

Final Project Setup and Exploitation Instructions

Install Kali Linux VM

1. See WinCre3 Document already uploaded to the github repo

Install Ubuntu 20.04 with Linux kernel 5.8.0

1. Navigate to https://releases.ubuntu.com/20.04/?_ga=2.158895674.1041833567.1668977229-1417524537.1668977229 and download the desktop image (~20 min)
2. Setup a new virtual box with the iso above using all the default settings
3. Start the new virtual box, install only minimal installation, and when prompted to create a user, name the system 'target' and for continuity create a user with username 'setup' and password 'setup'
4. Unless otherwise specified, always log into the virtual box you launch from Ubuntu using the Target::setup:setup credentials
5. With the virtual box configured to connect to the internet, download the following packages via apt:
 - a. vim
 - b. ssh
 - c. net-tools
 - d. gcc
6. Downgrade the kernel from 5.15 to 5.8.0 using Mainline Kernel Installer
 - a. Add the repository with the command `sudo add-apt-repository ppa:cappelikan/ppa`
 - b. Add the repository to your source lists `sudo apt update`
 - c. Install Mainline Kernel Installer `sudo apt install mainline`
 - d. Once installed, launch Mainline Kernel Installer and navigate to the 5.8.0 kernel and click 'install'
 - e. When the installation is finished close the window but don't yet reboot.
 - f. Edit the grub configuration to set the default kernel as the last one you boot into
 - i. Edit the config file `sudo vim /etc/default/grub` and add the following lines:
`GRUB_SAVEDEFAULT=true`
`GRUB_DEFAULT=saved`
 - ii. Update grub `sudo update-grub`
 - g. Reboot your machine using the command `reboot`
 - h. On the splash screen for Virtual Box, press and hold the 'Shift' key to open the Grub Menu
 - i. Select 'Advanced Options'

- j. Find, select, and boot the 5.8.0 kernel
- k. Once your virtual box launches, confirm you are on the correct kernel by querying `uname -a`
- l. Reboot your virtual machine once more using the `reboot` command and verify it automatically boots into the 5.8.0 kernel using `uname -a`
7. Create a read-only user in the target VM
 - a. Create a new user that does not have root privileges, for continuity use the username 'readonly' and password 'readonly', run the command `sudo useradd readonly` and follow the prompts to add the password and hit 'Enter' to bypass all other fields

Ensure both VM's are on the same local host-only network

1. With both VM's closed (not currently launched) select one of the VM's and then select settings in Virtual Box
2. Go to the 'Network' tab
3. Select 'Host-Only' from the drop down
4. Repeat for the second VM
5. Ensure the setup is correct and VM's are appropriately linked
 - a. Launch both VM's
 - b. Run the `ifconfig` command on both VM's to ensure they are on the same subnet, for example both are on 192.168.XX.XXX
 - c. Try pinging the other VM from each VM using `ping <IP ADDR>`
 - d. If not successful, repeat above steps or consider using a 'Bridged Adapter' instead of 'Host-Only'

Performing the Exploit

1. Launch both VM's
 - a. Log into Kali with default credentials
 - b. Log into Ubuntu target with setup credentials
2. Get the IP address from the target machine by running the command `ifconfig` in the target machine
3. SSH into the target machine from the Kali machine by using the command `ssh readonly@<IPADDRESS>`
4. Find a location that you want to and can write in from the 'readonly' account, I recommend `~/Documents/` and copy the exploits c file from the github repo to this location
5. NOTE: Take the time to understand what the exploit file is doing – it is going to replace the password hash for the root user with the corresponding hash for 'SecurePassword' in the cached `/etc/passwd` file, allowing the attacking machine to use that new password for root until the cache is cleared or the machine is rebooted
6. Compile the c file using the command `gcc dirtypipe_passwd.c -lcrypt -o dirtypipe_passwd` to create the executable

7. Run the command `id` from the ssh terminal in the Kali machine to verify the user and permissions for your current shell, you should see something like

```
readonly@target:~/Documents$ id
uid=1001(readonly) gid=1001(readonly) groups=1001(readonly)
readonly@target:~/Documents$
```

8. Execute the exploit `./dirtypipe_passwd`

```
readonly@target:~/Documents$ ./dirtypipe_passwd
/etc/passwd successfully backed up to /tmp/passwd.bak
SaWJv12bNyQ5I
New passwd line: oot:SaWJv12bNyQ5I:0:0:Pwned:/root:/bin/bash
It worked!
You can now login with root:SecurePassword
```

9. You can verify the corresponding entry for the passwd file has been changed by running `head -n 1 /etc/passwd` from the ssh shell on the Kali machine

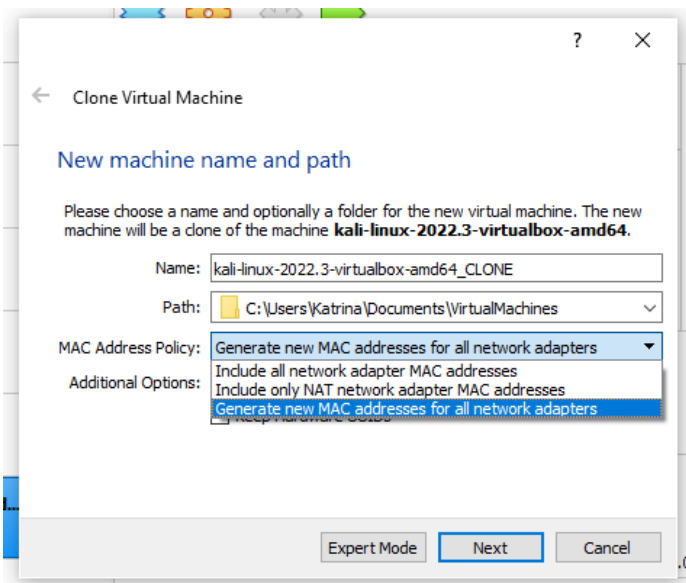
```
readonly@target:~/Documents$ head -n 1 /etc/passwd
root:SaWJv12bNyQ5I:0:0:Pwned:/root:/bin/bash
```

10. Log into root from the ssh shell by running `su -` and when prompted for the password use the new password 'SecurePassword'
11. You are now logged in as root, use the `id` command to verify you are now logged in as root

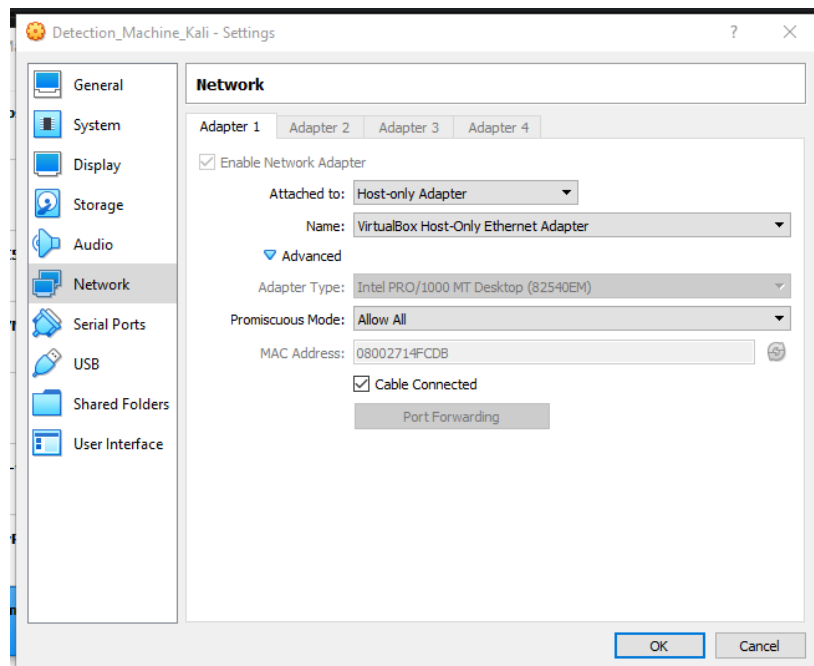
```
root@target:~# id
uid=0(root) gid=0(root) groups=0(root)
```

Creating a Detection Kali VM

1. Power off the Ubuntu target machine and the Kali attack machine
2. Create a copy of the attack machine in Oracle VM VirtualBox by right clicking on the Kali machine and selecting 'Clone'
3. Select the folder location where you want your Virtual Machine to be stored, and VERY IMPORTANT – select "Generate new MAC addresses for all network adapters"



4. When prompted, select “Full clone”, and complete the cloning process
5. Once complete, change the network settings for this detection system to be ‘Host-only’, and under the Advanced section ensure the Promiscuous Mode is set to ‘Allow All’



6. Launch the detection machine, remember the default login credentials will be
username: kali
password: kali

Using nmap to Map Network IP's

You will be performing all commands on the detection machine unless specifically noted otherwise.

1. Make sure all three VM's are running – target Ubuntu, attacker Kali, and detection Kali; in this section
2. Run this nmap command in the terminal `sudo nmap -sn 192.168.56.0/24 -oG - | awk '/Up/{print $2}'` to report all IP addresses from devices on the same local network as your detection machine, these are the IP's you'll need to keep track of

```
(kali@kali)-[~]  
$ sudo nmap -sn 192.168.56.0/24 -oG - | awk '/Up/{print $2}'  
192.168.56.1  
192.168.56.100  
192.168.56.101  
192.168.56.105  
192.168.56.106
```

192.168.56.1 and 192.168.56.100 belong to the VM host computer, which you can confirm by going to File > Host Network Manager in the main VirtualBox application window

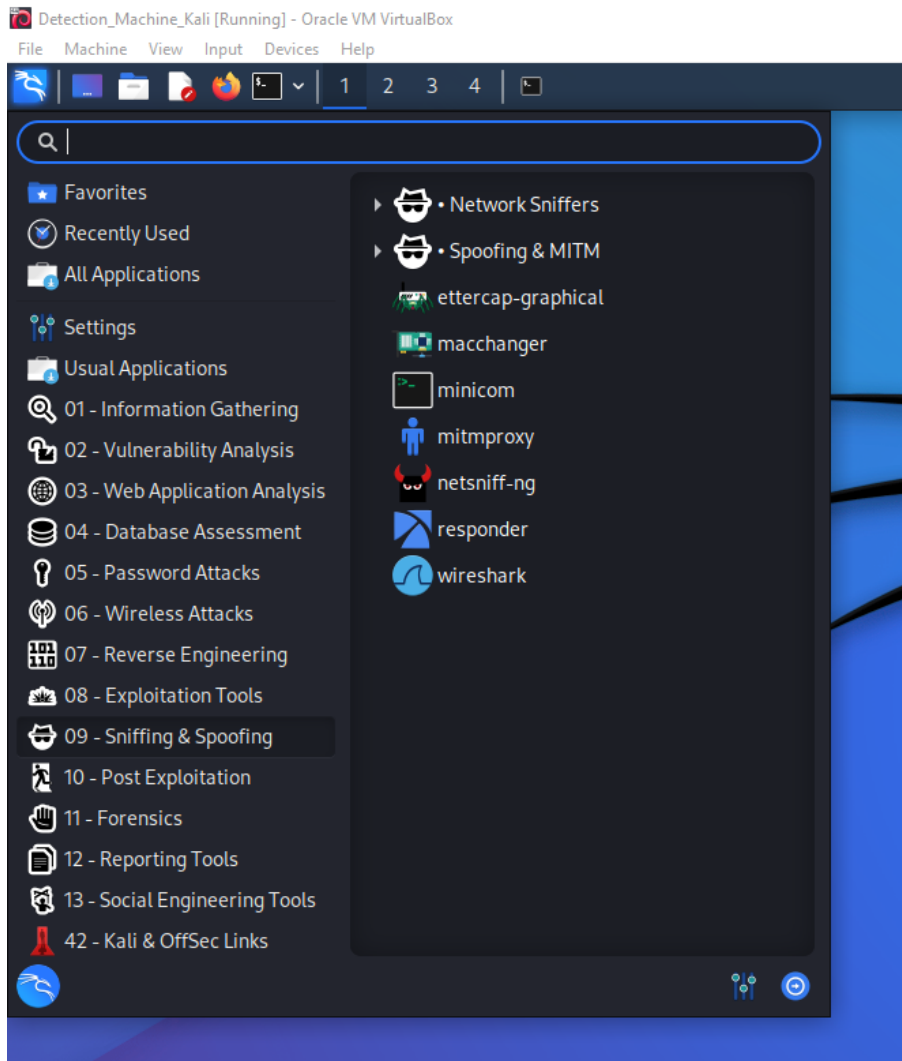
3. Confirm your detection machine's IP by running the `ifconfig` command in the terminal, it should be one of the remaining three IP's identified in step 2

4. Now you have 2 IP's left that you need to keep track of – one is your attacker machine and one is your target machine, assuming you are entrusted to monitor the target machine you'll be able to confirm this as well by running the `ifconfig` command on the target machine
5. The last remaining IP is by default the potential attacker, or unknown

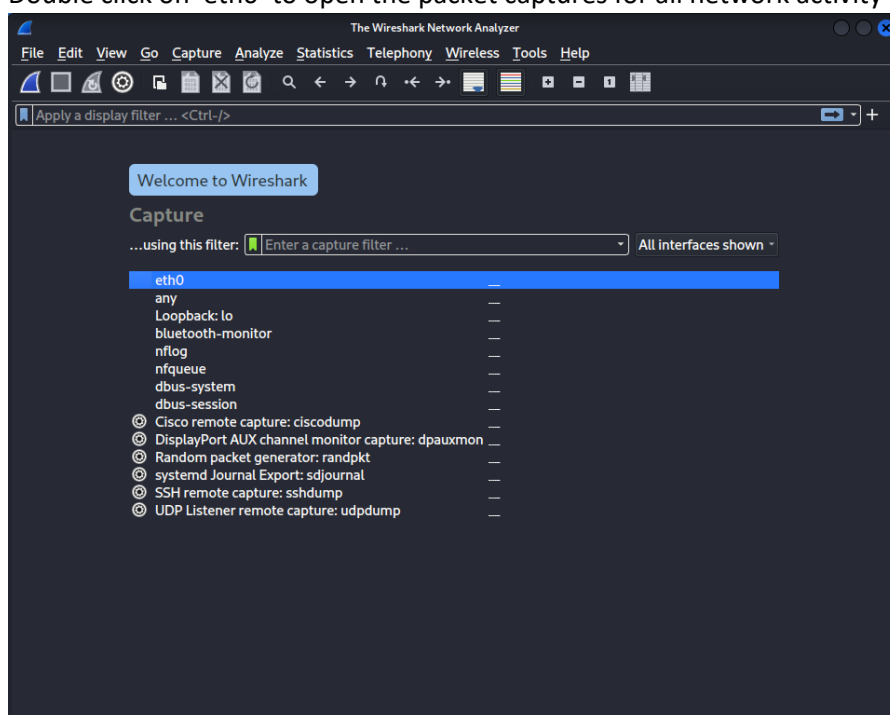
Using wireshark to Examine Packet Captures on the Network

The work in this section is done using the detection Kali VM unless otherwise stated.

1. With all three VM's still running, open your detection Kali machine, and open Wireshark from the applications menu under '09 – Sniffing & Spoofing'



- Double click on 'eth0' to open the packet captures for all network activity



- Try ssh'ing from the attacking system to the target or pinging the various IP's to see the packets picked up from the different machines

No.	Time	Source	Destination	Protocol	Length	Info
1	0.000000000	192.168.56.1	239.255.255.250	SSDP	217	M-SEARCH * HTTP/1.1
2	0.431018457	192.168.56.1	239.255.255.250	SSDP	217	M-SEARCH * HTTP/1.1
3	1.005493592	192.168.56.1	239.255.255.250	SSDP	217	M-SEARCH * HTTP/1.1
4	1.437784549	192.168.56.1	239.255.255.250	SSDP	217	M-SEARCH * HTTP/1.1
5	2.013760007	192.168.56.1	239.255.255.250	SSDP	217	M-SEARCH * HTTP/1.1
6	2.445608661	192.168.56.1	239.255.255.250	SSDP	217	M-SEARCH * HTTP/1.1
7	3.021539928	192.168.56.1	239.255.255.250	SSDP	217	M-SEARCH * HTTP/1.1
8	3.454460107	192.168.56.1	239.255.255.250	SSDP	217	M-SEARCH * HTTP/1.1
9	31.548451905	192.168.56.106	192.168.56.100	DHCP	324	DHCP Request - Transaction ID=00000000
10	31.555924536	192.168.56.100	192.168.56.106	DHCP	590	DHCP ACK - Transaction ID=00000000
11	36.570755163	PcsCompu_14:fc:db	PcsCompu_63:f6:a0	ARP	42	Who has 192.168.56.100? T
12	36.571109941	PcsCompu_63:f6:a0	PcsCompu_14:fc:db	ARP	60	192.168.56.100 is at 08:0
13	81.586712163	192.168.56.101	192.168.56.105	SSH	102	Client: Encrypted packet
14	81.587123026	192.168.56.105	192.168.56.101	SSH	102	Server: Encrypted packet
15	81.587284338	192.168.56.101	192.168.56.105	TCP	66	52306 → 22 [ACK] Seq=37 A
16	81.702692351	192.168.56.101	192.168.56.105	SSH	102	Client: Encrypted packet
17	81.703048009	192.168.56.105	192.168.56.101	SSH	102	Server: Encrypted packet
18	81.703247837	192.168.56.101	192.168.56.105	TCP	66	52306 → 22 [ACK] Seq=73 A
19	81.826957953	192.168.56.101	192.168.56.105	SSH	102	Client: Encrypted packet
20	81.827530353	192.168.56.105	192.168.56.101	SSH	102	Server: Encrypted packet
21	81.827677792	192.168.56.101	192.168.56.105	TCP	66	52306 → 22 [ACK] Seq=109

- Proposal for Live Mitigation: Could be used to create a repository for frequent pattern mining to identify suspicious network activity over a particular network

Monitoring Users for Live Detection

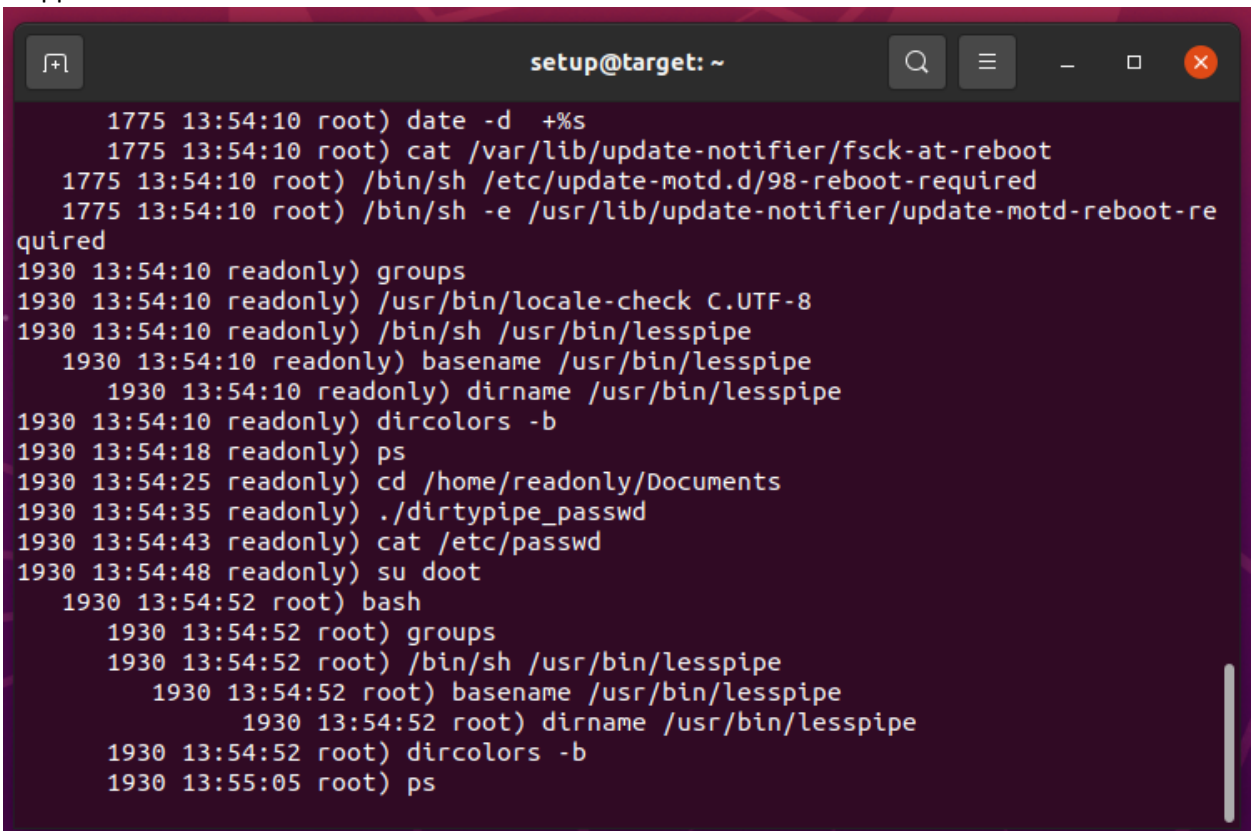
All of the following in this section will be done on the target system logged in to the 'setup' account unless otherwise specified.

1. Shut down the target Ubuntu VM, change the network settings to use 'NAT' and restart
2. Download the sysdig program by running `sudo apt-get install sysdig` in the terminal, and follow prompts to install
3. Shut down the machine, change the network setting back to 'Host-only' and restart
4. Running the `w <username>` command will tell you what each user on the machine is doing, what IP they are from, and when they've logged in, on the Ubuntu target machine logged in to the 'setup' account run `w readonly`

```
setup@target:~$ w readonly
13:34:33 up 52 min,  2 users,  load average: 0.37, 0.08, 0.03
USER      TTY      FROM          LOGIN@      IDLE   JCPU   PCPU WHAT
readonly pts/0    192.168.56.101 13:29        5:05    0.04s   0.04s -bash
```

We know that the IP of the target VM is not the IP listed above, we can confirm this to ourselves by running the `ifconfig` command again

5. Running the `w` command all on its own will show all users which may be more informative in other use cases but considering we only have the two users it won't show much more
6. We can gather more information using the command `sudo sysdig -c spy_users`
7. Repeat the exploit while sysdig is running, you'll notice that all commands the user enters are mapped



```
1775 13:54:10 root) date -d +%s
1775 13:54:10 root) cat /var/lib/update-notifier/fsck-at-reboot
1775 13:54:10 root) /bin/sh /etc/update-motd.d/98-reboot-required
1775 13:54:10 root) /bin/sh -e /usr/lib/update-notifier/update-motd-reboot-re
quired
1930 13:54:10 readonly) groups
1930 13:54:10 readonly) /usr/bin/locale-check C.UTF-8
1930 13:54:10 readonly) /bin/sh /usr/bin/lesspipe
1930 13:54:10 readonly) basename /usr/bin/lesspipe
1930 13:54:10 readonly) dirname /usr/bin/lesspipe
1930 13:54:10 readonly) dircolors -b
1930 13:54:18 readonly) ps
1930 13:54:25 readonly) cd /home/readonly/Documents
1930 13:54:35 readonly) ./dirtypipe_passwd
1930 13:54:43 readonly) cat /etc/passwd
1930 13:54:48 readonly) su doot
1930 13:54:52 root) bash
1930 13:54:52 root) groups
1930 13:54:52 root) /bin/sh /usr/bin/lesspipe
1930 13:54:52 root) basename /usr/bin/lesspipe
1930 13:54:52 root) dirname /usr/bin/lesspipe
1930 13:54:52 root) dircolors -b
1930 13:55:05 root) ps
```

8. Proposal for Live Detection: Could be used to create a repository for frequent pattern mining to identify suspicious user access, could actively look for suspicious escalation to root from non-root users

```
1930 13:54:48 readonly) su doot
1930 13:54:52 root) bash
```

9. Proposal for Mitigation: restrict IP address access for remote users, enforce a VPN, write script to immediately kill ssh connections or disable non-root users who escalate to root

Monitoring Command Execution for Live Detection

Taking the above a step further we can create a small program that can read the sysdig log on a system and alert us if there is user escalation to root. The proof-of-concept code has been uploaded to the github repo. Simply execute this demo program on one of the admin accounts of your target machine to monitor across all users for unexpected escalation to root.

COTS Scanning Tools to Consider

The following are commercial off-the-shelf (COTS) solutions which may also provide similar insights for live detection that could be explored if we have enough time and if there is an opportunity to do a free trial

- Sysdig Secured