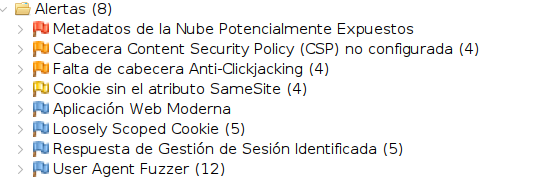
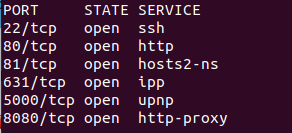
****

### **PASOS ATAQUE**

### Instalar nmap

Ejecutar comando para ver los puertos abiertos: nmap 127.0.0.1



### esos son los puertos que tiene abiertos

### **- descargar hydra y tener los archivos de texto en el escritorio**

****COMANDO A UTILIZAR****

**hydra -L usuarios.txt -P contrasennas.txt 127.0.0.1 http-post-form "/login.php:user=^USER^&pass=^PASS^:F=incorrect"**

### ****1. Metadatos de la Nube Potencialmente Expuestos****

#### ****¿Qué es?****

Los **metadatos de la nube** son datos que los servicios de nube (como AWS, Google Cloud, Azure, etc.) proporcionan para gestionar instancias, redes y otros recursos. Si estos metadatos están mal protegidos o expuestos en la web, pueden permitir que un atacante obtenga credenciales o información sensible.

#### ****Ataque****

* **Cómo un atacante puede explotar**: Un atacante podría acceder a los metadatos expuestos para obtener claves de API o credenciales de servicios en la nube. Esto podría permitirles realizar acciones no autorizadas en la infraestructura de la nube, como el robo de datos o la manipulación de recursos.
* **Ejemplo de ataque**: Un atacante podría hacer una solicitud a una URL específica, como http://169.254.169.254/latest/meta-data/ (si el servidor está en AWS) para obtener información de la instancia y posibles credenciales.

### **2. **Cabecera Content Security Policy (CSP) no configurada****

#### ****¿Qué es?****

La **Content Security Policy (CSP)** es una cabecera HTTP que ayuda a prevenir ataques como **Cross-Site Scripting (XSS)** y la inyección de contenido no deseado. Una política CSP adecuada puede bloquear la ejecución de scripts maliciosos inyectados en la página.

#### ****Ataque****

* **Cómo un atacante puede explotar**: Si CSP no está configurada, los atacantes pueden inyectar scripts maliciosos en formularios o parámetros de URL, lo que podría permitirles robar cookies, redirigir al usuario a sitios maliciosos o ejecutar código arbitrario en el navegador del usuario.
* **Ejemplo de ataque**: Si no hay CSP, un atacante podría insertar un script como este en un formulario o URL:

**<script>alert('XSS');</script>**

**Si no se bloquea mediante CSP, el navegador ejecutaría el script.**

### **Conclusión: **El ataque más sencillo****

De las vulnerabilidades que mencionas, **el ataque más sencillo probablemente sería un ataque de XSS (Cross-Site Scripting)** debido a la falta de una **Content Security Policy (CSP)** configurada. Un atacante solo necesitaría encontrar un punto de entrada en tu sitio (como un formulario o URL), inyectar un script malicioso y robar información sensible (como cookies de sesión). Esto es relativamente sencillo y no requiere mucha habilidad técnica.

**Inyección SQL (SQL Injection)**

Es un ataque en el que se introducen comandos SQL maliciosos en los formularios de entrada de la web, como campos de búsqueda o formularios de login. Esto puede permitir al atacante acceder, modificar o eliminar datos de la base de datos. INYECCIÓN ESTARÁ CUBIERTO

**Cross-Site Scripting (XSS)**

En este ataque, el atacante inyecta scripts maliciosos en páginas web que son luego ejecutados en el navegador de otros usuarios. Esto puede robar cookies, secuestrar sesiones o redirigir a los usuarios a sitios maliciosos.

**Cross-Site Request Forgery (CSRF)**

Un ataque en el que el atacante engaña a un usuario autenticado para que realice una acción no deseada en una web, como cambiar su contraseña o realizar una transferencia, sin su consentimiento.

**Denegación de servicio (DoS)**

Este tipo de ataque intenta hacer que un servidor o red sea inaccesible, generalmente mediante el envío de grandes cantidades de tráfico malicioso para saturar los recursos del servidor.

**Denegación de servicio distribuido (DDoS)**

Similar al DoS, pero en este caso el ataque es llevado a cabo por una red de máquinas comprometidas (bots) distribuidas, lo que aumenta la potencia del ataque. JODIDO

**Ataques de fuerza bruta**

Los atacantes intentan adivinar credenciales de acceso (como contraseñas) mediante un proceso de prueba y error masivo. Utilizan software automatizado para intentar múltiples combinaciones hasta que una funciona.

**Phishing**

A través de correos electrónicos o mensajes fraudulentos, el atacante engaña al usuario para que entregue información sensible, como contraseñas o datos bancarios, pensando que está interactuando con una página legítima.

**Sniffing**

Implica interceptar y analizar el tráfico de red entre el cliente y el servidor para capturar información sensible, como contraseñas, cookies o datos de sesiones no cifrados.

**Man-in-the-Middle (MitM)**

Un atacante intercepta y potencialmente modifica las comunicaciones entre el cliente y el servidor sin que ninguna de las partes lo sepa. Este ataque es efectivo cuando las comunicaciones no están cifradas.

**Directory Traversal (o Path Traversal)**

Permite a un atacante acceder a directorios y archivos fuera de la carpeta web pública de un servidor, lo que puede llevar a la exposición de información confidencial del sistema.

**Ransomware**

Un malware que cifra los archivos de la web o el sistema y exige un rescate para descifrarlos. Si un servidor es comprometido, el atacante puede bloquear el acceso a los archivos de la web. UN POCO INÚTIL PORQUE ACCEDEMOS DE MANERA LOCAL

**Exploit de vulnerabilidades en software**

Consiste en aprovechar las vulnerabilidades de un software (como un CMS o una librería usada en la web) para ejecutar código malicioso y tomar control del servidor o acceder a datos sensibles.

**Privilege Escalation**

En este ataque, el atacante gana privilegios elevados en un sistema o servidor, lo que le permite acceder a áreas restringidas, realizar cambios en el sistema o robar información.

**Clickjacking**

El atacante engaña al usuario para que haga clic en un objeto invisible o disfrazado como algo legítimo, lo que puede desencadenar acciones no deseadas, como compartir información o realizar acciones en una web.

**Session Hijacking (Secuestro de sesión)**

Este ataque implica el robo de una sesión activa de un usuario para obtener acceso a sus privilegios, como en el caso de servicios web o aplicaciones donde la autenticación se maneja mediante cookies.

**TCP/IP Spoofing**

El atacante falsifica direcciones IP para hacerse pasar por otro sistema en la red, con el objetivo de interceptar comunicaciones o lanzar ataques de DoS o DDoS.

**LDAP Injection**

Similar a la inyección SQL, este ataque manipula las consultas LDAP (Lightweight Directory Access Protocol) para acceder a datos no autorizados, como las credenciales de un sistema o información privada. INYECCIÓN ESTARÁ CUBIERTO

**Local File Inclusion (LFI)**

En este tipo de ataque, el atacante explota vulnerabilidades en la web para incluir archivos locales del servidor en las respuestas del servidor, lo que puede permitir la ejecución de código malicioso.

**Remote File Inclusion (RFI)**

Similar al LFI, pero en este caso, el atacante incluye archivos remotos (por ejemplo, desde un servidor controlado por el atacante), lo que puede llevar a la ejecución de código malicioso.

**Buffer Overflow**

Este ataque ocurre cuando se sobrescribe la memoria de un programa debido a una entrada de datos excesiva, lo que puede permitir al atacante ejecutar código arbitrario en el servidor y tomar control del sistema.