

**INFORME: EJECUTIVO Y TÉCNICO**

**SISTEMA KEVGIR**

* Fecha: 2 de octubre de 2024
* Cliente: Reto 15 – Team Challenge
* Consultora de Ciberseguridad: The Bridge - Accelerator
* Control de Cambios

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Versión | Documento | Fecha | Cambios | Autor | revisor | visto bueno |
| 1.1 | Informe de resultados | 09/10/2024 | 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 ---------------------------------------------------8

1. Informe Técnico: ----------------------------------------------------------------------------9

* Presentación------------------------------------------------------------------------------8
* Fase de exploración – Servidor – web---------------------------------------------8
* Fase de explotación-------------------------------------------------------------------11
* Conclusiones ---------------------------------------------------------------------------16

1. Bibliografía ----------------------------------------------------------------------------------19
2. Anexos ---------------------------------------------------------------------------------------12

1. INTRODUCCIÓN

El presente informe está formado por 2 partes: un **informe ejecutivo**, menos técnico y dirigido 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, así como funciones de detección y respuesta ante amenazas, **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. PRESENTACIÓN. – Este informe tiene como **objetivo** mostrar los resultados de las **vulnerabilidades detectadas y explotadas** en el equipo SAR, de acuerdo con el contrato firmado entre ambas partes, en el que permiten la explotación del sistema con la finalidad de conseguir la **autenticación y elevación de privilegios por atacantes externos,** consiguiendo **ser usuario con** **privilegios root, logrando,** además **persistencia** en el sistema explotado. El equipo no cuenta entorno gráfico (CLI)*,* el cual necesita para acceder una contraseña que no aportan, habiendo usado para su explotación diversas herramientas de ciberseguridad, destacando alguno de sus resultados:

* **Algo destacable y muy significativo**, es que al consultar en la web el nombre de la maquina *“KEVGIR*”, ofrece como primera opción la página de descarga de este tipo de máquinas, encontrando como credenciales “***user:resu”***, que, posteriormente probadas en el sistema analizado, son validadas, permitiendo el acceso al sistema.

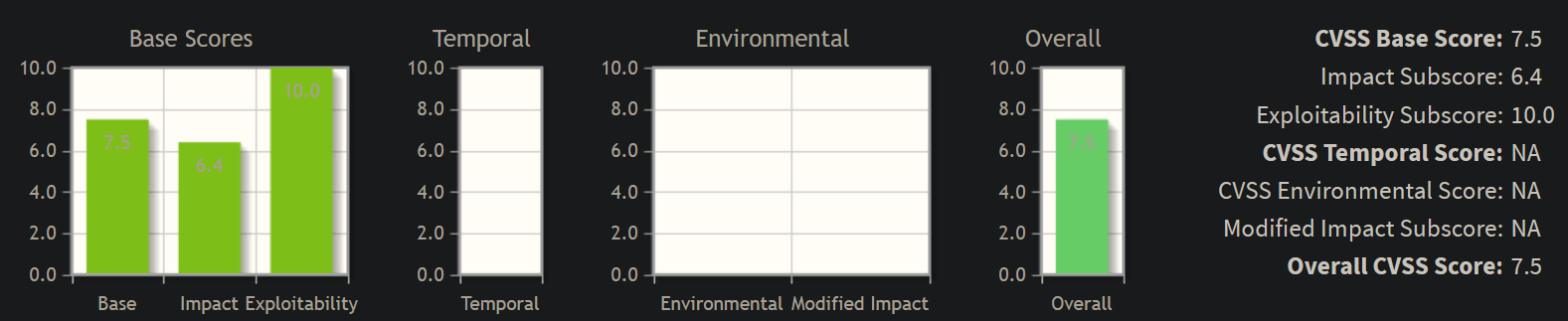


Esto de una **falla de seguridad muy importante**, por no haber modificado las credenciales de seguridad que vienen por defecto, aunque siendo de severidad media, debido a que el usuario posee muy pocos privilegios y el teclado esta predeterminado para el idioma turco.

* Mediante herramientas de **escaneo** de servidores web para descubrir **directorios, archivos, subdominios** y otros puntos de entrada ocultos y menos evidentes, se ha encontrado numerosos puertos abiertos y servicios, **destacand**o en este análisis los **puertos 80, 8080, 8081 y 9000**, encontrando en los puertos 9000 y 8081 directorios que pueden aportan información sensible para del sistema, concretamente, se ha podido acceder a los archivos que debería estar oculto a las búsquedas, denominado **“robots.txt”.** Este archivo, normalmente, es incluido por los administradores web para limitar la información que puede ser indexada por los buscadores web, **teniendo acceso**, si es visible, a **la estructura interna de la web, así como a cualquier dato importante** que haya podido anotar en el mismo el administrador.
* Se ha conseguido explotar varias vulnerabilidades graves, debido a una **falta de actualización** en las tecnologías o aplicaciones, por **contraseñas inseguras y débiles**, asi como en los **accesos a zonas sensibles**, que no deberían estar abiertas al público que iremos desarrollando en el próximo punto.

1. 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**:

* Se ha podido acceder al servicio **web “Joomla[[1]](#footnote-1)”,** el cual, presenta una **versión** esta **desactualizada**, a través de la cual es **posible** su **explotación**, permitiendo **cambiar** la **contraseña del administrador** **de manera remota** sin autenticación y, una vez dentro de la web, mediante **modificación de código HTML**, **acceso al sistema** con el usuario Joomla con escasos privilegios.

[[2]](#footnote-2)

* Una vez en dentro del sistema a través de CLI[[3]](#footnote-3), se comprueba que el **comando de Linux** **“cp[[4]](#footnote-4)”**, tiene unos permisos especiales conocidos como “Bit SUID[[5]](#footnote-5), que permite que cualquier usuario use el comando con privilegios root.
* Utilizando herramientas PEAS[[6]](#footnote-6), se ha conseguido numerosa información del sistema, y junto al privilegio root del comando de Linux *“cp”,* se ha podido acceder a la **herramienta web “Jenkins[[7]](#footnote-7)”**, encontrando su “hash[[8]](#footnote-8)” de contraseña, siendo descifrada, accediendo al servicio con las credenciales **“admin:hello”**, y una vez dentro, a través de una consola para **subir scripts[[9]](#footnote-9)**, se ha conseguido acceso al sistema con el **usuario**
* Se ha explotado el **servicio “Tomcat7[[10]](#footnote-10)”** y aprovechando el comando “cp”, se ha podido **acceder a un archivo de configuración** del servicio, pudiendo visualizar **sin cifrar** el usuario y la contraseña de Apache-Tomcat: **“tomcat:tomcat”.**
* Continuando con el aprovechamiento de los archivos Bit SUID, se ha conseguido explotar el comando **pkexec**[[11]](#footnote-11) y conseguir acceso al sistema con **privilegios máximos (root**) utilizandouna colección de archivos binarios para el citado escalamiento.
* Finalmente, ya con el acceso al sistema con máximo privilegios, se ha procedido a realizar **persistencia en el sistema[[12]](#footnote-12)**, modificando una tarea programada del sistema y reemplazando el archivo original por un script malicioso.

1. 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 y conseguir la persistencia en el mismo, aportando detalles más técnicos más adelante.
2. RECOMENDACIONES GENERALES.- Con base al análisis reciente de seguridad de su equipo KEVGIR, se han detectado varias vulnerabilidades que requieren atención para proteger los datos y garantizar el funcionamiento seguro de los sistemas. A continuación, se presentan una serie de recomendaciones claras y acciones a seguir para mitigar los riesgos identificados, en un lenguaje accesible para facilitar su comprensión:

1. El **archivo "robots.txt"** es visible en los puertos 8081 y 9000, contiendo información comprometida:

* **Problema**: El archivo **"robots.txt",** está diseñado para guiar a los motores de búsqueda sobre qué secciones del sitio web deben ignorar, actualmente está expuesto y contiene información que podría ser explotada para acceder a áreas sensibles del sitio.
* **Recomendación**: Solicitar a los desarrolladores web que **oculten el archivo de los escáneres públicos** y eliminen cualquier información confidencial que pueda ayudar a terceros a acceder a secciones no autorizadas, mediante la configuración adecuada del servidor web.

2. **Desactualizaciones de servicios o sistemas**.- Representan un riesgo crítico para la ciberseguridad, ya que el software desactualizado puede quedar vulnerable a exploits[[13]](#footnote-13) y ataques que se aprovechan de fallas de seguridad conocidos:

* **Problema:** EL servicio web “Joomla” desactualizado carece de parches de seguridad esenciales, lo cual lo hace vulnerable a explotaciones y ataques dirigidos a versiones antiguas.
* **Recomendación: Actualizar Joomla y otros servicios** a las versiones más recientes para garantizar la aplicación de parches de seguridad y reducir el riesgo de ataques.

3. **Bit SUID** en el Comando **“cp”**.- Tener activado el bit SUID en este comando, implica un riesgo significativo para la ciberseguridad, ya que, permite que cualquier usuario ejecute el comando con privilegios de otro usuario (normalmente root), lo cual puede **comprometer** la integridad del **sistema:**

* **Problema:** Activar SUID en el comando utilizado para copiar archivos, podría permitir que **un atacante copie o modifique archivos críticos** del sistema, escalando sus privilegios y comprometiendo el sistema.
* **Recomendación:** **Desactivar el bit SUID** de este comando y aplicar SUID sólo cuando sea absolutamente necesario, para evitar la escalada de privilegios no autorizada.

4. **Acceso no autorizado** a **“Jenkins”.-** El acceso a “Jenkins” a través del usuario *“admin”* y la contraseña descifrada a partir del hash de contraseña y el uso de la consola de script, una vez iniciado sesión en la web, representan una escalada de ataques críticos, lo que permite la manipulación de procesos de CI/CD[[14]](#footnote-14) y la integración de código malicioso:

* **Problema:** Al comprometer esta herramienta, un atacante puede modificar código, integrar malware y obtener credenciales sensibles.
* **Recomendación:** Utilizar contraseñas seguras y cifradas, restringir el acceso a la consola de scripts, y mejorar la configuración de seguridad del servidor Jenkins para mitigar ataques.

5. **Credenciales sin Cifrar en “tomcat-users.xml”.**-El citado archivo contiene credenciales para acceder a la aplicación “Apache Tomcat 7”, y mantener esas credenciales sin cifrar constituye un serio problema de ciberseguridad.

* **Problema:** Las credenciales sin cifrar en este o cualquier archivo, facilitan el acceso no autorizado, lo cual podría permitir a un atacante controlar el servidor Tomcat y desplegar aplicaciones maliciosas.
* **Recomendación:** Cifrar las credenciales del archivo y restringir su acceso sólo a usuarios autorizados, utilizando gestores de contraseñas seguros y aplicando buenas prácticas de segmentación de accesos.

6. **Servicio “pkexec”** con Bit SUID.-Este servicio tiene activado el bit SUID, lo que permite a usuarios sin privilegios ejecutar comandos con permisos elevados, representando un riesgo crítico, ya que el acceso puede ser escalado a máximos privilegios.

* **Problema:** Puede ser explotado mediante herramientas como “GTFOBins[[15]](#footnote-15)” para obtener máximos privilegios, tomando el control del sistema debido a la activación Bit SUID.
* **Recomendación:** Actualizar el servicio a la versión más reciente para que corrija la vulnerabilidad y reducir el uso de SUID, siempre que sea posible, para evitar la escalada de privilegios.

6. Importancia del **cifrado de credenciales** y la **segmentación de los accesos[[16]](#footnote-16)**.- La falta de estas medidas adecuadas de seguridad, facilitan la exposición de datos críticos y el compromiso de la infraestructura.

* **Problema:** La configuración inadecuada y la falta de cifrado provocó la exposición de datos sensibles, permitiendo la escalada de privilegios por parte del pentester.
* **Recomendación:** Usar buenas prácticas, como gestores de contraseñas seguras, segmentar accesos y aplicar una estricta minimización de privilegios para proteger los archivos de configuración y evitar accesos indebidos.

7. Adopción del **Modelo "Zero Trust".-** Además de las acciones mencionadas, y con carácter general, se recomienda evaluar y actualizar la política de seguridad de la empresa hacia el modelo de seguridad *“Zero Trust”[[17]](#footnote-17),* el cual, fortalecerá significativamente la postura de seguridad de la empresa al reducir la superficie de ataque y garantizar que sólo los usuarios autorizados puedan acceder a los datos críticos.

Si bien algunas de estas recomendaciones requieren un enfoque más técnico, es vital entender la importancia de estas acciones para evitar riesgos graves y potenciales violaciones de seguridad, sugiriendo que los equipos técnicos, desarrolladores y de seguridad trabajen de manera conjunta para implementar estas soluciones a la mayor brevedad posible.

1. 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)[[18]](#footnote-18) y la Ley Orgánica de Protección de Datos Personales y garantía de los derechos digitales (LOPDGDD)[[19]](#footnote-19). - 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)[[20]](#footnote-20). - 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.
* Directiva NIS2[[21]](#footnote-21). - En caso de comprometer a infraestructuras críticas o servicios esenciales, las empresas pueden enfrentarse a sanciones administrativas y reputacionales por no cumplir con los estándares mínimos de ciberseguridad exigidos.
* ISO - 27001[[22]](#footnote-22).- Ayuda a las empresas a identificar, gestionar y mitigar riesgos de ciberseguridad, asegurando la triada CIA y facilitando el cumplimiento de la NIS2.

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.

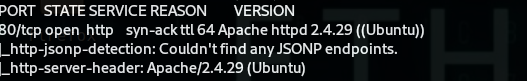
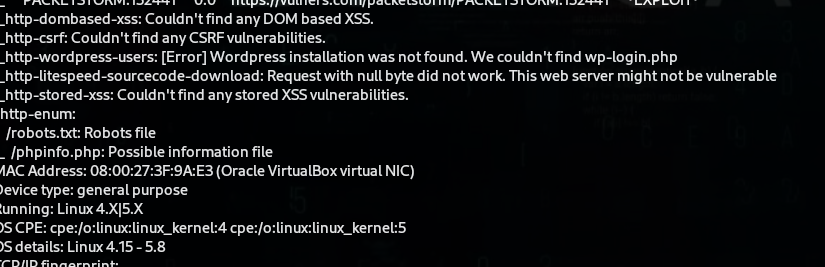
Además, la empresa está obligada a notificar a las autoridades y a los afectados en un plazo determinado las consecuencias del incidente, pudiendo agravar la repercusión pública del incidente a la reputación de la empresa.

3.- INFORME TÉCNICO

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

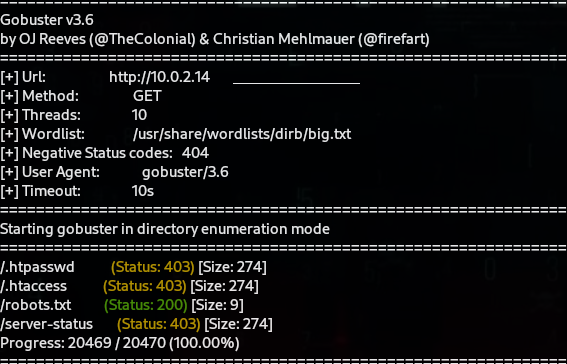
* El Equipo ha sido entregado con un sistema un kernel Linux sar 5.0.0-23-generic montado en un entorno GUI de Ubuntu 18.04.01 x86, sin aportar credenciales de inicio de sesión de la maquina denominada “SAR”, únicamente un nombre de usuario “LOVE”, por lo que el análisis y explotación será realizado 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.

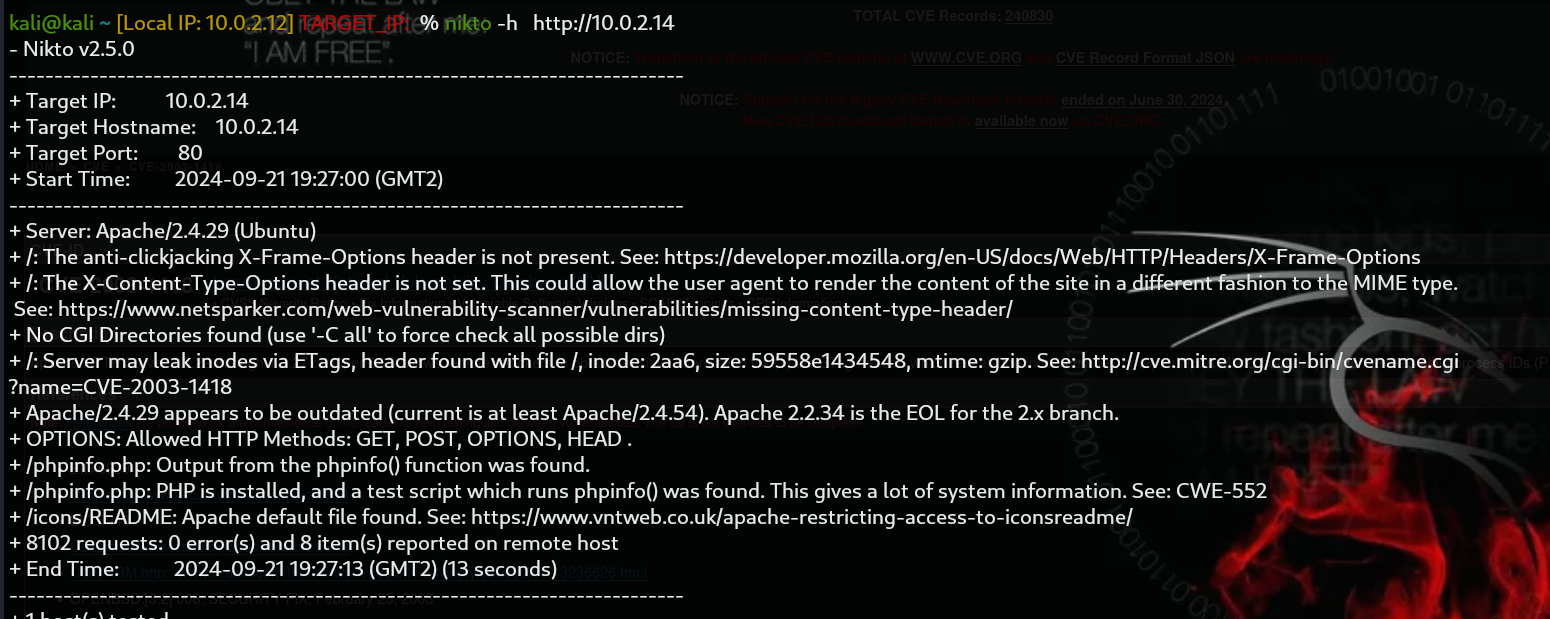
1. INFORMACIÓN INICIAL. - Se procede a consultar mediante Nmap*,* herramienta de código abierto utilizada para explorar y auditar la seguridad de redes y sistemas*,* el rango de IPs donde se encuentran ambas maquinas*,* siendo la de SAR: 10.0.2.14 y de la maquina atacante: 10.0.2.12. Además, la maquina objetivo únicamente tiene el puerto 80 abierto, el cual tiene asignado el servicio Apache 2.4.29(Ubuntu) y como directorios destacados del sistema: /robots y /phpinfo.php.

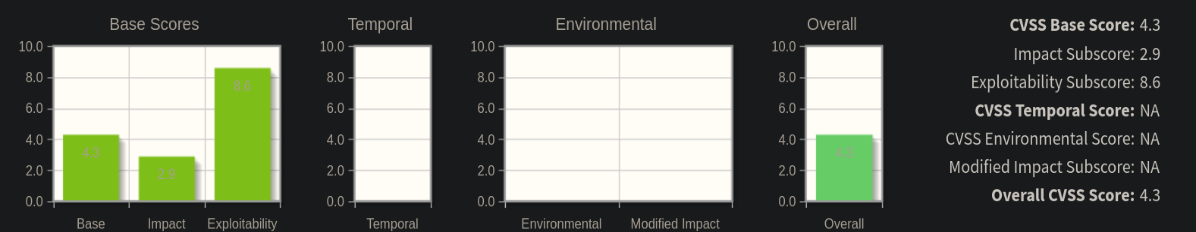
 

1. FASE DE EXPLORACIÓN – SERVIDOR WEB.

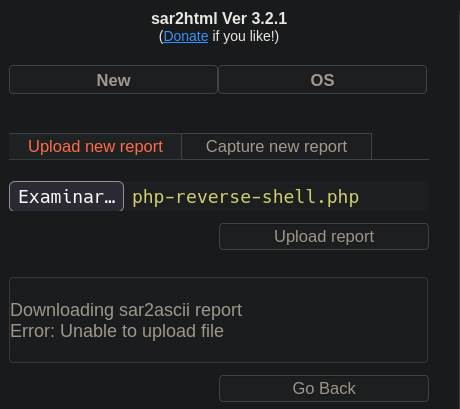
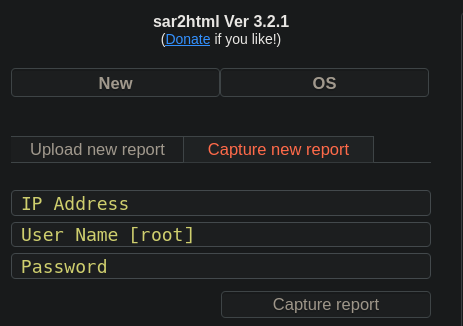
* Mediante el uso de *Gobuster*, siendo una herramienta de seguridad y hacking web, comúnmente utilizada durante las fases de reconocimiento en pruebas de penetración, que usa para descubrir objetos y directorios ocultos o no indexados en un servidor web, confirmando la presencia del directorio */robots.txt*, siendo éste, un archivo que los administradores de sitios web colocan en la raíz de su servidor, para dar instrucciones a los motores de búsqueda sobre cómo rastrear e indexar las páginas de la web, permitiendo si es visible, aportar información de la estructura de la web, asi como informaciones sensibles.

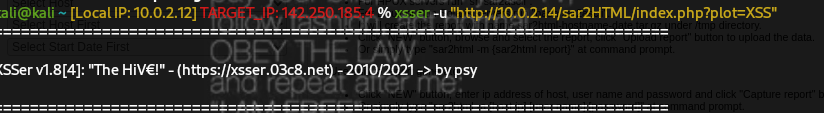
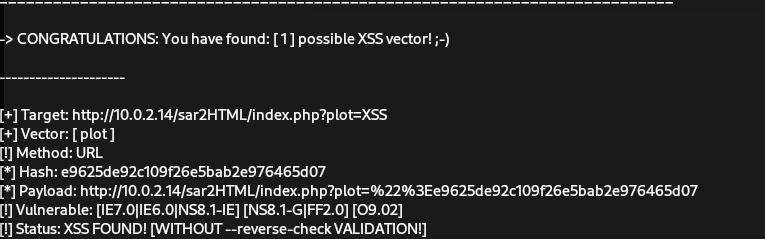
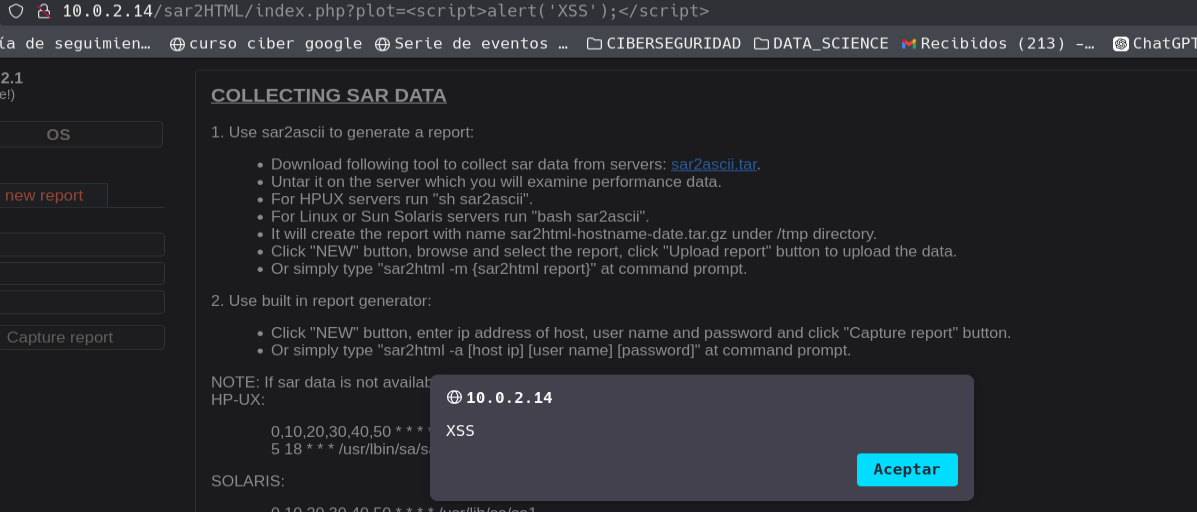


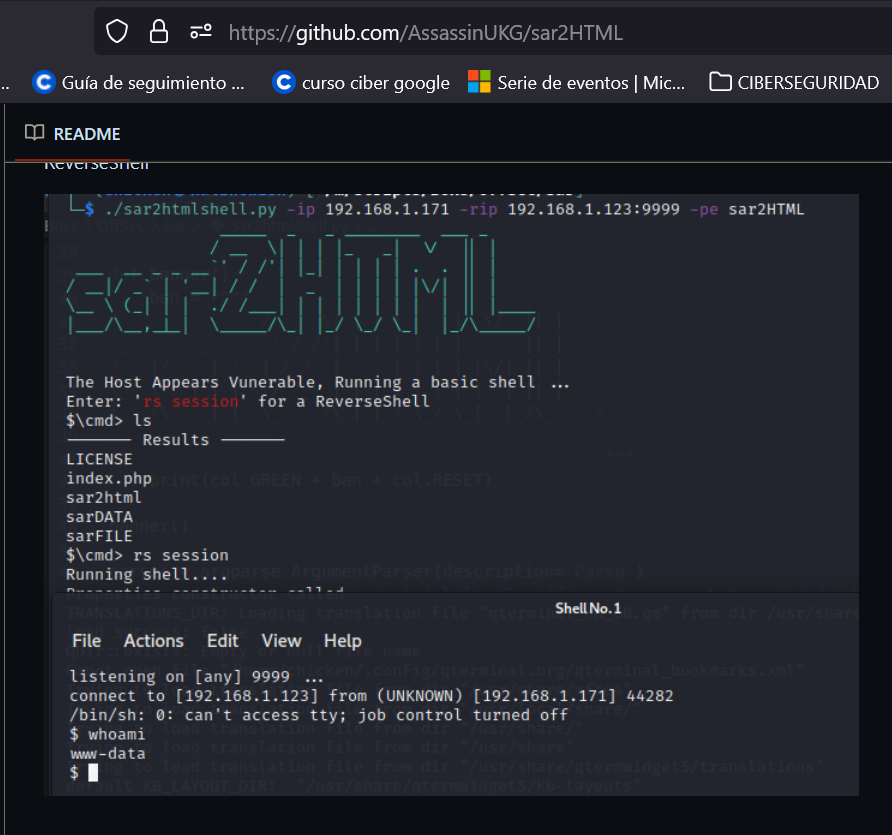
* Con la aplicación Nikto, herramienta de escaneo de vulnerabilidades web de código abierto, diseñada para analizar servidores web en busca de fallos de seguridad se procede a intentar encontrar vulnerabilidades que puedan ser explotadas, no encontrando ninguna aplicable directamente vía Metaexploit. No obstante, se podrían destacar algunas posibilidades: no presenta protección contra el clickjacking, el CVE-2003-1418, el cual puede permitir a atacantes remotos obtener informaciones sensibles a través de las cabeceras Etags, estando catalogada de media severidad y el CWE-552, el cual se refiere a archivos o directorios accesibles a actores no autorizados, concretamente al directorio */phpinfo.php*, que aporta mucha información del sistema.



* Se abre el directorio *[[23]](#footnote-23),* el cual se divide en 2 partes, en una parte de la web aparecen una serie de instrucciones e informaciones sobre la aplicación, y en la parte izquierda, información sobre su versión 3.2.1 y dos barras HTML, y tras hacer click en la etiquetada “New”, cambia la página, mostrando una parte que permite subir archivos a la web y además la URL se modifica <http://10.0.2.14/sar2HTML/index.php?plot=NEW> .

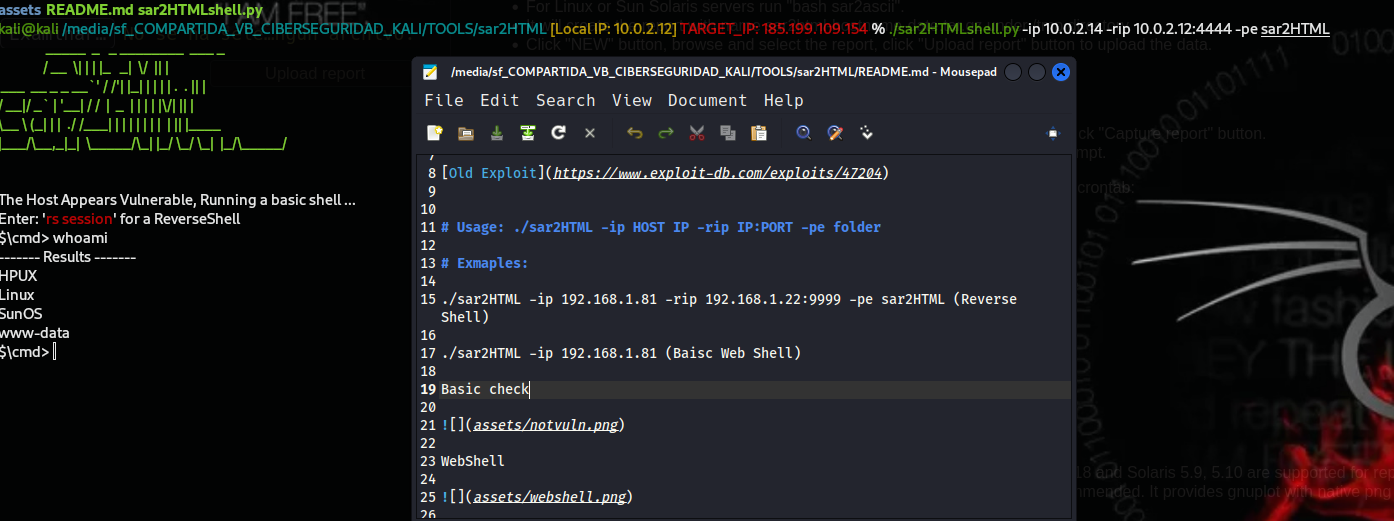


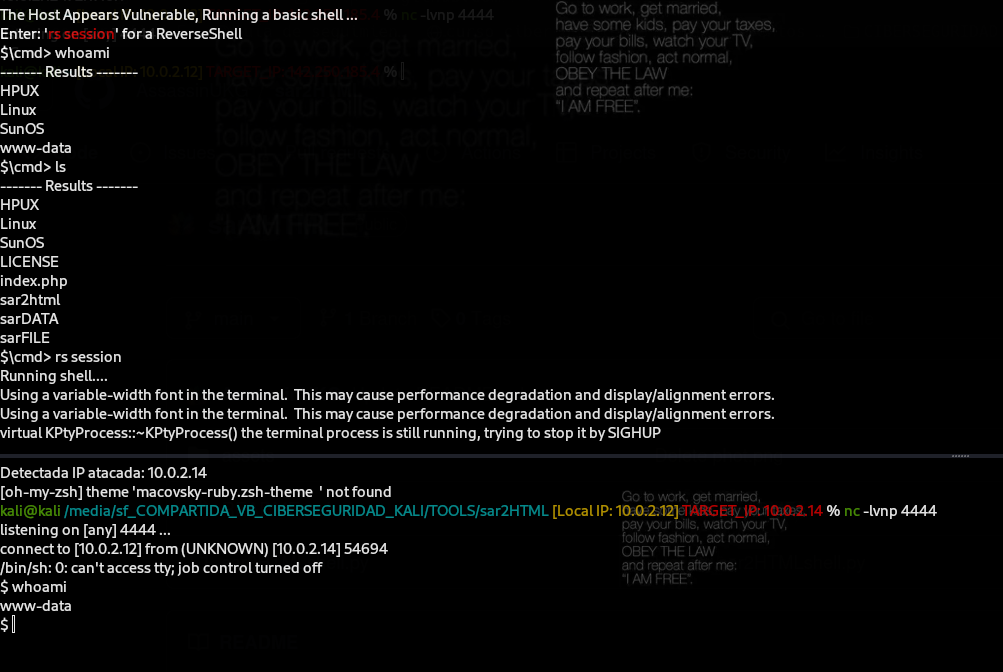
* Con la nueva URL, se procede a comprobar posibles vulnerabilidades (SQL, Path Traversal, LFI...), encontrando una vulnerabilidad XSS (Cross Site Scripting) de tipo reflejado, ya que se ejecuta como respuesta del servidor cuando el usuario interactúa con el URL, utilizando para ello la aplicación **“XSSer”,** la cual,es herramienta automatizada para detectar y explotar este tipo de vulnerabilidades en aplicaciones web, buscando formas de inyectar código malicioso y detectar XSS.
* Por todo lo anterior, se procede a la búsqueda en Google de la aplicación sar2HTML, apareciendo como primer enlace un exploit para esta aplicación, de la web exploit-db[[24]](#footnote-24), el cual permite la ejecución remota de comandos. Prosiguiendo con la búsqueda, se encuentra en GitHub[[25]](#footnote-25), un script de python que permite la explotación.



1. FASE DE EXPLOTACIÓN:

* Con el script mencionado, es posible abrir una shell o reverse shell, por lo que se procede a su descarga y ejecución, consiguiendo acceso a una shell básica a la maquina objetivo con privilegios básicos.

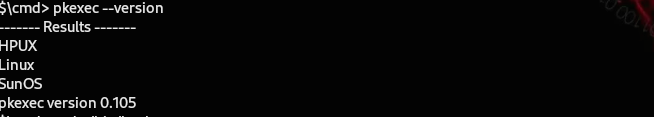
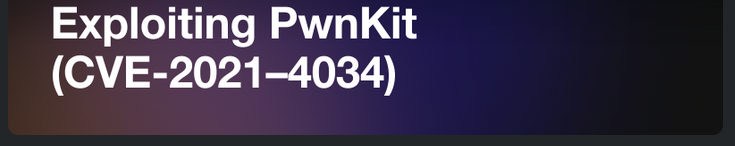


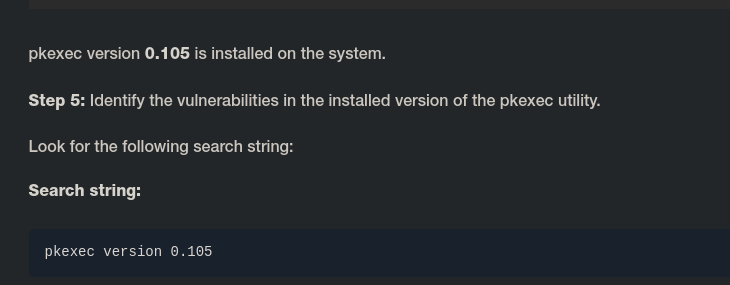


* Se procede a analizar posibles ficheros que se puedan ejecutar con permisos root, si no están correctamente configurados, los llamados *“bit SUID” (bit (4) Set User ID*), los cuales, pueden dar la capacidad de ejecutarse con los permisos del propietario del archivo, en lugar de con los permisos del usuario que lo ejecuta, es decir, si el usuario es root, podrías ejecutar archivos con ese permiso:



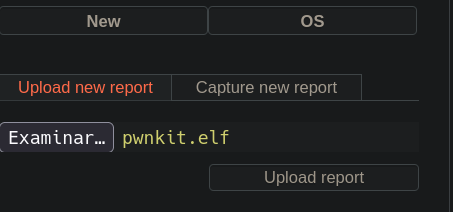
* Paralelamente, se realiza una búsqueda de información de los archivos “bit SUID” en la web, encontrando que, el situado en tercer lugar en la imagen anterior: *“pkexec”*, es una herramienta que permite a un usuario ejecutar programas con los privilegios de otro usuario, típicamente root, comprobando la versión en el sistema objetivo y buscando en la web posibles exploit[[26]](#footnote-26), encontrando una vulnerabilidad en esa versión con CVE-2021-4034.



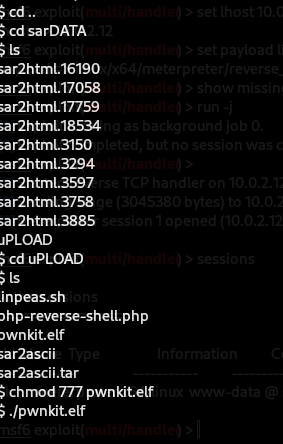
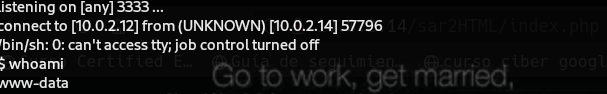


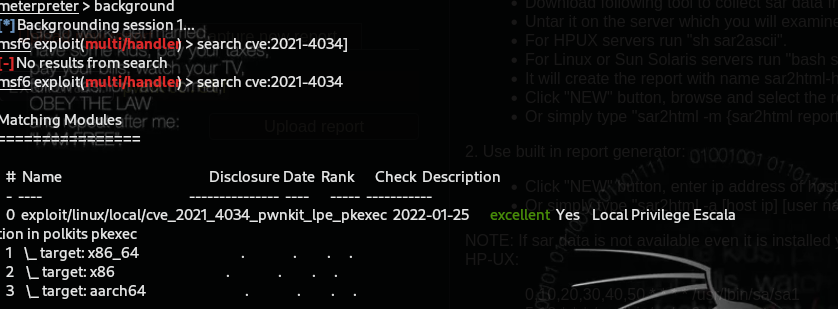
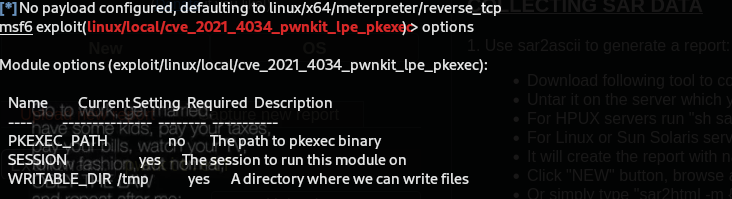
* Se procede a buscar la vulnerabilidad detectada anteriormente con CVE 2021-4034, siendo conocida como *“Pwnkit”,* la cual, es *una* vulnerabilidad crítica para la aplicación pkexec, que se produce por un error de desbordamiento de variables, permitiendo a un atacante manipular la forma en que se pasan los argumentos al programa, lo que puede ser explotado para ejecutar comandos maliciosos como root.

Este CVE se encuentra en Metaexploit, necesitando entre las distintas opciones necesarias para su ejecución, tener una sesión previa abierta, por lo que, aprovechando la sesión con privilegios limitados, que nos ha permitido abrir el exploit de GitHub y la posibilidad de subir archivos a través de la web al servidor, procedemos a aperturar un handler en MetaExploit y con esa sesión abierta, finalmente, se explotará la vulnerabilidad Pwnkit, la cual nos dará acceso root al sistema.

* Para ello, en primer lugar, a través de la herramienta MSFvenom, se realiza un payload con una shell interactiva, el cual procedemos a subir a través del enlace: <http://10.0.2.14/sar2HTML/index.php?plot=NEW>
* A través de la shell básica anteriormente explotada mediante un script de python para sar2HTML y cambiando a reverse shell con la opción que viene descrita, accediendo a “*/sarDATA/uPLOAD”* encontrando en esa carpeta el payload subido, dándole permisos adecuados y ejecutando el script “*./pwnkit.elf”*, no sin antes haber abierto un handler en Metaesploit a la escucha, consiguiendo la conexión con una meterpreter,





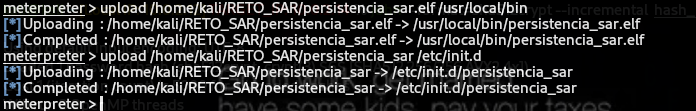
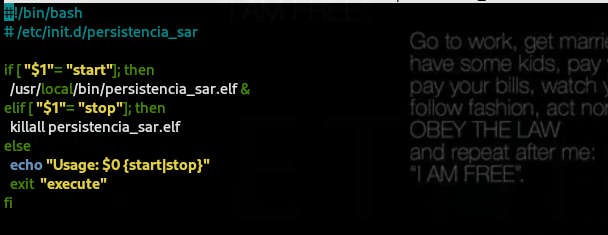
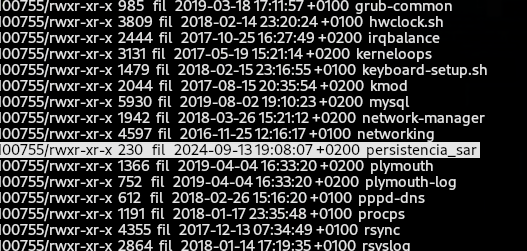
* Una vez que tenemos la sesión sin privilegios en MetaExploit, procedemos a explotar la vulnerabilidad con CVE 2021-4034 llamada *“Pwnkit”,* consiguiendo una shell interactiva (meterpreter) con privilegios root.

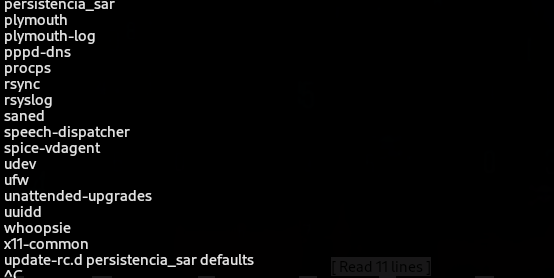




1. Ahora que hemos conseguido una shell interactiva con privilegios root, vamos a ganar persistencia, aprovechando para ello el archivo “*/etc/init.d”,* en el cual están las aplicaciones que se ejecutan al inicio de sesión de cualquier usuario. Para ello, realizamos un nuevo payload en otro puerto a la escucha, el cual será subido a “*usr/etc/bin”* y un script de Bash, para activar el payload como si fuera un servicio del sistema, que será el que se colocara en “*etc/init.d”*, estableciendo a ambos archivos los permisos necesarios de ejecución.

Cuando se inicie el sistema objetivo, los servicios y aplicaciones ubicadas en esa ruta, se iniciarán automáticamente(start), incluyendo el script, el cual establece que si el servicio se encuentra en “*start”* hace que se ejecute el payload malicioso, si está en *“stop”* que se pare el servicio y en cualquier otro caso, que pinte por pantalla: “execute”



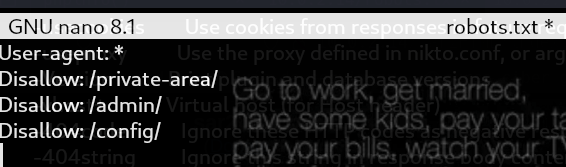


* Finalmente, se activa para que se inicie de manera automática en la configuración por defecto en el sistema objetivo, utilizando para ello el comando *“update-rc.d persitencia\_sar defaults”)* y reiniciamos con “reboot”, no sin antes haber preparado un nuevo handler en MetaExploit a la escucha de nuestro payload en el puerto 5555, resultando positivo, consiguiendo la persistencia en el sistema.

4.- CONCLUSIONES FINALES

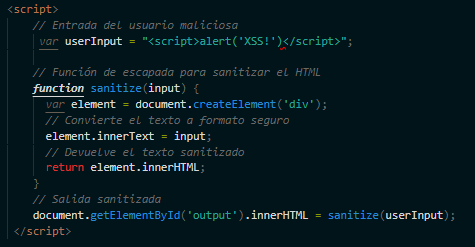
El análisis y explotación realizados en el sistema SAR han revelado varias vulnerabilidades críticas que podrían ser explotadas por actores maliciosos para obtener acceso no autorizado y elevación de privilegios en el sistema. Los siguientes puntos resumen los hallazgos clave y el proceso técnico que llevó a la explotación exitosa del sistema:

1. Exposición del archivo robots.txt:
   * La visibilidad del archivo *“robots.txt”* ha proporcionado información sensible que ha ayudado a dirigir la exploración hacia rutas específicas del sistema web, como el directorio /sar2HTML/, el cual, contenía vulnerabilidades críticas.
   * Recomendación: Limitar el acceso al archivo robots.txt y revisar el contenido para eliminar información confidencial o critica para el sistema, como, por ejemplo:



Aquí robots.txt, el cual debe estar oculto, bloquea el acceso a varios lugares de la web sensibles, que no serán indexados.

1. Vulnerabilidad XSS Reflejado:
   * Se ha encontrado una vulnerabilidad XSS reflejado en el sistema a través del parámetro *“plot”* de la URL en sar2HTML[[27]](#footnote-27), vulnerabilidad de seguridad web donde un atacante inyecta código malicioso (normalmente JavaScript) en una solicitud enviada al servidor, y este código es "reflejado" en la respuesta del servidor al cliente sin ser validado o sanitizado.
   * Esta vulnerabilidad ha permitido su explotación que podría comprometer la experiencia del usuario (cliente) y facilitar el acceso no autorizado al servidor web.
   * Recomendación: Implementar validaciones o sanitizaciones robustas en la entrada de datos para evitar la ejecución de código malicioso, como, por ejemplo:



1. Ejecución Remota de Código a través de sar2HTML:
   * Se ha explotado la vulnerabilidad en la versión 3.2.1 de sar2HTML, utilizando un exploit de GitHub, permitiendo abrir una reverse shell y obtener acceso a la máquina objetivo con privilegios básicos.
   * Recomendación: Actualizar la aplicación sar2HTML a su versión más reciente, siendo la 4.0.0 que fue lanzada el 25 de mayo de 2021 y limitar el acceso público a sus funcionalidades.
2. Explotación de la Vulnerabilidad CVE-2021-4034 (PwnKit):
   * Se ha explotado la vulnerabilidad de la aplicación pkexec, herramienta del paquete Polkit[[28]](#footnote-28), permitiendo una escalada de privilegios debido a un mal manejo de las variables de entorno, consiguiendo elevar los privilegios de una sesión básica obtenida primeramente, a una sesión con permisos root.
   * Esta vulnerabilidad crítica, ha permitido ejecutar código arbitrario con privilegios administrativos, comprometiendo completamente la seguridad del sistema.
   * Recomendación: Actualizar inmediatamente pkexec a la versión 0.12 o superior para corregir la vulnerabilidad CVE-2021-4034 y revisar la configuración de seguridad de todos los servicios con privilegios elevados.
3. Persistencia en el Sistema:
   * Se ha conseguido persistencia en el sistema objetivo, mediante la creación de un servicio que se ejecuta al inicio de sesión, garantizando el acceso continuado al sistema incluso después de reinicios, lo que representa una amenaza crítica de seguridad.
   * Recomendación: Tener instalados IDS/IPS que se encarguen de proteger, detectar y eliminar actividades sospechosas en el sistema, incluyendo la instalación de EDR en cada endpoint de la red, monitorizadores de la integridad de los archivos (FIM) que verifican la integridad de los mismos en el sistema, antivirus y antimalware en tiempo real, entre otros.

**Evaluación Final:**

Las vulnerabilidades descubiertas y explotadas demuestran la necesidad de mejorar la postura de seguridad del sistema SAR. Estas incluyen vulnerabilidades web (XSS), exposición de archivos sensibles (robots.txt), y graves problemas de configuración en servicios críticos como pkexec, siendo recomendable seguir las mejoras propuestas para reducir la superficie de ataque, corregir los fallos de seguridad detectados, y mejorar la gestión de privilegios en el sistema, implementando un modelo de seguridad “Zero Trust” y la adopción de mejores prácticas en la gestión de vulnerabilidades, actualizaciones y configuraciones seguras para evitar futuros compromisos del sistema.

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/>

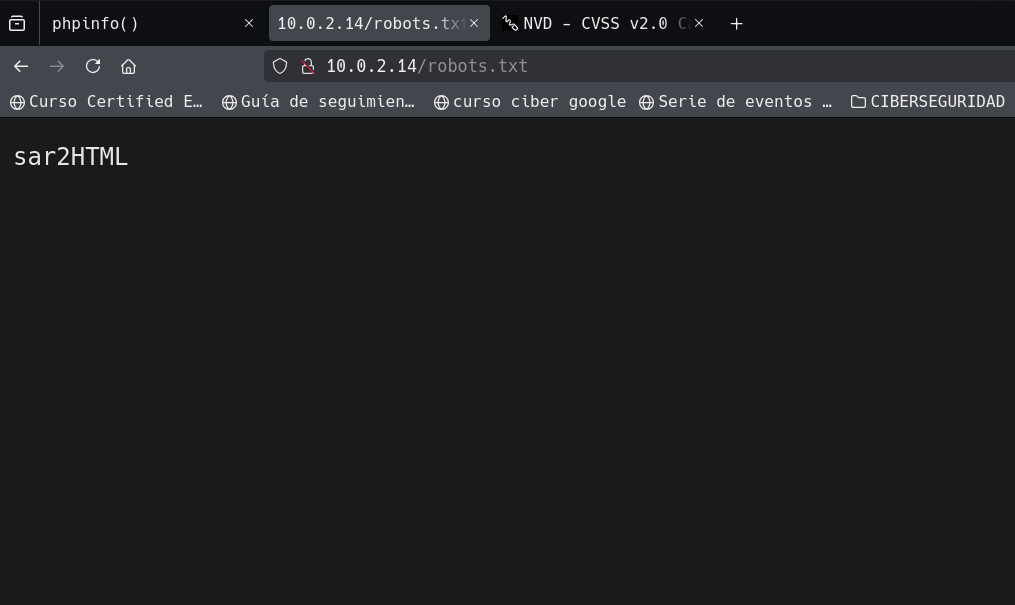
<https://nvd.nist.gov/vuln/detail/cve-2021-4034>

<https://www.ccn.cni.es/es/normativa/directiva-nis2>

6.- ANEXOS

1.- Se adjunta pdf con la gran información extraída de la web <http://10.0.2.14/phpinfo.php>.

2.- [http://10.0.2.14/](http://10.0.2.14/phpinfo.php)robots.txt



1. Sistema de gestión de contenidos (CMS) que permite crear y gestionar sitios web y aplicaciones en línea, ofreciendo herramientas para crear sitios web dinámicos sin necesidad de conocimientos avanzados de programación [↑](#footnote-ref-1)
2. CVE-2008-3681 [↑](#footnote-ref-2)
3. Interfaz de línea de comandos que permite a los usuarios interactuar con el sistema operativo o aplicaciones escribiendo **comandos de texto** [↑](#footnote-ref-3)
4. En sistemas Unix/Linux se utiliza para copiar archivos o directorios de una ubicación a otra, siendo similar al "copiar y pegar" [↑](#footnote-ref-4)
5. permiso especial en sistemas **Unix/Linux** que permite a un archivo ejecutable ser ejecutado con los privilegios del propietario, en lugar de los del usuario que lo ejecuta. [↑](#footnote-ref-5)
6. Colección de scripts diseñados para ayudar en la escalada de privilegios durante auditorías de seguridad o pruebas de penetración, con versiones para Linux y Windows( Privilege Escalation Awesome Scripts) [↑](#footnote-ref-6)
7. Herramienta de automatización de integración continua (CI) y entrega continua (CD), que permite a los desarrolladores automatizar procesos de construcción, pruebas y despliegue de aplicaciones, facilitando el desarrollo de software de forma ágil y colaborativa [↑](#footnote-ref-7)
8. Representación cifrada que se obtiene aplicando una función hash a la contraseña original, almacenándose en lugar de las contraseñas reales para mejorar la seguridad, si están tienen un numero y tipo de caracteres adecuados. [↑](#footnote-ref-8)
9. Conjunto de instrucciones o comandos escritos en un lenguaje de programación que se ejecutan de forma automática para realizar tareas específicas, siendo utilizados también, con motivos maliciosos. [↑](#footnote-ref-9)
10. Versión del servidor de aplicaciones Apache Tomcat, que se utiliza para desplegar y ejecutar aplicaciones web escritas en Java [↑](#footnote-ref-10)
11. Comando en sistemas Linux que permite ejecutar programas con los privilegios de otro usuario, similar a sudo, pero con una gestión de permisos más granular mediante PolicyKit, framework que gestiona permisos para permitir a aplicaciones no privilegiadas realizar tareas que requieren privilegios administrativos, sin otorgar acceso completo al sistema. [↑](#footnote-ref-11)
12. Son técnicas usadas para garantizar que un atacante pueda mantener acceso a un sistema comprometido incluso después de reinicios, cambios de contraseña, o intentos de desinfección, mediante la instalación de puertas traseras, modificaciones en archivos de configuración, o tareas programadas, permitiendo el acceso continuo al sistema. [↑](#footnote-ref-12)
13. Un código o técnica que aprovecha una vulnerabilidad en un sistema, software o aplicación para realizar acciones no autorizadas, como el acceso no permitido, la ejecución de código arbitrario o la escalada de privilegios [↑](#footnote-ref-13)
14. **CI (Integración continua)** implica integrar el trabajo de los desarrolladores frecuentemente para detectar errores rápidamente, mientras que **CD (Entrega continua)** automatiza el despliegue de aplicaciones después de pruebas exitosas. [↑](#footnote-ref-14)
15. Colección de binarios de Unix que se pueden utilizar para escalar privilegios o evadir restricciones de seguridad en sistemas mal configurados, pudiendo ejecutar comandos con privilegios elevados, sin necesidad de exploits adicionales. [↑](#footnote-ref-15)
16. La segmentación de accesos consiste en dividir los permisos y recursos dentro de un sistema para limitar el acceso de los usuarios únicamente a lo que necesitan, minimizando el riesgo, ya que, los permisos se restringen según roles y necesidades específicas. [↑](#footnote-ref-16)
17. 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. [↑](#footnote-ref-17)
18. El RGPD es un reglamento de la Unión Europea que establece normas estrictas para la protección de datos personales [↑](#footnote-ref-18)
19. La LOPDGDD es ley española que desarrolla el RGPD y que establece normas específicas para la protección de datos personales en España [↑](#footnote-ref-19)
20. 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 incumplimiento., [↑](#footnote-ref-20)
21. Directiva NIS2 (Seguridad de Redes y Sistemas de Información 2) es una actualización de la Directiva NIS original, aprobada por la Unión Europea, con el objetivo de fortalecer la ciberseguridad en los sectores esenciales y en las infraestructuras críticas de los Estados miembros de la UE. [↑](#footnote-ref-21)
22. Norma internacional que define los requisitos para establecer, implementar, mantener y mejorar un sistema de gestión de seguridad de la información (SGSI), cubriendo aspectos como, el control de acceso, la gestión de incidentes de seguridad y la continuidad del negocio, siendo ampliamente utilizada para demostrar el compromiso de una organización con la ciberseguridad y la protección de datos. [↑](#footnote-ref-22)
23. Herramienta de monitoreo del rendimiento del sistema en Unix/Linux, que recopila datos sobre la actividad del sistema, convirtiendo los informes generados por la utilidad **sar** (System Activity Report) en un formato HTML, más fácil de visualizar y entender [↑](#footnote-ref-23)
24. <https://www.exploit-db.com/exploits/47204> [↑](#footnote-ref-24)
25. <https://github.com/AssassinUKG/sar2HTML> [↑](#footnote-ref-25)
26. <https://ine.com/blog/exploiting-pwnkit-cve-20214034> [↑](#footnote-ref-26)
27. <http://10.0.2.14/sar2HTML/index.php?plot=NEW> [↑](#footnote-ref-27)
28. paquete de software que permite gestionar permisos y privilegios en sistemas basados en Linux. [↑](#footnote-ref-28)