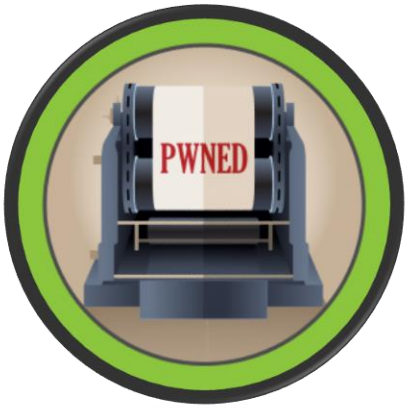


MAQUINA ANTIQUE – HACK THE BOX

Raptor-Attack

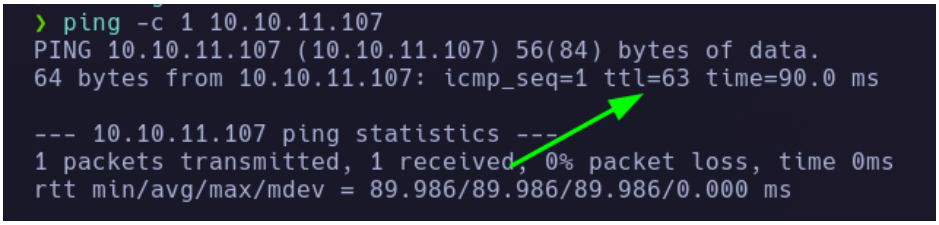
TOPICS

- SNMP Enum
- Hexadecimal Password Decoding to Plain Text
- Command Execution in HP JetDirect (RCE)
- Internal Port Discovery Cups 1.6.1 (Local Port Forwarding)
- Reading Root Files with Script-Exploit cups-root-file-read.sh
- Extra - Abusing the Pkexec Binary (Privilege Escalation)

Enumeración y Reconocimiento

Inicamos comprobando conectividad con el host victima

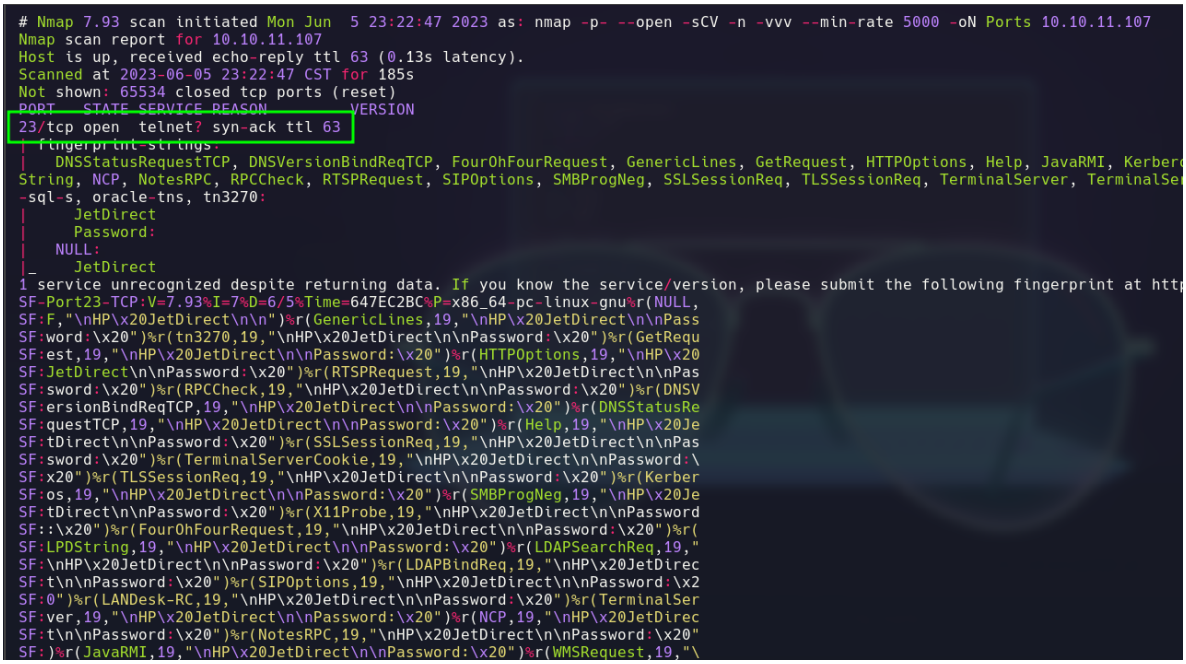
$ping -c 1 10.10.11.107



Tenemos un ttl 64 = Maquina Linux

Realizare un escaneo de puertos, esto me ayudara a reconocer los puertos abiertos en la maquina víctima.

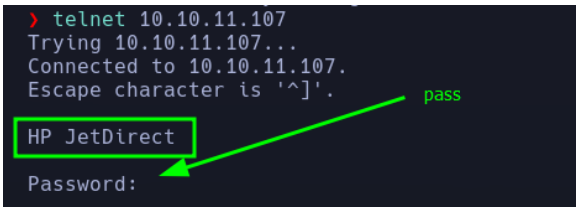
#nmap -p- --open -sCV -n -v –min-rate 5000 10.10.11.107 -oN Ports



Raptor-Attack

Solo encontré el puerto 23/tcp telnet

Nota. Por lo que veo es un puerto que, al intentar conectarme, me conecto a una impresora HP JetDirect y me pide un usuario y contraseña.



Por el momento no tengo alguna contraseña, busqué directamente por internet si existe alguna contraseña por default para esto y encontré la siguiente información

HP Jetdirect es el nombre de una tecnología vendida por Hewlett-Packard que permite que las impresoras de las computadoras se conecten directamente a una red de área local.

Veo que es el único puerto abierto por TCP, por lo cual intentare encontrar mas puertos por el protocolo UDP.

$ nmap -sU -sCV -n -v --top-ports 200 10.10.11.107 -oN UDPPorts

```
cat UDPPorts -l Java
       File: UDPPorts
1      # Nmap 7.93 scan initiated Tue Jun  6 20:33:00 2023 as: nmap -sU -sCV -n -v --top-ports 200 -oN UDPPorts 10.10.11.107
2      Increasing send delay for 10.10.11.107 from 800 to 1000 due to 11 out of 23 dropped probes since last increase.
3      Nmap scan report for 10.10.11.107
4      Host is up (0.080s latency).
5      Not shown: 199 closed udp ports (port-unreach)
6      PORT    STATE SERVICE VERSION
7      161/udp open  snmp    SNMPv1 server (public)
8
9      Read data files from: /usr/bin/../share/nmap
10     Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
11     # Nmap done at Tue Jun  6 20:36:26 2023 -- 1 IP address (1 host up) scanned in 205.81 seconds
```

Al parecer existe un puerto 162/udp snmp. Realizare una búsqueda para realizar Pentesting sobre este puerto, primero veamos que es este puerto.

El puerto 161 es el puerto predeterminado de los dispositivos de red a los que se envían consultas SNMP durante los procesos de descubrimiento y supervisión. Se encuentra definido en la columna m_SnmpPort de la tabla de base de datos snmpStack. verSecurityTable.

Veamos qué más podemos encontrar.

https://book.hacktricks.xyz/network-services-pentesting/pentesting-snmp

Pentesting SNMP

SNMP - Simple Network Management Protocol es un protocolo utilizado para monitorear diferentes dispositivos en la red (como enrutadores, conmutadores, impresoras, IoT...).

MIB (Base de Información de Gestión): Es un formato independiente para almacenar información del dispositivo. Consiste en un archivo de texto en el que se enumeran todos los objetos SNMP consultables de un dispositivo en una jerarquía de árbol estandarizada. La MIB contiene al menos un Identificador de Objeto (OID), que proporciona una dirección única, un nombre, información sobre el tipo de objeto, los derechos de acceso y una descripción del objeto correspondiente.

OID (Identificadores de Objeto): Los OID identifican de manera única los objetos administrados en una jerarquía MIB. Se representan como un árbol, con niveles asignados por diferentes organizaciones. Los OID de nivel superior pertenecen a organizaciones estándar, mientras que los proveedores definen sus propias ramas privadas que incluyen objetos administrados para sus productos.

Un ejemplo de OID es el siguiente: 1.3.6.1.4.1.1452.1.2.5.1.3.21.1.4.7. Cada número en este OID tiene un significado específico. Por ejemplo, el primer número (1) representa ISO y establece que es un OID. El segundo número (3) es ORG, que especifica la organización que construyó el dispositivo. El tercer número (6) es DOD (Departamento de Defensa), la organización que estableció Internet. Los siguientes números (1 y 4) indican que es un dispositivo fabricado por una organización privada y comercial. Los demás números dan información específica sobre el dispositivo, como el nombre de la organización (1452) y el tipo de dispositivo (despertador).

Versiones de SNMP: SNMP (Protocolo Simple de Administración de Red) tiene dos versiones principales: SNMPv1 y SNMPv3.

SNMPv1: Es la versión más común y su autenticación se basa en una cadena comunitaria que viaja en texto plano. Toda la información se envía en texto plano, lo que puede presentar riesgos de seguridad.

SNMPv3: Utiliza una forma de autenticación más segura y la información se transmite cifrada. Esto mejora la seguridad en comparación con SNMPv1, ya que protege la información de posibles ataques de diccionario.

Cadenas comunitarias: Las cadenas comunitarias son utilizadas para acceder a la información almacenada en la MIB en las versiones SNMPv1 y SNMPv2/2c. Hay dos tipos de cadenas comunitarias:

Public: Principalmente de solo lectura, lo que significa que solo se pueden realizar consultas y no se permite modificar los valores de los objetos.

Private: Permite tanto la lectura como la escritura de valores en general, lo que brinda más flexibilidad para realizar cambios en los objetos.

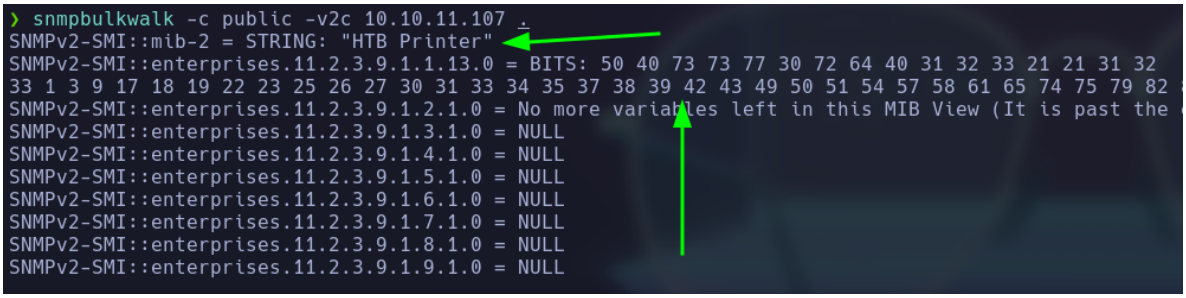
La capacidad de escritura de un OID depende de la cadena comunitaria utilizada. Algunos objetos pueden ser de solo lectura y no se les puede modificar, incluso si se utiliza la cadena comunitaria "public". Si intentas escribir en un objeto de solo lectura, recibirás un error como "noSuchName" o "readOnly".

En las versiones SNMPv1 y SNMPv2/2c, si se utiliza una cadena comunitaria incorrecta, el servidor no responder

Ahora que sabemos un poco acerca de cómo opera SNMP, existen herramientas como snmpwalk o snmpbulkwalk para enumerar este servicio.

Iniciare enumeración para ver si puedo obtener información de la maquina

$snmpbulkwalk -c public -v2c 10.10.11.107

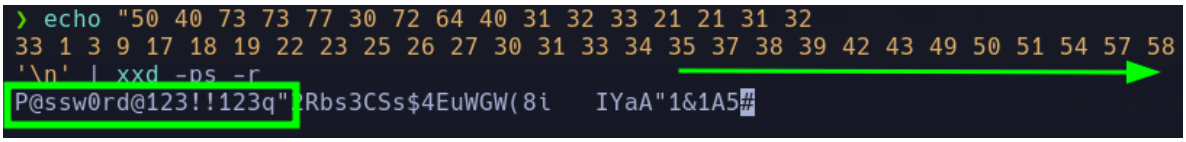


Si ponemos atención en los resultados, podemos identificar un nombre “HTB Printer” y una cade extraña en el apartado BITS:, que por su estructura podemos llegar a determinar que hexadecimal.

50 40 73 73 77 30 72 64 40 31 32 33 21 21 31 32 33 1 3 9 17 18 19 22 23 25 26 27 30 31 33 34 35 37 38 39 42 43 49 50 51 54 57 58 61 65 74 75 79 82 83 86 90 91 94 95 98 103 106 111 114 115 119 122 123 126 130 131 134 135

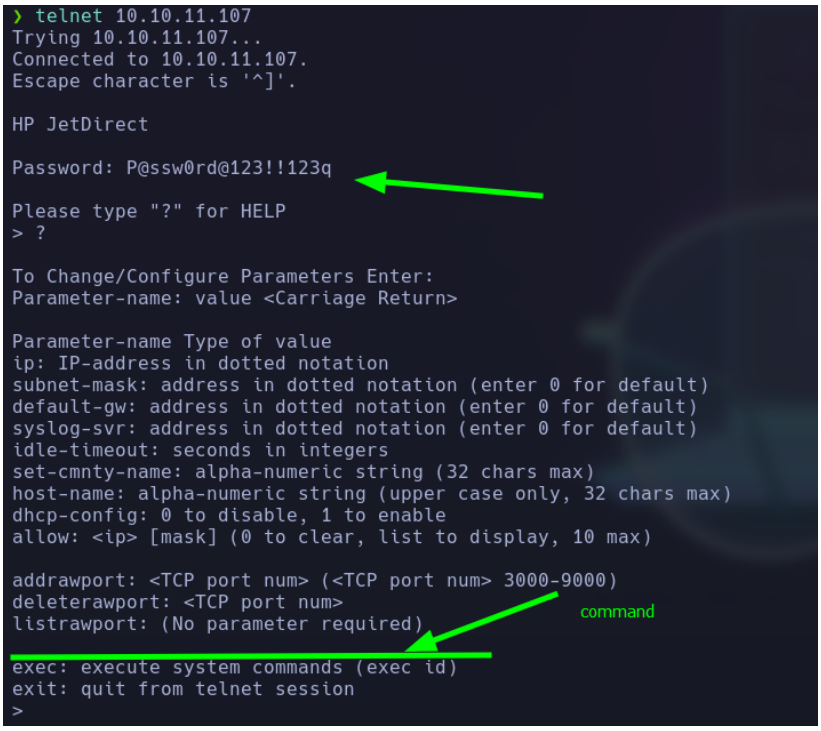
Voy a realizar la decodificación para tratar de determinar si existe un mensaje sobre esta cade en hexadecimal

$ snmpbulkwalk -c public -v2c 10.10.11.107 .

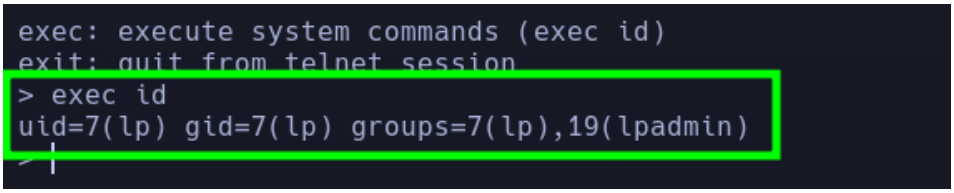


Explotación

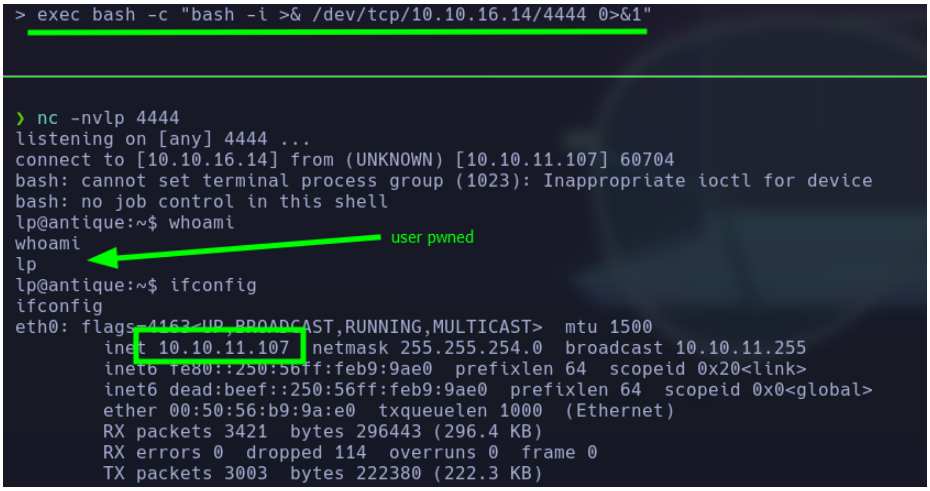
Si mis cálculos no me fallan y estoy en lo correcto, la contraseña que acabo de encontrar es la misma que me pide la impresora



Al parecer si son las credenciales, pero noto que existe una manera de ejecutar comandos (¿¿¿¿¿IMPRESORA CON EJECUCIÓN DE COMANDOS?????)



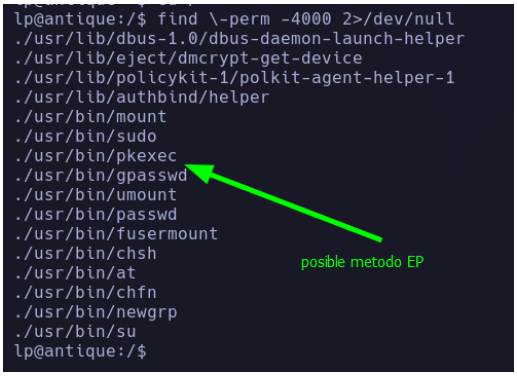
Ahora veo que, si puedo ejecutar comandos, me entablare una reverse shell para ganar acceso a la maquina de la siguiente manera.



Escala de Privilegios

Después de realizar enumeración para obtener una posible forma de escalar privilegios, encontré 2 activos potencialmente atractivos, el primero es el binario pkexec, lo cual podemos abusar de este binario ya que tiene permisos especiales (SUID) y al otra, pude encontrar un puerto corriendo internamente por lo cual puedo realizar Local Port Forwarding

1-

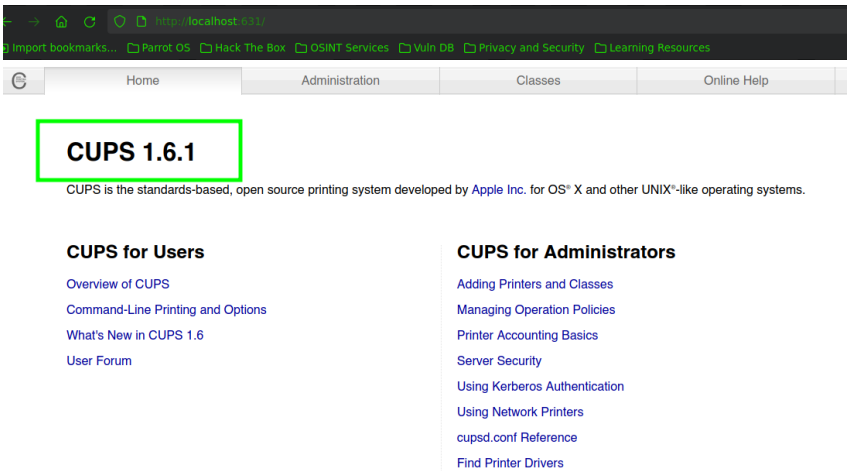


2-

Realizando LPF con chisel y enumerando el puerto 631 en local encuentro lo siguiente

```
cat NewPort -l java
File: NewPort
1   # Nmap 7.93 scan initiated Tue Jun  6 22:35:36 2023 as: nmap -p631 -sCV -n -v --min-rate 5000 -oN NewPort localhost
2   Nmap scan report for localhost (127.0.0.1)
3   Host is up (0.000064s latency).
4   Other addresses for localhost (not scanned): ::1
5
6   PORT    STATE SERVICE VERSION
7   631/tcp open  ipp     CUPS 1.6
8   |_http-title: Home - CUPS 1.6.1
9   | http-robots.txt: 1 disallowed entry
10  |_/
11  | http-methods:
12  |   Supported Methods: GET HEAD OPTIONS POST PUT
13  |_  Potentially risky methods: PUT
14  |_http-server-header: CUPS/1.6
15
16  Read data files from: /usr/bin/../share/nmap
17  Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
18  # Nmap done at Tue Jun  6 22:35:47 2023 -- 1 IP address (1 host up) scanned in 11.25 seconds
```

Tenemos un servicio ipp donde podemos observar la versión cups 1.6.1 y un servicio http



Buscare que es cups y si existe algún exploit relacionado.

CUPS (sistema de impresión común de UNIX)

https://github.com/p1ckzi/CVE-2012-5519

Al parecer existe una vulnerabilidad que permite a los usuarios del grupo lpadmin realizar cambios en el archivo cupsd.conf, con el cupsctl dominio.
este comando también permite al usuario especificar una ruta de ErrorLog.
cuando el usuario visita la página '/admin/log/error_log', el demonio cupsd que se ejecuta con un SUID de root lee la ruta de ErrorLog y la repite en texto sin formato.
en resumen, los archivos propiedad del usuario raíz se pueden leer si la ruta de ErrorLog se dirige allí.

Raptor-Attack

Voy a descargarme este recurso para después ejecutarlo directamente en la maquina víctima.

```
cups-root-file-read.sh
rver 80
```

La forma de ejecutarlo según la descripción del exploit

echo “/root/root.txt” | ./cups-root-file-read.sh

```
[i] ./cups-root-file-read.sh commands:
        type 'info' for exploit details.
        type 'help' for this dialog text.
        type 'quit' to exit the script.
[i] for more information on the limitations
[i] of the script and exploit, please visit:
[i] https://github.com/0zvxr/CVE-2012-5519/blob/main/README.md
[>] [+] contents of /root/root.txt:
fb29e2228e9d216d69cbaf4f3570aeeb
[>] lp@antique:/tmp$
```

EXTRA

Recordando que también podíamos ganar acceso abusando del binario pkexec(Polkit), podemos descargar un recurso de git hub que nos permitirá convertirnos en el usuario root ejecutando un binario compilado del siguiente repositorio

https://github.com/berdav/CVE-2021-4034

Polkit (anteriormente PolicyKit) es un componente para controlar los privilegios de todo el sistema en sistemas operativos similares a Unix. Proporciona una forma organizada para que los procesos no privilegiados se comuniquen con los procesos privilegiados. También es posible usar polkit para ejecutar comandos con privilegios elevados usando el comando pkexec seguido del comando que se pretende ejecutar (con permiso de root).

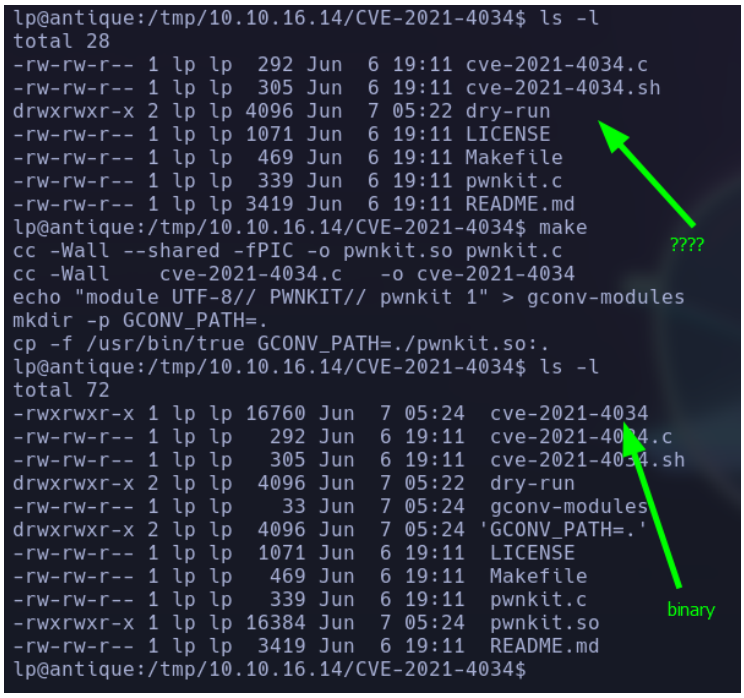
1- Clonamos el repositorio en la maquina acatante

Git clone https://github.com/berdav/CVE-2021-4034

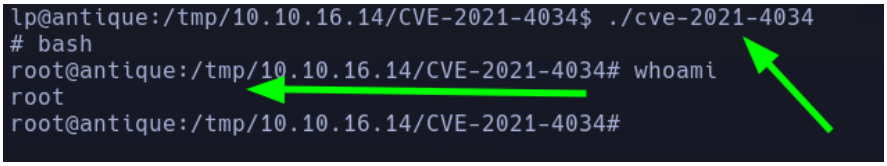
2- Subimos el recurso a la maquina victima (Asegurar de que en la maquina victima tienen gcc y make)

```
lp@antique:/tmp/10.10.16.14$ ls -l
total 4
drwxrwxr-x 3 lp lp 4096 Jun  7 05:22 CVE-2021-4034
lp@antique:/tmp/10.10.16.14$
```

3- Nos metemos a la carpeta donde posteriormente compilaremos el binario ejecutando el comando make



Ejecutamos el binario



PWNED