# 🧠 THM / Ice

## 📌 General Info

- **\*\*Nombre\*\***: Ice

- **\*\*Plataforma\*\***: TryHackMe

- **\*\*Dificultad\*\***: Fácil

- **\*\*Fecha\*\***: [16/08/2025]

- **\*\*Autor del reto\*\***: [Opcional]

- **\*\*Tu nick\*\***: [0ldboy]

---

## 🛠️ Used tools

- Nmap

- Xsltproc

- Metasploit

- Remmina

---

## 🔍 1. Connect

IMPORTANT: I recommended previously to install ***xsltproc***, this tool will allowed to convert the xml file to html.

```bash

**sudo apt install xsltproc**

### 🔎 NMAP SCAN

```bash

**sudo nmap 10.10.74.87 -sS -sC -sV --stats-every=5s -oX ice.xml && xsltproc ice.xml -o ice.html && xdg-open ice.html**

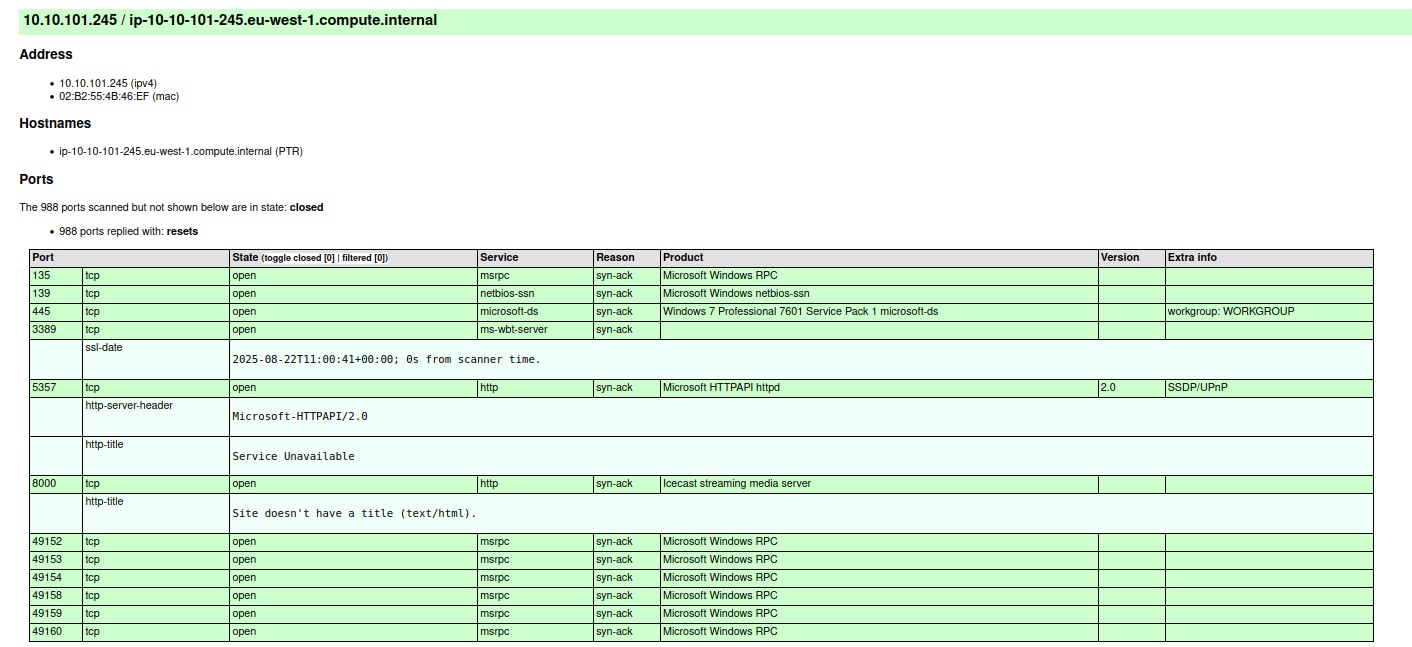
OPTION 2: Create an authomatation script:

```bash

**alias nmapscan='sudo nmap 10.10.74.87 -sS -sC -sV --stats-every=5s -oX ice.xml && xsltproc ice.xml -o ice.html && xdg-open ice.html'**

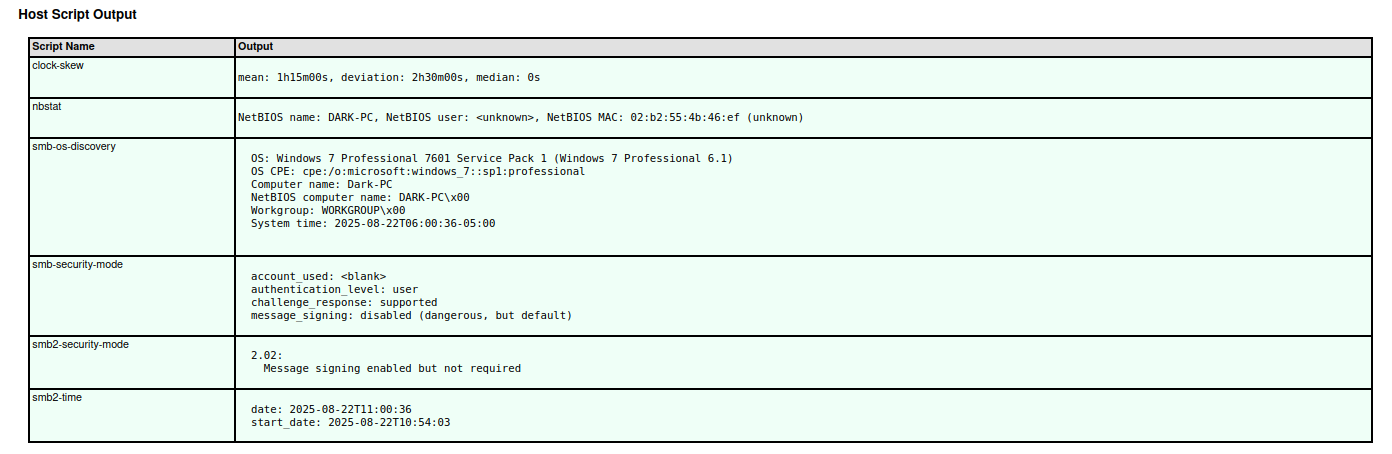
Guardar en archivo “[ice.sh](http://ice.sh)”

## 🔍 3. RECON

* 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**

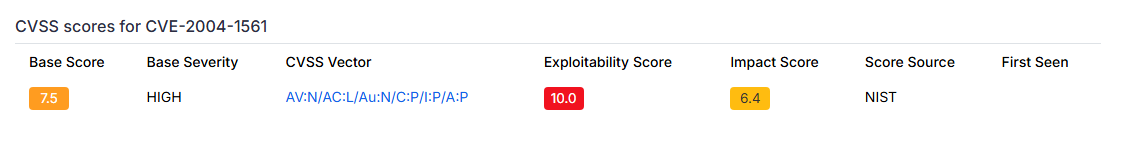
## 🔍 4. Gain Access

* What service did nmap identify as running on port 8000? (First word of this service) Check ***Product***
* What does Nmap identify as the hostname of the machine? (All caps for the answer)Check ***nbstat Output***

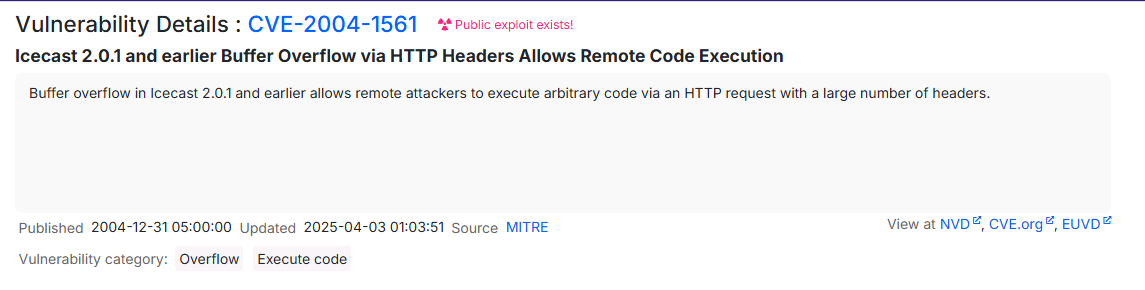
****

## 🔍 4. Gain Access

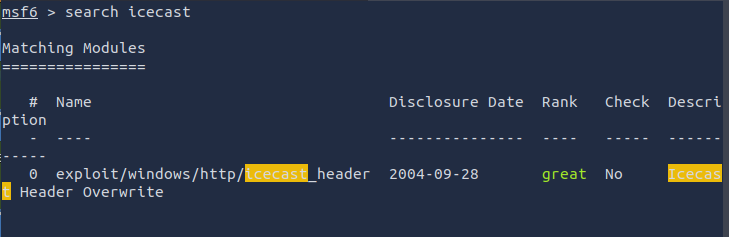
* Now that we've identified some interesting services running on our target machine, let's do a little bit of research into one of the weirder services identified: Icecast. Icecast, or well at least this version running on our target, is heavily flawed and has a high level vulnerability with a score of 7.5 (7.4 depending on where you view it). What is the **Impact Score** for this vulnerability? Use [https://www.cvedetails.com](https://www.cvedetails.com/cve/CVE-2004-1561/) for this question and the next.



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

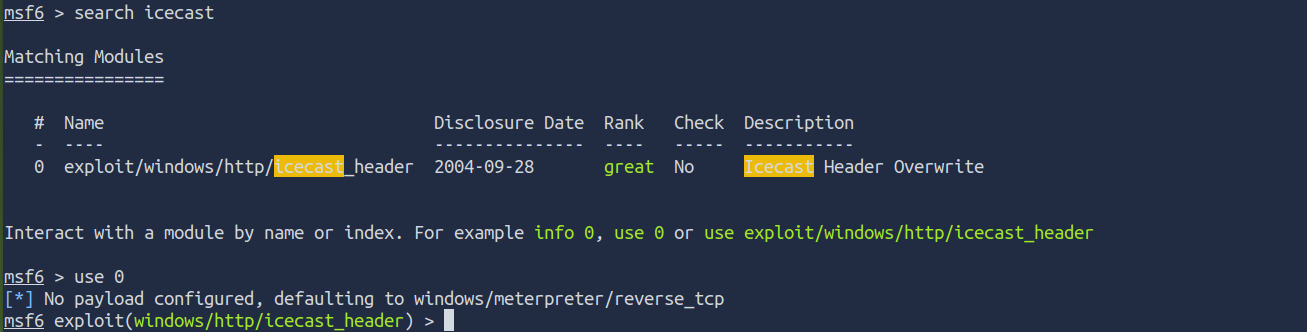


* After Metasploit has started, let's search for our target exploit using the command 'search icecast'. What is the full path (starting with exploit) for the exploitation module? If you are not familiar with metasploit, take a look at the [Metasploit](https://tryhackme.com/module/metasploit) module.

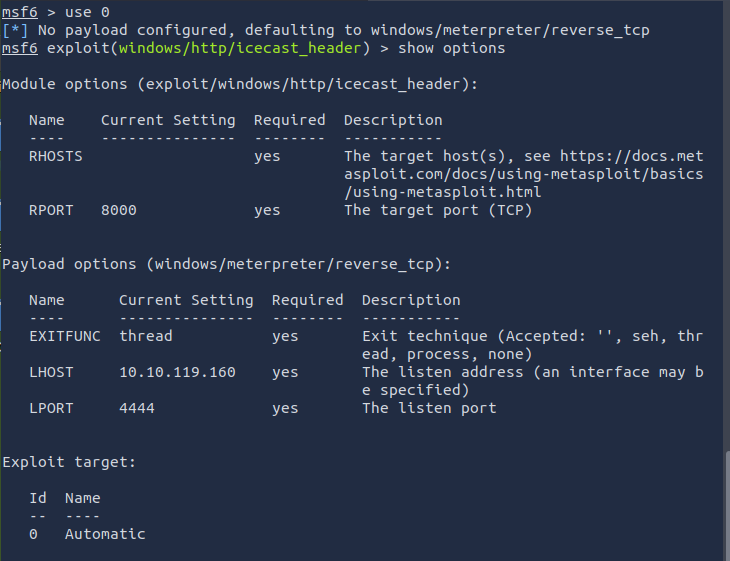


* Let's go ahead and select this module for use. Type either the command `***use icecast***` or `***use 0***` to select our search result.

**Check print**

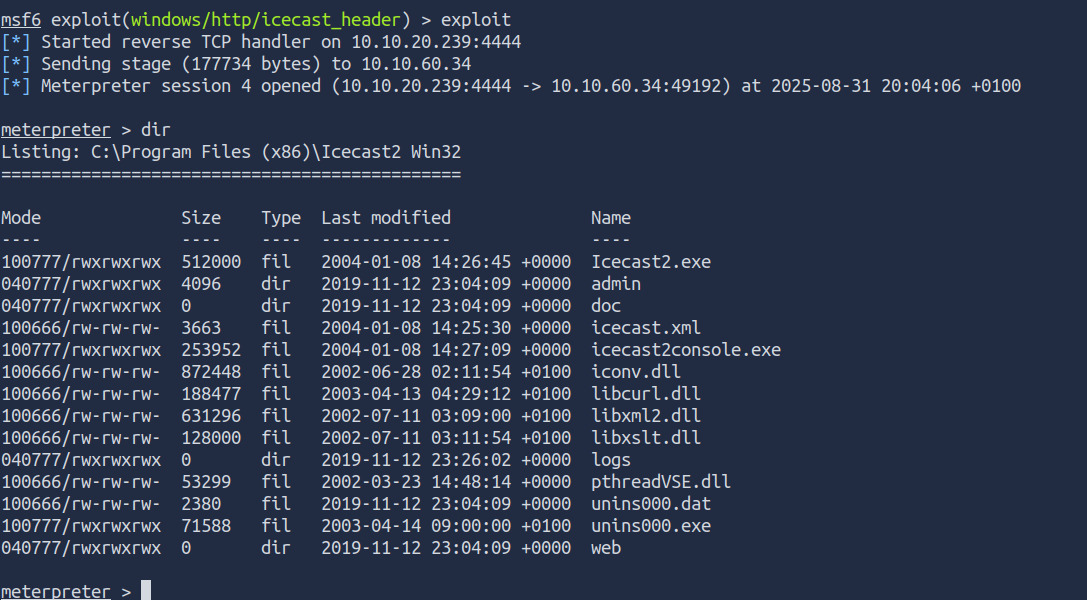
****

* 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?**Check print**



* First let's check that the LHOST option is set to our tun0 IP (which can be found on the [access](https://tryhackme.com/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***`

**Check print**



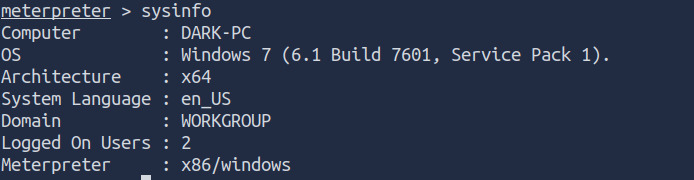
## 🔍 5. ESCALATE

* Woohoo! We've gained a foothold into our victim machine! What's the name of the shell we have now?***Check the name of tool me used to run the exploit***
* What user was running that Icecast process? The commands used in this question and the next few are taken directly from the '[Metasploit](https://tryhackme.com/module/metasploit)' module.

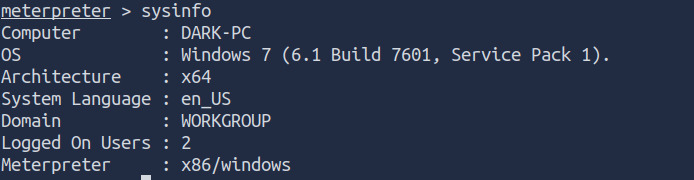
***Check print***

******

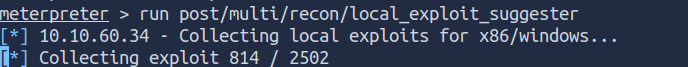
* ***What build of Windows is the system?***

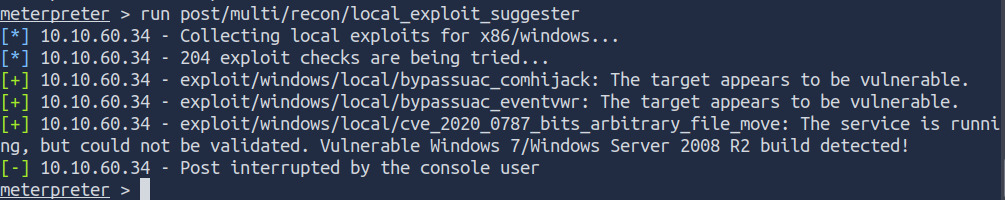
******

* ***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?***

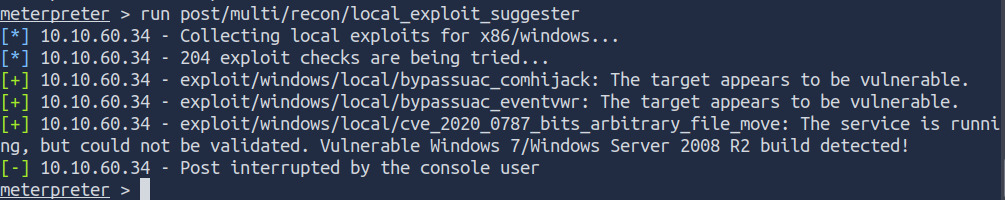
******

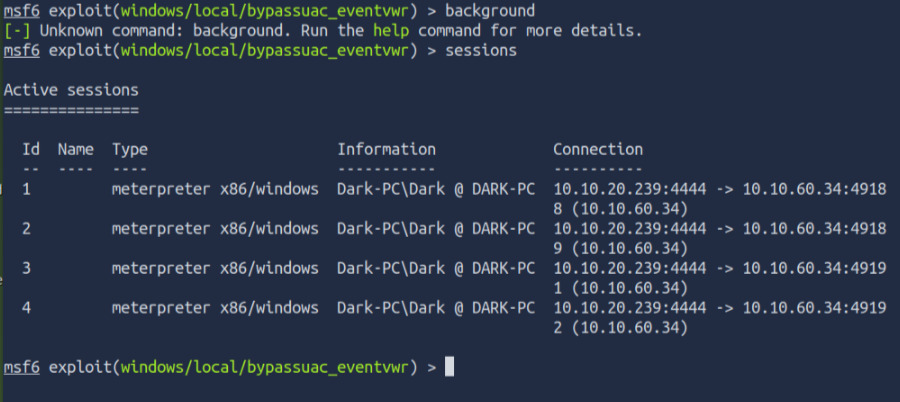
* *Now that we know the architecture of the process, let's perform some further recon. While this doesn't work the best on x64 machines, let's now run the following command `****run post/multi/recon/local\_exploit\_suggester****`. \*This can appear to hang as it tests exploits and might take several minutes to complete\**

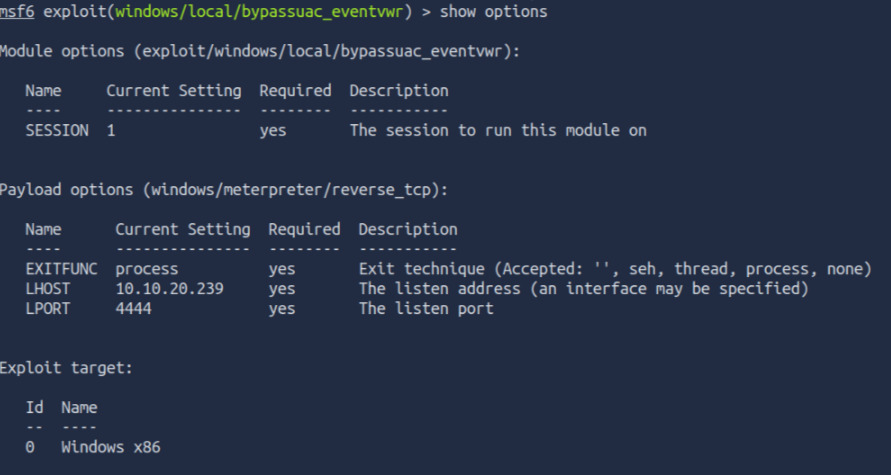
**

**

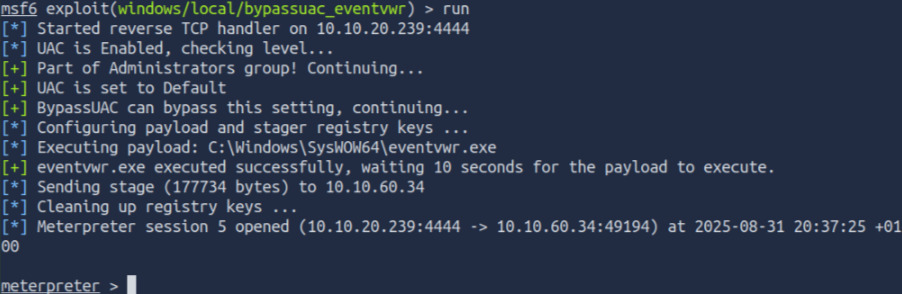
* *Running the local exploit suggester will return quite a few results for potential escalation exploits. What is the full path (starting with exploit/) for the first returned exploit?*

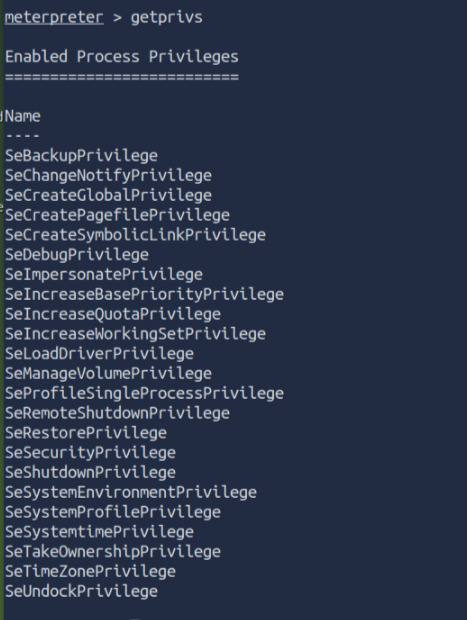
**

* *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.*
* *Go ahead and select our previously found local exploit for use using the command `use FULL\_PATH\_FOR\_EXPLOIT`*

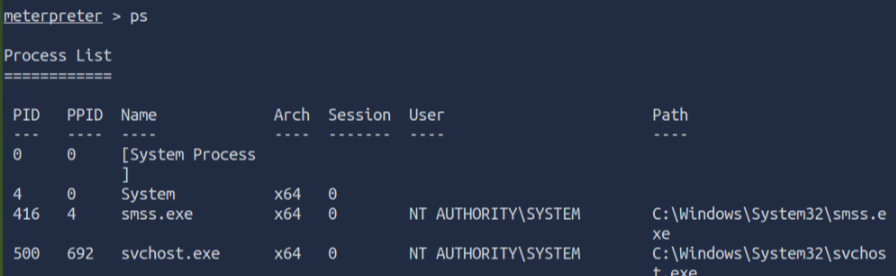
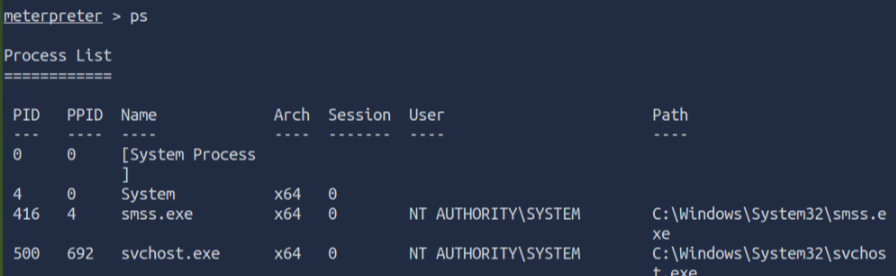
**

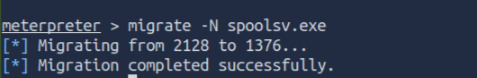
* *Following completion of the privilege escalation a new session will be opened. Interact with it now using the command `****sessions SESSION\_NUMBER****`*

**

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

## 🔍 6. 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).
* 
* 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.
* 
* Migrate to this process now with the command `migrate -N PROCESS\_NAME`



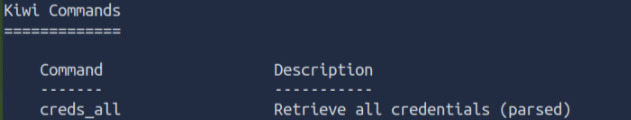
* Let's check what user we are now with the command `getuid`.What user is listed?



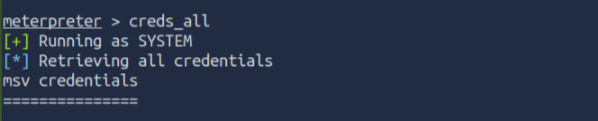
Now that we've made our way to full administrator permissions we'll set our sights on looting. Mimikatz is a rather infamous password dumping tool that is incredibly useful. Load it now using the command `load kiwi` (Kiwi is the updated version of Mimikatz)



Which command allows up to retrieve all credentials?

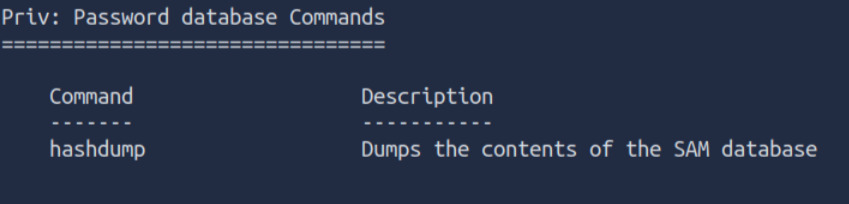


Run this command now. What is Dark's password? Mimikatz allows us to steal this password out of memory even without the user 'Dark' logged in as there is a scheduled task that runs the Icecast as the user 'Dark'. It also helps that Windows Defender isn't running on the box ;) (Take a look again at the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

****

## 🔍 7. POST-EXPLOITATION

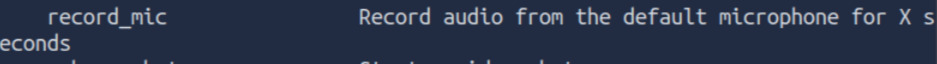
What command allows us to dump all of the password hashes stored on the system? We won't crack the Administrative password in this case as it's pretty strong (this is intentional to avoid password spraying attempts)



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



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



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.

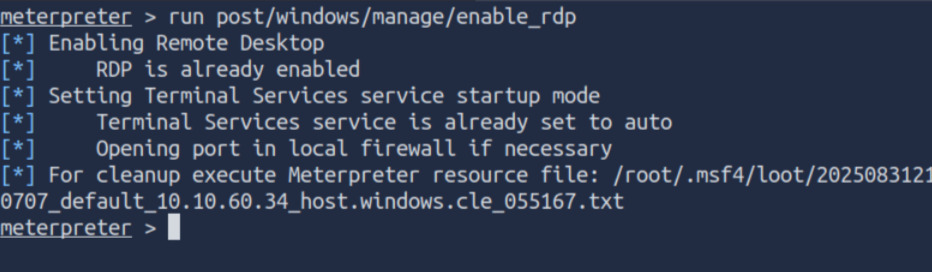


Mimikatz allows us to create what's called a `golden ticket`, allowing us to authenticate anywhere with ease. What command allows us to do this?

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.



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***`



## 🔍 8. EXTRA CREDIT

As you advance in your pentesting skills, you will be faced eventually with exploitation without the usage of Metasploit. Provided above is the link to one of the exploits found on Exploit DB for hijacking Icecast for remote code execution. While not required by the room, it's recommended to attempt exploitation via the provided code or via another similar exploit to further hone your skills.

Feel free to contact me through the following profiles:

[Linkedin](https://www.linkedin.com/in/andrespds/) | [IG](https://www.instagram.com/tob_project/) | [GItHub](https://github.com/tobproject)