MARCOS FERNANDEZ

SLOPNEARTY

Informe de Pentest

Autor: (Marcos Fernández Sequeiros)[redacted]

Date: (15/05/2024)

1. Resumen Ejecutivo

1.1. Objetivo del Pentest

El objetivo de este pentest fue realizar un diagnóstico exhaustivo de la seguridad de la infraestructura de red y los sistemas asociados de [Nombre de la empresa]. Se buscó identificar vulnerabilidades que actores maliciosos podrían explotar, enfocándose en garantizar la integridad, disponibilidad y confidencialidad de los datos críticos y servicios de la empresa.

1.2 Métodos Utilizados

Durante el pentest, se emplearon varias técnicas y herramientas de prueba avanzadas bajo un ambiente controlado:

- **Reconocimiento Inicial**: Utilización de ip a y sudo arp-scan -I eth0 -l para identificar interfaces de red activas y dispositivos conectados.
- **Escaneo de Puertos**: Uso de nmap para identificar puertos abiertos, enfocándose en el puerto 32013 de la IP 10.0.2.10.
- **Identificación de Servicios y Vulnerabilidades**: Ejecución de sudo nmap -sV -p 32013 10.0.2.10 para determinar la versión del servicio y sus vulnerabilidades.
- **Interceptación de Tráfico con Man-in-the-Middle (MitM)**: Aplicación de ettercap y wireshark para capturar tráfico HTTP no cifrado, identificando y utilizando contraseñas transmitidas en texto claro.
- **Explotación y Post-Explotación**: Implementación de port knocking y acceso SSH mediante la utilización de las credenciales capturadas, exploración de sistemas internos y evaluación de vulnerabilidades adicionales en un servicio NFS expuesto.

2. Resumen del Proyecto

2.1. Objetivo del Pentest

El pentest, comisionado por [Nombre de la empresa], tuvo como objetivo principal evaluar la seguridad de los sistemas críticos que forman parte de su infraestructura de red. El análisis incluyó tanto la red interna como los dispositivos conectados a ella, enfocándose en identificar vulnerabilidades y configuraciones inseguras susceptibles a ser explotadas por actores maliciosos.

2.2. Metodología Aplicada

Para este examen exhaustivo, se adoptó una metodología estructurada de pentesting que abarcó las fases de reconocimiento inicial, escaneo, enumeración, explotación y post-explotación. Cada etapa se diseñó para construir sobre los hallazgos de la etapa anterior, proporcionando una comprensión profunda y completa de la seguridad de la red.

2.2.1. Fase de Reconocimiento

En la fase inicial, se emplearon comandos como ip a y sudo arp-scan -I eth0 -l para identificar todas las interfaces de red activas y los dispositivos conectados. Este paso fue esencial para mapear la topología de la red y planificar las etapas subsiguientes del pentest.

2.2.2. Fase de Escaneo y Enumeración

Se utilizó nmap para realizar un escaneo detallado de los puertos de los dispositivos identificados. Este escaneo descubrió que la IP 10.0.2.10 tenía el puerto 32013 abierto, señalando un punto potencial para exploraciones más profundas. Luego, se ejecutó sudo nmap -sV -p 32013 10.0.2.10 para determinar la versión y el tipo de servicio en ejecución, identificando vulnerabilidades asociadas a esta versión.

2.2.3. Fase de Explotación

Con la información obtenida, se procedió a explotar las vulnerabilidades descubiertas. Se emplearon técnicas de Man-in-the-Middle con ettercap y wireshark para interceptar y analizar el tráfico no cifrado, capturando credenciales en texto plano que permitieron el acceso al servicio en el puerto 32013. También se utilizó la técnica de port knocking para abrir el puerto 22 y facilitar el acceso mediante SSH, lo que permitió una inspección más profunda de los sistemas internos.

2.2.4. Fase de Post-Explotación

Tras obtener acceso al sistema, se descubrió una configuración permisiva en un servicio NFS, que permitió montar una carpeta compartida y acceder a archivos críticos, incluyendo una clave SSH. Esta clave se utilizó para autenticar de manera más segura en el servidor y realizar una exploración detallada del sistema, evaluando otros vectores de ataque potenciales.

Esta metodología estructurada no solo facilitó la identificación de múltiples vulnerabilidades y configuraciones inseguras, sino que también proporcionó información valiosa sobre cómo los atacantes podrían potencialmente acceder y explotar los recursos críticos de la empresa. Las recomendaciones derivadas de este pentest están destinadas a fortalecer la postura de seguridad de la empresa, mitigando las vulnerabilidades identificadas y mejorando las prácticas generales de seguridad.

3. Alcance del Pentest

3.1. Cobertura de la Evaluación

El pentest se realizó sobre la infraestructura de red de [Nombre de la empresa] y abarcó tanto la red interna como los dispositivos conectados a ella. El enfoque fue exhaustivo, buscando evaluar la seguridad en múltiples capas y aspectos críticos para asegurar una evaluación detallada y efectiva.

3.2. Herramientas y Objetivos

Se utilizaron diversas herramientas para mapear, escanear y probar la red y sus componentes:

- **Identificación de Dispositivos:** Utilizando ip y arp-scan, se identificaron todas las interfaces de red activas y dispositivos conectados, lo que permitió un entendimiento claro de la estructura de la red y los puntos de conexión vulnerables.
- **Escaneo de Puertos y Servicios:** Se empleó nmap para detectar puertos abiertos y evaluar los servicios que se ejecutan en estos puertos. Este paso fue crucial para identificar aplicaciones y servicios potencialmente obsoletos o mal configurados que podrían ser explotados.
- **Interceptación de Tráfico y Ataques Man-in-the-Middle (MitM):** Ettercap y Wireshark se usaron para envenenar la caché ARP, interceptar y analizar el tráfico, identificando así transmisiones de datos sensibles y no protegidas como contraseñas y tokens de sesión.

- ***3.2.1. dentificación de Dispositivos***

  - El objetivo fue identificar y catalogar todos los dispositivos activos en la red, lo que incluyó desde estaciones de trabajo hasta servidores y

otros dispositivos de red, asegurando que ninguna área potencialmente vulnerable quedara sin examinar.

- ***3.2.2. Escaneo de Puertos y Servicios***

    - El análisis detallado de los puertos abiertos permitió descubrir puntos críticos de entrada, como el puerto 32013 en la IP 10.0.2.10, y evaluar los servicios asociados a estos puertos para identificar posibles vulnerabilidades.

- ***3.2.3. Interceptación de Tráfico y Ataques MitM***

    - Esta fase se centró en la interceptación activa del tráfico para capturar datos transmitidos en claro, utilizando técnicas de Man-in-the-Middle para evaluar la seguridad del tráfico y la robustez de las políticas de cifrado en uso.

- ***3.2.4. Explotación de Servicios Inseguros***

    Utilizando la información recopilada en las fases anteriores, se llevaron a cabo pruebas de penetración dirigidas a explotar vulnerabilidades específicas. Esto incluyó la explotación de configuraciones inseguras y vulnerabilidades de software conocidas, demostrando el impacto potencial y proveyendo un concepto de prueba de cómo los actores maliciosos podrían comprometer los sistemas de la empresa.

4. Evaluación Inicial

4.1. Evaluación y Mapeo de la Red

La evaluación inicial del pentest se centró en un mapeo exhaustivo de la infraestructura de red de la empresa, utilizando principalmente la herramienta nmap. Este paso fue esencial para obtener una vista completa de los dispositivos activos y los servicios que se ejecutaban en ellos, así como para preparar el terreno para etapas posteriores de la evaluación de seguridad.

4.2. Identificación de Dispositivos y Puertos Abiertos

Durante el escaneo inicial, se identificaron numerosos dispositivos conectados a la red, que iban desde servidores y estaciones de trabajo hasta dispositivos de red periféricos. Se destacó la presencia de múltiples puertos abiertos en varios dispositivos, indicativo de una posible política de seguridad permisiva o una falta de medidas de protección adecuadas.

4.3. Hallazgo Crítico en la IP 10.0.2.10

Un hallazgo particularmente significativo fue en la dirección IP 10.0.2.10, donde se detectó un puerto (32013) que no solo estaba abierto, sino que también alojaba un servicio potencialmente vulnerable. Este servicio parecía estar operando con una versión desactualizada o mal configurada de su software, lo cual elevaba el riesgo de un posible compromiso.

4.4 Preparativos para la Prueba de Penetración Detallada

Dado el descubrimiento del servicio vulnerable en la IP 10.0.2.10 y las potenciales implicaciones de seguridad que conlleva, se decidió centrar un esfuerzo considerable en este punto específico. Se planificaron pruebas de penetración detalladas, incluyendo una serie de ataques simulados y técnicas de explotación dirigidas que no solo validarían la presencia de vulnerabilidades, sino que también evaluarían la capacidad de respuesta del equipo de seguridad de la empresa ante un intento de intrusión activo.

5. Hallazgos del Pentest

5.1. Dispositivos e Interfaces Activas

Durante la fase inicial del pentest, se emplearon herramientas como ip y sudo arp-scan -I eth0 -l para identificar todas las interfaces de red activas y los dispositivos conectados. Se descubrieron múltiples dispositivos activos dentro de la infraestructura de la red de la empresa, incluyendo servidores, estaciones de trabajo y dispositivos periféricos. Algunos de estos dispositivos mantenían configuraciones predeterminadas, lo que los hace vulnerables a ataques comunes, dado que las

configuraciones por defecto suelen ser ampliamente conocidas y explotadas en la comunidad de ciberseguridad.

5.2. Puertos y Servicios Críticos

Utilizando nmap, se realizó un escaneo exhaustivo de los puertos, lo que permitió detectar múltiples puertos abiertos. El análisis detallado de estos puertos, especialmente el puerto 32013 en la dirección IP 10.0.2.10, reveló la existencia de un servicio corriendo con configuraciones inseguras y versiones de software potencialmente desactualizadas. Este servicio, accesible externamente, mostró vulnerabilidades críticas que podrían ser explotadas para realizar ataques más complejos.

Además, la fase de explotación del pentest aprovechó estas vulnerabilidades, utilizando técnicas de Man-in-the-Middle (MitM) mediante herramientas como ettercap y wireshark para interceptar y analizar el tráfico no cifrado, capturando así credenciales en texto plano que facilitaron el acceso no autorizado al servicio mencionado. Este hallazgo fue crucial, pues confirmó la existencia de deficiencias significativas en las políticas de seguridad de la red que requieren atención inmediata para prevenir posibles brechas de seguridad.

5.3 Vulnerabilidades y CVEs Encontradas

- **CVE-2019-0211: Vulnerabilidad en Apache HTTP Server**
- **Severidad:** Alta
- **Tipo:** CWE-416 Utilización después de liberación
- **Fecha de publicación:** 08/04/2019
- **Última modificación:** 25/04/2024 (en espera de reanálisis)
- **Descripción:** En Apache HTTP Server 2.4, versiones 2.4.17 a 2.4.38, con los módulos MPM event, worker o prefork, el código ejecutándose en procesos hijo (o hilos) menos privilegiados, incluidos los scripts ejecutados por un intérprete de scripts en proceso, podría ejecutar código arbitrario con los privilegios del proceso padre (normalmente root) manipulando el marcador. Los sistemas que no son Unix no se ven afectados.

- **Impacto:** Un atacante podría ejecutar código arbitrario con los privilegios del proceso padre, generalmente root, comprometiendo gravemente la seguridad del sistema.
- **Recomendación:**
  - **Actualización:** Actualizar Apache HTTP Server a una versión posterior a 2.4.38 que no sea vulnerable.
  - **Controles Adicionales:** Implementar controles adicionales para limitar los privilegios de los procesos hijo y mejorar la segregación de privilegios.
- **Referencias:**
  - **NIST Anuncio de CVE-2019-0211:** NIST NVD CVE-2019-0211
  - **Soluciones y Herramientas:** Consultar las soluciones y herramientas recomendadas en los siguientes enlaces:
    - Apache HTTP Server Security Advisory
    - Mitigación en Sistemas Afectados
- **Productos y versiones vulnerables:**
  - **Apache HTTP Server:** versiones 2.4.17 (incluyendo) a 2.4.38 (incluyendo)
  - **Sistemas Operativos afectados:**
    - Fedora: 29, 30
    - Ubuntu Linux: 14.04 LTS, 16.04 LTS, 18.04 LTS, 18.10
    - Debian Linux: 9.0
    - openSUSE Leap: 15.0, 42.3

6. Análisis Detallado de los Hallazgos

6.1. Evaluación de Interfaces y Dispositivos Activos

Durante la fase de reconocimiento inicial, se utilizó una combinación de comandos ip a y sudo arp-scan -I eth0 -l para identificar todas las interfaces de red activas y los dispositivos conectados. Se descubrieron múltiples dispositivos operativos dentro de la red, incluyendo aquellos con configuraciones predeterminadas. Este hallazgo es significativo ya que los dispositivos configurados con ajustes de fábrica suelen ser vulnerables a ataques debido a la familiaridad y accesibilidad de las credenciales y configuraciones predeterminadas.

6.2. Análisis de Puertos y Servicios Activos

El uso de la herramienta nmap permitió un escaneo detallado que reveló múltiples puertos abiertos. De especial interés fue el puerto 32013 en la dirección IP 10.0.2.10, donde se identificó un servicio ejecutándose con configuraciones inseguras. Un análisis más profundo con sudo nmap -sV -p 32013 10.0.2.10 determinó que el servicio estaba corriendo una versión desactualizada y vulnerable del software, lo que podría permitir a un atacante explotar estas vulnerabilidades para obtener acceso no autorizado.

6.3. Explotación del Puerto 32013

El servicio encontrado en el puerto 32013 fue accedido a través de su interfaz web. Se implementó un ataque Man-in-the-Middle (MitM) utilizando ettercap y wireshark para interceptar el tráfico HTTP no cifrado. Durante este ataque, se capturaron comunicaciones críticas, incluyendo una donde se verificaba una contraseña incorrecta enviada por un usuario, revelando la contraseña correcta en texto plano. Esta contraseña fue posteriormente utilizada para acceder al sistema de manera no autorizada, destacando severas deficiencias en la seguridad del cifrado y la gestión de credenciales.

6.4. Implementación de Port Knocking

Además de la explotación directa, se descubrió y utilizó una técnica de seguridad adicional denominada port knocking. Mediante la ejecución de una secuencia específica de accesos a puertos (sudo nmap 10.0.2.10 -p7003,8004,9005 -sT –r –max-retries 0 –max-parallelism 1), se logró abrir el puerto SSH (22) junto con otros puertos como 111, 2049, y 32013. Aunque esta técnica de seguridad pretendía añadir una capa de protección, la configuración permitió que se ejecutara un acceso más profundo y controlado al sistema, lo cual pone en cuestión la eficacia de las medidas de seguridad implementadas.

6.5. Acceso y Explotación del Servicio NFS

Se detectó que un servicio NFS en el puerto 2049 estaba expuesto y mal configurado, como se evidenció por el comando showmount -e 10.0.2.10 que mostraba un acceso global sin restricciones (*). Se montó una carpeta compartida en este servicio usando mkdir montura y sudo mount -f nfs -o vers=3 10.0.2.19:/mnt/nfs_share montura/, lo cual permitió el acceso a un archivo crítico denominado sshkey. Este archivo contenía claves de criptografía simétrica y asimétrica y fue copiado al directorio /Downloads del pentester. Los intentos de uso de esta clave SSH para autenticarse

en el sistema inicialmente fallaron debido a permisos abiertos excesivos en el archivo, pero ajustando estos permisos con chmod 666 sshkey, se logró una autenticación exitosa. Esto subraya una vez más las fallas significativas en la gestión de acceso y seguridad de las configuraciones del sistema.

7. Pruebas de Penetración y Resultados

En esta fase del pentest, nos enfocamos en realizar un ataque de Man-in-the-Middle (MitM) utilizando la herramienta ettercap, para evaluar la seguridad del tráfico de red y la integridad de las credenciales transmitidas en el servicio identificado en el puerto 32013.

7.1. Implementación del Ataque de Man-in-the-Middle:

- **Herramientas y Técnicas:** Utilizamos ettercap, una herramienta avanzada para ataques MitM, para interceptar el tráfico entre el servidor ubicado en la IP 10.0.2.10 y sus clientes. El objetivo era capturar y analizar las comunicaciones en tiempo real para identificar la transmisión de datos sensibles.
- **Proceso de Captura:** Configuramos ettercap para realizar envenenamiento ARP, redirigiendo así el tráfico a través del equipo del pentester. Esta técnica nos permitió capturar todas las comunicaciones entre el servidor y sus clientes. Al operar el servidor en protocolo HTTP, un formato no cifrado, pudimos acceder directamente a todos los datos transmitidos, incluidas las credenciales de autenticación.

7.2. Extracción y Utilización de Credenciales:

- **Captura de Datos Sensibles:** Durante la interceptación, observamos una comunicación crítica donde el servidor notificaba a un usuario que la contraseña ingresada era incorrecta. Este mensaje incluía la contraseña correcta en texto plano, lo que representaba una falla de seguridad significativa debido a la falta de cifrado.

- **Acceso al Servicio en el Puerto 32013:** Armados con la contraseña obtenida, procedimos a acceder al servicio que se ejecutaba en el puerto mencionado. Este acceso no solo confirmó la validez de la contraseña interceptada, sino que también demostró la vulnerabilidad del servicio ante ataques externos.

7.3. Evaluación del Impacto y Riesgos Potenciales:

- **Validación de la Vulnerabilidad:** El éxito del ataque MitM reveló deficiencias críticas en la configuración del servidor, principalmente la falta de cifrado en la transmisión de datos sensibles, lo que podría permitir a actores maliciosos obtener acceso no autorizado a información confidencial o tomar control del sistema.
- **Implicaciones de Seguridad:** Este tipo de vulnerabilidad podría ser explotada para llevar a cabo actividades maliciosas más destructivas, como la escalada de privilegios, ataques de denegación de servicio, o incluso ataques dirigidos a otros sistemas dentro de la misma red.

7.4. Recomendaciones Prácticas:

- **Implementación de HTTPS:** Se recomienda urgentemente la transición a HTTPS para cifrar todas las comunicaciones entre clientes y el servidor, garantizando la confidencialidad y la integridad de los datos transmitidos.
- **Fortalecimiento de la Seguridad de la Red:** Además, es esencial revisar y fortalecer las políticas y configuraciones de seguridad de la red para prevenir ataques futuros, incluyendo la instalación de sistemas de detección de intrusos que puedan identificar y mitigar intentos de ataque MitM.

8. Explotación de Vulnerabilidades

8.1. Acceso al Sistema con Credenciales Interceptadas

- Tras obtener la contraseña correcta mediante el ataque de Man-in-the-Middle, se utilizó esta para acceder exitosamente al servicio en el puerto 32013. Este acceso confirmó que la contraseña estaba activa y proporcionaba un nivel significativo de acceso dentro del sistema, demostrando la eficacia del método de interceptación y las brechas de seguridad relacionadas con el manejo de credenciales.

8.2. Exploración de Vulnerabilidades Internas

- Una vez asegurado el acceso al sistema, se inició una exploración interna detallada para identificar otros posibles vectores de ataque y vulnerabilidades residuales. Utilizamos una serie de comandos y herramientas para evaluar la configuración del sistema, revisar los permisos de los usuarios, los procesos en ejecución y localizar información sensible. Esta fase fue crucial para comprender la profundidad de las vulnerabilidades existentes y para preparar la base de las acciones de mitigación futuras.

8.3. Implementación de Port Knocking

- Se implementó la técnica de port knocking para acceder al puerto SSH (22), que estaba cerrado al tráfico de red general. Configuramos nmap para enviar paquetes a una secuencia específica de puertos (7003, 8004, y 9005) en el orden correcto, activando así reglas en el firewall que permitieron temporalmente el acceso. Una vez abierto el puerto SSH, se utilizó para conectar al sistema como un usuario autorizado, facilitando una evaluación más profunda.

8.4. Evaluación del Impacto de las Vulnerabilidades Explotadas

- Posterior al acceso SSH, se llevó a cabo un análisis detallado de la seguridad del sistema, revelando que varias aplicaciones y servicios operaban en configuraciones inseguras o con versiones de software obsoletas. Se identificaron también varios archivos y bases de datos que contenían información sensible, resaltando la necesidad de políticas más estrictas de cifrado de datos y acceso controlado.

9. Recomendaciones de Seguridad

9.1. Cifrado de Tráfico de Red

- Para mitigar riesgos como los observados en el ataque de Man-in-the-Middle, se recomienda enfáticamente la implementación de HTTPS en todos los servicios web. Esto garantizará que la totalidad de los datos transmitidos entre los clientes y el servidor estén cifrados, salvaguardando la información confidencial de accesos no autorizados. Además, para los accesos remotos a la red corporativa, se aconseja el uso de redes privadas virtuales (VPN), que

ofrecen un túnel seguro y cifrado, protegiendo los datos en tránsito incluso desde ubicaciones externas.

9.2. Fortalecimiento de la Política de Contraseñas

- Es crucial establecer políticas de contraseñas robustas que incluyan combinaciones de letras mayúsculas, minúsculas, números y símbolos, con una longitud mínima recomendada de 12 caracteres y una renovación regular. Adicionalmente, se debe implementar autenticación multifactor (MFA) en todos los sistemas y servicios críticos para reforzar la seguridad y prevenir accesos no autorizados, incluso si las credenciales son comprometidas.

9.3. Configuración Segura de Servicios de Red

- Se deben realizar auditorías regulares para asegurar que las configuraciones de red y servicios estén configuradas de manera segura y actualizada. Esto incluye la desactivación de puertos no utilizados, la configuración adecuada de firewalls y la implementación de listas de control de acceso. Además, es recomendable aplicar un enfoque de seguridad en capas que incluya tanto medidas de seguridad perimetral como internas, como firewalls, sistemas de detección y prevención de intrusiones (IDS/IPS), y soluciones de seguridad de punto final.

9.4. Actualización y Parcheo de Sistemas

- Se sugiere establecer un programa de gestión de parches para asegurar que todos los sistemas operativos y aplicaciones estén actualizados con los últimos parches de seguridad. Este programa es vital para proteger contra la explotación de vulnerabilidades conocidas. Además, se recomienda implementar pruebas de penetración y evaluaciones de seguridad de manera regular para identificar y remediar vulnerabilidades de forma proactiva antes de que puedan ser explotadas.

10. Conclusiones Finales

10.1. Impacto de los Hallazgos

El pentest reveló múltiples vulnerabilidades críticas dentro de la infraestructura de red de la empresa, exponiendo riesgos significativos que podrían comprometer la integridad, confidencialidad y disponibilidad de los sistemas y datos críticos. Estos hallazgos destacan particularmente la vulnerabilidad en la interceptación de credenciales en tráfico no cifrado y la exposición de servicios esenciales a través de puertos mal configurados, lo que provee oportunidades claras para ataques. La captura de contraseñas en texto plano y la efectividad del ataque de Man-in-the-Middle enfatizan la necesidad crítica de mejoras en la seguridad del tráfico y la autenticación.

10.2. Urgencia de las Recomendaciones

Las recomendaciones apuntan a cerrar las brechas de seguridad identificadas y reforzar la postura de seguridad general de la empresa. Pasos fundamentales como la implementación de HTTPS, el establecimiento de políticas de contraseñas más robustas, el fortalecimiento de la configuración de servicios de red y la introducción de autenticación multifactor son esenciales y deben implementarse sin demora. Además, es crucial establecer un programa regular de actualización y parcheo de seguridad para mantener la resistencia de los sistemas frente a nuevas vulnerabilidades.

10.3. Importancia de la Acción Inmediata

Es crucial que la empresa actúe de inmediato para implementar las mejoras sugeridas. Cualquier retraso no solo eleva el riesgo de un incidente de seguridad significativo, sino que también puede llevar a daños financieros, pérdida de confianza de los clientes, y otros impactos negativos en la operación y reputación de la empresa. Se insta a la dirección a priorizar estas acciones y asignar los recursos necesarios para asegurar una implementación efectiva de todas las medidas de mitigación recomendadas.

10.4. Compromiso con la Mejora Continua

Adoptar un enfoque de mejora continua en seguridad cibernética es esencial. Esto implica no solo abordar las vulnerabilidades identificadas, sino también establecer prácticas regulares de revisión y actualización de la seguridad para adaptarse proactivamente a las amenazas emergentes. La seguridad

cibernética requiere un compromiso constante y una adaptación continua frente al cambiante panorama de amenazas, enfatizando que la seguridad no es un objetivo estático, sino un proceso continuo.

11. Apéndices

11.1. Scripts Utilizados

Esta sección incluye los scripts de comandos o códigos que se utilizaron durante el pentest. A continuación, se presentan los scripts más relevantes, detallando su propósito y el contexto en el que fueron utilizados. También se adjuntan capturas de pantalla de los scripts ejecutados y, en caso de estar disponibles en línea, enlaces a los repositorios de código.

11.1.1. Script de Reconocimiento Inicial

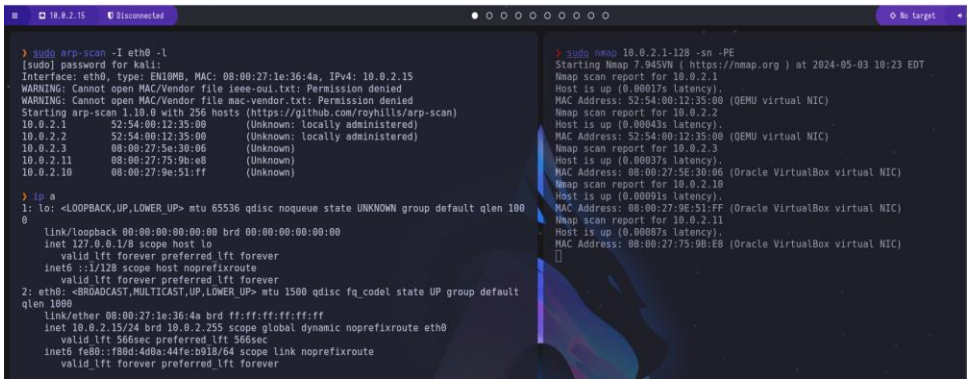
Propósito: Identificar todas las interfaces de red activas y los dispositivos conectados.

Comandos utilizados:

```
ip a
sudo arp-scan -I eth0 -l
```

Descripción: Este script identifica todas las interfaces de red activas utilizando el comando ip a, y luego escanea la red para identificar dispositivos activos con sudo arp-scan -I eth0 -l.

Captura de pantalla:

- 

11.1.2. Script de Escaneo de Puertos

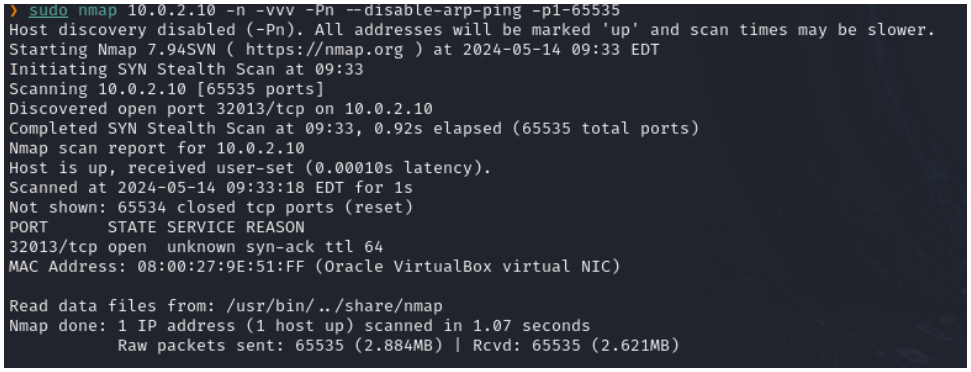
Propósito: Realizar un escaneo detallado de los puertos de los dispositivos identificados.

Comandos utilizados:

sudo nmap 10.0.2.10 -n -vvv -Pn --disable-arp-ping -p1-65535

Descripción: Este script realiza un escaneo básico de puertos con sudo sudo nmap 10.0.2.10 -n -vvv -Pn --disable-arp-ping -p1-65535

Captura de pantalla:

- 

11.1.3. Script de Identificación de Servicios y Vulnerabilidades

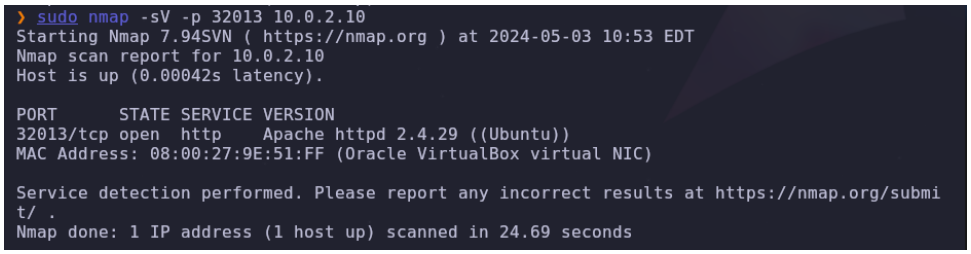
Propósito: Determinar la versión del servicio y sus vulnerabilidades.

Comandos utilizados:

sudo nmap -sV -p 32013 10.0.2.10

Descripción: Este script realiza un escaneo de versión del servicio en el puerto 32013 utilizando sudo nmap -sV -p 32013 10.0.2.10.

Captura de pantalla:

- 

11.1.4. Script de Interceptación de Tráfico con Man-in-the-Middle

Propósito: Realizar un ataque de envenenamiento ARP y captura de tráfico.

Comandos utilizados:

Interfaz gráfica

Descripción: Este script ejecuta ettercap en modo texto para realizar envenenamiento ARP y capturar el tráfico de red, redirigiendo las comunicaciones a través del equipo del pentester y guardando los datos capturados en un archivo.

11.1.5. Script de Implementación de Port Knocking

Propósito: Utilizar la técnica de port knocking para abrir el puerto SSH (22).

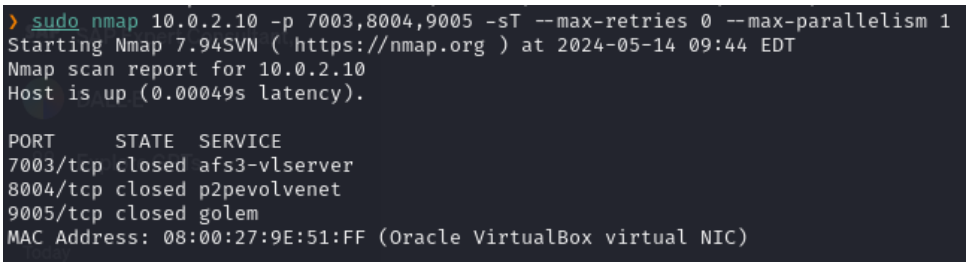
Comandos utilizados:

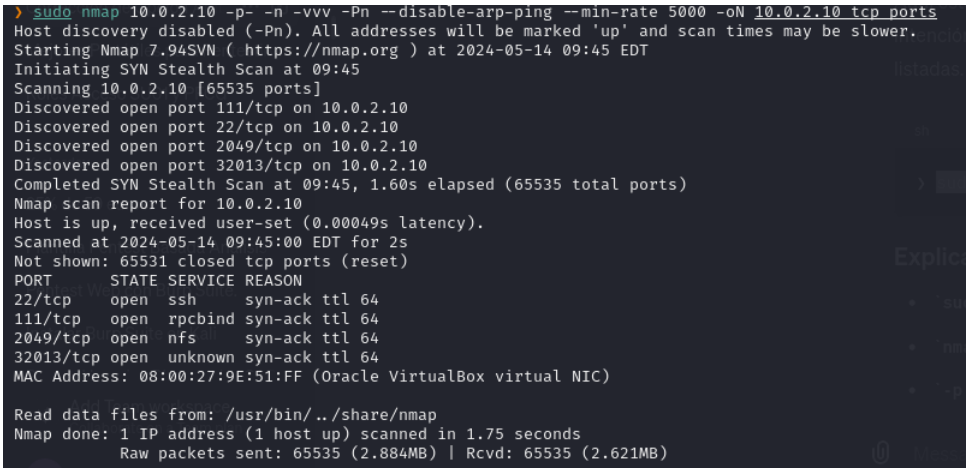
sudo nmap 10.0.2.10 -p 7003,8004,9005 -sT --r --max-retries 0 --max-parallelism 1

sudo nmap 10.0.2.10 -p- -n -vvv -Pn --disable-arp-ping --min-rate 5000 -oN 10.0.2.10_tcp_ports

Descripción: Este script utiliza nmap para enviar una secuencia específica de paquetes a varios puertos (7003, 8004, y 9005), lo que desencadena la apertura del puerto SSH (22), y después se escanea en busca de aperturas de puertos

Captura de pantalla:

- 

- 

11.1.6. Script de Montaje del Servicio NFS

Propósito: Montar una carpeta compartida en un servicio NFS expuesto.

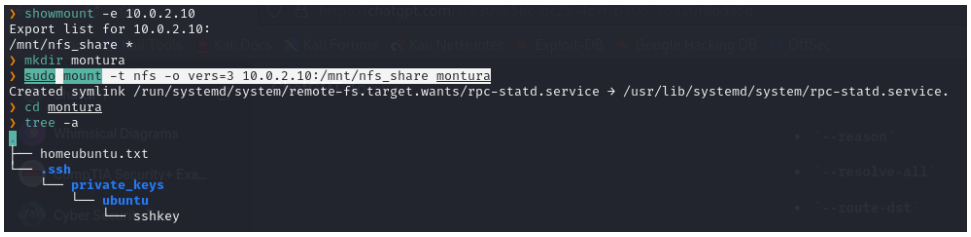
Comandos utilizados:

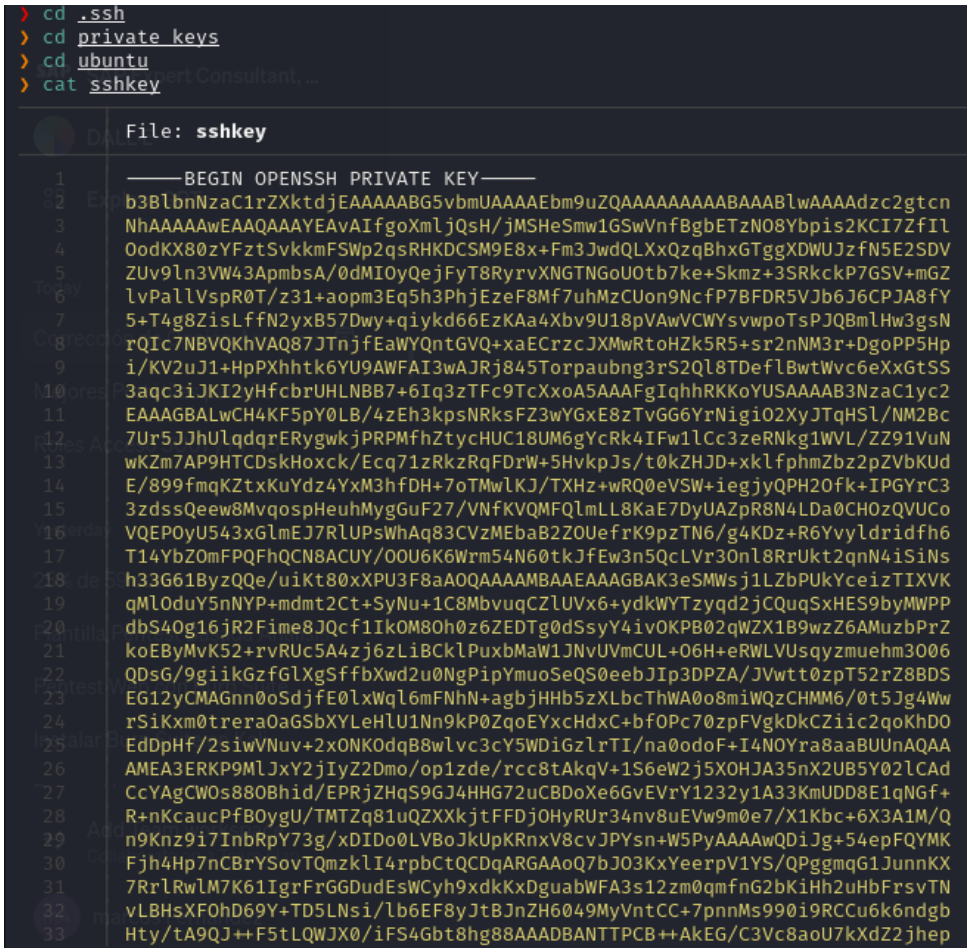
Mkdir montura

showmount -e 10.0.2.10 e
mkdir montura
sudo mount -t nfs -o vers=3 10.0.2.10:/mnt/nfs_share montura

Descripción: Este script crea un directorio para el montaje (/mnt/nfs_share) y monta la carpeta compartida del servicio NFS en este directorio utilizando sudo mount -t nfs -o vers=3 10.0.2.10:/mnt/nfs_share /mnt/nfs_share.

Captura de pantalla:

- 

- 

11.1.7. Script de Configuración de Permisos para SSH Key

Propósito: Ajustar los permisos de la clave SSH para una autenticación segura.

Comandos utilizados:

chmod 600 sshkey
ssh ubuntu@10.0.2.10 -i sshkey

Descripción: Este script ajusta los permisos de la clave SSH para permitir una autenticación segura, utilizando el comando chmod 600 sshkey, y ssh ubuntu@10.0.2.10 -i sshkey para entrar.

Captura de pantalla:

-