CTF Hacking ético

Por
David Fernández Alejo

Dirigido por
Raimundo Alcázar Quesada

CTF Ethical Hacking

Madrid, 2024–2025

CTF Hacking ético

CTF Ethical hácking

Memoria que se presenta para el Trabajo de Fin de Grado

David Fernández Alejo

Dirigido por

Raimundo Alcázar Quesada

Madrid, 2025

Agradecimientos

Resumen

Palabras clave
*****

Índice general

Página

Índice de figuras

Índice de tablas

Parte A

Contenido de la memoria

Capítulo 1

Introducción

En este capítulo se explica la motivación, los objetivos y la estructura de trabajo.

1.1. Motivación

Para profundizar en los conceptos estudiados durante el máster, se realizará el despliegue de los principales servicios que la mayoría de las empresas utilizan para gestionar su infraestructura. Estos servicios se instalarán con una serie de vulnerabilidades intencionadas, de modo que la máquina resultante pueda ser utilizada por otros usuarios para identificar y mitigar dichas vulnerabilidades. Para automatizar el despliegue de los servicios se utilizará ansible [1], una herramienta de automatización de configuración, gestión de sistemas y orquestación de software. Dicha herramienta se usa en entornos de administración de sistemas y DevOps debido a su simplicidad, uno de los rasgos distintivos de Ansible es su capacidad para funcionar sin agentes adicionales instalados en los nodos gestionados.Esto significa que Ansible se comunica con los servidores a través de protocolos estándar, como SSH en sistemas Unix o WinRM en Windows, sin requerir software especial en cada máquina. [2]

1.2. Objetivos

Este Trabajo Fin de Máster se centra en el diseño y creación de una máquina tipo Capture The Flag (CTF), un formato muy utilizado hoy en día para poner a prueba habilidades prácticas en ciberseguridad. La idea general es montar un servidor que actúe como entorno realista de pruebas, simulando una pequeña infraestructura con algunos de los servicios que normalmente se encuentran en redes corporativas: DNS, LDAP, SSH, SMB, un firewall y un servidor web.

En este servidor se instalarán y configurarán esos servicios, pero de forma que presenten ciertas vulnerabilidades específicas, con el objetivo de que puedan ser identificadas y explotadas durante una fase posterior del trabajo. Cada uno de estos servicios ocultará una flag, que servirá como prueba de que la vulnerabilidad ha sido descubierta y aprovechada con éxito.

El desarrollo del proyecto se plantea en dos partes. Por un lado, se explicará el proceso de despliegue de todos estos servicios, lo que permitirá dejar documentada una forma

sencilla y repetible de construir el entorno. Por otro lado, se dedicará una sección a la resolución práctica del entorno, documentando paso a paso cómo se lleva a cabo el análisis y la explotación de cada servicio, desde los primeros reconocimientos hasta la obtención de acceso y la localización de las flags.

Este enfoque permitirá explorar de forma técnica las principales debilidades que pueden presentar servicios básicos de cualquier sistema y, al mismo tiempo, aplicar de forma práctica los conocimientos adquiridos durante el máster. La intención es que el resultado no solo sea útil como experiencia formativa, sino también como recurso reutilizable para otras personas que quieran practicar o aprender sobre seguridad informática en un entorno realista y controlado.

Como máquina base para desarrollar el CTF se utilizará la imagen de Ubuntu 22.04 LTS (Jammy Jellyfish) [3].

1.3. Organización de la memoria

A continuación se describe de manera breve la estructura de la memoria:

- **Capítulo 1:**
- **Capítulo 2:**
- **Capítulo 3:**
- **Capítulo 4:**
- **Capítulo 5:**
- **Capítulo 6:**
- **Capítulo 7:**
- **Capítulo 8:**
- **Capítulo 9:**
- **Capítulo 10:**
- **Anexo I:**
- **Anexo II:**

Capítulo 2

Instalación de los principales servicios

En este capítulo se describe la instalación de los principales servicios.

2.1. DNS

2.1.1. Descripción del servicio

2.1.2. Instalación del servicio

2.2. Firewall

2.2.1. Descripción del servicio

2.2.2. Instalación del servicio

2.3. LDAP

2.3.1. Descripción del servicio

2.3.2. Instalación del servicio

2.4. OpenSSH

2.4.1. Descripción del servicio

OpenSSH [4] es una potente colección de herramientas para controlar remotamente ordenadores en red y transferir datos entre ellos. Aquí describiremos algunos de los ajustes de configuración posibles con la aplicación de servidor OpenSSH y cómo cambiarlos en su sistema Ubuntu.

OpenSSH es una versión de libre acceso de la familia de herramientas del protocolo Secure Shell (SSH). Las herramientas tradicionales, como telnet o rcp, son inseguras y transmiten la contraseña del usuario en texto claro cuando se utilizan. OpenSSH ofrece un demonio

servidor y herramientas cliente que facilitan las operaciones seguras y cifradas de control remoto y transferencia de archivos, sustituyendo así a las herramientas tradicionales.

El componente servidor de OpenSSH, sshd, está continuamente a la escucha de conexiones cliente procedentes de cualquiera de las herramientas cliente. Cuando se produce una solicitud de conexión, sshd establece la conexión correcta en función del tipo de herramienta cliente que se conecte. Por ejemplo, si el equipo remoto se conecta con la aplicación cliente SSH, el servidor OpenSSH establece una sesión de control remoto tras la autenticación. Si un usuario remoto se conecta a un servidor OpenSSH con scp, el demonio del servidor OpenSSH inicia una copia segura de archivos entre el servidor y el cliente después de la autenticación. OpenSSH puede utilizar muchos métodos de autenticación, incluyendo contraseña simple, clave pública y tickets Kerberos.

2.4.2. Instalación del servicio

2.5. Apache server

2.5.1. Descripción del servicio

2.5.2. Instalación del servicio

Capítulo 3

Tecnología empleada

En este capítulo se van a comentar las diferentes tecnologías que se han empleado para la realización de este proyecto.

3.1. Herramientas utilizadas para desplegar la maáquina

3.1.1. Ansible

Ansible es un motor open source que automatiza una gran cantidad de procesos informáticos, como la preparación de la infraestructura, la gestión de la configuración, la implementación de las aplicaciones y la organización de los sistemas.

Puede utilizarse para instalar software, automatizar tareas cotidianas, preparar elementos de infraestructura y de red, mejorar la seguridad y el cumplimiento normativo, aplicar parches a los sistemas y organizar flujos de trabajo complejos [5].

Instalación y configuración

En primer lugar para realizar la instalación de ansible se seguirá la guia de instalación descrita en su página web Ansible installation on ubuntu .

Se ejecutarán los siguientes comandos para realizar la instalación de ansible siguiendo la guia mencionada anteriormente:

```
sudo apt update
sudo apt install software-properties-common
sudo add-apt-repository --yes --update ppa:ansible/ansible
sudo apt install ansible
```

Se creará una carpeta para albergar la estructura principal de ansible:

```
mkdir -p ansible/{inventories,group_vars,host_vars,roles,
   playbooks}
ansible-galaxy init roles/dns_role
ansible-galaxy init roles/ldap_role
ansible-galaxy init roles/firewall_role
ansible-galaxy init roles/openssh_role
ansible-galaxy init roles/smb_role
ansible-galaxy init roles/apache_role
ansible-galaxy init playbooks/dns
ansible-galaxy init playbooks/ldap
ansible-galaxy init playbooks/firewall
ansible-galaxy init playbooks/apache
ansible-galaxy init playbooks/openssh
ansible-galaxy init playbooks/smb
```

3.1.2. Vagrant

Vagrant [6] una herramienta de línea de comandos que permite construir y gestionar entornos de desarrollo virtualizados de forma reproducible y portátil. Utiliza proveedores de virtualización como VirtualBox o VMware y herramientas de configuración como Ansible, Chef o Puppet para automatizar la creación y provisión de máquinas virtuales. Está diseñada para facilitar el trabajo colaborativo y asegurar que los entornos de desarrollo sean consistentes en distintos equipos.

Instalación y configuración

3.2. Otras herramientas

3.2.1. Git

Git [7] es un software de control de versiones de código abierto y gratuito. Inicialmente fue planeado para trabajar con varios desarrolladores en el núcleo de Linux. Se trata de un rastreador de contenido que se usa principalmente para almacenar código. Git posee un sistema de control de versiones para que varios desarrolladores puedan trabajar en paralelo sobre la misma aplicación permitiéndoles revertir y regresar a una versión anterior de su código

Esta herramienta se ha utilizado para trabajar en un repositorio donde almacenar las distintas versiones del CTF.

3.2.2. Overleaf

Overleaf [8] es una herramienta de publicación y redacción colaborativa en línea que hace más eficiente el proceso de redacción, edición y publicación de documentos.

Overleaf ofrece un editor LaTeX fácil de usar, con posibilidad de colaboración en tiempo real y una vista previa cargada automáticamente en segundo plano a medida que escribe.

Ha sido utilizado, junto con el libro BASIX [9] de LaTeX, para la creación de la documentación relativa al proyecto.

3.2.3. Visual Studio Code

Visual Studio Code [10] es un editor de código gratuito y de código abierto desarrollado por Microsoft. Posee soporte para la depuración, control integrado de Git, resaltado de sintaxis, finalización inteligente de código, fragmentos y refactorización de código. Es altamente personalizable ya que admite la instalación de distintas extensiones, cambios de temas, atajos de teclado y/o preferencias.

Esta herramienta se ha utilizado para configurar los distintos playbooks y tareas de ansible para el correcto despliegue de la máquina asi como para acceder a la máquina virtual mediante la terminal de visual.

3.2.4. Virtual Box

VirtualBox [11] es un software de virtualización de código abierto que permite ejecutar múltiples sistemas operativos como máquinas virtuales dentro de un sistema anfitrión. Es compatible con Windows, Linux y macOS, y permite simular entornos completos de hardware para instalar y probar sistemas operativos sin necesidad de hardware adicional. Es ampliamente utilizado para pruebas, desarrollo y entornos de laboratorio.

Esta herramienta ha sido utilizada como software de virutlización para poder desplegar los servicios indicados en una máquina virtual.

Bibliografía

[1] https://docs.ansible.com/.

[2] https://www.grupocastilla.es/ansible/.

[3] https://releases.ubuntu.com/jammy/.

[4] https://documentation.ubuntu.com/server/how-to/security/openssh-server/.

[5] https://www.redhat.com/es/topics/automation/learning-ansible-tutorial.

[6] Vagrant. https://developer.hashicorp.com/vagrant/docs.

[7] Git. https://git-scm.com/. 2022.

[8] Overleaf. https://es.overleaf.com/. 2022.

[9] David Pacios Izquierdo. https://www.ucm.es/data/cont/docs/1346-2019-04-12-BaSix%20LaTeX%20ba%CC%81sico%20con%20ejercicios%20resueltos27.pdf. 2022.

[10] Visual Studio Code. https://code.visualstudio.com/. 2022.

[11] https://www.virtualbox.org/.

“Todo lo que tenemos que decidir es qué
hacer con el tiempo que se nos da“
Gandalf

David Fernández Alejo
Lunes 30 de mayo de 2025
Ult. actualización 14 de julio de 2025
LaTeX lic. LPPL & powered by TEFLON CC-BY-NC-ND