**Informe Técnico**

**Escaneo de Subred con Detección de Servicios y Reporte – Lección 6**

**1. Escenario**

El ejercicio consistió en automatizar un **escaneo de subred** para detectar hosts activos, identificar puertos abiertos, reconocer servicios en ejecución y obtener sus versiones. Se trabajó sobre un rango tipo **192.168.1.0/24** en un entorno de laboratorio controlado, con el fin de generar un resumen legible y útil para un reporte técnico de ciberseguridad.

**2. Metodología**

La práctica se desarrolló aplicando las siguientes fases:

1. **Herramientas utilizadas:**
   * nmap para el escaneo de red.
   * Librería python-nmap para integrar resultados en Python.
2. **Parámetros clave:**
   * -sV para detectar versiones de servicios.
   * Escaneo completo de la subred indicada.
3. **Script en Python:**
   * Ejecución del escaneo.
   * Extracción de IPs activas, puertos abiertos, servicios y versiones.
   * Generación de salida ordenada por host.

**3. Implementación Técnica**

**Código en Python (escaneo\_subred.py)**

import nmap

def escanear\_subred(rango="192.168.1.0/24"):

nm = nmap.PortScanner()

print(f"[+] Escaneando subred: {rango}\n")

nm.scan(hosts=rango, arguments="-sV")

hosts\_activos = nm.all\_hosts()

for host in hosts\_activos:

print(f"Host: {host}")

for proto in nm[host].all\_protocols():

puertos = nm[host][proto].keys()

for puerto in sorted(puertos):

servicio = nm[host][proto][puerto]['name']

version = nm[host][proto][puerto]['version']

print(f" - Puerto {puerto}: {servicio} ({version})")

print()

print(f"Total de hosts activos: {len(hosts\_activos)}")

if \_\_name\_\_ == "\_\_main\_\_":

escanear\_subred("192.168.1.0/24")

**4. Ejemplo de Ejecución**

[+] Escaneando subred: 192.168.1.0/24

Host: 192.168.1.10

- Puerto 22: ssh (OpenSSH 7.6)

- Puerto 80: http (Apache httpd 2.4.29)

Host: 192.168.1.15

- Puerto 139: netbios-ssn

- Puerto 445: microsoft-ds

Total de hosts activos: 2

**5. Análisis de Seguridad**

* Los servicios detectados evidencian la **superficie de ataque** disponible.
* **SSH** expuesto requiere verificación de versiones y políticas de contraseñas.
* **HTTP sin cifrar** representa riesgo de intercepción y manipulación de tráfico.
* **NetBIOS y SMB** pueden abrir la puerta a ataques de enumeración de usuarios o explotación de vulnerabilidades conocidas.

**6. Recomendaciones de Ciberseguridad**

1. Limitar la exposición de servicios críticos únicamente a redes internas confiables.
2. Usar **firewalls** y segmentación para reducir la superficie de ataque.
3. Sustituir **HTTP por HTTPS** con certificados válidos.
4. Mantener actualizado el software de los servicios detectados.
5. Monitorizar y registrar accesos a puertos sensibles como 22, 139 y 445.
6. Realizar escaneos periódicos para detectar nuevos servicios expuestos.

**7. Reflexión**

Este laboratorio permitió experimentar con el uso de **nmap y Python** para la automatización de escaneos en ciberseguridad. Se evidenció que los resultados deben interpretarse cuidadosamente para priorizar medidas de mitigación. Además, resaltó la importancia de realizar estos análisis solo en entornos autorizados, respetando los principios de ética profesional.

**8. Conclusión**

La automatización del escaneo de subred con **nmap + python-nmap** permitió obtener un panorama detallado de hosts activos y servicios en ejecución. El ejercicio reforzó la relevancia del reconocimiento en pruebas de penetración y la necesidad de reportar los hallazgos con enfoque en riesgos y mitigaciones. Esta práctica demuestra que la combinación de **herramientas de escaneo y scripting** constituye una estrategia eficaz para fortalecer la seguridad en redes empresariales.