



MARCOS FERNANDEZ
SLCP N EARTY

Informe de Pentest

Autor: (Marcos Fernández Sequeiros) [REDACTED]

Date: (15/05/2024)

1. Resumen Ejecutivo

- 1.1. Objetivo del pentest
- 1.2. Métodos utilizados

2. Resumen del Proyecto

- 2.1. [Objetivo del pentest](#)
- 2.2. [Metodología aplicada](#)
 - 2.2.1. [Fase de Reconocimiento](#)
 - 2.2.2. [Fase de Escaneo y Enumeración](#)
 - 2.2.3. [Fase de Explotación](#)
 - 2.2.4. [Fase de Post-Explotación](#)

3. Alcance del Pentest

- 3.1. [Cobertura de la evaluación](#)
- 3.2. [Herramientas y objetivos](#)
 - 3.2.1. [Identificación de dispositivos](#)
 - 3.2.2. [Escaneo de puertos y servicios](#)
 - 3.2.3. [Interceptación de tráfico y ataques MitM](#)
 - 3.2.4. [Explotación de servicios inseguros](#)

4. Evaluación Inicial

- 4.1. [Evaluación y mapeo de la red](#)
- 4.2. [Identificación de dispositivos y puertos abiertos](#)
- 4.3. [Hallazgo crítico en la IP 10.0.2.10](#)
- 4.4. [Preparativos para la prueba de penetración detallada](#)

5. Hallazgos del Pentest

- 5.1. [Dispositivos e interfaces activas](#)
- 5.2. [Puertos y servicios críticos](#)
- 5.3. **5.3 Vulnerabilidades y CVEs Encontradas**

6. Análisis Detallado de los Hallazgos

- 6.1. [Evaluación de interfaces y dispositivos activos](#)
- 6.2. [Análisis de puertos y servicios activos](#)
- 6.3. [Explotación del puerto 32013](#)
- 6.4. [Implementación de Port Knocking](#)
- 6.5. [Acceso y Explotación del Servicio NFS](#)

7. Pruebas de Penetración y Resultados

- 7.1. [Implementación del ataque Man-in-the-Middle](#)
- 7.2. [Extracción y utilización de credenciales](#)
- 7.3. [Evaluación del impacto y riesgos potenciales](#)
- 7.4. [Recomendaciones prácticas](#)

8. Explotación de Vulnerabilidades

- 8.1. [Acceso al sistema con credenciales interceptadas](#)
- 8.2. [Exploración de vulnerabilidades internas](#)
- 8.3. [Implementación de Port Knocking](#)
- 8.4. [Evaluación del impacto de las vulnerabilidades explotadas](#)

9. Recomendaciones de Seguridad

- 9.1. [Cifrado de tráfico de red](#)
- 9.2. [Fortalecimiento de la política de contraseñas](#)
- 9.3. [Configuración segura de servicios de red](#)
- 9.4. [Actualización y parcheo de sistemas](#)

10. Conclusiones Finales

- 10.1. [Impacto de los hallazgos](#)
- 10.2. [Urgencia de las recomendaciones](#)
- 10.3. [Importancia de la acción inmediata](#)
- 10.4. [Compromiso con la mejora continua](#)

11. Apéndices

- 11.1. [Scripts utilizados](#)
- 11.2. [Logs de las pruebas](#)
- 11.3. [Detalles técnicos adicionales](#)

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.
- **Escanear 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. identificación 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 -l 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:

```
18.0.2.15 Disconnected No target

> sudo arp-scan -I eth0 -l
[sudo] password for kali:
Interface: eth0, type: EN10MB, MAC: 08:00:27:1e:36:4a, IPv4: 10.0.2.15
WARNING: Cannot open MAC/Vendor file ieee-oui.txt: Permission denied
WARNING: Cannot open MAC/Vendor file mac-vendor.txt: Permission denied
Starting arp-scan 1.10.0 with 256 hosts (https://github.com/royhills/arp-scan)
10.0.2.1 52:54:00:12:35:00 (Unknown: locally administered)
10.0.2.2 52:54:00:12:35:00 (Unknown: locally administered)
10.0.2.3 08:00:27:5e:39:06 (Unknown)
10.0.2.11 08:00:27:75:9b:e8 (Unknown)
10.0.2.10 08:00:27:9e:51:ff (Unknown)

> ip a
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
    link/loopback 08:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
        valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host noprefixroute
        valid_lft forever preferred_lft forever
2: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default qlen 1000
    link/ether 08:00:27:1e:36:4a brd ff:ff:ff:ff:ff:ff
    inet 10.0.2.15/24 brd 10.0.2.255 scope global dynamic noprefixroute eth0
        valid_lft 565sec preferred_lft 565sec
    inet6 fe80::f8d4:40a:44fe:b918/64 scope link noprefixroute
        valid_lft forever preferred_lft forever

> sudo nmap 10.0.2.1-128 -sn -PE
Starting Nmap 7.94SVN ( https://nmap.org ) at 2024-05-03 10:23 EDT
Nmap scan report for 10.0.2.1
Host is up (0.00017s latency).
MAC Address: 52:54:00:12:35:00 (QEMU virtual NIC)
Nmap scan report for 10.0.2.2
Host is up (0.00043s latency).
MAC Address: 52:54:00:12:35:00 (QEMU virtual NIC)
Nmap scan report for 10.0.2.3
Host is up (0.00037s latency).
MAC Address: 08:00:27:5e:39:06 (Oracle VirtualBox virtual NIC)
Nmap scan report for 10.0.2.10
Host is up (0.00091s latency).
MAC Address: 08:00:27:9e:51:ff (Oracle VirtualBox virtual NIC)
Nmap scan report for 10.0.2.11
Host is up (0.00087s latency).
MAC Address: 08:00:27:75:9b:e8 (Oracle VirtualBox virtual NIC)
```

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

Captura de pantalla:

```
> sudo nmap 10.0.2.10 -n -vvv -Pn --disable-arp-ping -p1-65535
Host discovery disabled (-Pn). All addresses will be marked 'up' and scan times may be slower.
Starting Nmap 7.94SVN ( https://nmap.org ) at 2024-05-14 09:33 EDT
Initiating SYN Stealth Scan at 09:33
Scanning 10.0.2.10 [65535 ports]
Discovered open port 32013/tcp on 10.0.2.10
Completed SYN Stealth Scan at 09:33, 0.92s elapsed (65535 total ports)
Nmap scan report for 10.0.2.10
Host is up, received user-set (0.00010s latency).
Scanned at 2024-05-14 09:33:18 EDT for 1s
Not shown: 65534 closed tcp ports (reset)
PORT      STATE SERVICE REASON
32013/tcp open  unknown syn-ack ttl 64
MAC Address: 08:00:27:9E:51:FF (Oracle VirtualBox virtual NIC)

Read data files from: /usr/bin/./share/nmap
Nmap done: 1 IP address (1 host up) scanned in 1.07 seconds
Raw packets sent: 65535 (2.884MB) | Rcvd: 65535 (2.621MB)
```

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:

```
■ > sudo nmap -sV -p 32013 10.0.2.10
Starting Nmap 7.94SVN ( https://nmap.org ) at 2024-05-03 10:53 EDT
Nmap scan report for 10.0.2.10
Host is up (0.00042s latency).

PORT      STATE SERVICE VERSION
32013/tcp  open  http    Apache httpd 2.4.29 ((Ubuntu))
MAC Address: 08:00:27:9E:51:FF (Oracle VirtualBox virtual NIC)

Service detection performed. Please report any incorrect results at https://nmap.org/submi
t/ .
Nmap done: 1 IP address (1 host up) scanned in 24.69 seconds
```

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 --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:

```
> sudo nmap 10.0.2.10 -p 7003,8004,9005 -sT --max-retries 0 --max-parallelism 1
Starting Nmap 7.94SVN ( https://nmap.org ) at 2024-05-14 09:44 EDT
Nmap scan report for 10.0.2.10
Host is up (0.00049s latency).

PORT      STATE SERVICE
7003/tcp  closed afs3-vlserver
8004/tcp  closed p2pevolvenet
9005/tcp  closed golem
MAC Address: 08:00:27:9E:51:FF (Oracle VirtualBox virtual NIC)

■

> sudo nmap 10.0.2.10 -p- -n -vvv -Pn --disable-arp-ping --min-rate 5000 -oN 10.0.2.10_tcp_ports
Host discovery disabled (-Pn). All addresses will be marked 'up' and scan times may be slower.
Starting Nmap 7.94SVN ( https://nmap.org ) at 2024-05-14 09:45 EDT
Initiating SYN Stealth Scan at 09:45
Scanning 10.0.2.10 [65535 ports]
Discovered open port 111/tcp on 10.0.2.10
Discovered open port 22/tcp on 10.0.2.10
Discovered open port 2049/tcp on 10.0.2.10
Discovered open port 32013/tcp on 10.0.2.10
Completed SYN Stealth Scan at 09:45, 1.60s elapsed (65535 total ports)
Nmap scan report for 10.0.2.10
Host is up, received user-set (0.00049s latency).
Scanned at 2024-05-14 09:45:00 EDT for 2s
Not shown: 65531 closed tcp ports (reset)
PORT      STATE SERVICE REASON
22/tcp    open  ssh     syn-ack ttl 64
111/tcp   open  rpcbind syn-ack ttl 64
2049/tcp  open  nfs     syn-ack ttl 64
32013/tcp open  unknown syn-ack ttl 64
MAC Address: 08:00:27:9E:51:FF (Oracle VirtualBox virtual NIC)

■

Read data files from: /usr/bin/./share/nmap
Nmap done: 1 IP address (1 host up) scanned in 1.75 seconds
Raw packets sent: 65535 (2.884MB) | Rcvd: 65535 (2.621MB)
```

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:

```
> showmount -e 10.0.2.10
Export list for 10.0.2.10:
/mnt/nfs_share *
> mkdir montura
> sudo mount -t nfs -o vers=3 10.0.2.10:/mnt/nfs_share montura
Created symlink /run/systemd/system/remote-fs.target.wants/rpc-statd.service → /usr/lib/systemd/system/rpc-statd.service.
> cd montura
> tree -a
.
├── homeubuntu.txt
├── .ssh
│   ├── private_keys
│   ├── ubuntu
│   └── sshkey
└──
```

```
> cd .ssh
> cd private_keys
> cd ubuntu
> cat sshkey
```

```
-----BEGIN OPENSSH PRIVATE KEY-----
b3BlbnNzaC1rZXktbjEAAAAABG5vbUUAUAAAEbm9uZQAAAAAAAAABAAABlwAAAAAdzc2gtcn
NhAAAAAwEAAQAAAEAvAIfgoXmLjQsH/jMSHeSmw1GSwVnFBgbETzN08Ybpis2KC17Zf1L
OodKX80zYFztSvkkmFSWp2qsRHKDCSM9E8x+Fm3JwdQLXxQzqBhxGTggXDWUJzfN5E2SDV
ZUv9ln3VW43ApmbSA/0dMIOyQejFyT8RyrvXNGTNGoU0tb7ke+Skmz+3SRckP7GSV+mGZ
lvPaLLvSpR0T/z31+aopm3Eq5h3PhjEzeF8Mf7uhMzCUon9NcfP7BFDR5VJb6J6CPJA8fY
5+T4g8ZisLffN2yx857Dwy+qiykd66EzKAa4Xbv9U18pVAwVCWYsvwpoTsPJBmLHw3gsN
rQIc7NBVQKhVAQ87JTnjfEaWYQntGVQ+xaECrzcJXMwRtoH2k5R5+sr2nNM3r+DgoPP5Hp
i/KV2uJ1+HpPXhhtk6YU9AWFAI3wAJRj845Torpaubng3rS2QL8TDefLBwtWvc6eXxGtSS
3aqc3iJKI2yHfcbUHLNBB7+6Iq3zTfc9TcXxoA5AAAFgIqhhrKKoYUAAAAAB3NzaC1yc2
EAAAGBALwCh4KF5pY0LB/4zEh3kpsNRksFZ3wYGxE8zTvGG6YrNigi02XyJTqHSL/NM2Bc
7Ur5JJhUlqdrERYgwkjPRPMfhZtyCHUC18UM6gYcRk4IFw1lCc3zeRNkg1WVL/ZZ91VuN
wKZm7AP9HTCDskHoxck/Ecq71zRkzRqFDrW+5HvKpJs/t0kZHD+xlfpHmZbz2pZVbKUD
E/899fmqKZtxKuYdz4YxM3hFDH+7oTMwLKJ/TXHz+wRQ0eVSW+iegjyQPH20fk+IPGYrC3
3zdssQeeW8MvqospHeuhMyGuF27/VNfKVQMFQlMLL8KaE7DyUAZPr8N4LDa0CHOzQVUCo
VQEP0yU543xGLmEJ7RLUPsWhAq83CVzMEbaB2Z0UefrK9pzTN6/g4KDz+R6Yvylridrfh6
T14YbZomFPQFHqCN8ACUY/00U6K6Wrm54N60tkJfEw3n5QcLvr30n18RrUkt2qnN4iSiNs
h33G61ByzQqe/uiKt80xXPU3F8aAOQAAAAAMBAEAAAGBAK3eSMWsj1LZbPUkYceizTIXVK
qML0duY5nNYP+mdmt2Ct+SyNu+1C8MbvuqCZLUVx6+ydkWYtzyqd2jCQuqSxHES9bYMWPP
dbS40g16jR2Fime8JQcf1IkOM80h0z6ZEDTg0dSsyY4ivOKPB02qWZX189wzZ6AMuzbPrZ
koEByMvK52+rvRUc5A4zj6zLiBCKLPuxbMaW1JNvUVmCUL+O6H+eRWLVUSqyzmuehm3006
QDsG/9giikGzfGLXgSfbXwd2u0NgPipYmuoSeQS0eebJIp3DPZA/JVwt0zpT52rZ8BDS
EG12yCMAGnn0oSdJfE0lxWql6mFNhN+agbjHHb5zXLbcThWA0o8miWqzCHMM6/0t5Jg4Ww
rSiXm0treraOaGSbXYLeHLU1Nn9kP0ZqoEYxcHdxC+bf0Pc70zpFVgkDkCZiic2qoKhD0
EdDpHf/2siwVNuv+2xONK0dqB8wlv3cY5WDiGzLrTI/na0odoF+I4NOYra8aaBUUAAQAA
AMEA3ERKP9MLJxY2jIyZ2Dmo/op1zde/rcc8tAkqV+1S6eW2j5X0HJA35nX2UB5Y02LCad
CcYAgCW0s880Bhid/EPRjZHqS9GJ4HHG72uCBDoXe6GvEVrY1232y1A33KmUDD8E1qNGf+
R+nKcaucPfB0ygu/TMTZq81uQZXXkjTFFDj0HyRUr34nv8uEVw9m0e7/X1Kbc+6X3A1M/Q
n9Knz9i7InbRpY73g/xDIDo0LVBoJkUpKRnxV8cvJPYsn+W5PyAAAAWQDiJg+54epFQYMK
Fjh4Hp7nCBrySovTQmzklI4rpbCtQCDqARGAAoQ7bJ03KxYeerpV1S/QPggmqG1JunnKX
7RrLrWlM7K61IgrFrGGDudEsWCyH9xdKXDGuabWFA3s12zm0qmFnG2bKiHh2uHbFrsvTN
vLBHsXF0hD69Y+TD5LNsI/lb6EF8YJtBJnZH6049MyVntCC+7pnnMs990i9RCCu6k6ndgb
Hty/tA9QJ++F5tLQWJX0/iFS4Gbt8hg88AAADBANTTPCB++AkEG/C3Vc8aoU7kXdZ2jhep
```

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:

```
> ssh ubuntu@10.0.2.10 -i sshkey  
@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@  
@                WARNING: UNPROTECTED PRIVATE KEY FILE!          @  
@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@  
Permissions 0644 for 'sshkey' are too open.  
It is required that your private key files are NOT accessible by others.  
This private key will be ignored.  
Load key "sshkey": bad permissions  
ubuntu@10.0.2.10's password:  
  
> chmod 600 sshkey  
> ssh ubuntu@10.0.2.10 -i sshkey  
Welcome to Ubuntu 18.04 LTS (GNU/Linux 4.15.0-20-generic i686)  
  
 * Documentation:  https://help.ubuntu.com  
 * Management:    https://landscape.canonical.com  
 * Support:       https://ubuntu.com/advantage  
  
Pueden actualizarse 593 paquetes.  
385 actualizaciones son de seguridad.  
  
Last login: Fri Nov  5 22:46:13 2021 from 10.0.40.25  
ubuntu@ubuntu:~$
```

