Made By Hao Ding

hd945

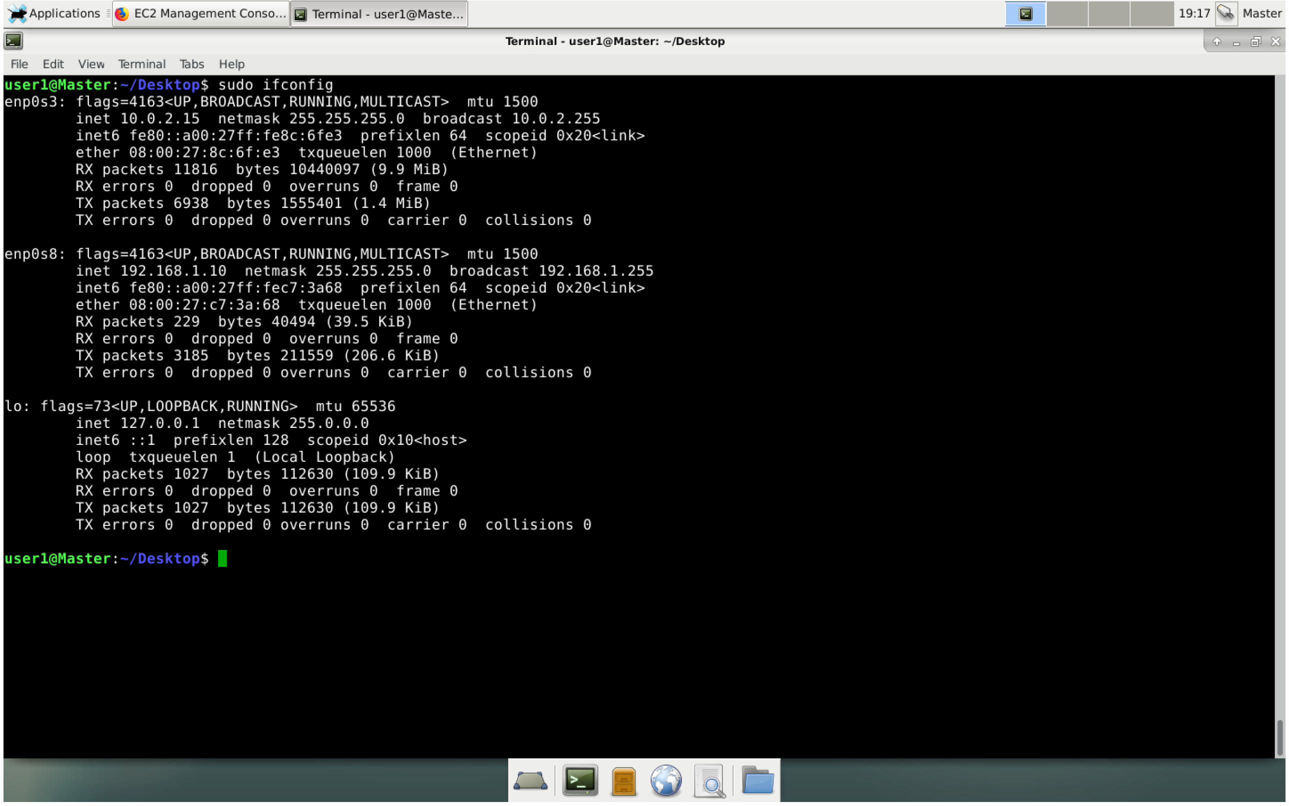
12/1/2018

# 3530FinalProject

## Manual process:

### On Debian

```
#Add users to Debian.
 sudo adduser user1
#Change host name.
 sudo hostnamectl set-hostname Master
 sudo reboot
#Add them to soduers.
 sudo visudo
#Give them new ip address.
 sudo ifconfig enp0s8 192.168.1.10 netmask 255.255.255.0
#Show its ip adress.
 sudo ifconfig
```

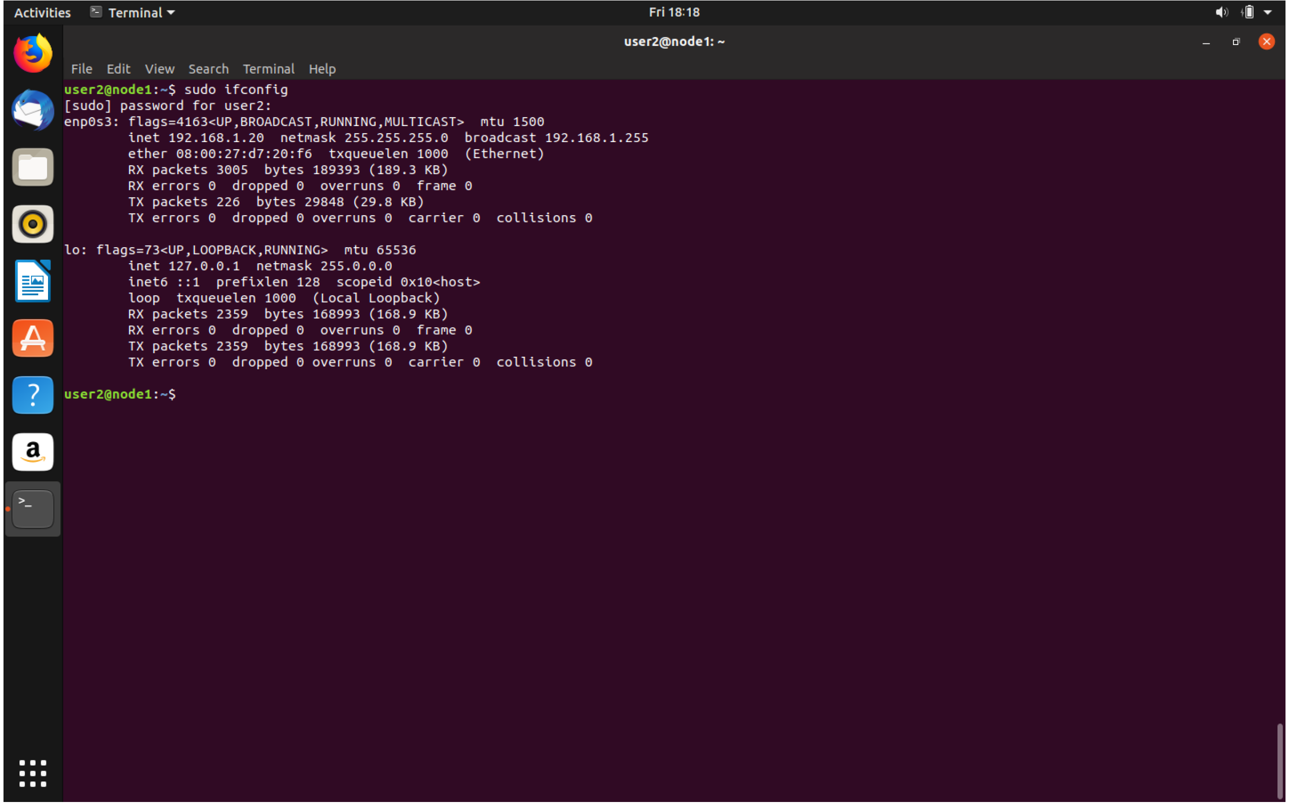

Applications
EC2 Management Conso...
Terminal - user1@Maste...
19:17
Master
Terminal - user1@Master: ~/Desktop
File Edit View Terminal Tabs Help

```
user1@Master:~/Desktop$ sudo ifconfig
enp0s3: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        inet 10.0.2.15  netmask 255.255.255.0  broadcast 10.0.2.255
        inet6 fe80::a00:27ff:fe8c:6fe3  prefixlen 64  scopeid 0x20<link>
        ether 08:00:27:8c:6f:e3  txqueuelen 1000  (Ethernet)
        RX packets 11816  bytes 10440097 (9.9 MiB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 6938  bytes 1555401 (1.4 MiB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

enp0s8: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        inet 192.168.1.10  netmask 255.255.255.0  broadcast 192.168.1.255
        inet6 fe80::a00:27ff:fec7:3a68  prefixlen 64  scopeid 0x20<link>
        ether 08:00:27:c7:3a:68  txqueuelen 1000  (Ethernet)
        RX packets 229  bytes 40494 (39.5 KiB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 3185  bytes 211559 (206.6 KiB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

lo: flags=73<UP,LOOPBACK,RUNNING>  mtu 65536
        inet 127.0.0.1  netmask 255.0.0.0
        inet6 ::1  prefixlen 128  scopeid 0x10<host>
        loop  txqueuelen 1  (Local Loopback)
        RX packets 1027  bytes 112630 (109.9 KiB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 1027  bytes 112630 (109.9 KiB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

user1@Master:~/Desktop$
```

### On Ubuntu

```
#Add users to Ubuntu.
 sudo adduser user2
#Change host name.
 sudo hostnamectl set-hostname node1
 sudo reboot
#Add them to soduers.
 sudo visudo
```

```
#Give them new ip address.
 sudo ifconfig enp0s3 192.168.1.20 netmask 255.255.255.0
#Show its ip adress.
 sudo ifconfig
```

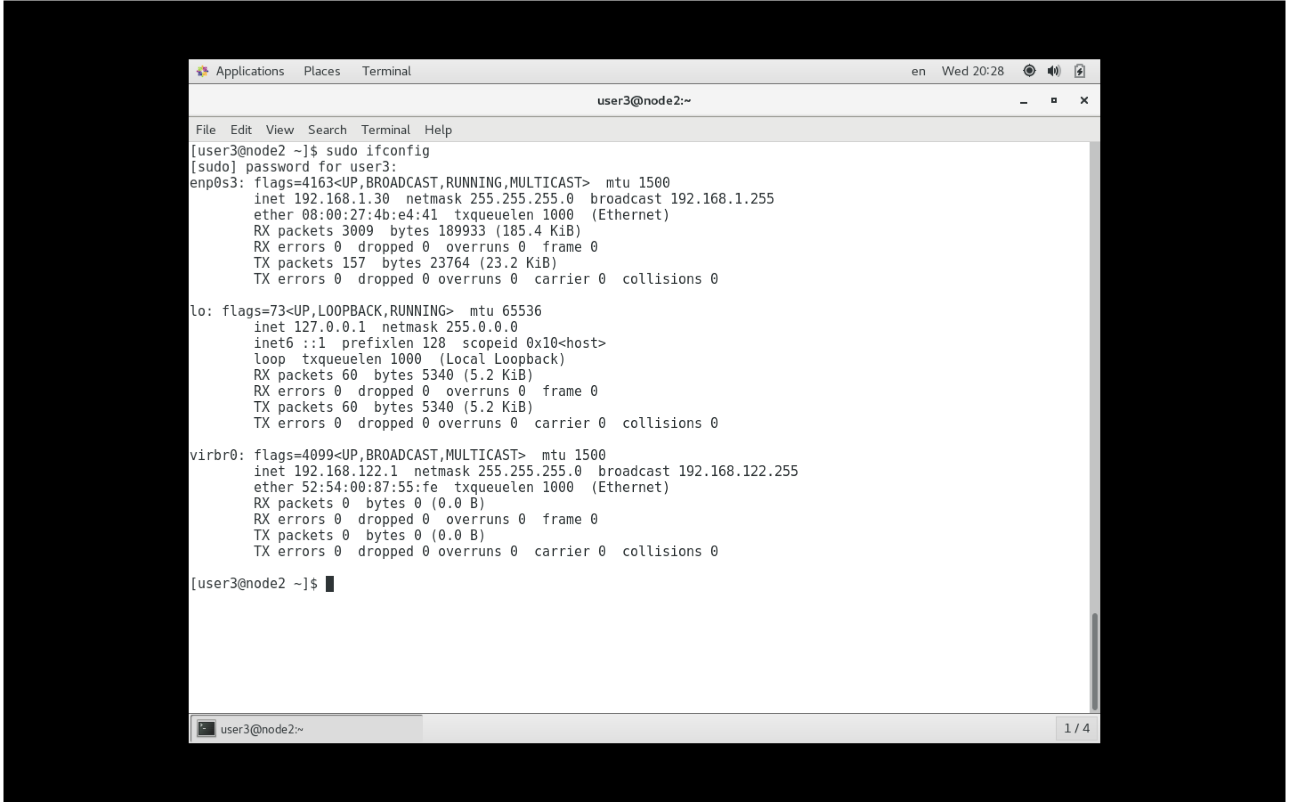


```
user2@node1:~$ sudo ifconfig
[sudo] password for user2:
enp0s3: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        inet 192.168.1.20  netmask 255.255.255.0  broadcast 192.168.1.255
        ether 08:00:27:d7:20:f6  txqueuelen 1000  (Ethernet)
        RX packets 3005  bytes 189393 (189.3 KB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 226  bytes 29848 (29.8 KB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

lo: flags=73<UP,LOOPBACK,RUNNING>  mtu 65536
        inet 127.0.0.1  netmask 255.0.0.0
        inet6 ::1  prefixlen 128  scopeid 0x10<host>
        loop  txqueuelen 1000  (Local Loopback)
        RX packets 2359  bytes 168993 (168.9 KB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 2359  bytes 168993 (168.9 KB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

user2@node1:~$
```

### On Centos

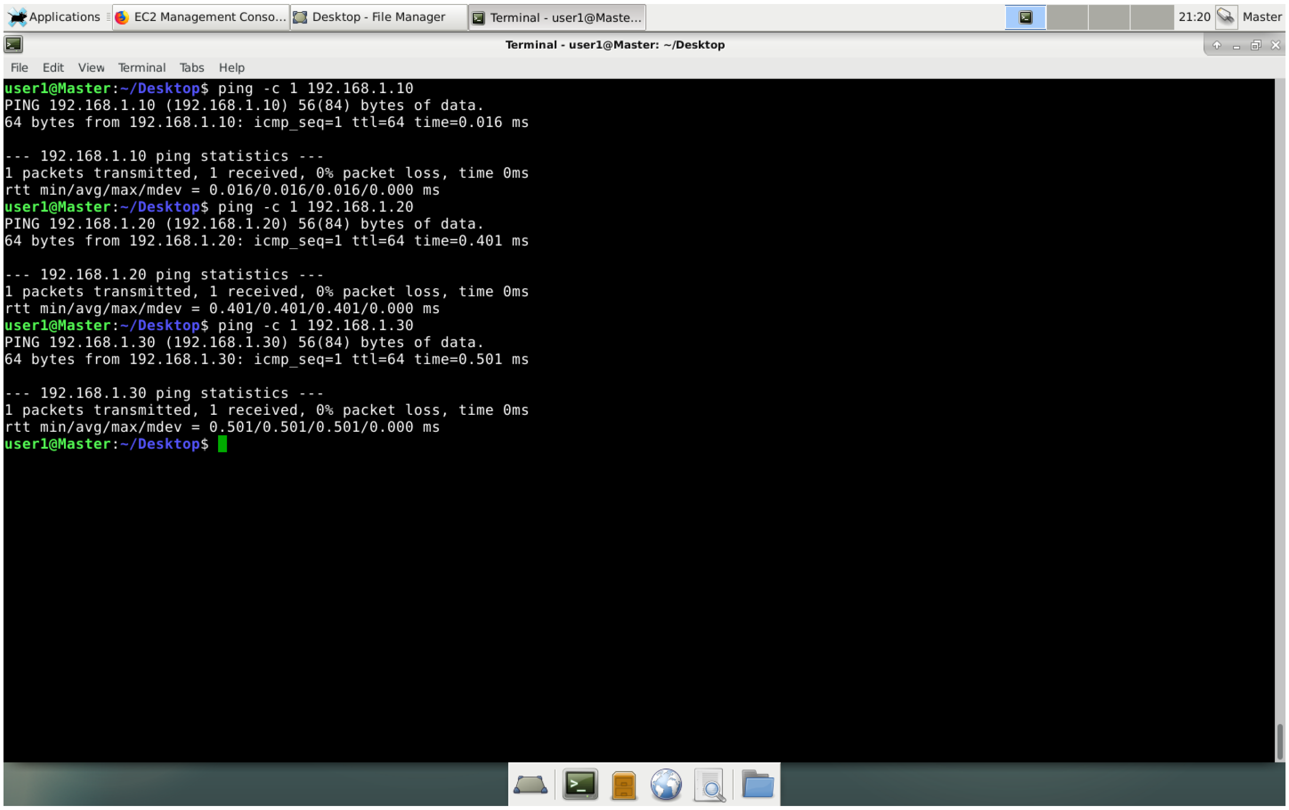
```
#Add users to Centos.
 sudo adduser user3
#Change host name.
 sudo hostnamectl set-hostname node2
 sudo reboot
#Add them to soduers.
 sudo visudo
#Give them new ip address.
 sudo ifconfig enp0s3 192.168.1.30 netmask 255.255.255.0
#Show its ip adress.
```

```
sudo ifconfig
```



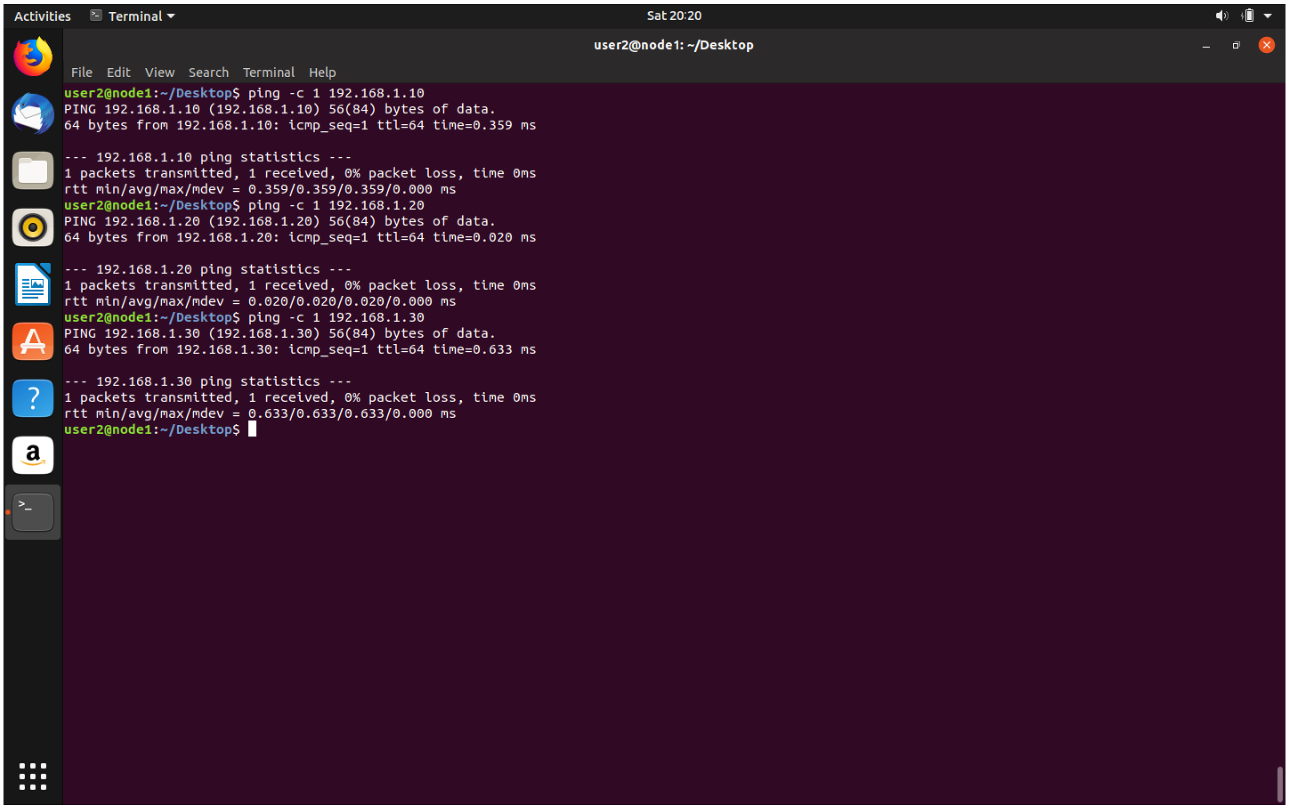
```
[user3@node2 ~]$ sudo ifconfig
[sudo] password for user3:
enp0s3: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        inet 192.168.1.30  netmask 255.255.255.0  broadcast 192.168.1.255
        ether 08:00:27:4b:e4:41  txqueuelen 1000  (Ethernet)
        RX packets 3009  bytes 189933 (185.4 KiB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 157  bytes 23764 (23.2 KiB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

lo: flags=73<UP,LOOPBACK,RUNNING>  mtu 65536
        inet 127.0.0.1  netmask 255.0.0.0
        inet6 ::1  prefixlen 128  scopeid 0x10<host>
        loop  txqueuelen 1000  (Local Loopback)
        RX packets 60  bytes 5340 (5.2 KiB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 60  bytes 5340 (5.2 KiB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

virbr0: flags=4099<UP,BROADCAST,MULTICAST>  mtu 1500
        inet 192.168.122.1  netmask 255.255.255.0  broadcast 192.168.122.255
        ether 52:54:00:87:55:fe  txqueuelen 1000  (Ethernet)
        RX packets 0  bytes 0 (0.0 B)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 0  bytes 0 (0.0 B)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

[user3@node2 ~]$
```

```
#After each host has a right ip adress, ping each other.
ping -c 1 each_ip
#Master ping:
```



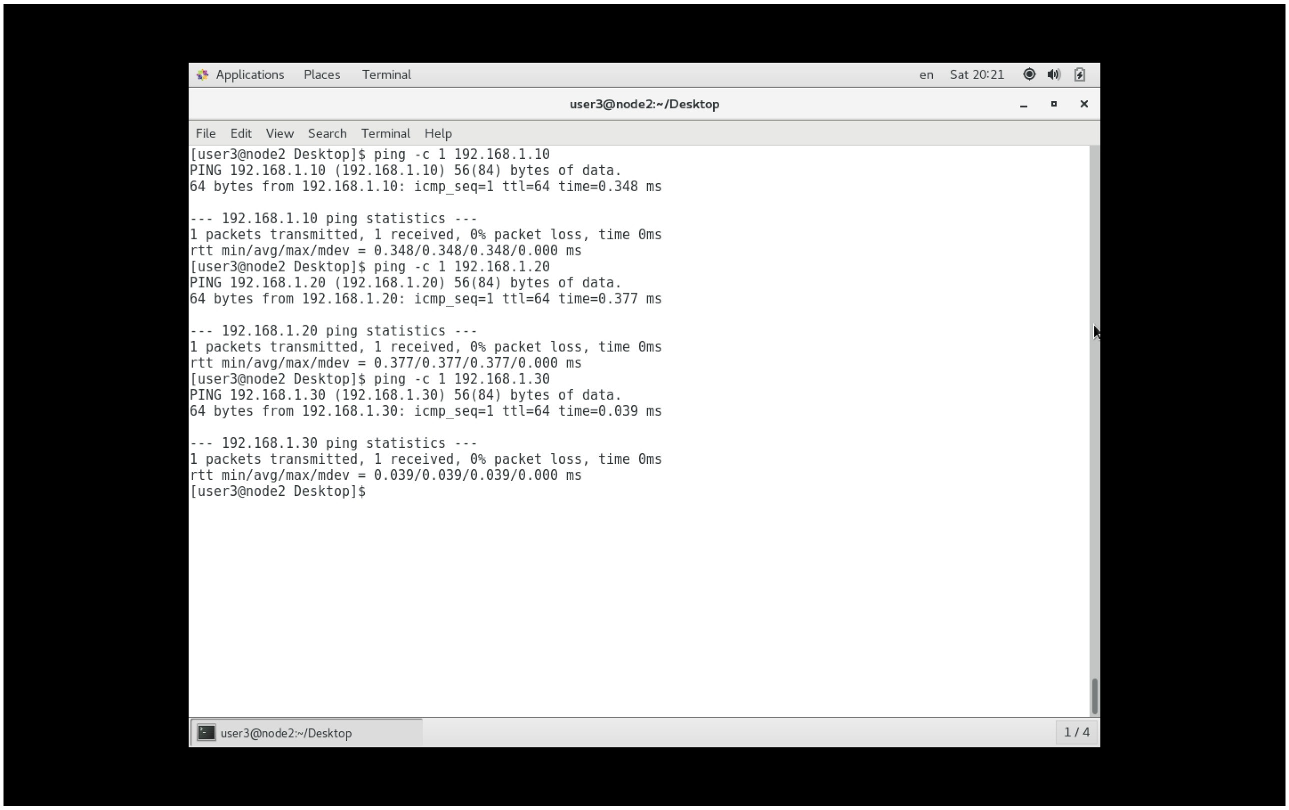
```
user1@Master:~/Desktop$ ping -c 1 192.168.1.10
PING 192.168.1.10 (192.168.1.10) 56(84) bytes of data.
64 bytes from 192.168.1.10: icmp_seq=1 ttl=64 time=0.016 ms

--- 192.168.1.10 ping statistics ---
1 packets transmitted, 1 received, 0% packet loss, time 0ms
rtt min/avg/max/mdev = 0.016/0.016/0.016/0.000 ms
user1@Master:~/Desktop$ ping -c 1 192.168.1.20
PING 192.168.1.20 (192.168.1.20) 56(84) bytes of data.
64 bytes from 192.168.1.20: icmp_seq=1 ttl=64 time=0.401 ms

--- 192.168.1.20 ping statistics ---
1 packets transmitted, 1 received, 0% packet loss, time 0ms
rtt min/avg/max/mdev = 0.401/0.401/0.401/0.000 ms
user1@Master:~/Desktop$ ping -c 1 192.168.1.30
PING 192.168.1.30 (192.168.1.30) 56(84) bytes of data.
64 bytes from 192.168.1.30: icmp_seq=1 ttl=64 time=0.501 ms

--- 192.168.1.30 ping statistics ---
1 packets transmitted, 1 received, 0% packet loss, time 0ms
rtt min/avg/max/mdev = 0.501/0.501/0.501/0.000 ms
user1@Master:~/Desktop$
```

```
#Node1 ping:
```



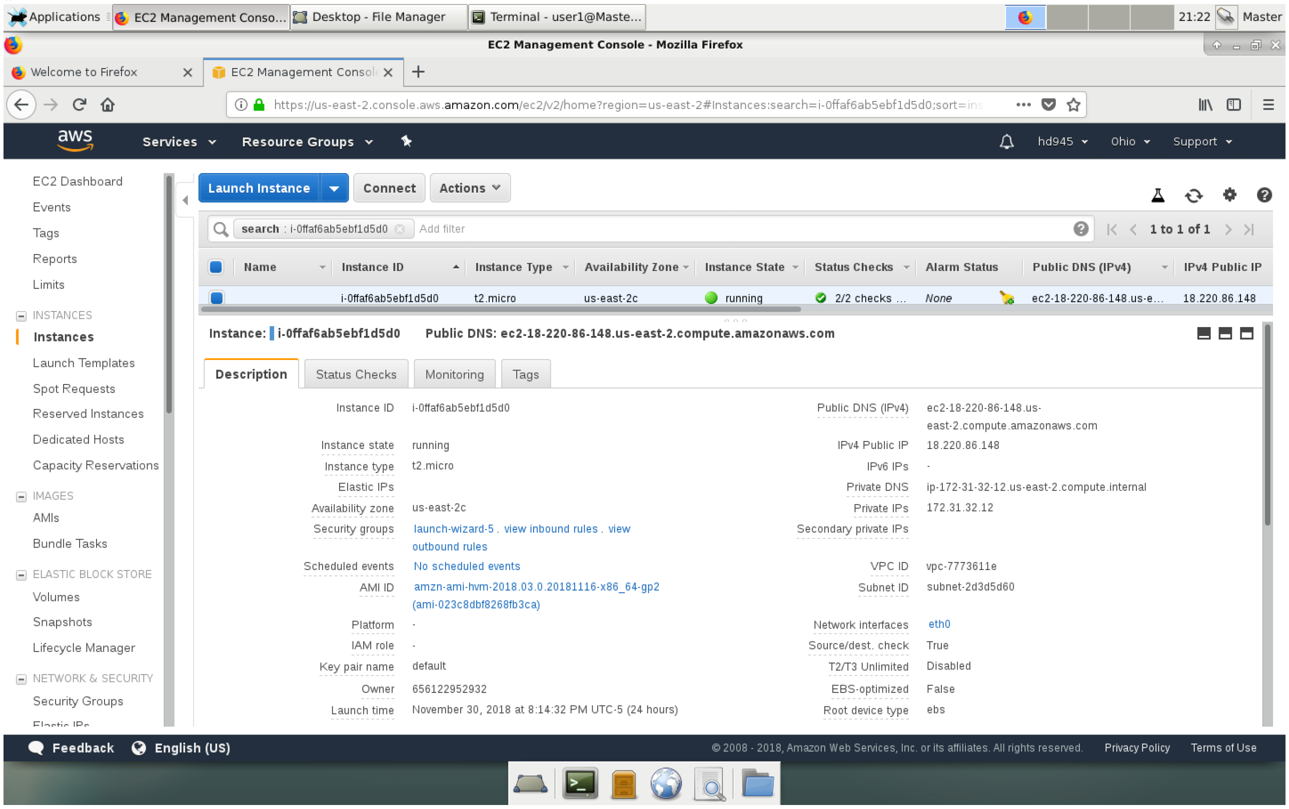
```
user2@node1:~/Desktop$ ping -c 1 192.168.1.10
PING 192.168.1.10 (192.168.1.10) 56(84) bytes of data.
64 bytes from 192.168.1.10: icmp_seq=1 ttl=64 time=0.359 ms

--- 192.168.1.10 ping statistics ---
1 packets transmitted, 1 received, 0% packet loss, time 0ms
rtt min/avg/max/mdev = 0.359/0.359/0.359/0.000 ms
user2@node1:~/Desktop$ ping -c 1 192.168.1.20
PING 192.168.1.20 (192.168.1.20) 56(84) bytes of data.
64 bytes from 192.168.1.20: icmp_seq=1 ttl=64 time=0.020 ms

--- 192.168.1.20 ping statistics ---
1 packets transmitted, 1 received, 0% packet loss, time 0ms
rtt min/avg/max/mdev = 0.020/0.020/0.020/0.000 ms
user2@node1:~/Desktop$ ping -c 1 192.168.1.30
PING 192.168.1.30 (192.168.1.30) 56(84) bytes of data.
64 bytes from 192.168.1.30: icmp_seq=1 ttl=64 time=0.633 ms

--- 192.168.1.30 ping statistics ---
1 packets transmitted, 1 received, 0% packet loss, time 0ms
rtt min/avg/max/mdev = 0.633/0.633/0.633/0.000 ms
user2@node1:~/Desktop$
```

#Node2 ping:


Applications Places Terminal
en Sat 20:21
user3@node2:~/Desktop
File Edit View Search Terminal Help
user3@node2:~/Desktop
1 / 4

```
[user3@node2 Desktop]$ ping -c 1 192.168.1.10
PING 192.168.1.10 (192.168.1.10) 56(84) bytes of data.
64 bytes from 192.168.1.10: icmp_seq=1 ttl=64 time=0.348 ms

--- 192.168.1.10 ping statistics ---
1 packets transmitted, 1 received, 0% packet loss, time 0ms
rtt min/avg/max/mdev = 0.348/0.348/0.348/0.000 ms
[user3@node2 Desktop]$ ping -c 1 192.168.1.20
PING 192.168.1.20 (192.168.1.20) 56(84) bytes of data.
64 bytes from 192.168.1.20: icmp_seq=1 ttl=64 time=0.377 ms

--- 192.168.1.20 ping statistics ---
1 packets transmitted, 1 received, 0% packet loss, time 0ms
rtt min/avg/max/mdev = 0.377/0.377/0.377/0.000 ms
[user3@node2 Desktop]$ ping -c 1 192.168.1.30
PING 192.168.1.30 (192.168.1.30) 56(84) bytes of data.
64 bytes from 192.168.1.30: icmp_seq=1 ttl=64 time=0.039 ms

--- 192.168.1.30 ping statistics ---
1 packets transmitted, 1 received, 0% packet loss, time 0ms
rtt min/avg/max/mdev = 0.039/0.039/0.039/0.000 ms
[user3@node2 Desktop]$
```

```
#Create an AWS instance.
```



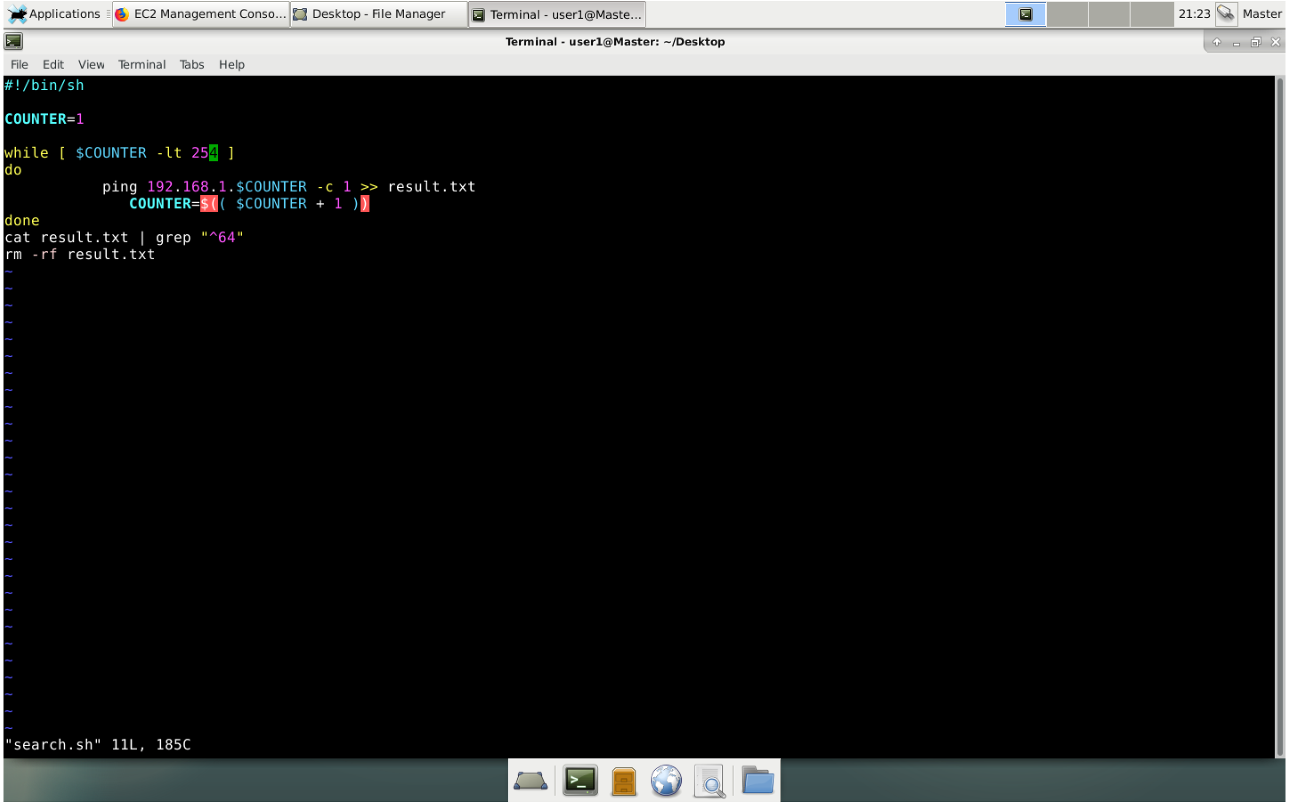
| Name | Instance ID | Instance Type | Availability Zone | Instance State | Status Checks | Alarm Status | Public DNS (IPv4) | IPv4 Public IP |
|---|---|---|---|---|---|---|---|---|
| | i-0ffaf6ab5ebf1d5d0 | t2.micro | us-east-2c | ● running | ✓ 2/2 checks ... | None | ec2-18-220-86-148.us-e... | 18.220.86.148 |

Instance: i-0ffaf6ab5ebf1d5d0 Public DNS: ec2-18-220-86-148.us-east-2.compute.amazonaws.com

| | | | |
|---|---|---|---|
| Instance ID | i-0ffaf6ab5ebf1d5d0 | Public DNS (IPv4) | ec2-18-220-86-148.us-east-2.compute.amazonaws.com |
| Instance state | running | IPv4 Public IP | 18.220.86.148 |
| Instance type | t2.micro | IPv6 IPs | - |
| Elastic IPs | | Private DNS | ip-172-31-32-12.us-east-2.compute.internal |
| Availability zone | us-east-2c | Private IPs | 172.31.32.12 |
| Security groups | launch-wizard-5 . view inbound rules . view outbound rules | Secondary private IPs | |
| Scheduled events | No scheduled events | VPC ID | vpc-7773611e |
| AMI ID | amzn-ami-hvm-2018.03.0.20181116-x86_64-gp2 (ami-023c8dbf8268fb3ca) | Subnet ID | subnet-2d3d5d60 |
| Platform | - | Network interfaces | eth0 |
| IAM role | - | Source/dest. check | True |
| Key pair name | default | T2/T3 Unlimited | Disabled |
| Owner | 656122952932 | EBS-optimized | False |
| Launch time | November 30, 2018 at 8:14:32 PM UTC-5 (24 hours) | Root device type | ebs |

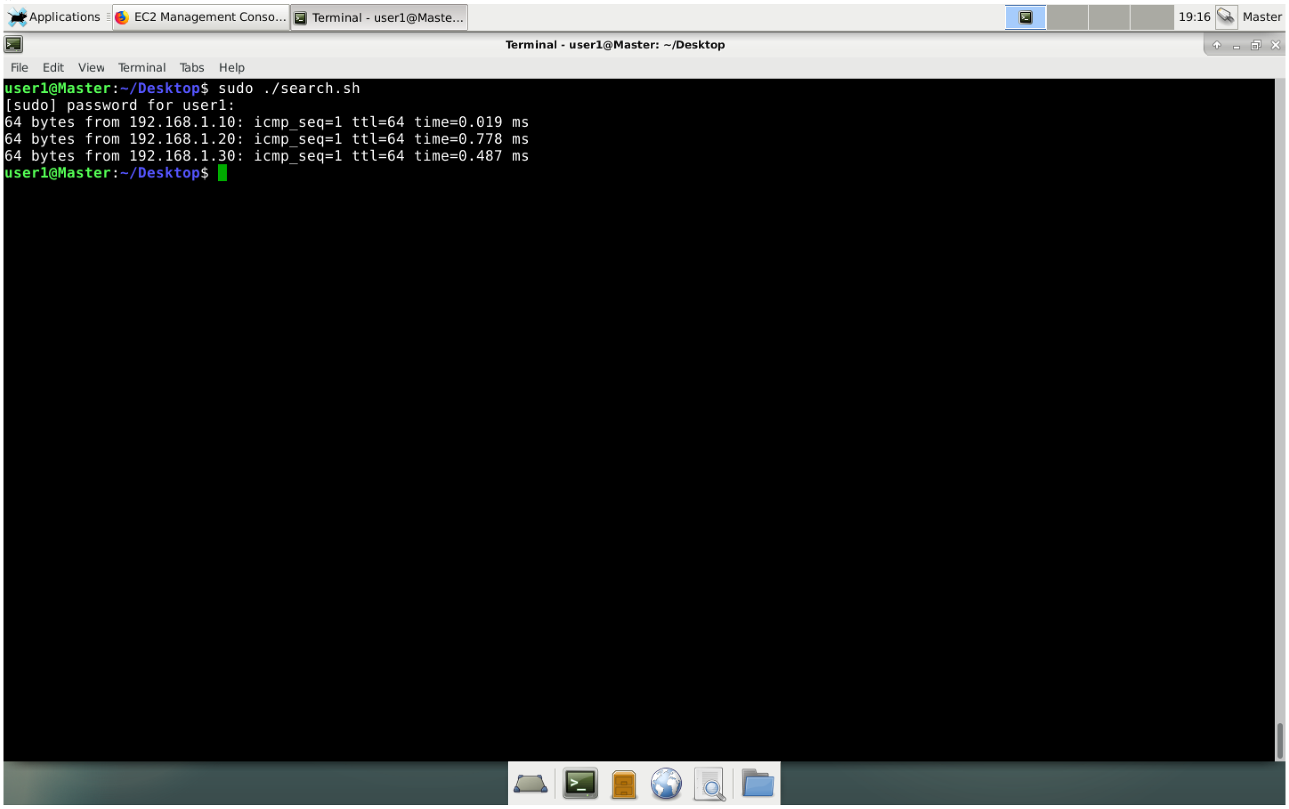
## Automation process in local resources:

```
#Search all ip address in Class C by search.sh
```



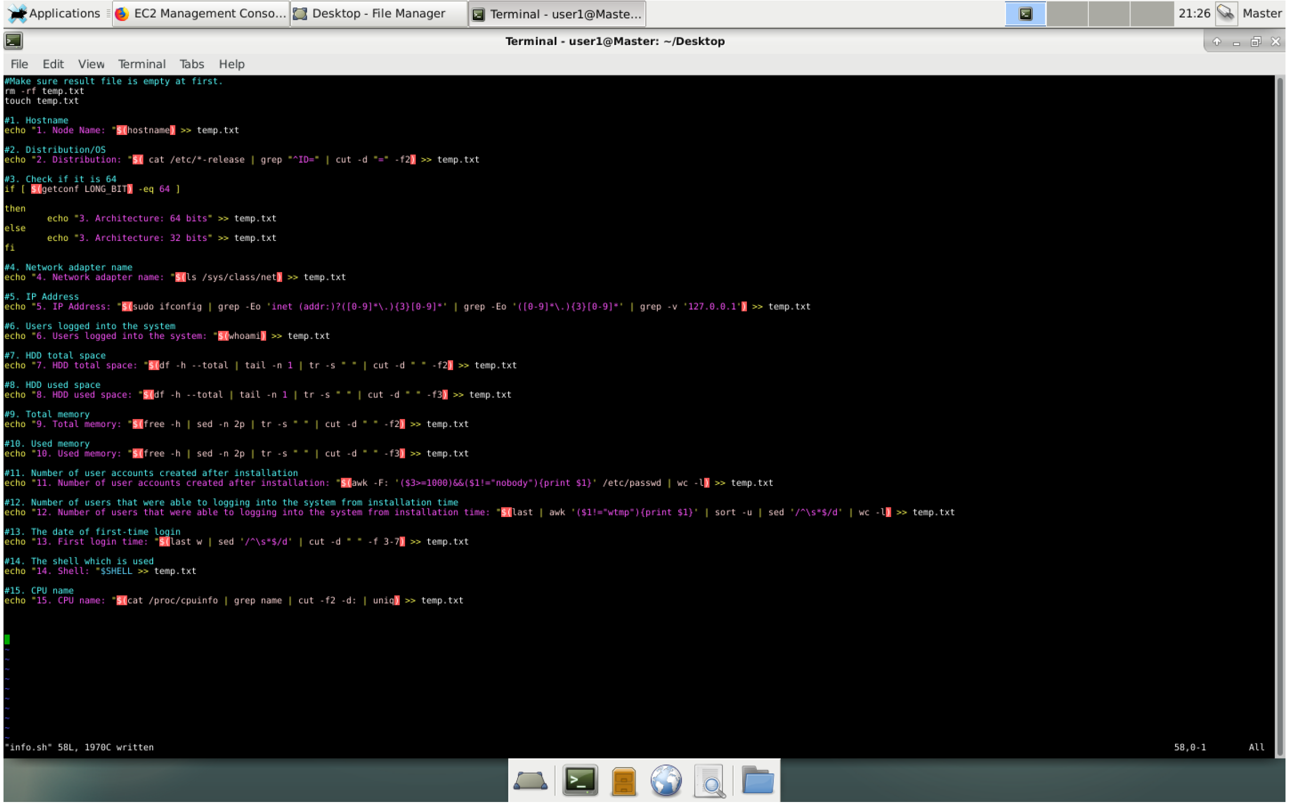
```
#!/bin/sh

COUNTER=1

while [ $COUNTER -lt 254 ]
do
           ping 192.168.1.$COUNTER -c 1 >> result.txt
              COUNTER=$(( $COUNTER + 1 ))
done
cat result.txt | grep "^64"
rm -rf result.txt
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
"search.sh" 11L, 185C
```

#Get a result.


Applications
EC2 Management Conso...
Terminal - user1@Maste...
19:16
Master
Terminal - user1@Master: ~/Desktop
File Edit View Terminal Tabs Help

```
user1@Master:~/Desktop$ sudo ./search.sh
[sudo] password for user1:
64 bytes from 192.168.1.10: icmp_seq=1 ttl=64 time=0.019 ms
64 bytes from 192.168.1.20: icmp_seq=1 ttl=64 time=0.778 ms
64 bytes from 192.168.1.30: icmp_seq=1 ttl=64 time=0.487 ms
user1@Master:~/Desktop$
```

#Get host information by info.sh, and it will be used in auto.sh.

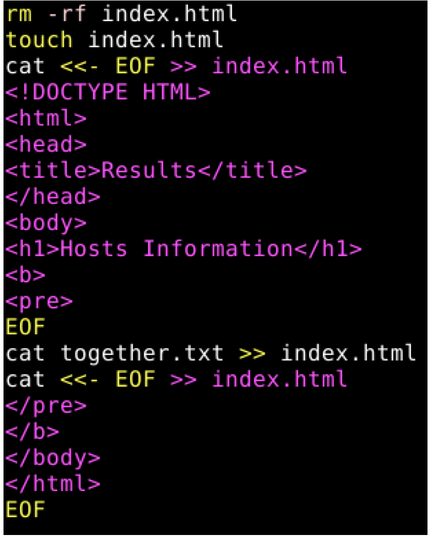

Applications
EC2 Management Conso...
Desktop - File Manager
Terminal - user1@Maste...
21:26
Master
Terminal - user1@Master: ~/Desktop
File Edit View Terminal Tabs Help

```
#Make sure result file is empty at first.
rm -rf temp.txt
touch temp.txt

#1. Hostname
echo "1. Node Name: "$(hostname) >> temp.txt

#2. Distribution/OS
echo "2. Distribution: "$( cat /etc/*-release | grep "^ID=" | cut -d "=" -f2) >> temp.txt

#3. Check if it is 64
if [ $(getconf LONG_BIT) -eq 64 ]

then
        echo "3. Architecture: 64 bits" >> temp.txt
else
        echo "3. Architecture: 32 bits" >> temp.txt
fi

#4. Network adapter name
echo "4. Network adapter name: "$(ls /sys/class/net) >> temp.txt

#5. IP Address
echo "5. IP Address: "$(sudo ifconfig | grep -Eo 'inet (addr:)?([0-9]*\.){3}[0-9]*' | grep -Eo '([0-9]*\.){3}[0-9]*' | grep -v '127.0.0.1') >> temp.txt

#6. Users logged into the system
echo "6. Users logged into the system: "$(whoami) >> temp.txt

#7. HDD total space
echo "7. HDD total space: "$(df -h --total | tail -n 1 | tr -s " " | cut -d " " -f2) >> temp.txt

#8. HDD used space
echo "8. HDD used space: "$(df -h --total | tail -n 1 | tr -s " " | cut -d " " -f3) >> temp.txt

#9. Total memory
echo "9. Total memory: "$(free -h | sed -n 2p | tr -s " " | cut -d " " -f2) >> temp.txt

#10. Used memory
echo "10. Used memory: "$(free -h | sed -n 2p | tr -s " " | cut -d " " -f3) >> temp.txt

#11. Number of user accounts created after installation
echo "11. Number of user accounts created after installation: "$(awk -F: '($3>=1000)&&($1!="nobody"){print $1}' /etc/passwd | wc -l) >> temp.txt

#12. Number of users that were able to logging into the system from installation time
echo "12. Number of users that were able to logging into the system from installation time: "$(last | awk '($1!="wtmp"){print $1}' | sort -u | sed '/^\s*$/d' | wc -l) >> temp.txt

#13. The date of first-time login
echo "13. First login time: "$(last w | sed '/^\s*$/d' | cut -d " " -f 3-7) >> temp.txt

#14. The shell which is used
echo "14. Shell: "$SHELL >> temp.txt

#15. CPU name
echo "15. CPU name: "$(cat /proc/cpuinfo | grep name | cut -f2 -d: | uniq) >> temp.txt



~
~
~
~
~
~
~
~
~
~
"info.sh" 58L, 1970C written                                                    58,0-1        All
```

## Automation process in remote resources:

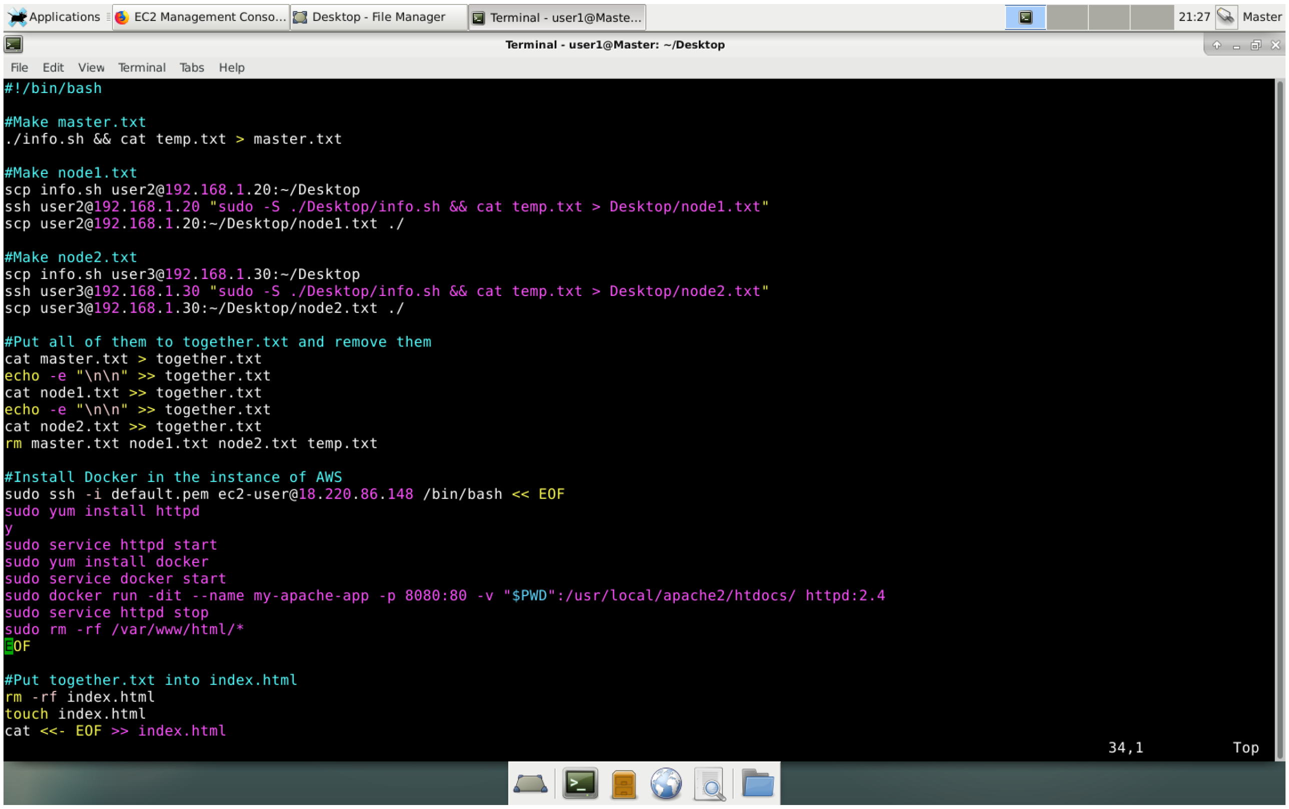
#The program is completed automaticly by auto.sh.
#Firstly, auto.sh call info.sh three times to get three TXT files which contain information o
#Their names are master.txt, node1.txt and node2.txt
#For master.txt

```
./info.sh && cat temp.txt > master.txt
#For node1.txt
scp info.sh user2@192.168.1.20:~/Desktop
ssh user2@192.168.1.20 "sudo -S ./Desktop/info.sh && cat temp.txt > Desktop/node1.txt"
scp user2@192.168.1.20:~/Desktop/node1.txt ./
#For node2.txt
scp info.sh user3@192.168.1.30:~/Desktop
ssh user3@192.168.1.30 "sudo -S ./Desktop/info.sh && cat temp.txt > Desktop/node2.txt"
scp user3@192.168.1.30:~/Desktop/node2.txt ./

#Secondly, collect these three TXT files together into together.txt, and remove these three f
cat master.txt > together.txt
echo -e "\n\n" >> together.txt
cat node1.txt >> together.txt
echo -e "\n\n" >> together.txt
cat node2.txt >> together.txt
rm master.txt node1.txt node2.txt temp.txt

#Thirdly, install Docker in the instance of AWS
sudo ssh -i default.pem ec2-user@18.220.86.148 /bin/bash << EOF
sudo yum install httpd
y
sudo service httpd start
sudo yum install docker
sudo service docker start
sudo docker run -dit --name my-apache-app -p 8080:80 -v "$PWD":/usr/local/apache2/htdocs/ htt
sudo service httpd stop
sudo rm -rf /var/www/html/*
EOF

#Fourthly, build index.html by together.txt
```

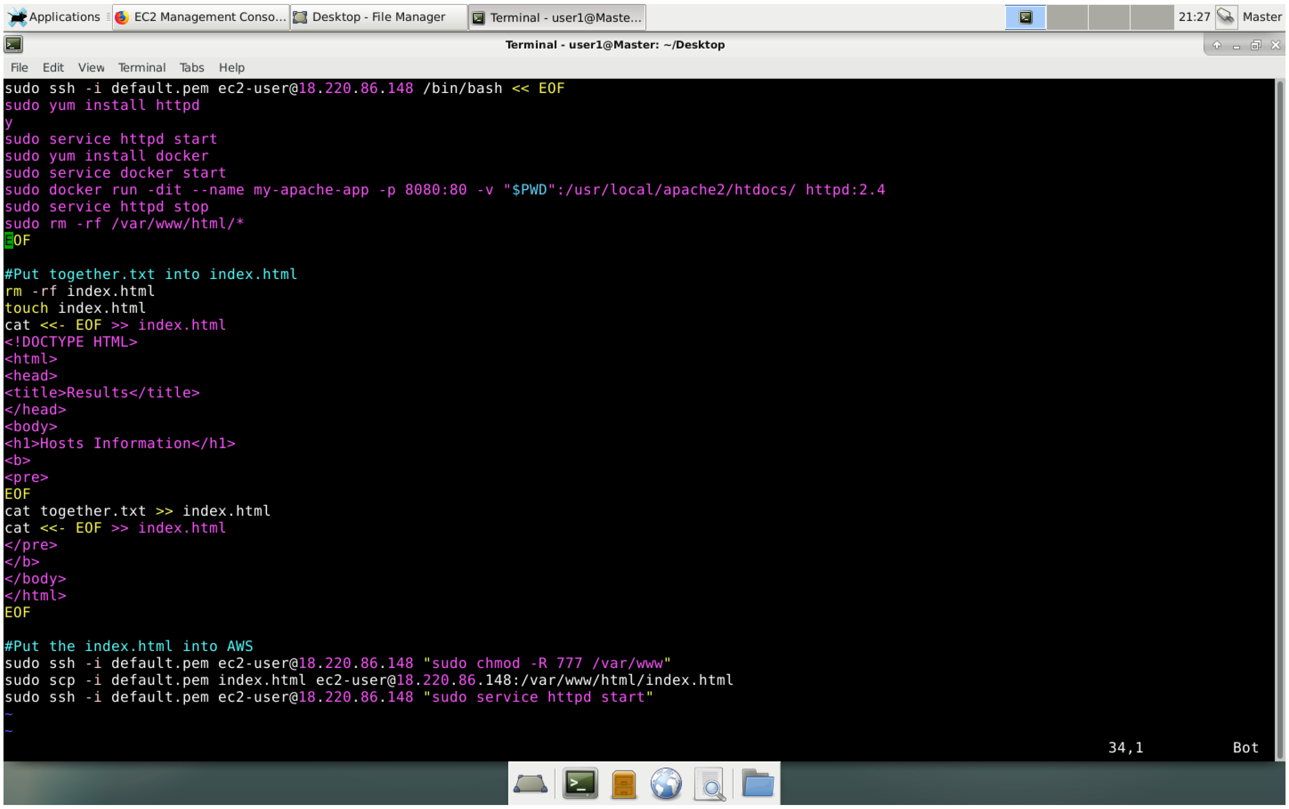


```
rm -rf index.html
touch index.html
cat <<- EOF >> index.html
<!DOCTYPE HTML>
<html>
<head>
<title>Results</title>
</head>
<body>
<h1>Hosts Information</h1>
<b>
<pre>
EOF
cat together.txt >> index.html
cat <<- EOF >> index.html
</pre>
</b>
</body>
</html>
EOF
```

```
#Fifthly, display the index.html to AWS
```

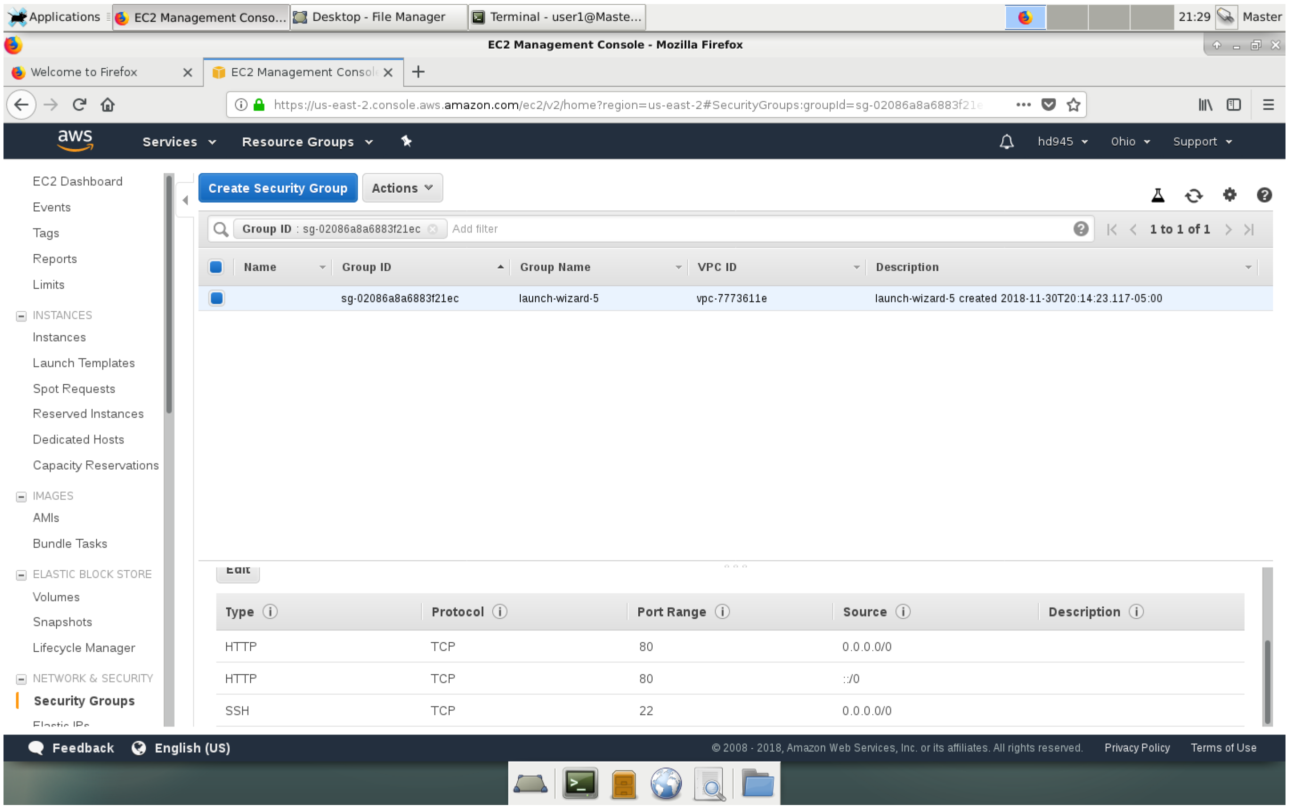

Applications
EC2 Management Conso...
Desktop - File Manager
Terminal - user1@Maste...
21:27
Master
Terminal - user1@Master: ~/Desktop
File Edit View Terminal Tabs Help

```
#!/bin/bash

#Make master.txt
./info.sh && cat temp.txt > master.txt

#Make node1.txt
scp info.sh user2@192.168.1.20:~/Desktop
ssh user2@192.168.1.20 "sudo -S ./Desktop/info.sh && cat temp.txt > Desktop/node1.txt"
scp user2@192.168.1.20:~/Desktop/node1.txt ./

#Make node2.txt
scp info.sh user3@192.168.1.30:~/Desktop
ssh user3@192.168.1.30 "sudo -S ./Desktop/info.sh && cat temp.txt > Desktop/node2.txt"
scp user3@192.168.1.30:~/Desktop/node2.txt ./

#Put all of them to together.txt and remove them
cat master.txt > together.txt
echo -e "\n\n" >> together.txt
cat node1.txt >> together.txt
echo -e "\n\n" >> together.txt
cat node2.txt >> together.txt
rm master.txt node1.txt node2.txt temp.txt

#Install Docker in the instance of AWS
sudo ssh -i default.pem ec2-user@18.220.86.148 /bin/bash << EOF
sudo yum install httpd
y
sudo service httpd start
sudo yum install docker
sudo service docker start
sudo docker run -dit --name my-apache-app -p 8080:80 -v "$PWD":/usr/local/apache2/htdocs/ httpd:2.4
sudo service httpd stop
sudo rm -rf /var/www/html/*
EOF

#Put together.txt into index.html
rm -rf index.html
touch index.html
cat <<- EOF >> index.html
                                                                                 34,1          Top
```


Applications
EC2 Management Conso...
Desktop - File Manager
Terminal - user1@Maste...
21:27
Master
Terminal - user1@Master: ~/Desktop
File Edit View Terminal Tabs Help

```
sudo ssh -i default.pem ec2-user@18.220.86.148 /bin/bash << EOF
sudo yum install httpd
y
sudo service httpd start
sudo yum install docker
sudo service docker start
sudo docker run -dit --name my-apache-app -p 8080:80 -v "$PWD":/usr/local/apache2/htdocs/ httpd:2.4
sudo service httpd stop
sudo rm -rf /var/www/html/*
EOF

#Put together.txt into index.html
rm -rf index.html
touch index.html
cat <<- EOF >> index.html
<!DOCTYPE HTML>
<html>
<head>
<title>Results</title>
</head>
<body>
<h1>Hosts Information</h1>
<b>
<pre>
EOF
cat together.txt >> index.html
cat <<- EOF >> index.html
</pre>
</b>
</body>
</html>
EOF

#Put the index.html into AWS
sudo ssh -i default.pem ec2-user@18.220.86.148 "sudo chmod -R 777 /var/www"
sudo scp -i default.pem index.html ec2-user@18.220.86.148:/var/www/html/index.html
sudo ssh -i default.pem ec2-user@18.220.86.148 "sudo service httpd start"
~
~
                                                                                 34,1          Bot
```

#The http port is opened.



| Name | Group ID | Group Name | VPC ID | Description |
|---|---|---|---|---|
| | sg-02086a8a6883f21ec | launch-wizard-5 | vpc-7773611e | launch-wizard-5 created 2018-11-30T20:14:23.117-05:00 |

| Type | Protocol | Port Range | Source | Description |
|---|---|---|---|---|
| HTTP | TCP | 80 | 0.0.0.0/0 | |
| HTTP | TCP | 80 | ::/0 | |
| SSH | TCP | 22 | 0.0.0.0/0 | |

```
#The website is http://18.220.86.148/
#All the files can be found in https://github.com/HHHHHHHHHHHHHHHHHHHHHHHHHHHHHH/3530FinalPro
```