

**TEAM CHALLENGE**

**SPRING 10 – RETO OBIOBÁ**

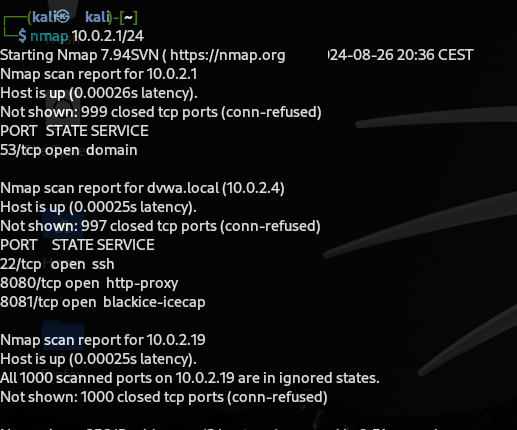
EXPLOTACIÓN VULNERABILIDADES

MAQUINA ODIOBA

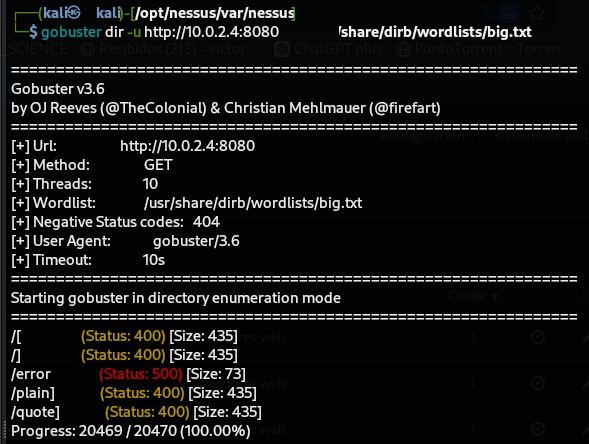
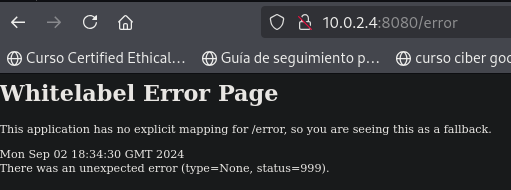
**INTRODUCCIÓN**

Para este ejercicio se han usado diversas aplicaciones y servicios para encontrar vulnerabilidades que permitan el acceso no autorizado al sistema, exponiendo únicamente los que han resultado positivos para conseguir tal fin, los cuales son:

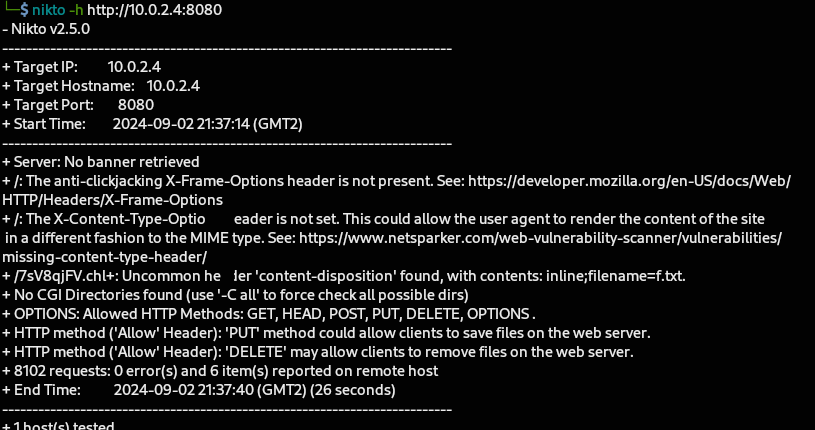
* 1.- EXPLORACIÓN IP DEL OBJETIVO



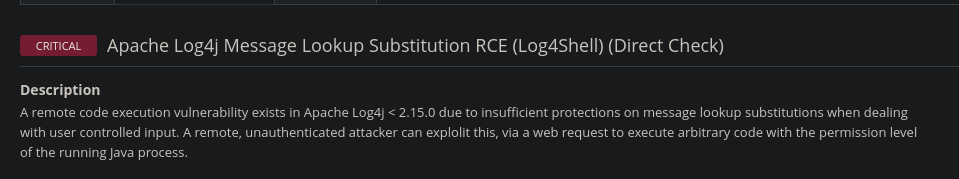
* En este caso se ha usado la aplicación nmap, la cual es una herramienta potente de escáner con multitud opciones. En este caso simplemente se ha usado para buscar las IP activas en el rango de IPs 10.0.2.1/24, siendo la 10.0.2.4 la del objetivo.
* 2.- EXPLORACIÓN WEB
* *A.-PUERTO 8080*
* En primer lugar, se ha procedido con la aplicación Gobuster para obtener información del servidor web, no encontrando nada, solo un directorio denominado /error con da status 500, es decir es un error por parte del servidor. Además, se consulta en el navegador, dando como resultado, un error por defecto en la web:



* En segundo lugar, se consulta la aplicación Nikto, con la finalidad de encontrar vulnerabilidades que puedan ser explotadas, no encontrando ninguna aplicable directamente vía Metaexploit. No obstante, si se muestran algunas como: el clickjacking, me todos “*put” y “delete”* permitidos, XSS (mediante XSStrike) etc.



* En tercer lugar, se prueban la mayoría de exploits encontrados en Metaexploit para el servicio Nagios, con resultado infructuoso, por lo que se procede al uso de herramientas externas: NESSUS

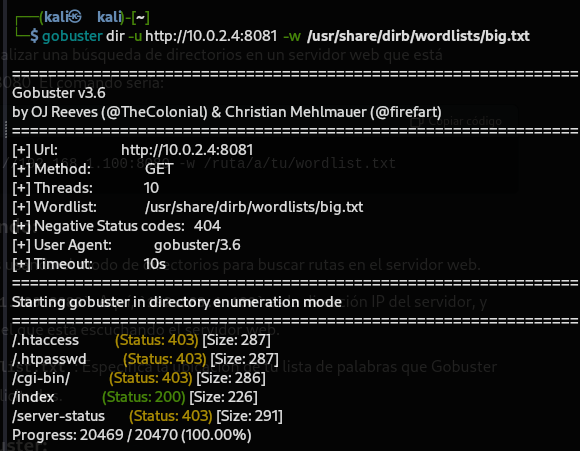


Esta vulnerabilidad está asociada al CVE - 2021- 45046 para Apache log4j (biblioteca de registros de mensajes de Java), la cual en las anteriores actualizaciones no era completamente efectiva para ciertas configuraciones, por lo que podría ser explotada por actores maliciosos.

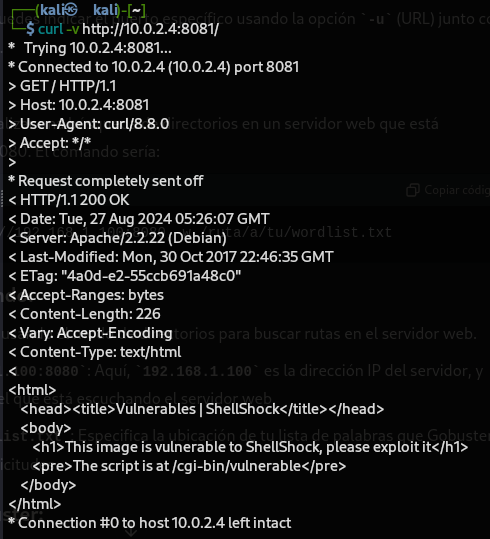
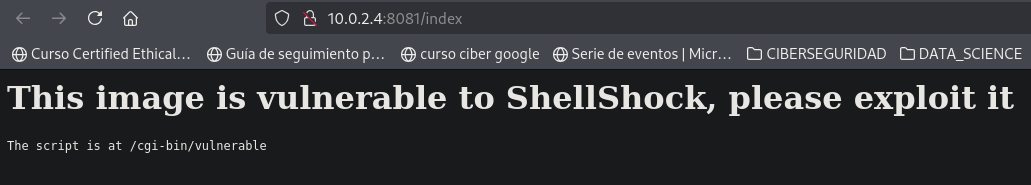
* En cuarto lugar, se procede a probar con el modulo auxiliar de Metaesploit que existe para este CVE, para ver si es vulnerable, siendo positiva:



* *B.- PUERTO 8081*
* En primer lugar, se ha usado la aplicación Gobuster, herramienta diseñada para descubrir directorios, archivos, subdominios y otros puntos de entrada ocultos en servidores web, al hacer solicitudes repetitivas fuerza bruta, basadas en listas de palabras.



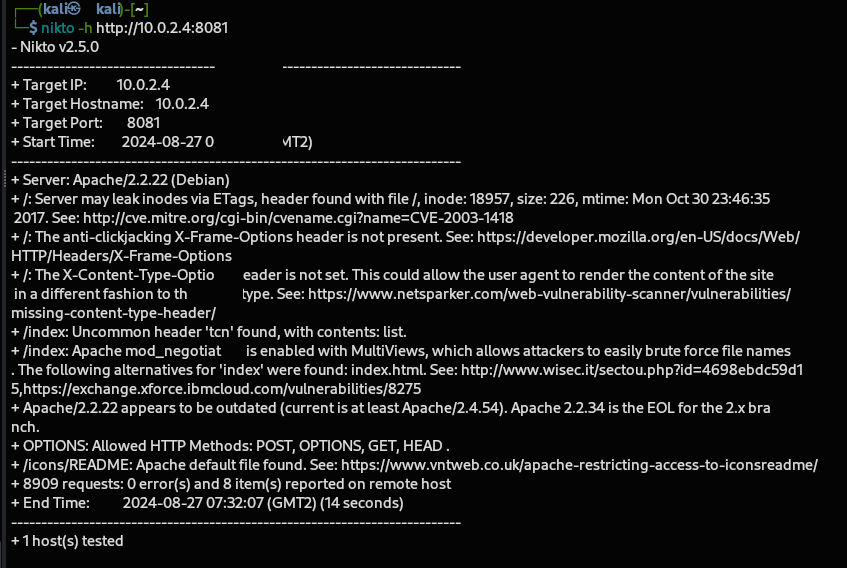
* En segundo lugar, se ha utilizado la aplicación Curl, herramienta CLI que permite la transferencia de datos desde o hacia un servidor, utilizando diversos protocolos (HTTP, HTTPS, FTP, entre otros), siendo muy versátil para realizar solicitudes web y obtener respuestas directamente desde la terminal.



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 de código abierto que ayuda a identificar fallos de seguridad en servidores web proporcionando una visión general de posibles puntos débiles (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.

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

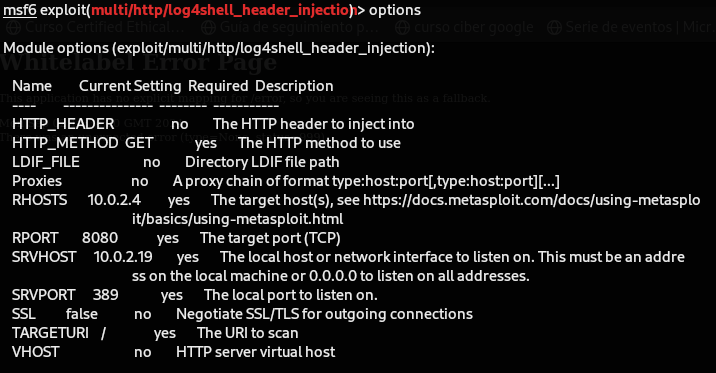
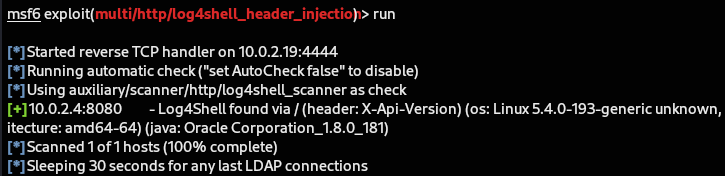
Además, el directorio */índex/* es vulnerable a ataques por fuerza bruta.

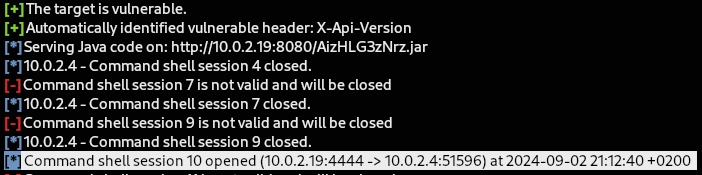
3.- FASE DE EXPLOTACIÓN

* Se ejecuta Metaesploit, configurando un workspace específico para ir guardando los progresos. En primer lugar, se utiliza una herramienta nmap aportando datos importantes:
* Puertos abiertos: 22(ssh), el cual presenta estas vulnerabilidades:

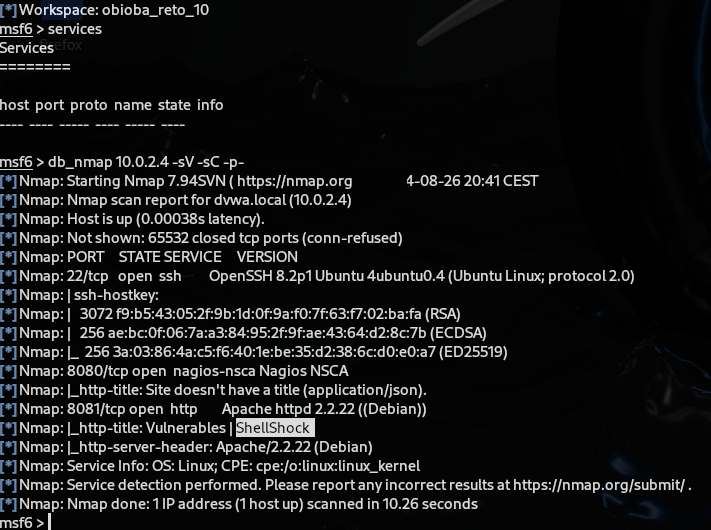


* El puerto 8080 donde está el servicio Nagios NSCA (Nagios Service Check Acceptor), una popular herramienta de monitorización de redes y sistemas, permitiendo a los servidores y dispositivos enviar resultados de verificación pasivos (como estado de servicios o dispositivos) al servidor Nagios central, 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](https://nvd.nist.gov/vuln/detail/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.

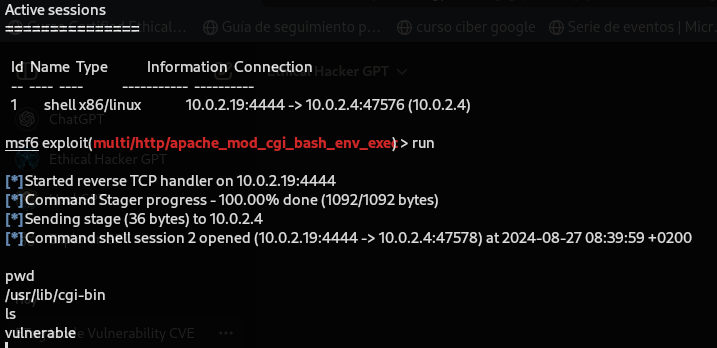




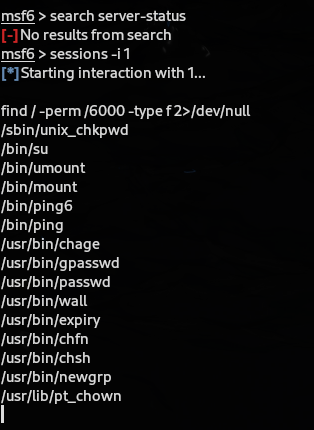
* El puerto 8081, el cual es ejecutando un servidor apache 2.2, el cual es vulnerable a ShellShock, la cual afecta a versiones del intérprete de comandos Bash, presentes en muchos sistemas Unix y Linux, permitiendo al atacante ejecutar comandos arbitrarios en un sistema afectado.



* 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:

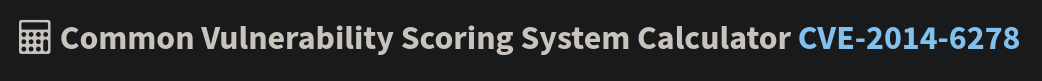


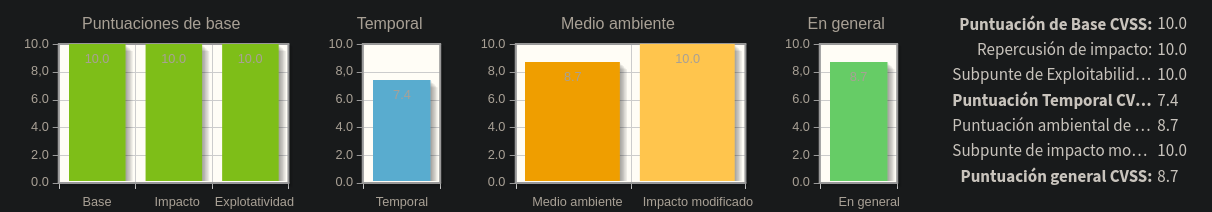
* **RESUMEN DE LAS EXPLOTACIÓNES PUERTO 8080 Y 8081:**
* 4.- FASE DE POST- EXPLOTACIÓN. - El siguiente paso consistiría en escalar privilegios en el sistema, encontrando varios archivos a través de los cuales conseguirlos, que son los SUID (Set User ID) y SGID (Set Group ID), los cuales presentan permisos especiales que pueden ser utilizados para permitir que un programa o script se ejecute con los privilegios del propietario del archivo o del grupo, independientemente de quién lo ejecute.

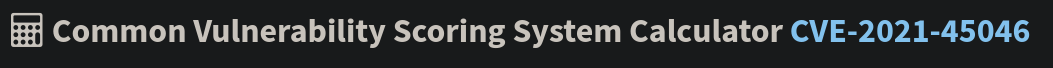


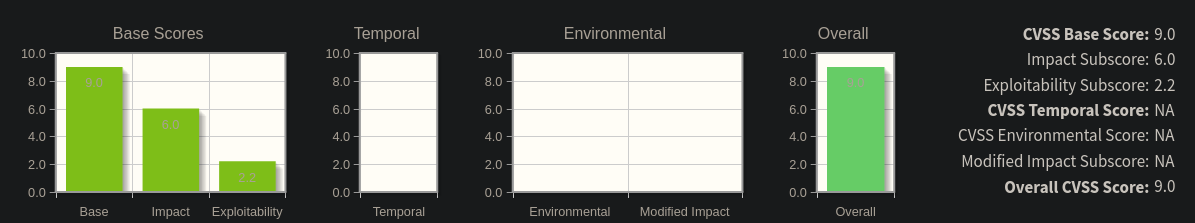
Archivos bit (suid y sgid)

Puerto8081 Puerto 8080

* 5.- PUNTUACIÓN DE LA VULNERABILIDAD SEGÚN NIST:







**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.