

# IP routing

1. Create virtual machines connection according to figure 1:

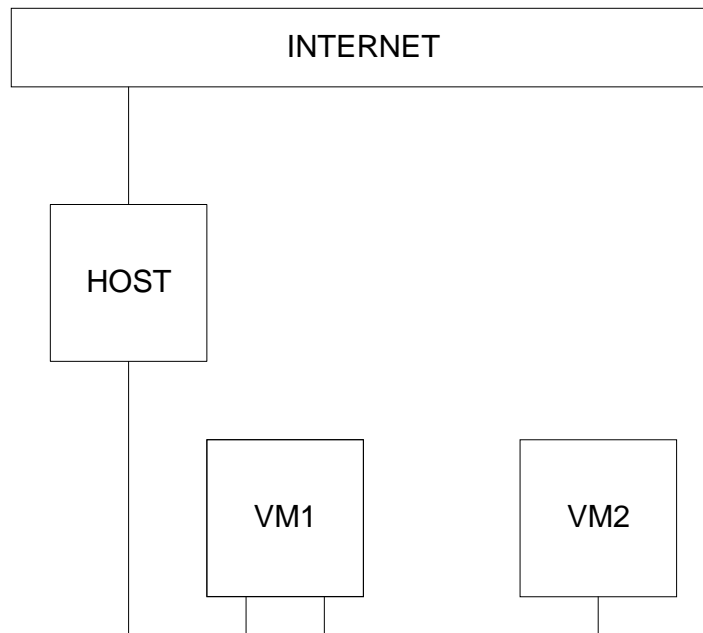
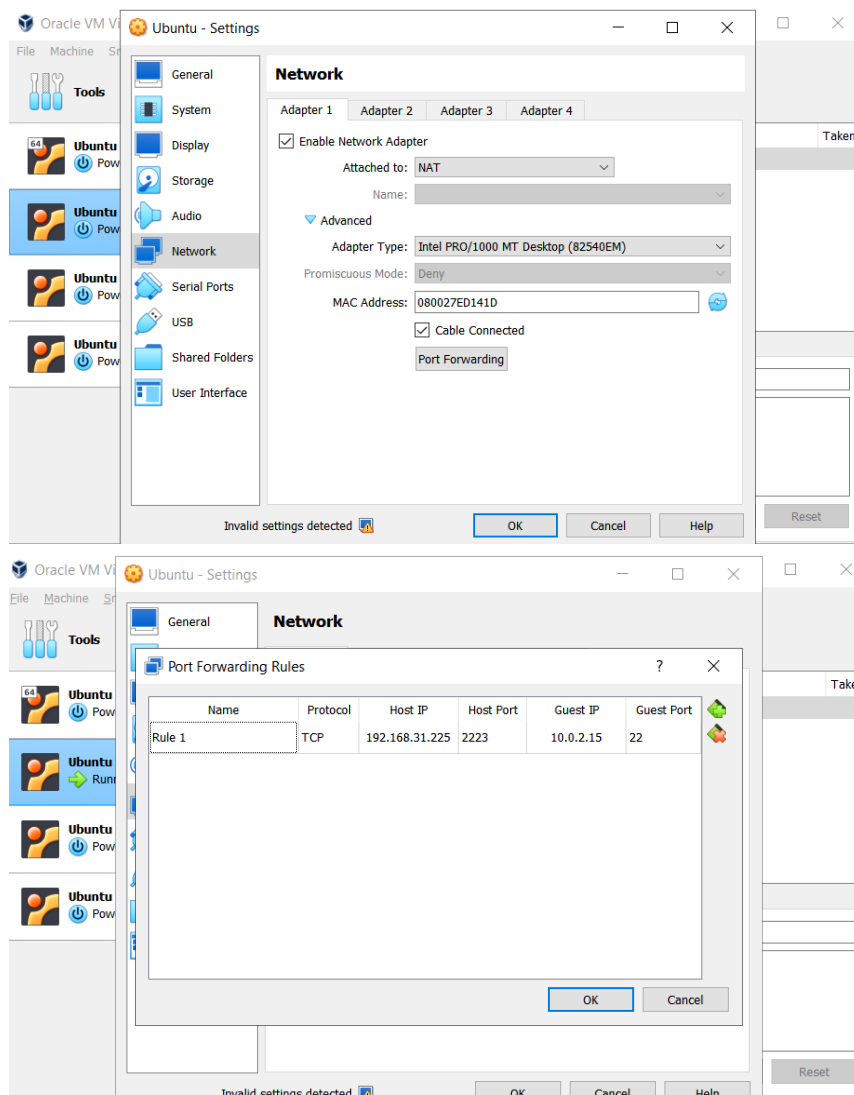
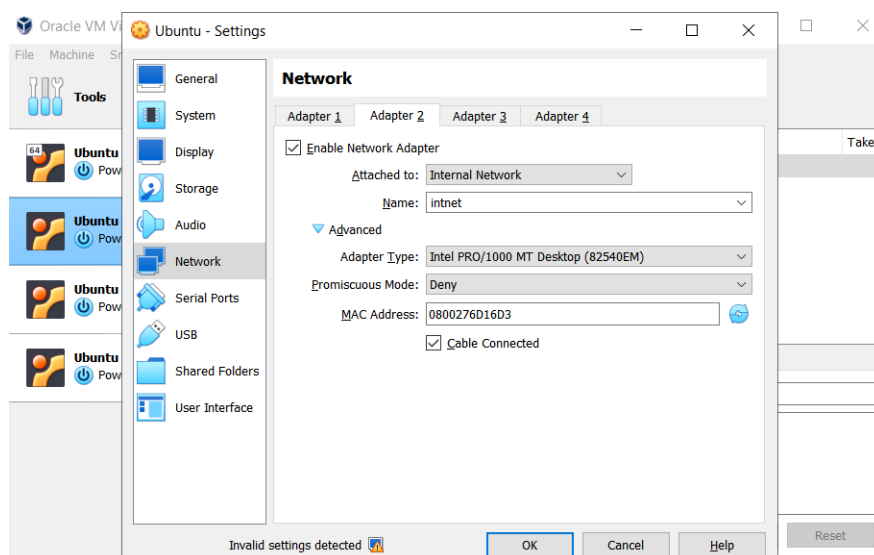


Figure 1 – VMs connection

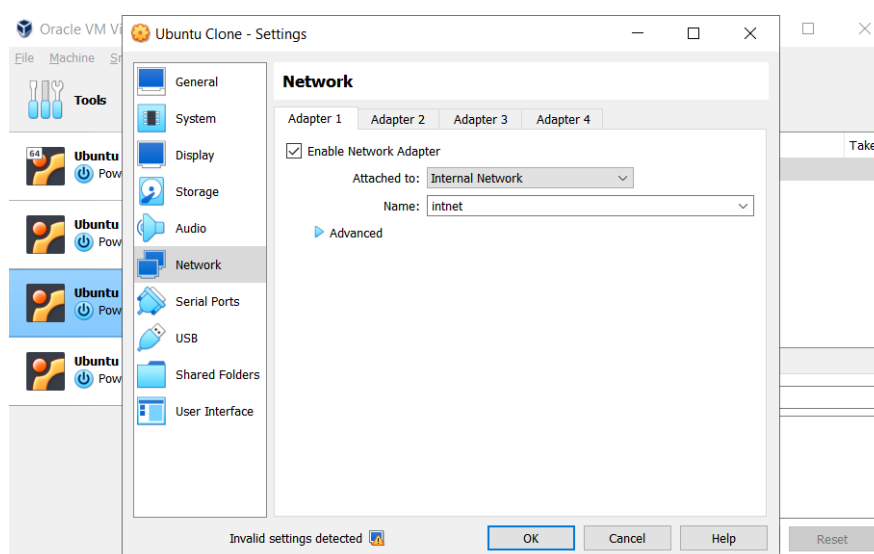
## VM1 NAT:



## VM1 Internal Network:



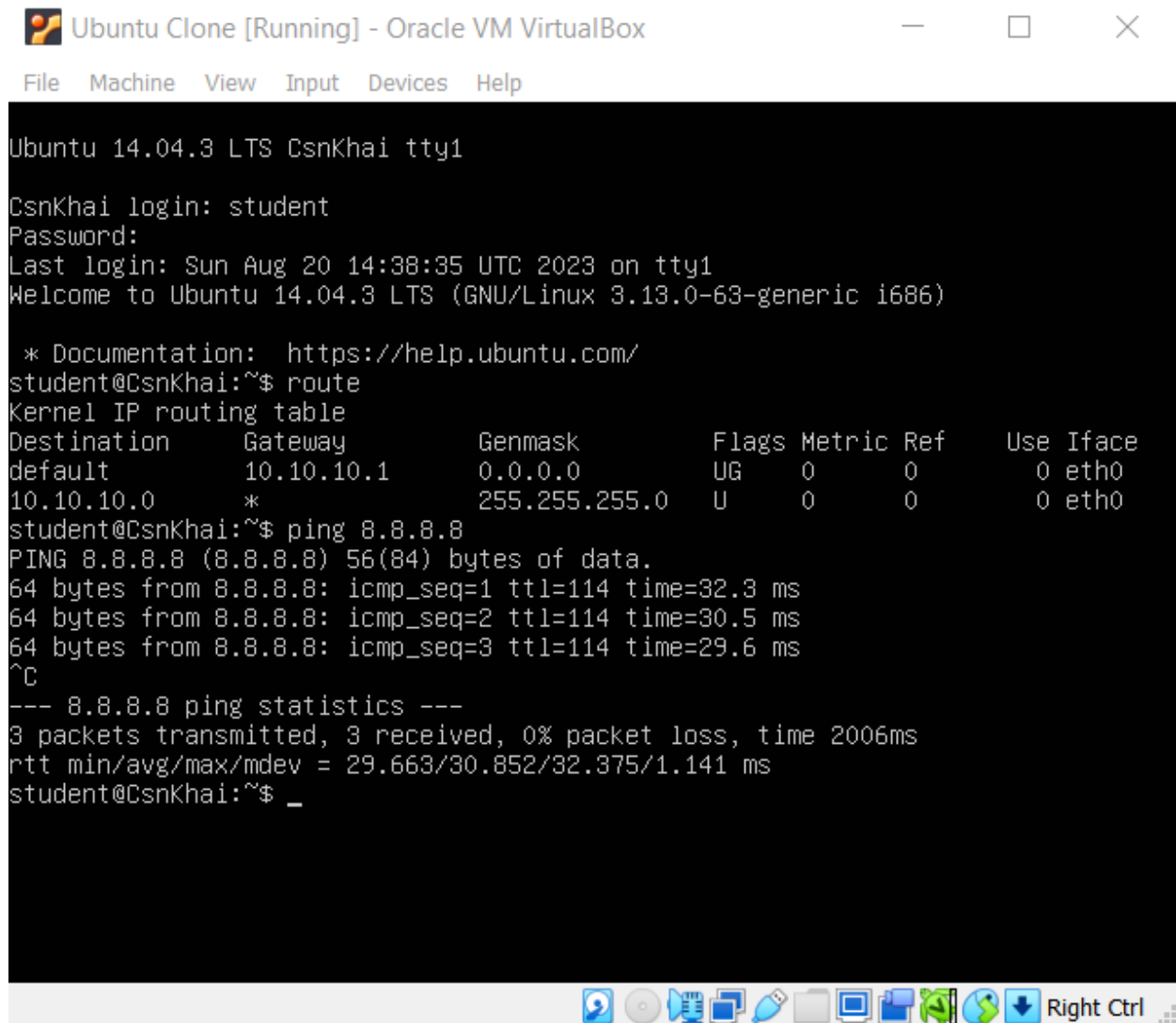
## VM2 Internal Network:



```
GNU nano 2.2.6 File: /

127.0.0.1    localhost
127.0.1.1    CsnKhai
192.168.0.120 MyVMServer

# The following lines are desirable for IPv6 capable hosts
::1        localhost ip6-localhost ip6-loopback
ff02::1    ip6-allnodes
ff02::2    ip6-allrouters
```

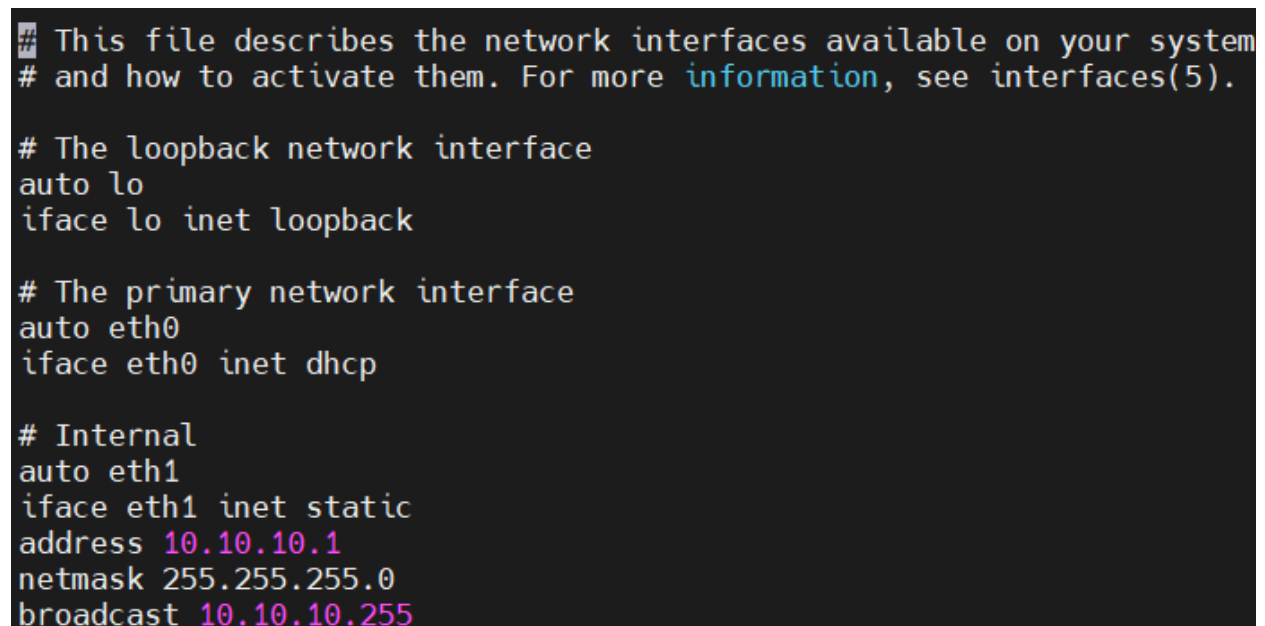


```
Ubuntu 14.04.3 LTS CsnKhai tty1
CsnKhai login: student
Password:
Last login: Sun Aug 20 14:38:35 UTC 2023 on tty1
Welcome to Ubuntu 14.04.3 LTS (GNU/Linux 3.13.0-63-generic i686)

 * Documentation:  https://help.ubuntu.com/
student@CsnKhai:~$ route
Kernel IP routing table
Destination      Gateway          Genmask          Flags Metric Ref    Use Iface
default          10.10.10.1       0.0.0.0          UG    0      0      0 eth0
10.10.10.0        *                255.255.255.0    U     0      0      0 eth0
student@CsnKhai:~$ ping 8.8.8.8
PING 8.8.8.8 (8.8.8.8) 56(84) bytes of data.
64 bytes from 8.8.8.8: icmp_seq=1 ttl=114 time=32.3 ms
64 bytes from 8.8.8.8: icmp_seq=2 ttl=114 time=30.5 ms
64 bytes from 8.8.8.8: icmp_seq=3 ttl=114 time=29.6 ms
^C
--- 8.8.8.8 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2006ms
rtt min/avg/max/mdev = 29.663/30.852/32.375/1.141 ms
student@CsnKhai:~$ _
```

2. VM2 has one interface (internal), VM1 has 2 interfaces (NAT and internal). Configure all network interfaces in order to make VM2 has an access to the Internet (iptables, forward, masquerade).

Adding Internal interface to VM1:



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

# The loopback network interface
auto lo
iface lo inet loopback

# The primary network interface
auto eth0
iface eth0 inet dhcp

# Internal
auto eth1
iface eth1 inet static
address 10.10.10.1
netmask 255.255.255.0
broadcast 10.10.10.255
```

## Adding ip forwarding in VM1:

```
GNU nano 2.2.6 File: /etc/sysctl.conf
#
# /etc/sysctl.conf - Configuration file for setting system variables
# See /etc/sysctl.d/ for additional system variables.
# See sysctl.conf(5) for information.
#
#kernel.domainname = example.com
# Uncomment the following to stop low-level messages on console
#kernel.printk = 3 4 1 3
#####3
# Functions previously found in netbase
#
# Uncomment the next two lines to enable Spoof protection (reverse-path filter)
# Turn on Source Address Verification in all interfaces to
# prevent some spoofing attacks
#net.ipv4.conf.default.rp_filter=1
#net.ipv4.conf.all.rp_filter=1
#
# Uncomment the next line to enable TCP/IP SYN cookies
# See http://lwn.net/Articles/27746/
# Note: This may impact IPv6 TCP sessions too
#net.ipv4.tcp_syncookies=1
#
# Uncomment the next line to enable packet forwarding for IPv4
net.ipv4.ip_forward=1
#
# Uncomment the next line to enable packet forwarding for IPv6
# Enabling this option disables Stateless Address Autoconfiguration
# based on Router Advertisements for this host
#net.ipv6.conf.all.forwarding=1
#####
# Additional settings - these settings can improve the network
# security of the host and prevent against some network attacks
# including spoofing attacks and man in the middle attacks through
# redirection. Some network environments, however, require that these
#
^G Get Help ^O WriteOut ^R Read File ^Y Prev Page ^K Cut Text ^C Cur Pos
^X Exit ^J Justify ^W Where Is ^V Next Page ^U UnCut Text ^T To Spell

ort MobaXterm by subscribing to the professional edition here: https://mobaxterm.mobatek.net
```

## Checking availability of iptables:

```
student@CsnKhai:~$ iptables
iptables v1.4.21: no command specified
Try `iptables -h' or 'iptables --help' for more information.
```

## Configuring interfaces with iptables:

```
student@CsnKhai:~$ sudo iptables -F
-P INPUT ACCEPT
-P FORWARD ACCEPT
-P OUTPUT ACCEPT
student@CsnKhai:~$ sudo iptables -t nat -A POSTROUTING -o eth0 -j MASQUERADE
student@CsnKhai:~$ sudo iptables -A FORWARD -i eth1 -o eth0 -m state --state REL
ATED,ESTABLISHED -j ACCEPT
student@CsnKhai:~$ sudo iptables -A FORWARD -i eth1 -o eth0 -j ACCEPT
student@CsnKhai:~$ sudo iptables -S
-P INPUT ACCEPT
-P FORWARD ACCEPT
-P OUTPUT ACCEPT
-A FORWARD -i eth1 -o eth0 -m state --state RELATED,ESTABLISHED -j ACCEPT
-A FORWARD -i eth1 -o eth0 -j ACCEPT
```

## Editing eth0 for Internal Network in VM2:

```
Ubuntu Clone [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
GNU nano 2.2.6 File: /etc/network/interfaces

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

# The loopback network interface
auto lo
iface lo inet loopback

# The primary network interface
auto eth0
iface eth0 inet static
address 10.10.10.2
netmask 255.255.255.0
broadcast 10.10.10.255
gateway 10.10.10.1

[ Read 14 lines ]
^G Get Help ^O WriteOut ^R Read File ^Y Prev Page ^K Cut Text ^C Cur Pos
^X Exit ^J Justify ^W Where Is ^V Next Page ^U UnCut Text ^T To Spell
```

3. Check the route from VM2 to Host.

```
student@CsnKhai:~$ traceroute 192.168.31.225
traceroute to 192.168.31.225 (192.168.31.225), 30 hops max, 60 byte packets
 1  10.10.10.1 (10.10.10.1)  1.132 ms  1.145 ms  1.141 ms
 2  10.0.2.2 (10.0.2.2)    2.295 ms  1.557 ms  2.808 ms
 3  10.0.2.2 (10.0.2.2)    1.657 ms  3.666 ms  2.696 ms
```

4. Check the access to the Internet, (just ping, for example, 8.8.8.8).

VM1:

```
student@CsnKhai:~$ ping 8.8.8.8
PING 8.8.8.8 (8.8.8.8) 56(84) bytes of data.
64 bytes from 8.8.8.8: icmp_seq=1 ttl=115 time=29.1 ms
64 bytes from 8.8.8.8: icmp_seq=2 ttl=115 time=28.4 ms
64 bytes from 8.8.8.8: icmp_seq=3 ttl=115 time=29.9 ms
64 bytes from 8.8.8.8: icmp_seq=4 ttl=115 time=28.6 ms
64 bytes from 8.8.8.8: icmp_seq=5 ttl=115 time=31.0 ms
^C
--- 8.8.8.8 ping statistics ---
5 packets transmitted, 5 received, 0% packet loss, time 4008ms
rtt min/avg/max/mdev = 28.433/29.439/31.056/0.966 ms
student@CsnKhai:~$
```

VM2:

```
student@CsnKhai:~$ ping 8.8.8.8
PING 8.8.8.8 (8.8.8.8) 56(84) bytes of data.
64 bytes from 8.8.8.8: icmp_seq=1 ttl=114 time=29.9 ms
64 bytes from 8.8.8.8: icmp_seq=2 ttl=114 time=30.4 ms
64 bytes from 8.8.8.8: icmp_seq=3 ttl=114 time=30.0 ms
64 bytes from 8.8.8.8: icmp_seq=4 ttl=114 time=30.3 ms
64 bytes from 8.8.8.8: icmp_seq=5 ttl=114 time=30.1 ms
^C
--- 8.8.8.8 ping statistics ---
5 packets transmitted, 5 received, 0% packet loss, time 4006ms
rtt min/avg/max/mdev = 29.993/30.192/30.474/0.258 ms
student@CsnKhai:~$
```

5. Determine, which resource has an IP address 8.8.8.8.

```
student@CsnKhai:~$ host 8.8.8.8
8.8.8.8.in-addr.arpa domain name pointer dns.google.
```

6. Determine, which IP address belongs to resource epam.com.

```
student@CsnKhai:~$ host epam.com
epam.com has address 3.214.134.159
epam.com mail is handled by 10 mxa-0039f301.gslb.pphosted.com.
epam.com mail is handled by 10 mxh-0039f301.gslb.pphosted.com.
```

7. Determine the default gateway for your HOST and display routing table.

```
student@CsnKhai:~$ route -n
Kernel IP routing table
Destination      Gateway          Genmask         Flags Metric Ref    Use Iface
0.0.0.0          10.0.2.2        0.0.0.0         UG    0      0        0 eth0
10.0.2.0         0.0.0.0         255.255.255.0   U      0      0        0 eth0
10.10.10.0       0.0.0.0         255.255.255.0   U      0      0        0 eth1
```

8. Trace the route to google.com.

```
student@CsnKhai:~$ traceroute google.com
traceroute to google.com (142.250.75.14), 30 hops max, 60 byte packets
 1  10.10.10.1 (10.10.10.1)  1.547 ms  0.855 ms  1.392 ms
 2  10.0.2.2 (10.0.2.2)    2.427 ms  3.424 ms  3.163 ms
 3  10.0.2.2 (10.0.2.2)    4.482 ms  3.649 ms  4.398 ms
student@CsnKhai:~$
```