**Informe Técnico**

**Reconocimiento Activo y Detección de Vulnerabilidades con Nmap y OpenVAS – Lección 1**

**1. Escenario**

Como parte de un equipo de **pruebas de penetración en laboratorio**, se asignó el análisis de la máquina con IP **192.168.56.110**. El objetivo fue identificar los **servicios expuestos**, sus **versiones**, y detectar vulnerabilidades mediante un escaneo activo con **Nmap** y un análisis profundo con **OpenVAS**, documentando los hallazgos y proponiendo mitigaciones.

**2. Metodología**

El ejercicio se desarrolló en las siguientes fases:

1. **Reconocimiento activo** con nslookup y dig para obtener información DNS del objetivo.
2. **Escaneo de puertos y servicios** con nmap -sS -sV -O y nmap --script=vuln para detectar servicios y vulnerabilidades asociadas.
3. **Escaneo de seguridad con OpenVAS** desde su interfaz web, configurando un análisis completo y profundo.
4. **Interpretación de resultados**: selección de vulnerabilidades críticas con referencia a CVEs y evaluación de impacto.
5. **Recomendaciones técnicas**: medidas de mitigación priorizadas según criticidad.

**3. Resultados**

**3.1 Reconocimiento**

* IP analizada: **192.168.56.110**
* Respuesta DNS: host identificado con nombre genérico del laboratorio.

**3.2 Escaneo con Nmap**

Ejemplo de hallazgos:

* **Puerto 22 (SSH):** OpenSSH 7.2 → posible vulnerabilidad CVE-2016-6210.
* **Puerto 80 (HTTP):** Apache httpd 2.4.29 → vulnerabilidad CVE-2019-0211 (escalada de privilegios).
* **Puerto 3306 (MySQL):** expuesto sin filtrado → riesgo de acceso no autorizado.

**3.3 Escaneo con OpenVAS**

El informe reportó múltiples vulnerabilidades, entre ellas:

* **Crítica (CVSS 9.8):** Apache httpd 2.4.29 – CVE-2019-0211.
* **Alta (CVSS 8.1):** OpenSSH 7.2 – vulnerabilidad de enumeración de usuarios.
* **Media (CVSS 6.5):** MySQL con configuraciones por defecto.

**3.4 Vulnerabilidad seleccionada**

* **CVE:** CVE-2019-0211
* **Servicio:** Apache httpd 2.4.29
* **Descripción:** permite a procesos con permisos de usuario no privilegiado ejecutar código arbitrario con privilegios root en el servidor.
* **Impacto:** compromiso completo del sistema, escalada de privilegios, control total del servidor web.
* **Explotación:** mediante ejecución controlada de módulos o scripts alojados en el servidor vulnerable.

**4. Recomendaciones Técnicas**

1. **Actualizar Apache httpd** a la última versión estable.
2. **Restringir accesos externos** a servicios críticos (MySQL, SSH) mediante firewall y segmentación.
3. Implementar **MFA** y endurecimiento de contraseñas en SSH.
4. Configurar **alertas y monitoreo** para detectar intentos de explotación.
5. Aplicar un **ciclo de gestión de vulnerabilidades** con escaneos periódicos y parches.

**5. Reflexión Ética**

El uso de herramientas como **Nmap y OpenVAS** demuestra el valor del análisis activo en auditorías de seguridad, pero también evidencia que este tipo de actividades solo deben ejecutarse en entornos controlados y con autorización expresa. El pentesting sin permiso constituye una actividad ilegal y puede comprometer la estabilidad de sistemas en producción.

**6. Conclusión**

El ejercicio permitió identificar servicios expuestos y vulnerabilidades críticas en el host **192.168.56.110**, confirmando la importancia de realizar auditorías preventivas. La integración de Nmap y OpenVAS brindó una visión detallada del riesgo, destacando la necesidad de aplicar parches y configuraciones seguras. Este laboratorio refuerza que el **pentesting ético** es una herramienta esencial para proteger la **confidencialidad, integridad y disponibilidad** de los sistemas.