 ms
64 bytes from 192.168.0.2: icmp_seq=2 ttl=64 time=0.517 ms
64 bytes from 192.168.0.2: icmp_seq=3 ttl=64 time=0.495 ms
--- 192.168.0.2 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2036ms
rtt min/avg/max/mdev = 0.495/0.526/0.566/0.029 ms
$ ping 192.168.0.3
PING 192.168.0.3 (192.168.0.3) 56(84) bytes of data:
64 bytes from 192.168.0.3: icmp_seq=1 ttl=64 time=0.028 ms
64 bytes from 192.168.0.3: icmp_seq=2 ttl=64 time=0.037 ms
64 bytes from 192.168.0.3: icmp_seq=3 ttl=64 time=0.044 ms
--- 192.168.0.3 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 1012ms
rtt min/avg/max/mdev = 0.028/0.036/0.044/0.008 ms
$ ping 192.168.0.4
PING 192.168.0.4 (192.168.0.4) 56(84) bytes of data:
64 bytes from 192.168.0.4: icmp_seq=1 ttl=64 time=0.314 ms
64 bytes from 192.168.0.4: icmp_seq=2 ttl=64 time=0.452 ms
64 bytes from 192.168.0.4: icmp_seq=3 ttl=64 time=0.459 ms
--- 192.168.0.4 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2029ms
rtt min/avg/max/mdev = 0.314/0.408/0.459/0.068 ms
$ _
```

Figure 25: Guestler Arası Ping

SCP

Gerek olmasada boş dosya göndermek yerine sıfır yazılmış 100M büyüklüğünde dosyalar gönderelim.

```
# dosyaları hazırlama
$ dd if=/dev/zero of=abdulhamit.txt count=100 bs=1M
$ cat abdulhamit.txt > kumru.txt > 1702020.txt
# scp komutu
$ scp *.txt 192.168.0.3:/home/kumru/
```



```
userver1 (Linked Base for userver1 and userver3) [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
$ scp 1702020.txt abdulhamit.txt kumru.txt 192.168.0.3:/home/kumru/
kumru192.168.0.3's password:
1702020.txt          100% 100MB 87.8MB/s   00:01
abdulhamit.txt       100% 100MB 83.8MB/s   00:01
kumru.txt            100% 100MB 82.3MB/s   00:01
$
```

Figure 26: scp Komutu Çıktısı

Senaryo 3

Bu senaryoda bir ağdan diğer ağa routing yapmamız istenmekte. Router yazılımı olarak Quagga'yı kullanacağız.

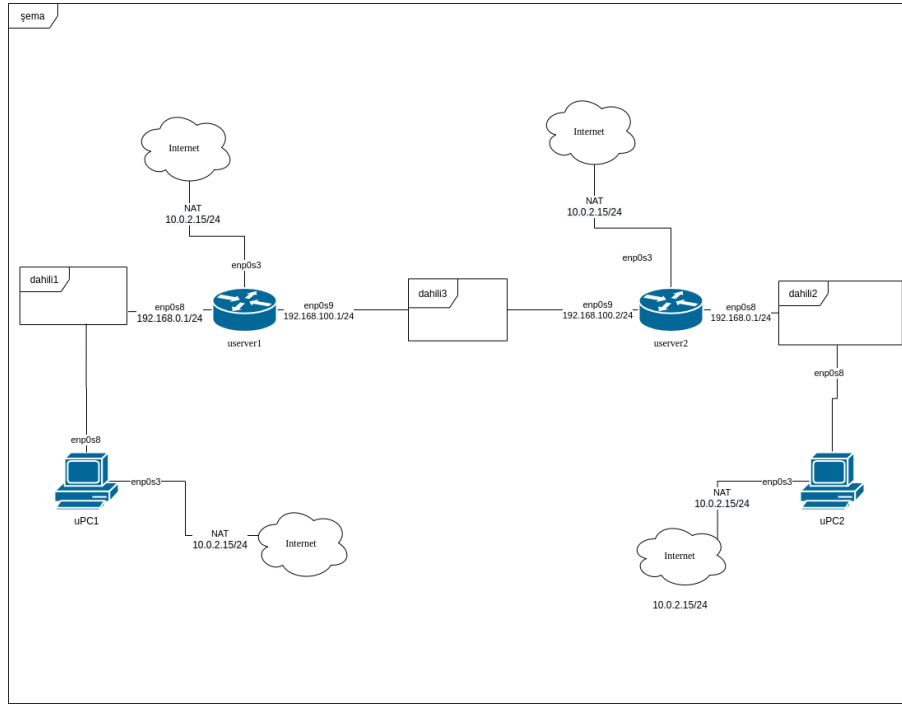


Figure 27: Şema

- dahili1
uPC1 192.168.1.1
userver1 192.168.1.254
- dahili2
uPC2 192.168.2.1
userver2 192.168.2.254
- dahili 3 userver1 192.168.100.1
userver2 192.168.100.2

Network Ayarları

Virtual Box Network Ayarları

Serverlardaki adapter4 hariç diğer adaptörler senaryo gerekleri için kullandık. Adapter4 ü ise sunucuları yönetmek için kullandık.

Network

Adapter 1: Intel PRO/1000 MT Desktop (NAT)

Adapter 2: Intel PRO/1000 MT Desktop (Internal Network, 'dahili1')

Figure 28: uPC1

Network

Adapter 1: Intel PRO/1000 MT Desktop (NAT)

Adapter 2: Intel PRO/1000 MT Desktop (Internal Network, 'dahili2')

Figure 29: uPC2

Network

Adapter 1: Intel PRO/1000 MT Desktop (NAT)

Adapter 2: Intel PRO/1000 MT Desktop (Internal Network, 'dahili1')

Adapter 3: Intel PRO/1000 MT Desktop (Internal Network, 'dahili100')

Adapter 4: Intel PRO/1000 MT Desktop (Bridged Adapter, enp24s0)

Figure 30: userver1

Network

Adapter 1: Intel PRO/1000 MT Desktop (NAT)

Adapter 2: Intel PRO/1000 MT Desktop (Internal Network, 'dahili2')

Adapter 3: Intel PRO/1000 MT Desktop (Internal Network, 'dahili100')

Adapter 4: Intel PRO/1000 MT Desktop (Bridged Adapter, enp24s0)

Figure 31: userver2

PC'lerin netplan Ayarları

uPC1 netplan ayarları

```
# uPC1
network:
  version: 2
  renderer: networkd
  ethernets:
    enp0s3:
      dhcp4: yes
    enp0s8:
      dhcp4: no
      addresses:
        - 192.168.1.1/24
      gateway4: 192.168.1.254
      nameservers:
        addresses: [8.8.8.8, 1.1.1.1]
      routes:
        - to: 192.168.0.0/16
          via: 192.168.1.254
```

uPC2 netplan ayarları

```
# uPC2
network:
  version: 2
  renderer: networkd
```



```

ethernets:
  enp0s3:
    dhcp4: yes
  enp0s8:
    dhcp4: no
    addresses:
      - 192.168.2.1/24
    gateway4: 192.168.2.254
    nameservers:
      addresses: [8.8.8.8, 1.1.1.1]
    routes:
      - to: 192.168.0.0/16
        via: 192.168.1.254

```

Server'lerin Network Ayarları

Serverlerin nat ayarları

serverler için ortak nat ayarı

```

network:
  ethernets:
    enp0s3:
      dhcp4: true
  version: 2

userver2
# userver2
network:
  version: 2
  renderer: networkd
  ethernets:
    enp0s8:
      dhcp4: no
      addresses:
        - 192.168.0.1/24
      #gateway4: 192.168.0.1
      nameservers:
        addresses: [8.8.8.8, 8.8.4.4]
    enp0s9:
      dhcp4: no
      addresses:
        - 192.168.100.2/24
      #gateway4: 192.168.0.1
      nameservers:
        addresses: [8.8.8.8, 8.8.4.4]
    enp0s10:
      dhcp4: no
      addresses:
        - 192.168.1.201/24
      #gateway4: 192.168.0.1
      nameservers:
        addresses: [8.8.8.8, 8.8.4.4]

```

userver1

```

# userver1
network:
  version: 2
  renderer: networkd
  ethernets:

```

```

enp0s8:
  dhcp4: no
  addresses:
    - 192.168.0.1/24
    #gateway4: 192.168.0.1
  nameservers:
    addresses: [8.8.8.8, 8.8.4.4]
enp0s9:
  dhcp4: no
  addresses:
    - 192.168.100.1/24
    #gateway4: 192.168.0.1
  nameservers:
    addresses: [8.8.8.8, 8.8.4.4]
enp0s10:
  dhcp4: no
  addresses:
    - 192.168.1.200/24
    #gateway4: 192.168.0.1
  nameservers:
    addresses: [8.8.8.8, 8.8.4.4]

```

Serverlere Quagga Kurulumu ve Ayarları

Buradaki bilgiler brianlinkletter.com , ixnfo.com dan alınmıştır.

Quagga Kurulum Scripti

```

#!/bin/bash
# quagga installer
if [ "$EUID" -ne 0 ]
then echo "Please run as root"
exit
fi

sudo apt install quagga quagga-doc
sudo cat > /etc/quagga/daemons << EOF
zebra=yes
bgpd=no
ospfd=yes
ospf6d=no
ripd=no
ripngd=no
isisd=no
babeld=no
EOF

echo "net.ipv4.ip_forward=1" >> /etc/sysctl.conf
cp /usr/share/doc/quagga-core/examples/vtysh.conf.sample /etc/quagga/vtysh.conf
cp /usr/share/doc/quagga-core/examples/zebra.conf.sample /etc/quagga/zebra.conf
cp /usr/share/doc/quagga-core/examples/bgpd.conf.sample /etc/quagga/bgpd.conf
chown quagga:quagga /etc/quagga/*.conf
chown quagga:quaggavty /etc/quagga/vtysh.conf
chmod 640 /etc/quagga/*.conf
service zebra start
service bgpd start
systemctl enable zebra.service
systemctl enable bgpd.service
echo 'VTYSH_PAGER=more' >>/etc/environment

```

```
echo 'export VTYSH_PAGER=more' >>/etc/bash.bashrc
```

Quagga Ayar Scriptleri

- userver1

```
#!/bin/bash
if [ "$EUID" -ne 0 ]
then echo "Please run as root"
exit
fi
cat >> /etc/quagga/ospfd.conf << EOF
interface enp0s8
interface enp0s9
interface lo
router ospf
  passive-interface enp0s8
  network 192.168.1.0/24 area 0.0.0.0
  network 192.168.100.0/24 area 0.0.0.0
line vty
EOF
cat >> /etc/quagga/zebra.conf << EOF
interface enp0s8
  ip address 192.168.1.254/24
  ipv6 nd suppress-ra
interface enp0s9
  ip address 192.168.100.1/24
  ipv6 nd suppress-ra
interface lo
ip forwarding
line vty
EOF
sudo service zebra restart
sudo service bgpd restart
```

- userver2

```
#!/bin/bash
if [ "$EUID" -ne 0 ]
then echo "Please run as root"
exit
fi
cat >> /etc/quagga/ospfd.conf << EOF
interface enp0s8
interface enp0s9
interface lo
router ospf
  passive-interface enp0s8
  network 192.168.2.0/24 area 0.0.0.0
  network 192.168.100.0/24 area 0.0.0.0
line vty
EOF
cat >> /etc/quagga/zebra.conf << EOF
interface enp0s8
  ip address 192.168.2.254/24
  ipv6 nd suppress-ra
interface enp0s9
  ip address 192.168.100.2/24
```

Ayar sontrası cihazların durumu

Figure 32: Ubuntu Server1 ip a, ip route Komut çıktısı

Figure 33: Ubuntu Server2 ip a, ip route Komut çıktısı

```
tmux
File Edit View Search Terminal Help
[ S:5505 ] [ kunru@upc1:~ ]
$ ip 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:a0:ee:ee brd ff:ff:ff:ff:ff:ff
    inet 10.0.2.15/24 brd 10.0.2.255 scope global dynamic enp0s3
        valid_lft 76657sec preferred_lft 76657sec
    inet6 fe80::a00:27ff:fe00:ee/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:f0:4b:99 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:fe00:4b/64 scope link
        valid_lft forever preferred_lft forever
[ S:5505 ] [ kunru@upc1:~ ]
$ ip route
default via 192.168.1.254 dev enp0s8 proto static
default via 10.0.2.2 dev enp0s3 proto dhcp src 10.0.2.15 metric 100
10.0.2.0/24 dev enp0s3 proto kernel scope link src 10.0.2.15
10.0.2.2 dev enp0s3 proto dhcp scope link src 10.0.2.15 metric 100
192.168.0.0/16 via 192.168.1.254 dev enp0s8 proto static
192.168.1.0/24 dev enp0s8 proto kernel scope link src 192.168.1.1
[ S:5505 ] [ kunru@upc1:~ ]
$
```

Figure 34: Ubuntu Desktop1 (uPC1) ip a, ip route Komut çıktısı

```
tmux
File Edit View Search Terminal Help
[ S:5605 ] [ kunru@upc2:~ ]
$ ip 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 08:00:27:03:69:d6 brd ff:ff:ff:ff:ff:ff
    inet 10.0.2.15/24 brd 10.0.2.255 scope global dynamic enp0s3
        valid_lft 76502sec preferred_lft 76502sec
    inet6 fe80::a00:27ff:fe03:69/64 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 08:00:27:18:c4:e4 brd ff:ff:ff:ff:ff:ff
    inet 192.168.2.1/24 brd 192.168.2.255 scope global enp0s8
        valid_lft forever preferred_lft forever
    inet6 fe80::a00:27ff:fe18:c4/64 scope link
        valid_lft forever preferred_lft forever
[ S:5605 ] [ kunru@upc2:~ ]
$ ip route
default via 192.168.2.254 dev enp0s8 proto static
default via 10.0.2.2 dev enp0s3 proto dhcp src 10.0.2.15 metric 100
10.0.2.0/24 dev enp0s3 proto kernel scope link src 10.0.2.15
10.0.2.2 dev enp0s3 proto dhcp scope link src 10.0.2.15 metric 100
192.168.2.0/24 dev enp0s8 proto kernel scope link src 192.168.2.1
[ S:5605 ] [ kunru@upc2:~ ]
$
```

Figure 35: Ubuntu Desktop2 (uPC2) ip a, ip route Komut çıktısı

traceroute, paket takibi

Traceroute komutu ile paketlerin izlediği yolun takibini yapabiliriz.

```
tmux
File Edit View Search Terminal Help
[ 3:470S ] [ kumru@upc1:~ ]
$ ip a | grep "inet "
    inet 127.0.0.1/8 scope host lo
    inet 10.0.2.15/24 brd 10.0.2.255 scope global dynamic enp0s3
    inet 192.168.1.1/24 brd 192.168.1.255 scope global enp0s8
[ 3:470S ] [ kumru@upc1:~ ]
$ traceroute 192.168.2.1
traceroute to 192.168.2.1 (192.168.2.1), 30 hops max, 60 byte packets
 1 _gateway (192.168.1.254)  0.378 ms  0.325 ms  0.234 ms
 2 192.168.100.2 (192.168.100.2)  0.818 ms  0.693 ms  0.599 ms
 3 192.168.2.1 (192.168.2.1)  1.165 ms  1.166 ms  1.093 ms
[ 3:480S ] [ kumru@upc1:~ ]
$ traceroute 192.168.1.254
traceroute to 192.168.1.254 (192.168.1.254), 30 hops max, 60 byte packets
 1 _gateway (192.168.1.254)  0.389 ms  0.334 ms  0.317 ms
[ 3:560S ] [ kumru@upc1:~ ]
$ traceroute 192.168.2.254
traceroute to 192.168.2.254 (192.168.2.254), 30 hops max, 60 byte packets
 1 _gateway (192.168.1.254)  0.333 ms  0.275 ms  0.255 ms
 2 192.168.2.254 (192.168.2.254)  0.788 ms  0.757 ms  0.730 ms
[ 3:560S ] [ kumru@upc1:~ ]
$
[0] 1:~* "upc1" 15:57 28-Haz-20
```

Figure 36: uPC1 -> uPC2, uPC1 -> userver1, uPC1 -> userver2