El objetivo de esta actividad es realizar un ejercicio de pentesting (pruebas de penetración) en una red utilizando el comando Nmap.

Nmap es una herramienta de código abierto que se utiliza para escanear redes, descubrir hosts y servicios en una red, y detectar posibles vulnerabilidades.

Lo primero que necesitas hacer es asegurarte de tener Nmap instalado en tu sistema. Dependiendo de tu sistema operativo, la instalación puede variar. Por ejemplo, en sistemas basados en Debian/Ubuntu, puedes instalar Nmap ejecutando el siguiente comando en la terminal: sudo apt-get install nmap

Antes de escribir cualquier código, es importante entender cómo funciona el escaneo con Nmap y qué opciones están disponibles. Puedes consultar la documentación de Nmap o ejecutar nmap --help en tu terminal para obtener más información sobre las opciones disponibles y cómo utilizarlas.

Sí, puedes instalar Nmap en sistemas Windows. Aquí tienes los pasos para hacerlo:

Descargar Nmap: Visita el sitio web oficial de Nmap en https://nmap.org/download.html y descarga la versión más reciente de Nmap para Windows. Puedes elegir entre la versión instalable (con interfaz gráfica) o la versión de línea de comandos. Para este ejercicio, te recomendaría descargar la versión de línea de comandos.

Instalar Nmap: Una vez que hayas descargado el archivo de instalación, ábrelo y sigue las instrucciones del instalador para completar la instalación de Nmap en tu sistema Windows. Durante la instalación, puedes elegir la ubicación donde deseas instalar Nmap en tu sistema.

Verificar la instalación: Después de completar la instalación, puedes abrir una ventana de línea de comandos (cmd) y ejecutar el siguiente comando para verificar que Nmap se haya instalado correctamente:

nmap --version

Si la instalación se realizó correctamente, este comando debería mostrar la versión de Nmap que has instalado. Ahora que tienes Nmap instalado en tu sistema Windows, puedes seguir adelante con el siguiente paso: escribir un script en Python para ejecutar un escaneo de Nmap.

Enfoque en Linux (usando una máquina virtual):

Configurar una máquina virtual:

Descarga e instala un software de virtualización como VirtualBox.

Descarga una imagen de una distribución de Linux como Ubuntu desde su sitio web oficial.

Crea una nueva máquina virtual en VirtualBox y configúrala para usar la imagen de Ubuntu descargada.

Instalar Nmap:

Una vez que la máquina virtual esté configurada y en funcionamiento, abre una terminal en Ubuntu.

Usa el comando sudo apt-get install nmap para instalar Nmap en la máquina virtual.

Ejecutar el script en Python:

Crea y ejecuta un script en Python similar al que te proporcioné anteriormente en la máquina virtual de Ubuntu.

Este script ejecutará un escaneo de Nmap en la red desde la máquina virtual de Linux y generará un reporte.

Enfoque en Windows:

Descargar e instalar Nmap:

Descarga la versión de Nmap para Windows desde el sitio web oficial de Nmap.

Ejecuta el archivo de instalación descargado y sigue las instrucciones del instalador para completar la instalación de Nmap en tu sistema Windows.

Ejecutar el script en Python:

Crea y ejecuta un script en Python similar al que te proporcioné anteriormente en tu sistema Windows.

Este script utilizará la instalación de Nmap en Windows para ejecutar un escaneo de Nmap en la red y generará un reporte.

Comparación y análisis:

Después de ejecutar los scripts en ambas plataformas, compara los resultados obtenidos en cada sistema. Observa si hay diferencias en el escaneo de la red, en la detección de vulnerabilidades y en la generación del reporte.

Este enfoque te permitirá comprender las diferencias entre realizar actividades de pentesting en sistemas Linux y Windows y te dará una visión más amplia sobre las herramientas y técnicas utilizadas en cada plataforma.

Paso 1: Descargar e instalar VMWare

Ve al sitio web oficial de VMWare (<https://www.vmware.com/>) y descarga la versión adecuada de VMWare Workstation Player para tu sistema operativo.

Una vez que se complete la descarga, sigue las instrucciones del instalador para instalar VMWare Workstation Player en tu sistema.

Paso 2: Descargar la imagen de Ubuntu

Visita el sitio web oficial de Ubuntu (<https://ubuntu.com/download/desktop>) y descarga la imagen de Ubuntu Desktop 64-bit.

Paso 3: Crear una nueva máquina virtual en VMWare

Abre VMWare Workstation Player en tu sistema.

Haz clic en "File" (Archivo) en la barra de menú y selecciona "New Virtual Machine" (Nueva máquina virtual).

En el asistente para nueva máquina virtual, selecciona "Typical" (Típico) y haz clic en "Next" (Siguiente).

Selecciona "Installer disc image file (iso)" (Archivo de imagen de disco del instalador ISO) y haz clic en "Browse" (Examinar) para seleccionar la imagen de Ubuntu que descargaste anteriormente. Luego, haz clic en "Next" (Siguiente).

Selecciona "Linux" como el sistema operativo y "Ubuntu" como la versión. Haz clic en "Next" (Siguiente).

Especifica un nombre y una ubicación para la máquina virtual y haz clic en "Next" (Siguiente).

Especifica el tamaño de disco deseado para la máquina virtual y haz clic en "Next" (Siguiente).

Haz clic en "Finish" (Finalizar) para crear la máquina virtual.

Paso 4: Instalar Ubuntu en la máquina virtual

Selecciona la máquina virtual que acabas de crear en la lista de VMWare y haz clic en "Play virtual machine" (Iniciar máquina virtual).

Sigue las instrucciones en pantalla para instalar Ubuntu en la máquina virtual. Durante la instalación, puedes elegir el idioma, la distribución del teclado, el nombre de usuario, la contraseña, etc.

Una vez que se complete la instalación, Ubuntu se iniciará en la máquina virtual.

¡Ahora has configurado con éxito una máquina virtual con Ubuntu en VMWare! En el próximo paso, instalaremos Nmap en Ubuntu y luego crearemos y ejecutaremos un script en Python para realizar un escaneo de red. ¿Listo para continuar?

Paso 5: Instalar Nmap en Ubuntu

Abre una terminal en Ubuntu. Puedes hacerlo buscando "Terminal" en el menú de aplicaciones o usando el atajo de teclado Ctrl+Alt+T.

En la terminal, ejecuta el siguiente comando para actualizar el índice de paquetes:

sudo apt update

Después de que se actualice el índice de paquetes, ejecuta el siguiente comando para instalar Nmap:

sudo apt install nmap

Ingresa la contraseña de administrador (sudo) cuando se te solicite y presiona Enter. Acepta cualquier mensaje de confirmación para completar la instalación.

Una vez que la instalación esté completa, verifica que Nmap se haya instalado correctamente ejecutando el siguiente comando:

nmap --version

Deberías ver la versión de Nmap instalada en tu sistema.

Paso 6: Crear y ejecutar un script en Python para realizar un escaneo de red

Abre un editor de texto en Ubuntu (puedes usar el editor de texto "Nano" que viene preinstalado o cualquier otro de tu preferencia Para abrir un archivo de texto usando Nano, simplemente puedes escribir en la terminal:nano nombre\_del\_archivo Esto abrirá el archivo especificado en Nano. Puedes editar el contenido del archivo, guardar los cambios y salir del editor utilizando los atajos de teclado que se muestran en la parte inferior de la ventana de Nano. Por ejemplo, para guardar los cambios y salir de Nano, puedes presionar Ctrl + O, luego Enter para confirmar el nombre del archivo, y finalmente Ctrl + X para salir de Nano).

Copia y pega el siguiente código en el editor de texto para crear un script en Python que ejecute un escaneo de red utilizando Nmap:

import os (Esta línea importa el módulo os, que proporciona funciones para interactuar con el sistema operativo, como ejecutar comandos en la terminal.)

def run\_nmap\_scan(ip\_address):

""" Esta función ejecuta un escaneo de Nmap en la dirección IP especificada. Esta función recibe una dirección IP como argumento y ejecuta un escaneo de Nmap en esa dirección IP.

"""

command = f'nmap -sV -Pn {ip\_address}'

((En Python, f'strings' es una sintaxis llamada "f-strings" que permite formatear cadenas de texto de manera más fácil e intuitiva. En este caso, {ip\_address} es un "placeholder" que será reemplazado por el valor de la variable ip\_address.

nmap -sV -Pn {ip\_address}

Esto es el comando de Nmap que se está construyendo. Desglosemos cada parte:

nmap: Es el comando de Nmap que se utilizará para realizar el escaneo.

-sV: Es una opción de Nmap que indica que se debe realizar un escaneo de versiones para detectar servicios y sus versiones.

-Pn: Es una opción de Nmap que indica que no se debe realizar el descubrimiento de hosts. Esto significa que Nmap no intentará determinar si los hosts están activos antes de escanearlos.

{ip\_address}: Este es el placeholder que se reemplazará por la dirección IP que el usuario ingresó cuando se le solicitó en la función main().

Entonces, en resumen, esta línea construye un comando de Nmap con opciones específicas (-sV para escanear versiones de servicios y -Pn para deshabilitar el descubrimiento de hosts) y una dirección IP específica que se proporciona como argumento (ip\_address). Este comando se utiliza para realizar el escaneo de red en la dirección IP especificada.

))

result = os.popen(command).read() (Utiliza la función os.popen() para ejecutar el comando de Nmap en la terminal y devuelve el resultado del escaneo.)

return result

def save\_report(report, filename):

""" Esta función guarda el reporte obtenido en un archivo de texto. Esta función recibe un reporte y un nombre de archivo como argumentos y guarda el reporte en un archivo de texto con el nombre especificado.

"""

with open(filename, 'w') as file: (Utiliza un bloque with para abrir el archivo en modo escritura y escribir el reporte en él.)

file.write(report) : (es una línea que se encuentra dentro de la función save\_report. Esta línea se encarga de escribir el contenido del reporte en un archivo de texto.

Desglosemos esta línea:

file: Es el objeto de archivo que se abrió previamente en modo escritura utilizando la función open(). Este objeto representa el archivo en el que se va a escribir el reporte.

write(report): Es un método del objeto de archivo que se utiliza para escribir datos en el archivo. En este caso, report es el contenido que se desea escribir en el archivo. report es el resultado del escaneo de Nmap que se obtuvo previamente y se proporciona como argumento a la función save\_report.

Por lo tanto, esta línea toma el contenido del reporte (que se pasa como argumento a la función save\_report) y lo escribe en el archivo de texto abierto en modo escritura.

En resumen, file.write(report) es la operación que realmente escribe el contenido del reporte en el archivo de texto.

)

def main():

""" Esta es la función principal del script. Solicita al usuario que ingrese la dirección IP que desea escanear.

"""

ip\_address = input("Introduce la dirección IP que deseas escanear: ")

report = run\_nmap\_scan(ip\_address) (Ejecuta un escaneo de Nmap en la dirección IP ingresada utilizando la función run\_nmap\_scan.)

save\_report(report, 'nmap\_report.txt') (Guarda el reporte obtenido en un archivo de texto)

print("El escaneo ha finalizado. El reporte se ha guardado en 'nmap\_report.txt'.")

(Imprime un mensaje indicando que el escaneo ha finalizado y el reporte se ha guardado correctamente.)

Llamar a la función main si el script se ejecuta directamente:

if \_\_name\_\_ == "\_\_main\_\_":: Esta línea verifica si el script se está ejecutando directamente (no importado como un módulo).

(Si es así, llama a la función main() para iniciar la ejecución del script.)

main()

Guarda el archivo con un nombre descriptivo y la extensión ".py" en tu directorio de trabajo.

Abre una terminal y navega hasta el directorio donde guardaste el script en Python.

Ejecuta el script utilizando el siguiente comando:

python nombre\_del\_script.py

Sigue las instrucciones en pantalla para ingresar la dirección IP que deseas escanear.

¡Y eso es todo! Ahora has creado y ejecutado un script en Python que utiliza Nmap para realizar un escaneo de red en Ubuntu. El script generará un reporte que se guardará en un archivo de texto llamado "nmap\_report.txt". ¿Hay algo más en lo que pueda ayudarte

Ahora con Windows

Por supuesto, vamos a realizar los pasos para configurar y ejecutar el escaneo de red utilizando Nmap en un sistema Windows. Aquí tienes los pasos:

Paso 1: Descargar e instalar Nmap en Windows

Ve al sitio web oficial de Nmap: https://nmap.org/download.html

Descarga el instalador de Nmap para Windows.

Una vez que se complete la descarga, haz doble clic en el archivo descargado para iniciar el instalador.

Sigue las instrucciones del instalador para completar la instalación de Nmap en tu sistema Windows.

Paso 2: Crear un nuevo script en Python

Abre tu editor de texto preferido en Windows. Puedes utilizar Notepad, Notepad++, Sublime Text, Visual Studio Code u otro editor de tu elección.

Copia y pega el siguiente código en el editor de texto para crear un nuevo script en Python:

python

Copy code

import subprocess

def run\_nmap\_scan(ip\_address):

"""

Esta función ejecuta un escaneo de Nmap en la dirección IP especificada.

"""

command = f'nmap -sV -Pn {ip\_address}'

result = subprocess.run(command, capture\_output=True, text=True, shell=True)

return result.stdout

def save\_report(report, filename):

"""

Esta función guarda el reporte obtenido en un archivo de texto.

"""

with open(filename, 'w') as file:

file.write(report)

def main():

"""

Esta es la función principal del script.

"""

ip\_address = input("Introduce la dirección IP que deseas escanear: ")

report = run\_nmap\_scan(ip\_address)

save\_report(report, 'nmap\_report.txt')

print("El escaneo ha finalizado. El reporte se ha guardado en 'nmap\_report.txt'.")

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

main()

Paso 3: Guardar y ejecutar el script en Python

Guarda el archivo con un nombre descriptivo y la extensión ".py" en una ubicación de tu elección en tu sistema Windows.

Abre una ventana de línea de comandos (CMD) en Windows.

Navega hasta el directorio donde guardaste el archivo del script en Python usando el comando cd.

Ejecuta el script usando el siguiente comando:

Copy code

python nombre\_del\_script.py

Sigue las instrucciones en pantalla para ingresar la dirección IP que deseas escanear.

Una vez que completes estos pasos, el script ejecutará un escaneo de Nmap en la dirección IP especificada y generará un reporte que se guardará en un archivo de texto llamado "nmap\_report.txt" en el mismo directorio que el script.

Sí, definitivamente puedes ejecutar el script en otro entorno además de Windows o Linux. El lenguaje de programación Python es multiplataforma, lo que significa que los scripts escritos en Python pueden ejecutarse en una variedad de sistemas operativos, incluyendo macOS, FreeBSD y otros sistemas UNIX-like, así como en sistemas embebidos y en la nube.

En estos sistemas, seguirías los mismos pasos básicos para ejecutar el script:

Instala Python si no está preinstalado en el sistema.

Instala Nmap si no está preinstalado en el sistema.

Ejecuta el script en Python usando el intérprete de Python en el sistema.

El proceso puede variar ligeramente dependiendo del sistema operativo específico, pero los conceptos básicos serían los mismos.

Si tienes alguna preferencia o requisito específico para ejecutar el script en un entorno diferente, por favor házmelo saber y estaré encantado de proporcionarte instrucciones específicas para ese entorno.

En el script se ha especificado la opción -Pn al ejecutar el comando de Nmap (nmap -sV -Pn {ip\_address}), lo que indica que no se realice el descubrimiento de hosts. Esto significa que Nmap no intentará determinar si los hosts están activos antes de escanearlos.

La razón principal para especificar esta opción puede ser la velocidad y eficiencia del escaneo. Cuando se realiza el descubrimiento de hosts, Nmap intenta determinar qué hosts están activos en la red antes de iniciar el escaneo de puertos en esos hosts. Esto implica enviar paquetes de solicitud (como paquetes ICMP Echo Request) a múltiples direcciones IP en la red y esperar respuestas.

En algunos casos, especialmente en redes grandes o redes donde la respuesta a las solicitudes de descubrimiento de hosts puede ser limitada o estar bloqueada, el descubrimiento de hosts puede llevar mucho tiempo o incluso no ser exitoso. Al especificar la opción -Pn, se evita este proceso de descubrimiento de hosts y se procede directamente al escaneo de puertos en la dirección IP especificada.

Es importante tener en cuenta que omitir el descubrimiento de hosts puede afectar la precisión de los resultados del escaneo, especialmente si no estás seguro de qué hosts están activos en la red. Sin embargo, en algunos casos, cuando se conoce la dirección IP objetivo y se desea realizar un escaneo rápido y directo, desactivar el descubrimiento de hosts puede ser una opción válida.