INFORME: EJECUTIVO Y TÉCNICO

Sistema Odiobá

- Fecha: 5 de septiembre de 2024
- Cliente: Reto 7 – Team Challenge
- Consultora de Ciberseguridad: The Bridge - Accelerator
- Control de Cambios

Versión	Documento	Fecha	Cambios	Autor	revisor	visto bueno
1.1	Informe de resultados	11/09/2024	Informe inicial	Victor Martínez	Ángel / Joseba	Javier Tomás

Índice de Contenidos

1. Introducción ----------3

2. Informe Ejecutivo ----------3

- Introducción ----------3
- Alcance ----------4
- Resumen de Actuaciones Practicadas ----------5
- Recomendaciones generales ----------5
- Normativa aplicable y sanciones ----------6

3. Informe Técnico: ----------7

- Introducción----------7
- Fase de exploración – Servidor – web----------7
    - Puerto 8080----------7
    - Puerto 8081----------10
- Fase de explotación----------12
- Resumen de explotaciones con MetaExploit----------15
- Conclusiones ----------15

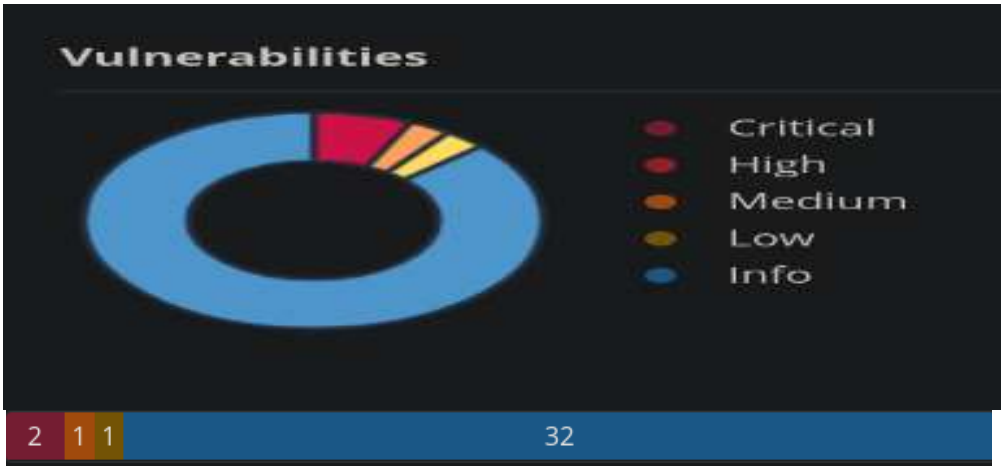
4. Bibliografía ----------16
5. Anexos ----------17

1. INTRODUCCIÓN

El presente informe está formado por 2 partes: un informe ejecutivo, menos técnico y dirigido a informar a cargos de toma de decisiones o ejecutivos de la compañía, y un informe técnico, dirigido a los analistas de ciberseguridad y programadores que tengan que crear y ejecutar tareas para mitigar las vulnerabilidades explotadas, con la finalidad de mejorar los manuales de estrategia de la compañía en la detección, contención y respuesta ante incidentes críticos en su sistema.

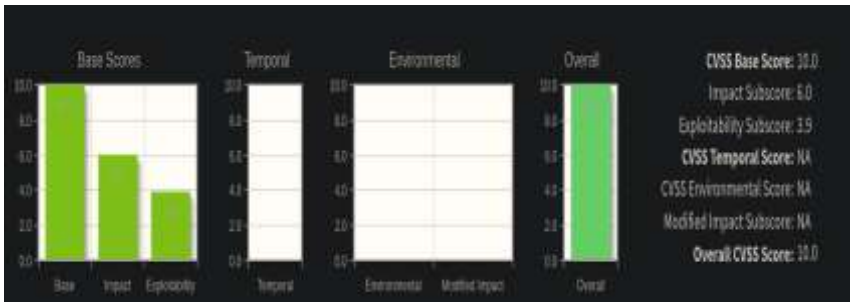
2. INFORME EJECUTIVO

1. Introducción. – Este informe tiene como objetivo presentar los resultados de las vulnerabilidades detectadas y explotadas en el equipo Odiobá, de acuerdo con el contrato firmado entre ambas partes, en el que permiten la explotación del sistema con la finalidad de conseguir la autenticación por atacantes externos con usuarios con privilegios root. El equipo no tiene entorno gráfico y para acceder en línea de comandos hace falta una clave y contraseña que no aportan, habiendo usado para su explotación diversas herramientas de ciberseguridad, destacando alguno de sus resultados:
   - Nessus Essentials. - Herramienta de escaneo de vulnerabilidades más populares y completas en el ámbito de la seguridad informática, que se utiliza para identificar vulnerabilidades en sistemas y redes, detectar configuraciones incorrectas y posibles puntos de entrada para ataques en una amplia gama de plataformas, clasificando estas en críticas, altas, medias e info.



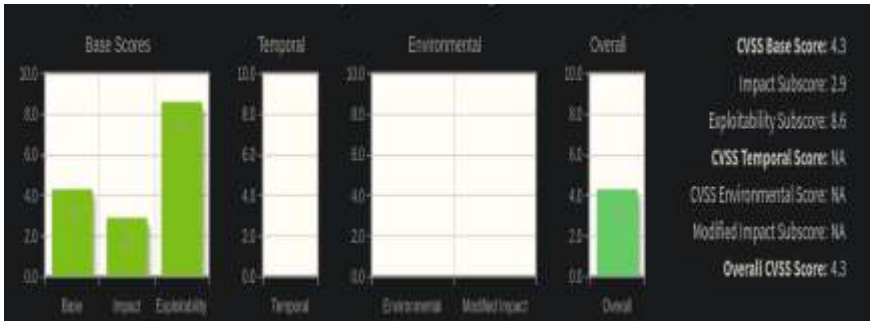
   - Como se puede observar el número de las vulnerabilidades encontradas se mantiene dentro de los límites que se pueden permitir por una organización. No obstante, seria interesante subsanarlas si fuera posible debido a la gravedad de alguna de ellas:

- CVE-2021-44228 para servidores Apache "*Log4j2 Remote Code (RCE)"*, detectada como crítica y alta, debido a que es extremadamente peligrosa permitiendo a cualquier atacante ejecutar código arbitrario en los sistemas infectados, comprometiendo la confidencialidad, integridad y disponibilidad de los datos del sistema infectado.



Puntuación del NIST (Instituto Nacional para Estándares y Tecnología

- CVE-2003-1418 para servidores Apache con el encabezado *"ETag Information Disclosure",* permitiendo a un atacante obtener información sensible sobre los archivos o recursos servidos por el servidor, sin permitir el acceso directo al servidor, por eso está catalogada como media.



Puntuación del NIST (Instituto Nacional para Estándares y Tecnología

2. Alcance. - El alcance se ha centrado en identificar y evaluar las debilidades de seguridad en el sistema, para lograr las finalidades expuestas en el contrato, explotando algunas de las vulnerabilidades encontradas, que pueden causar daños el sistema, así como comprometer la integridad, confidencialidad y disponibilidad de los datos del mismo, destacando:

- En colación al punto anterior, se ha procedido a realizar la explotación de la vulnerabilidad con CVE-2021-44228 llamada *"log4shell",* mediante un *framework* para pruebas de penetración y auditorias de seguridad, consiguiendo el acceso al sistema con el mas alto privilegio, por lo que podría poner en compromiso todo el sistema.

- Mediante herramientas de escaneo de servidores web para descubrir directorios, archivos, subdominios y otros puntos de entrada ocultos y menos evidentes, se ha encontrado un directorio que aporta información sensible para obtener acceso al sistema, mediante una vulnerabilidad que aprovecha del funcionamiento normal entre cliente-servidor web para la ejecución de código arbitrario mediante programas malicioso, consiguiendo mediante herramientas de explotación y penetración entrar nuevamente en el sistema con los mismos privilegios.

3. Resumen de actuaciones practicadas. – Se han realizado numerosas actuaciones, explotando ciertas debilidades / vulnerabilidades detectadas, algunas de las cuales han sido comentadas anteriormente, consiguiendo finalmente el objeto del contrato, es decir, la autenticación con usuario con privilegios root en el sistema, aportando detalles técnicos más adelante.

4. Recomendaciones generales

En el análisis efectuado de vulnerabilidades con el programa Nessus, se han encontrado 2 vulnerabilidades importantes, por lo que podría estar dentro de los riesgos permitidos dentro de las políticas de seguridad de ciertas empresas. No obstante, se recomienda actualizar, si es el caso, dicha política al modelo "Zero Trust"[1].

Por otro lado, tener un directorio web que incluya información sobre una posible vía de explotación del sistema, representa un riesgo significativo de seguridad para la empresa, debiendo ser subsanado lo antes posible.

[1] Zero Trust, parte de la premisa de no confiar en ningún usuario, dispositivo o sistema dentro o fuera de la red organizacional y se basa en los siguientes principios clave:

- **Verificación continua:** La identidad y la autorización de cada usuario y dispositivo se verifican constantemente.
- **Principio de Menos privilegios:** Los usuarios y dispositivos solo reciben acceso a los recursos que necesitan para realizar su trabajo.
- **Segmentación:** La red se segmenta en zonas para limitar el acceso, contención de amenazas y evitar el movimiento lateral de las mismas
- **Protección de datos:** Los datos se protegen con cifrado adecuado y otras medidas de seguridad.
- **Monitoreo y respuesta:** La actividad de la red se monitorea constantemente para detectar y responder a las amenazas.

5. Normativa aplicable y sanciones

Existen diversas normativas que regulan la protección de datos y la seguridad de la información, y que podrían ser aplicables en este caso:

- **Reglamento General de Protección de Datos (RGPD)[2] y la Ley Orgánica de Protección de Datos Personales y garantía de los derechos digitales (LOPDGDD)[3]**. - Si la información confidencial que se encuentra en los directorios bloqueados, incluye datos personales, su incumplimiento podría acarrear sanciones importantes para la empresa.
- **Ley de Servicios de la Sociedad de la Información y el Comercio Electrónico (LSSI)[4]. -** Los prestadores de servicios (corporaciones, empresas, etc) deben adoptar las medidas técnicas y organizativas necesarias para garantizar la seguridad de los datos de los usuarios, pudiendo su incumplimiento acarrear sanciones para la empresa.

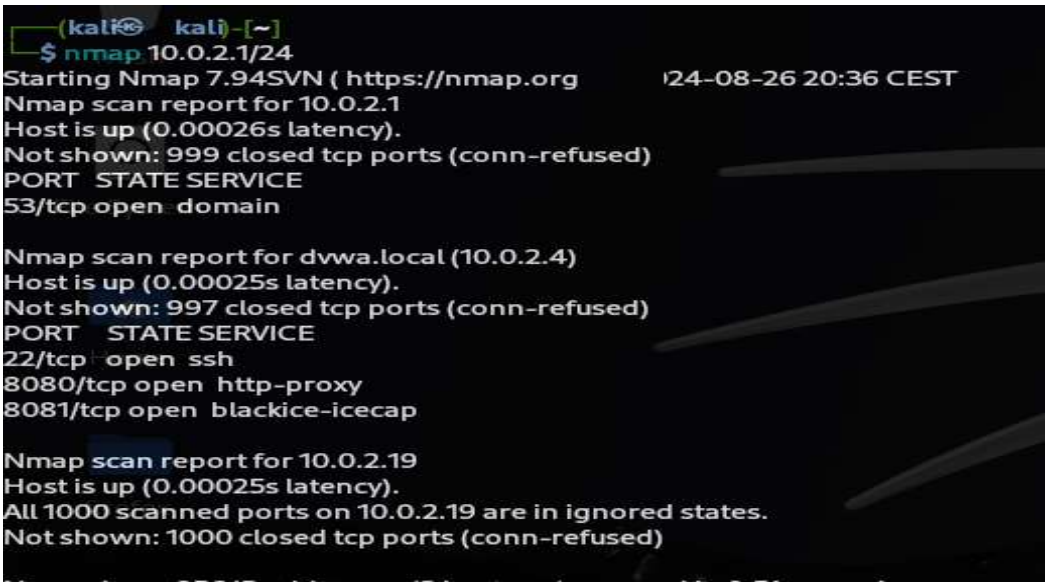
Las sanciones por el incumplimiento de las normativas de protección de datos y seguridad de la información pueden ser de elevado valor, por ejemplo, en el caso del RGPD, las multas pueden ascender hasta el 4% del volumen de negocio mundial anual de la empresa o 20 millones de euros, lo que sea mayor y en el caso de la LOPDGDD, las multas pueden ascender hasta 300.000 euros.

[2] El RGPD es un reglamento de la Unión Europea que establece normas estrictas para la protección de datos personales
[3] La LOPDGDD es ley española que desarrolla el RGPD y que establece normas específicas para la protección de datos personales en España
[4] La LSSI es una legislación española que regula la prestación de servicios de la sociedad de la información y el comercio electrónico, estableciendo una serie de obligaciones a las empresas e infracciones en caso de su incumpliento,

3.- INFORME TÉCNICO

1. Introducción. – Para conseguir el objetivo fijado en el contrato, se ha seguido la siguiente línea de investigación:

- El Equipo ha sido entregado con un sistema Linux Kernel 2.6 en un entorno CLI, sin aportar credenciales de inicio de sesión de la maquina denominada “ODIOBÁ”, por lo que el análisis y explotación será realizado en caja negra.
- Para esta explotación se ha usado como maquina atacante, un sistema Kali Linux virtualizado, en su versión .2 2024, conectando mediante Red NAT con la maquina objeto del presente.
- En primer lugar, se procede a consultar, mediante la herramienta *“Nmap”,* el rango de IPs donde se encuentran ambas maquinas*,* siendo la de Odiobá: 10.0.2.4 y de la maquina atacante: 10.0.2.19. Además *“Odiobá”* muestra que tiene abiertos el puerto SSH /22), el 8080 y el 8081, cada uno con un servicio.

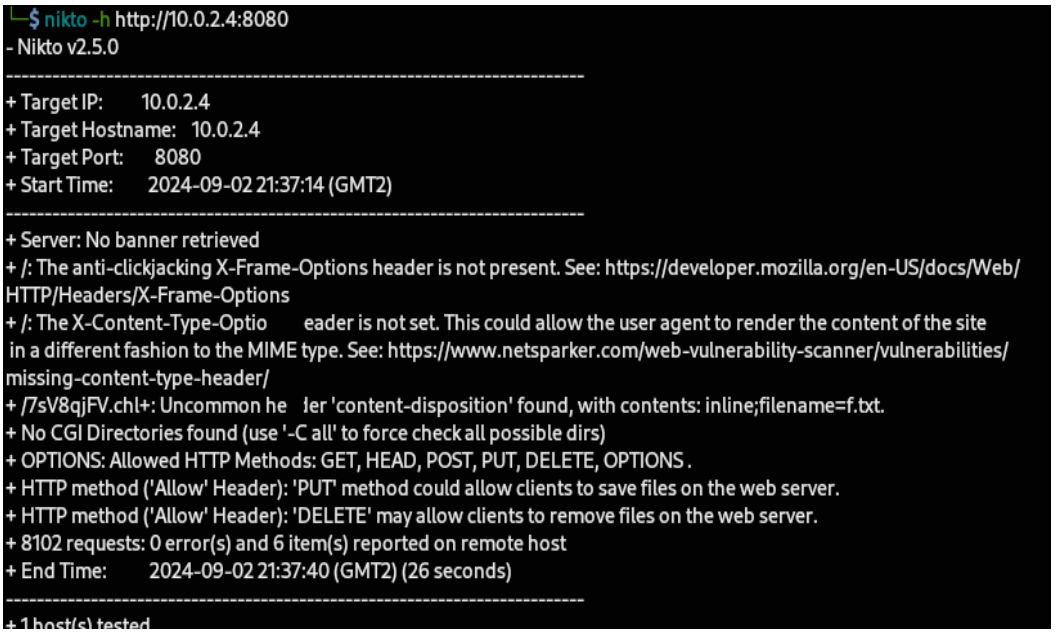


2. Fase de exploración – Servidor Web. -

- **- PUERTO 8080:**

- Mediante el uso de *Gobuster*, siendo una herramienta de seguridad y hacking web, comúnmente utilizada durante las fases de reconocimiento en pruebas de penetración, que usa para descubrir objetos y directorios ocultos o no indexados en un servidor web, no encontrando nada, sólo un directorio denominado /error con da status 500, es decir es un error por parte del servidor.

- Con la aplicación Nikto, se procede a intentar encontrar vulnerabilidades que puedan ser explotadas, no encontrando ninguna aplicable directamente vía Metaexploit. No obstante, si se muestran algunas como: el clickjacking, permite métodos HTTP *"put" y "delete"* permitidos, vulnerabilidades XSS (mediante XSStrike) etc.



- Mediante la aplicación Nessus, la cual, permite identificar fallos de seguridad en sistemas, redes y aplicaciones, como configuraciones incorrectas, vulnerabilidades conocidas y posibles puntos de acceso, generando informes detallados, como se puede ver en la imagen:

10.0.2.4

2	0	1	1	26
CRITICAL	HIGH	MEDIUM	LOW	INFO

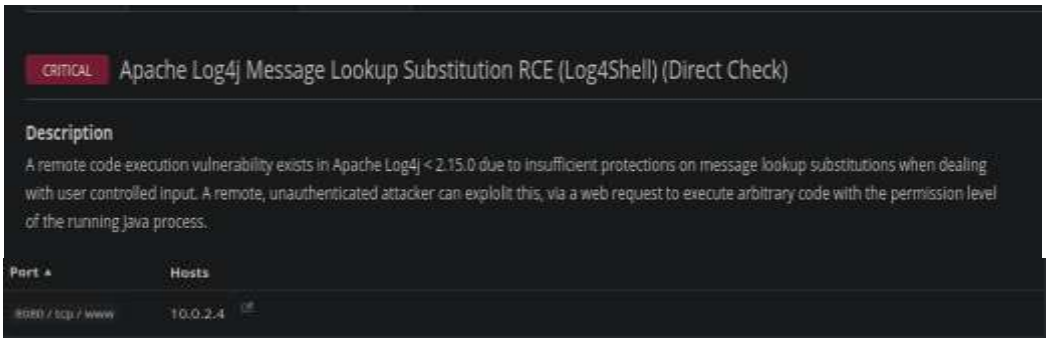
Vulnerabilities — Total: 30

SEVERITY	CVSS V3.0	VPR SCORE	PLUGIN	NAME
CRITICAL	10.0	10.0	156014	Apache Log4Shell RCE detection via callback correlation (Direct Check HTTP)
CRITICAL	10.0	10.0	155998	Apache Log4j Message Lookup Substitution RCE (Log4Shell) (Direct Check)
MEDIUM	5.3	1.4	88098	Apache Server ETag Header Information Disclosure
LOW	2.1*	4.2	10114	ICMP Timestamp Request Remote Date Disclosure
INFO	N/A	-	18261	Apache Banner Linux Distribution Disclosure
INFO	N/A	-	48204	Apache HTTP Server Version
INFO	N/A	-	39520	Backported Security Patch Detection (SSH)
INFO	N/A	-	39521	Backported Security Patch Detection (WWW)
INFO	N/A	-	45590	Common Platform Enumeration (CPE)
INFO	N/A	-	54615	Device Type

Si nos atenemos a las vulnerabilidades más importantes (críticas), encontramos **dos vulnerabilidades relacionadas con el "Log4Shell (CVE-2021-44228)"** pero con distintos métodos de detección y explotación, ambas por el puerto 8080:

** Apache Log4Shell RCE detection via callback correlation (Direct Check HTTP), es un método para detectar si una instancia "Log4j" es vulnerable mediante inyección de carga maliciosa, mediante la observación de un comportamiento específico de una solicitud (callback).

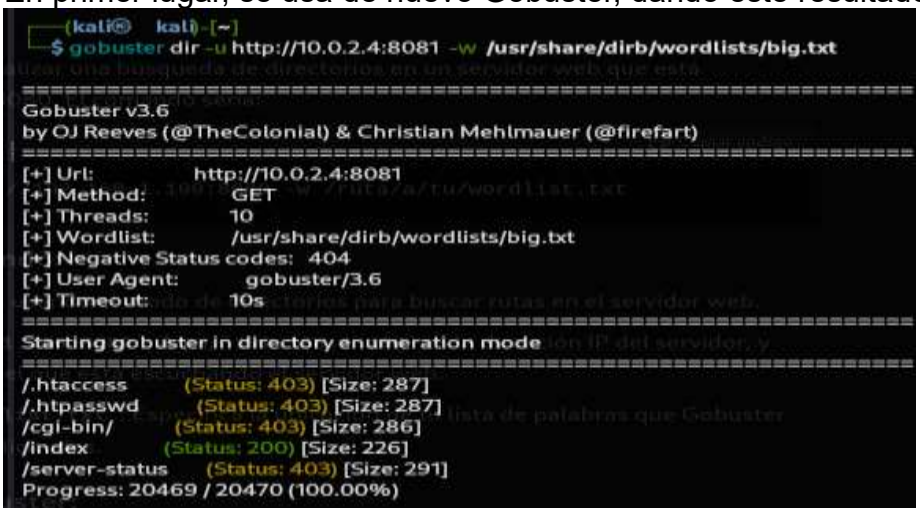
**_Apache Log4j2 Remote Code Execution (RCE)_", permite a los atacantes ejecutar código arbitrario en sistemas afectados, debido a la forma en que Log4j2 maneja las cadenas de texto al registrar datos, permitiendo a los atacantes enviar datos maliciosos a las aplicaciones que lo usan. Cuando Log4j2 procesa esas cadenas o datos maliciosos, el atacante puede inyectar código que luego se ejecuta en el servidor afectado, obteniendo control remoto sobre el sistema.



- Mediante la herramienta MetaExploit, la cual se utiliza en pruebas de penetración para identificar, probar, y explotar fallos de seguridad, permitiendo a los usuarios simular ataques reales y generar payloads personalizados para acceder a sistemas vulnerables, se procede a la detección de la vulnerabilidad anteriormente descrita *("log4shell")*, mediante un módulo auxiliar que incorpora, siendo positiva.

- **- PUERTO 8081:**

- En primer lugar, se usa de nuevo Gobuster, dando este resultado:

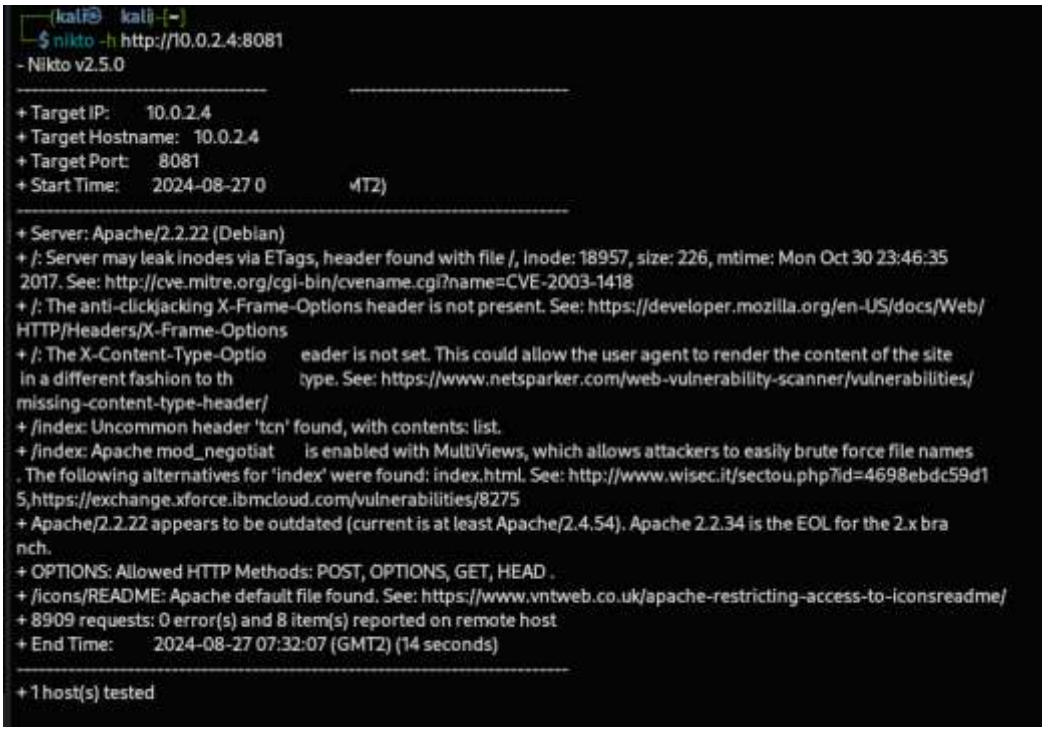


- En segundo lugar, se comprueba el directorio que ha contestado con código 200, ya que el resto son de acceso prohibido por falta de permisos:

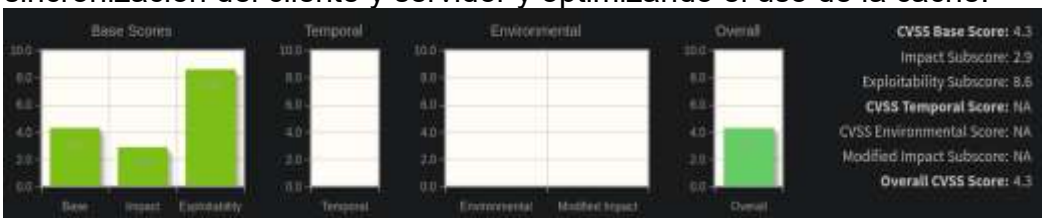
En este caso, en uno de los directorios ocultos y menos evidentes a usar desde la interfaz pública del sitio web, se ha hallado información o pista importante para una posterior explotación, indicando claramente que el script *"/cgi-bin/vulnerable"* presenta una vulnerabilidad conocida como "ShellShock".

El directorio */cgi-bin/* es la ubicación estándar donde se ubican los scripts CGI - Common Gateway Interface (Interfaz de puerta de enlace común), los cuales se ejecutan en el servidor en respuesta a las peticiones de los clientes, devolviendo a éste el resultado del script solicitado. Esta mecánica de funcionamiento, a menudo, se convierte en vulnerabilidades que permiten la ejecución de código arbitrario mediante la inyección de scripts maliciosos, como puede ser la "ShellShock"

- En tercer lugar, se ha utilizado la aplicación Nikto, otra herramienta de escaneo de vulnerabilidades web que ayuda a identificar fallos de seguridad en servidores web (archivos y directorios sensibles, versiones antiguas no actualizadas, configuraciones inseguras y otras vulnerabilidades más comunes) que podrían ser explotados por atacantes malintencionados.



Como se puede ver, ha encontrado una posible vulnerabilidad con un *CVE-2003-1418 relacionado con las etiquetas Etags (Entity Tags),* las cuales, se usan para identificar las versiones asignadas a un recurso en el servidor web. Cuando un cliente solicita un recurso, el servidor usa los Etags para determinar si la versión almacenada en la cache de ese recurso es la misma que está en el servidor, ayudando todo esto la sincronización del cliente y servidor y optimizando el uso de la caché.



Para una posible explotación, se ha realizado una búsqueda en Metaexploit del CVE, así como por la descripción o nombre del identificador, con resultado infructuoso.

Además, el directorio */índex/* es vulnerable a ataques por fuerza bruta.

3. Fase de Explotación:

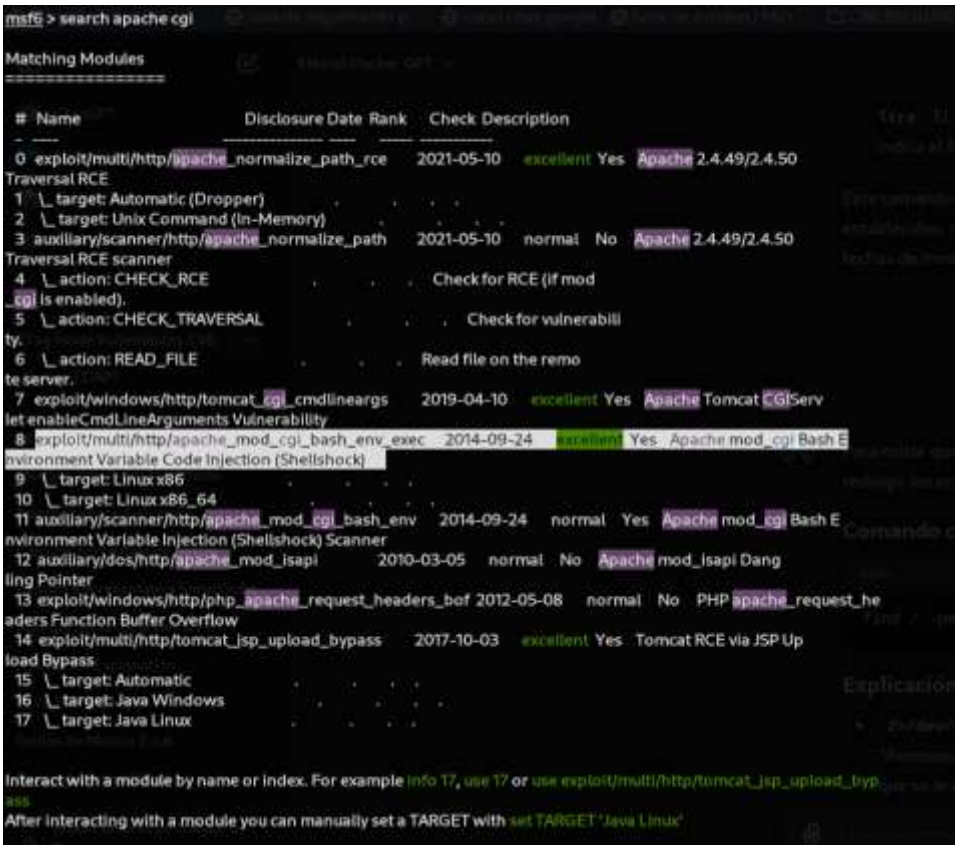
- Se ejecuta Metaesploit, configurando un workspace específico para ir guardando los progresos. En primer lugar, se utiliza una herramienta nmap aportando datos importantes:

  - El puerto 8080 donde está el servicio Nagios NSCA *(Nagios Service Check Acceptor)*, una popular herramienta de monitorización de redes y sistemas, permitiendo a los servidores y dispositivos enviar resultados de verificación pasivos (como estado de servicios o dispositivos) al servidor Nagios central, normalmente mediante conexión cifrada. Cuando NSCA está asociado con un puerto HTTP-Proxy, como el 8080, es posible que esté siendo utilizado para recibir y procesar informes de estado o resultados de monitorización desde dispositivos remotos a través de un proxy HTTP. No obstante, como ya se ha comentado, no ha encontrado forma de vulnerarlo por este medio, así que se ha recurrido a NESSUS, encontrando una vulnerabilidad explotable con CVE-2021-45046.

    Se ejecuta el exploit para esta vulnerabilidad CVE, consiguiendo acceso, utilizando como *"stager"* una *"shell_reverse",* siendo finalmente positiva el acceso a la maquina objetivo por el puerto 8080 con usuario no privilegiado.

- El puerto 8081, el cual es ejecutando un servidor apache 2.2, el cual es vulnerable a ShellShock, la cual afecta a versiones del intérprete de comandos Bash, presentes en muchos sistemas Unix y Linux, permitiendo al atacante ejecutar comandos arbitrarios en un sistema afectado.

- Se procede a la explotación del puerto 8081 con la vulnerabilidad critica ShellShock mediante el uso de Metaexploit, concretamente la numero 8:



- Ejecutamos el exploit cumplimentado los campos incluidos en "*Options"*, verificando que no dejamos ninguno sin rellenar con la opción *"show missing",* resultando positivo, consiguiendo iniciar sesión con user sin privilegios a través del servicio /cgi-bin/vulnerable:

- **RESUMEN DE LAS EXPLOTACIÓNES REALIZADAS CON METAEXPLOIT - PUERTOS 8080 Y 8081**

4.- CONCLUSIONES

En este ejercicio se han utilizado diversas herramientas y técnicas para identificar y explotar vulnerabilidades en un sistema objetivo, revelando, una exploración inicial, varios servicios potencialmente vulnerables, como SSH y un servidor web Apache que ejecuta Nagios NSCA.

La exploración web permitió descubrir una vulnerabilidad crítica de *"ShellShock"* en el puerto 8081, la cual fue explotada exitosamente para obtener acceso al sistema, identificando archivos SUID y SGID, que son cruciales para la escalada de privilegios, no siendo explotada esta última vía, al igual que con la vulnerabilidad encontrada en el puerto 8080 a través de Nessus, concretamente a la biblioteca de registros de java en Apache.

Sin embargo, los intentos de explotación en el puerto 8080 por el servicio Nagios, y el puerto 22 (SSH) no resultaron exitosos, debido a la inviabilidad de las vulnerabilidades identificadas, lo que pone de manifiesto la complejidad y los desafíos en la seguridad de sistemas y en pruebas de penetración.

5.- BIBLIOGRAFÍA

https://www.nist.gov/publications/zero-trust-architecture

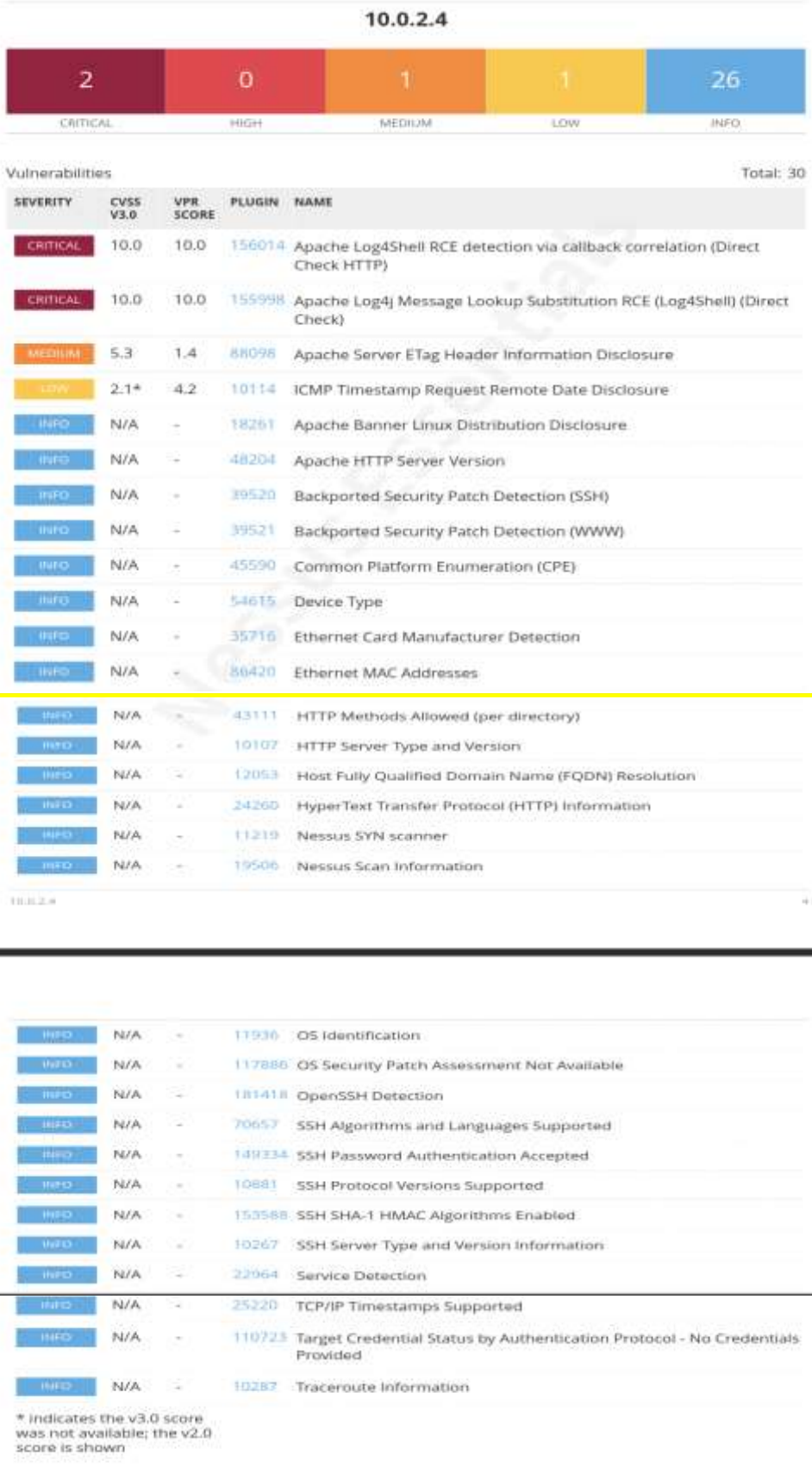
https://commission.europa.eu/law/law-topic/data-protection/reform/rules-business-and-organisations/enforcement-and-sanctions/sanctions/what-if-my-companyorganisation-fails-comply-data-protection-rules_es

https://ayudaleyprotecciondatos.es/2019/02/19/sanciones-rgpd-lopd-2019/

https://nvd.nist.gov/vuln-metrics/cvss/v2-calculator?name=CVE-2003-1418&vector=(AV:N/AC:M/Au:N/C:P/I:N/A:N)&version=2.0&source=NIST

https://nvd.nist.gov/vuln/detail/CVE-2021-44228

6.- ANEXOS



10.0.2.4

2	0	1	1	26
CRITICAL	HIGH	MEDIUM	LOW	INFO

Vulnerabilities — Total: 30

SEVERITY	CVSS V3.0	VPR SCORE	PLUGIN	NAME
CRITICAL	10.0	10.0	156014	Apache Log4Shell RCE detection via callback correlation (Direct Check HTTP)
CRITICAL	10.0	10.0	155998	Apache Log4j Message Lookup Substitution RCE (Log4Shell) (Direct Check)
MEDIUM	5.3	1.4	88098	Apache Server ETag Header Information Disclosure
LOW	2.1*	4.2	10114	ICMP Timestamp Request Remote Date Disclosure
INFO	N/A	-	18261	Apache Banner Linux Distribution Disclosure
INFO	N/A	-	48204	Apache HTTP Server Version
INFO	N/A	-	39520	Backported Security Patch Detection (SSH)
INFO	N/A	-	39521	Backported Security Patch Detection (WWW)
INFO	N/A	-	45590	Common Platform Enumeration (CPE)
INFO	N/A	-	54615	Device Type
INFO	N/A	-	35716	Ethernet Card Manufacturer Detection
INFO	N/A	-	86420	Ethernet MAC Addresses
INFO	N/A	-	43111	HTTP Methods Allowed (per directory)
INFO	N/A	-	10107	HTTP Server Type and Version
INFO	N/A	-	12053	Host Fully Qualified Domain Name (FQDN) Resolution
INFO	N/A	-	24260	HyperText Transfer Protocol (HTTP) Information
INFO	N/A	-	11219	Nessus SYN scanner
INFO	N/A	-	19506	Nessus Scan Information
INFO	N/A	-	11936	OS Identification
INFO	N/A	-	117886	OS Security Patch Assessment Not Available
INFO	N/A	-	181418	OpenSSH Detection
INFO	N/A	-	70657	SSH Algorithms and Languages Supported
INFO	N/A	-	149334	SSH Password Authentication Accepted
INFO	N/A	-	10881	SSH Protocol Versions Supported
INFO	N/A	-	153588	SSH SHA-1 HMAC Algorithms Enabled
INFO	N/A	-	10267	SSH Server Type and Version Information
INFO	N/A	-	22964	Service Detection
INFO	N/A	-	25220	TCP/IP Timestamps Supported
INFO	N/A	-	110723	Target Credential Status by Authentication Protocol - No Credentials Provided
INFO	N/A	-	10287	Traceroute Information

* indicates the v3.0 score was not available; the v2.0 score is shown