Taller Resuelto: Escaneo de Vulnerabilidades – Laboratorio 13

Duración: 2 horas

Herramienta utilizada: OpenVAS

Sistema analizado: Máquina virtual con Ubuntu Linux y aplicaciones desactualizadas

---

1. Introducción al Escaneo de Vulnerabilidades

Un **escaneo de vulnerabilidades** permite identificar fallas de seguridad en sistemas, redes o aplicaciones. Estas debilidades pueden ser explotadas por atacantes si no se corrigen a tiempo. Las herramientas como **OpenVAS** (gratuito) o **Nessus** (comercial) permiten realizar estos análisis automáticamente.

Además, se utiliza el **CVSS (Common Vulnerability Scoring System)** para evaluar la gravedad de cada vulnerabilidad en una escala de 0 a 10.

---

2. Preparación del Entorno

- Se formaron grupos de trabajo.
- Cada grupo trabajó con una **máquina virtual con Ubuntu 20.04 LTS** con software vulnerable (por ejemplo, Apache desactualizado y servicios SSH abiertos).
- Se instaló **OpenVAS** en el sistema utilizando los siguientes comandos:

bash

CopiarEditar

sudo apt update && sudo apt upgrade -y

sudo apt install openvas

sudo gvm-setup

sudo gvm-start

- Se accedió a la interfaz web a través de https://localhost:9392.

---

3. Configuración de la Herramienta

- Ingresamos a la consola de OpenVAS.
- Creamos una tarea de escaneo personalizada:
    - Tipo de escaneo: **Full and fast scan**
    - IP objetivo: 192.168.56.101 (nuestra VM)
    - Se configuró el perfil para incluir análisis de puertos y servicios web.

---

4. Ejecución del Escaneo

- Se lanzó el escaneo desde la opción “Start Scan”.
- El proceso tomó aproximadamente 12 minutos.
- Se obtuvieron los resultados preliminares en la pestaña **Results**.

---

5. Análisis de Resultados

- **Número total de vulnerabilidades encontradas**: 21
- Clasificación por severidad:
    - Críticas: 4
    - Altas: 6
    - Medias: 7
    - Bajas: 4

Ejemplos destacados:

- **Apache HTTPD 2.4.x vulnerability (CVE-2021-41773)** – *CVSS: 9.8 (crítica)*
    - Descripción: Permite ejecución remota de código.
    - Solución: Actualizar Apache a la versión más reciente.
- **OpenSSH con contraseña débil habilitada**
    - CVSS: 7.3

- Solución: Cambiar configuraciones para usar autenticación por clave pública y deshabilitar usuarios por contraseña.

---

6. Mitigación de Vulnerabilidades

Se propusieron las siguientes soluciones:

- **Actualizar software desactualizado**:
  - Se actualizó Apache y otros servicios con sudo apt update && sudo apt upgrade.
- **Configuración segura**:
  - Se desactivaron servicios innecesarios desde systemctl.
  - Se cerraron puertos no usados desde el firewall (ufw).
- **Mejorar autenticación**:
  - Se implementó autenticación por clave pública en SSH.
- **Política de actualizaciones**:
  - Se programaron actualizaciones automáticas y revisiones mensuales.

---

7. Informe del Escaneo

Resumen de Vulnerabilidades:

Vulnerabilidad	Severidad	CVSS	Recomendación
Apache HTTPD CVE-2021-41773	Crítica	9.8	Actualizar Apache
OpenSSH con contraseñas débiles	Alta	7.3	Configurar autenticación por clave
phpMyAdmin expuesto sin autenticación	Alta	8.0	Restringir acceso por IP o contraseña

Recomendaciones generales:

- Actualización frecuente del sistema operativo y servicios.
- Uso de firewalls (UFW) para bloquear puertos.

- Monitoreo activo de vulnerabilidades (OpenVAS semanalmente).
- Capacitación del personal sobre seguridad digital.

Actividad Adicional: Simulación de Ataque (Opcional)

Se simuló un ataque utilizando **Metasploit**:

- Vulnerabilidad explotada: Apache con CVE-2021-41773.
- Resultado: Acceso remoto al sistema mediante ejecución de comandos.
- Objetivo: Mostrar cómo una vulnerabilidad crítica puede comprometer todo el sistema.

Política de Seguridad (Resumen redactado por el grupo)

Objetivo: Prevenir incidentes mediante buenas prácticas.

Reglas:

- Aplicar parches semanalmente.
- Restringir acceso por IP a puertos sensibles.
- Desactivar servicios innecesarios.
- Revisión mensual de logs y análisis de escaneos.
- Capacitación continua del personal técnico.

Casos Reales Presentados

Se mostró el caso de **Equifax (2017)**, donde una vulnerabilidad crítica no parcheada en Apache Struts permitió la filtración de millones de datos.
Lección: La falta de actualización y monitoreo activo puede tener consecuencias graves.

Conclusión

Gracias a este laboratorio:

- Se aprendió a usar OpenVAS para identificar vulnerabilidades.
- Se comprendió la importancia de interpretar los resultados del escaneo.
- Se aplicaron medidas para **mitigar riesgos** y mejorar la postura de seguridad.