INFORME: EJECUTIVO Y TÉCNICO

- Fecha: 18 de septiembre de 2024
- Cliente: Reto 13 – Team Challenge
- Consultora de Ciberseguridad: The Bridge - Accelerator
- Control de Cambios

Versión	Documento	Fecha	Cambios	Autor	revisor	visto bueno
1.1	Informe de resultados	25/09/2024	Informe inicial	Victor Martínez	Ángel / Joseba	Javier Tomás

Índice de Contenidos

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

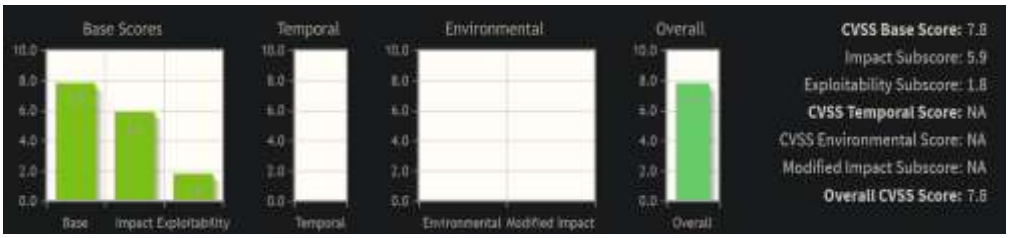
1. PRESENTACIÓN. – Este informe tiene como objetivo mostrar los resultados de las vulnerabilidades detectadas y explotadas en el equipo SAR, de acuerdo con el contrato firmado entre ambas partes, en el que permiten la explotación del sistema con la finalidad de conseguir la autenticación por atacantes externos con usuarios con privilegios root. El equipo cuenta con entorno gráfico con un usuario *"love",* el cual necesita para acceder una contraseña que no aportan, habiendo usado para su explotación diversas herramientas de ciberseguridad, destacando alguno de sus resultados:

    - 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, concretamente, se ha podido acceder a un archivo 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.
    - En este caso concreto, gracias al la información aportada en el archivo, hemos podido acceder a una parte de la web: http://10.0.2.14/sar2HTML/index.php?plot=NEW , a través de la cual se pueden subir archivos sin tener que acreditarse y además es vulnerable a una modalidad de ataque (XSS), a través de la cual se puede obtener acceso no autorizado al servidor con la posibilidad de ejecución de código arbitrario malicioso.
    - Teniendo en cuenta lo anterior, se ha explotado esta vulnerabilidad a través de un script con el cual podemos abrir una sesión no autorizada en el servidor web con privilegios muy limitados, pero mediante otra vulnerabilidad se han podido elevar esos privilegios a root.

2. 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 con las vulnerabilidades XSS y el script con acceso al sistema, se han analizado una seria de archivos que, si no están bien configurados, permiten a un atacante externo ejecutar código por privilegios root, encontrando concretamente un servicio llamado *"pkexec",* el cual es vulnerable a un CVE con la versión que tiene instalada el sistema SAR:

CVE-2021-4034



    - Esta vulnerabilidad de escalada de privilegios, afecta a la aplicación *pkexec*, siendo una herramienta diseñada para permitir a los usuarios de un sistema, que no poseen privilegios, ejecutar comandos como usuarios privilegiados de acuerdo con políticas predefinidas en el sistema, pudiendo un atacante aprovechar esto, para ejecutar código arbitrario y la consecución de privilegios root.
    - Esta ruta es la que se ha seguido para la explotación de su equipo, consiguiendo acceso root, y, además, se ha modificado el sistema, llegando a obtener persistencia en el mismo, con la posibilidad de poder entrar en el equipo siempre que el sistema se encuentre encendido.

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 y conseguir la persistencia en el mismo, aportando detalles más técnicos más adelante.

4. RECOMENDACIONES GENERALES

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.

Con base al análisis reciente de seguridad de su equipo SAR, 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 y contiene información comprometida:

- **Problema**: El archivo "robots.txt", 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. Esto puede hacerse mediante la configuración adecuada del servidor web.

2. Vulnerabilidad XSS (Cross-Site Scripting) detectada

- **Problema**: Se detectó una vulnerabilidad de XSS, lo que permite a un atacante inyectar código malicioso en la web, que puede comprometer la experiencia de los usuarios o robar información.
- **Recomendación**: Implementar filtros para validar los datos que los usuarios envían al sitio web. El equipo de desarrollo debe asegurarse de que todas las entradas de usuario sean debidamente limpiadas y no permitan la ejecución de código no deseado.

3. Subida de archivos sin autenticación

- **Problema**: Hay una funcionalidad en la página que permite a cualquiera subir archivos sin necesidad de ingresar credenciales. Esto podría ser aprovechado para cargar contenido malicioso en el servidor.
- **Recomendación**: Restringir el acceso a esta funcionalidad, requiriendo que los usuarios se autentiquen antes de subir cualquier archivo. Además, se deben limitar los tipos de archivos que se pueden subir y analizar cualquier archivo cargado en busca de malware.

4. Exposición de información sobre la herramienta SAR 3.2.1

- **Problema**: El sitio web revela información sobre la versión 3.2.1 de la herramienta "SAR". Esto es un riesgo, ya que los atacantes pueden usar esa información para buscar vulnerabilidades conocidas en esa versión.
- **Recomendación**: Ocultar esta información de las respuestas públicas de la web y actualizar la versión de SAR a la última disponible. Mantener todas las herramientas y software del servidor actualizados es esencial para prevenir ataques.

5. Ejecución de un script que permite acceso no autorizado

- **Problema**: La versión de SAR utilizada en su equipo, ha permitido que se ejecutara un script, proporcionando acceso no autorizado a la web.
- **Recomendación**: Solicitar al equipo técnico que revise todas las herramientas y componentes del sitio, asegurándose de que estén actualizados y que ninguna versión antigua esté comprometida.

6. Servicio vulnerable detectado: "pkexec" con escalada de privilegios

- **Problema**: Se encontró que el servicio "pkexec" tiene una vulnerabilidad crítica (CVE-2021-4034) que permite a usuarios no privilegiados obtener acceso de nivel "root" (máximo nivel de control en un servidor). Este problema puede ser explotado para tomar el control total del sistema.
- **Recomendación**: Actualizar de inmediato la herramienta "pkexec" a la última versión que corrige esta vulnerabilidad. Esta acción debe ser una prioridad, ya que la explotación de esta vulnerabilidad podría dar control total a un atacante sobre los servidores de la empresa.

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"*[1], 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] 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.
- Directiva NIS2[5]. - 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.

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.

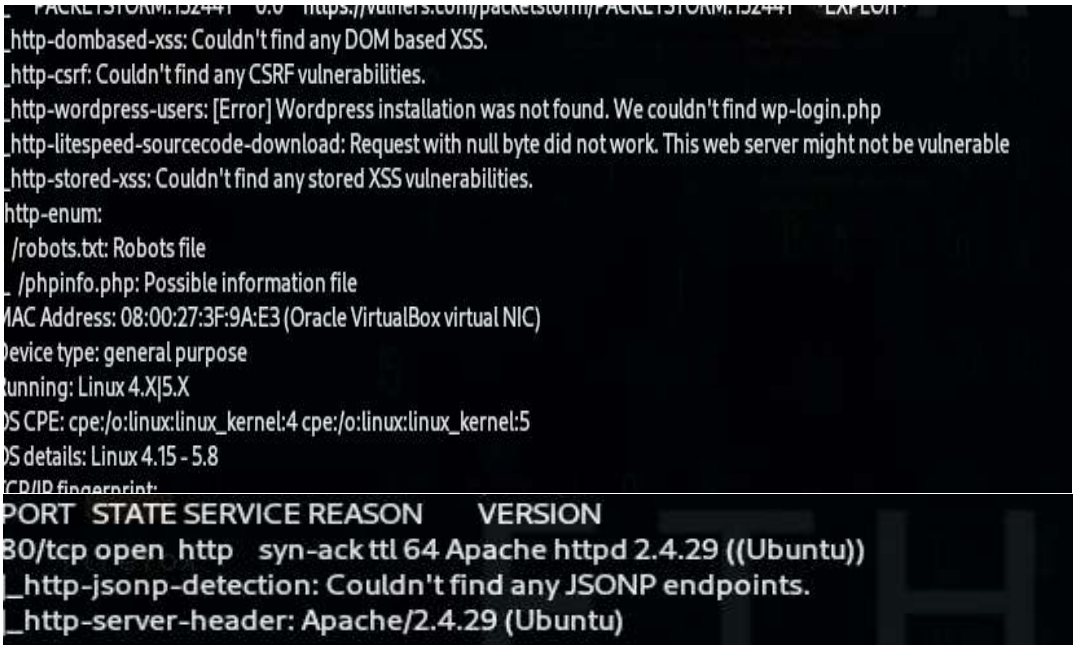
[2] El RGPD es un reglamento de la Unión Europea que establece normas estrictas para la protección de datos personales
[3] La LOPDGDD es ley española que desarrolla el RGPD y que establece normas específicas para la protección de datos personales en España
[4] La LSSI es una legislación española que 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 incumplimiento.,
5 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.

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.



2. FASE DE EXPLORACIÓN – SERVIDOR WEB.

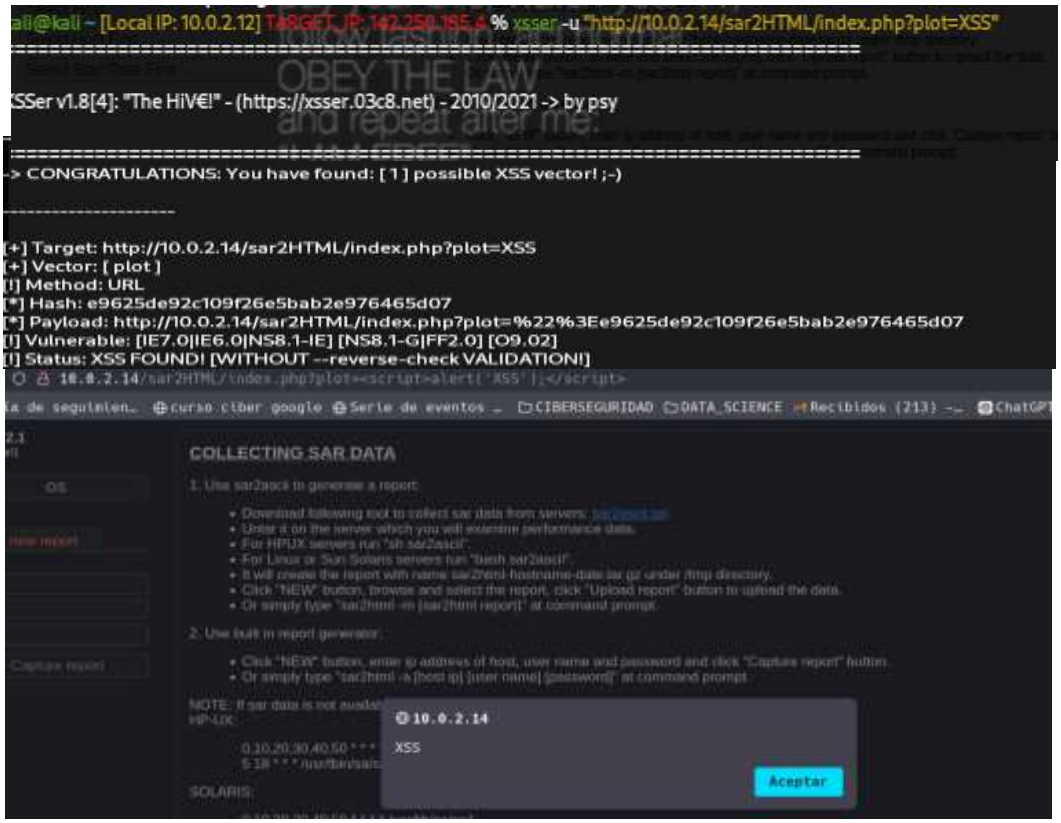
- Mediante el uso de *Gobuster*, siendo una herramienta de seguridad y hacking web, comúnmente utilizada durante las fases de reconocimiento en pruebas de penetración, que usa para descubrir objetos y directorios ocultos o no indexados en un servidor web, confirmando la presencia del directorio */robots.txt*, 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.

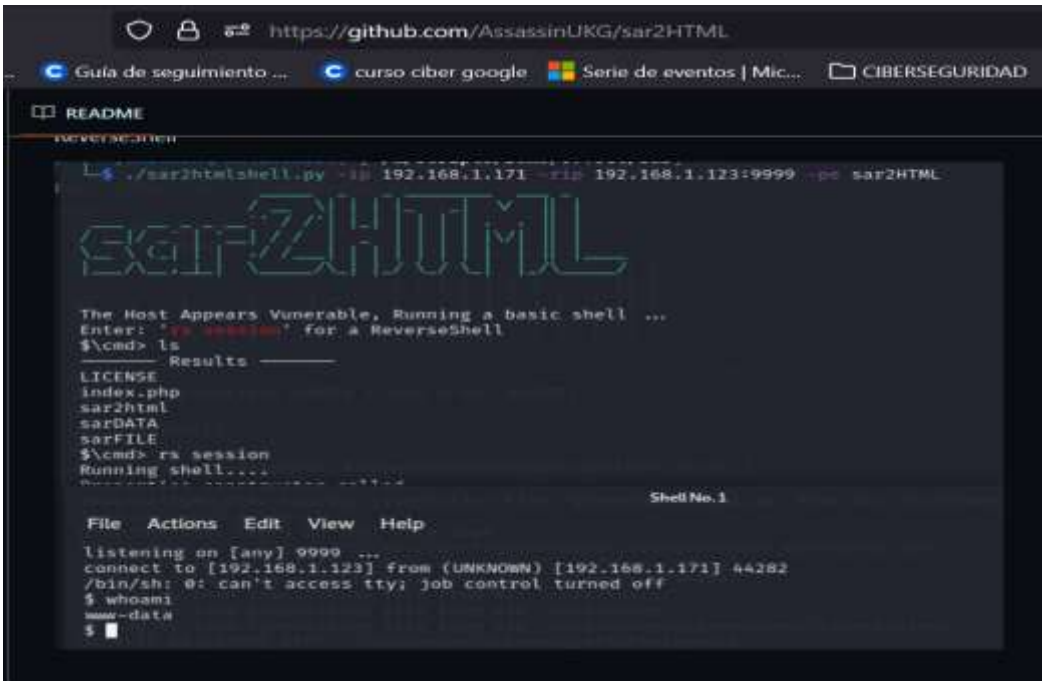
- Se abre el directorio http://10.0.2.14/sar2HTML/ [6], el cual se divide en 2 partes, en una parte de la web aparecen una serie de instrucciones e informaciones sobre la aplicación, y en la parte izquierda, información sobre su versión 3.2.1 y dos barras HTML, y tras hacer click en la etiquetada “New”, 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.



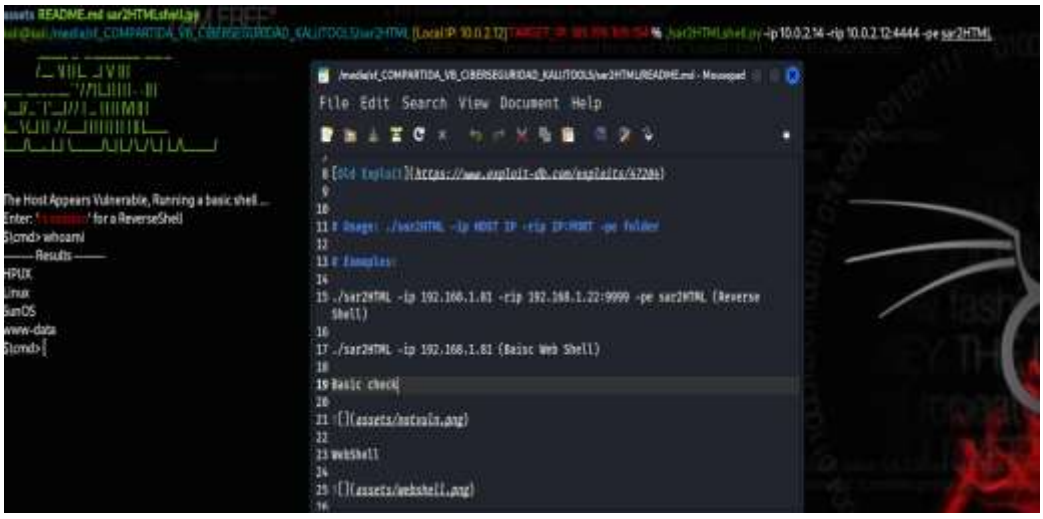
[6] Herramienta de monitoreo del rendimiento del sistema en Unix/Linux, que recopila datos sobre la actividad del sistema, convirtiendo los informes generados por la utilidad **sar** (System Activity Report) en un formato HTML, más fácil de visualizar y entender

- 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[7], el cual permite la ejecución remota de comandos. Prosiguiendo con la búsqueda, se encuentra en GitHub[8], un script de python que permite la explotación.



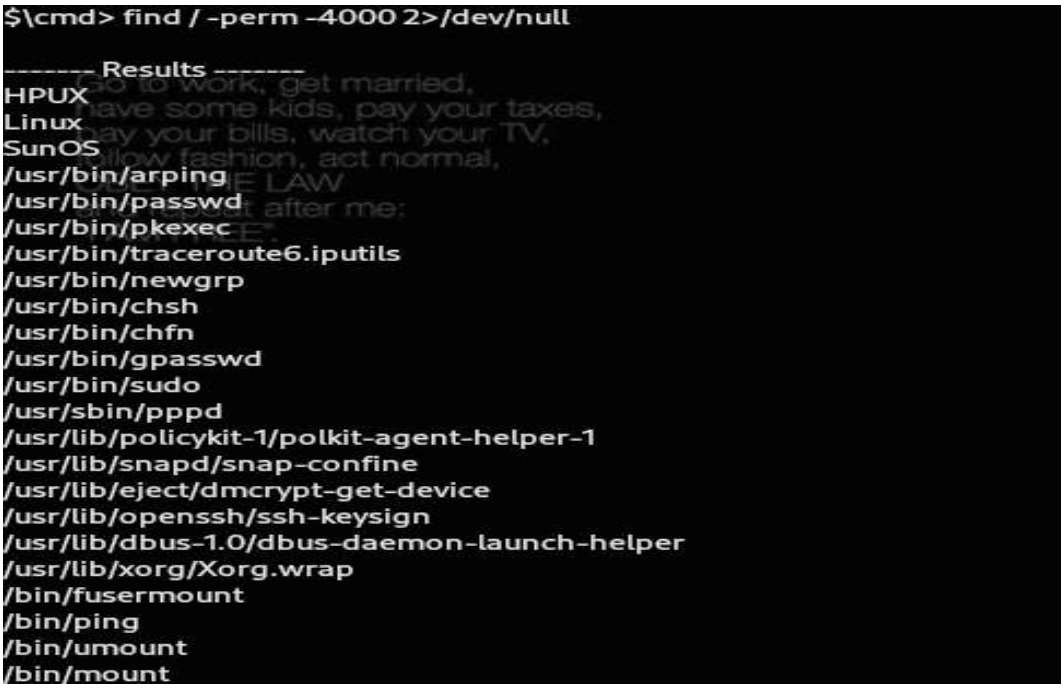
3. FASE DE EXPLOTACIÓN:

- 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.

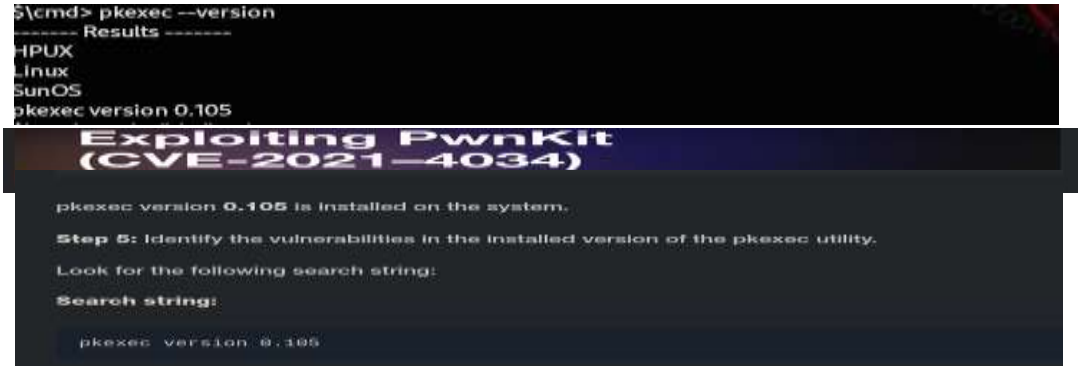


[7] https://www.exploit-db.com/exploits/47204
[8] https://github.com/AssassinUKG/sar2HTML

- 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:



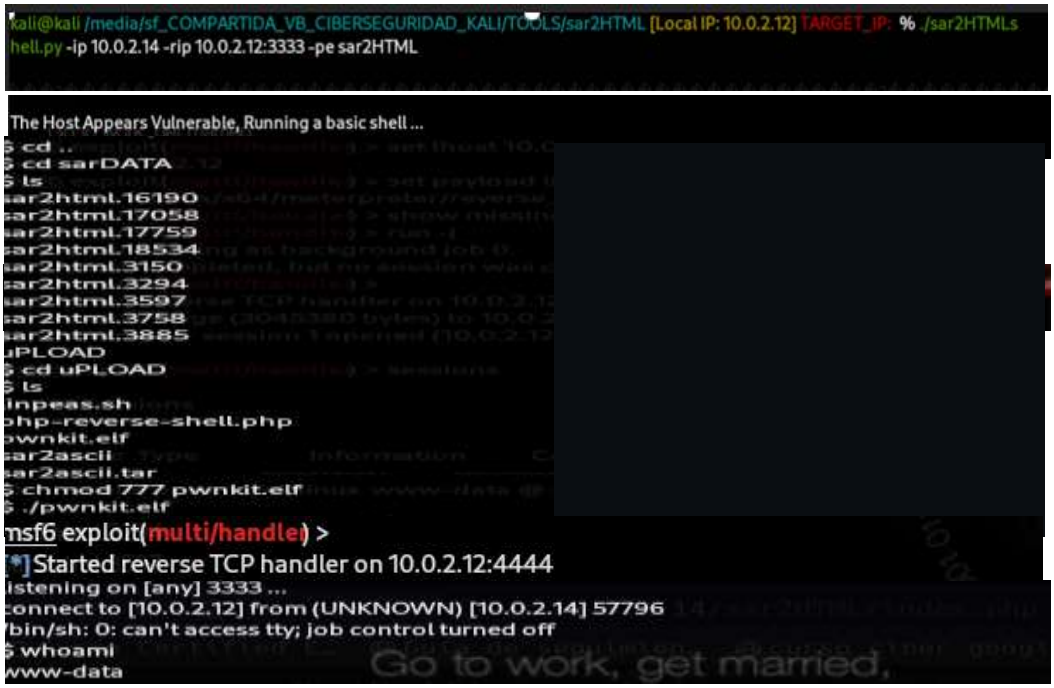
- Paralelamente, se realiza una búsqueda de información de los archivos "bit SUID" en la web, encontrando que, el situado en tercer lugar en la imagen anterior: *"pkexec"*, es una herramienta que permite a un usuario ejecutar programas con los privilegios de otro usuario, típicamente root, comprobando la versión en el sistema objetivo y buscando en la web posibles exploit[9], 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

[9] https://ine.com/blog/exploiting-pwnkit-cve-20214034

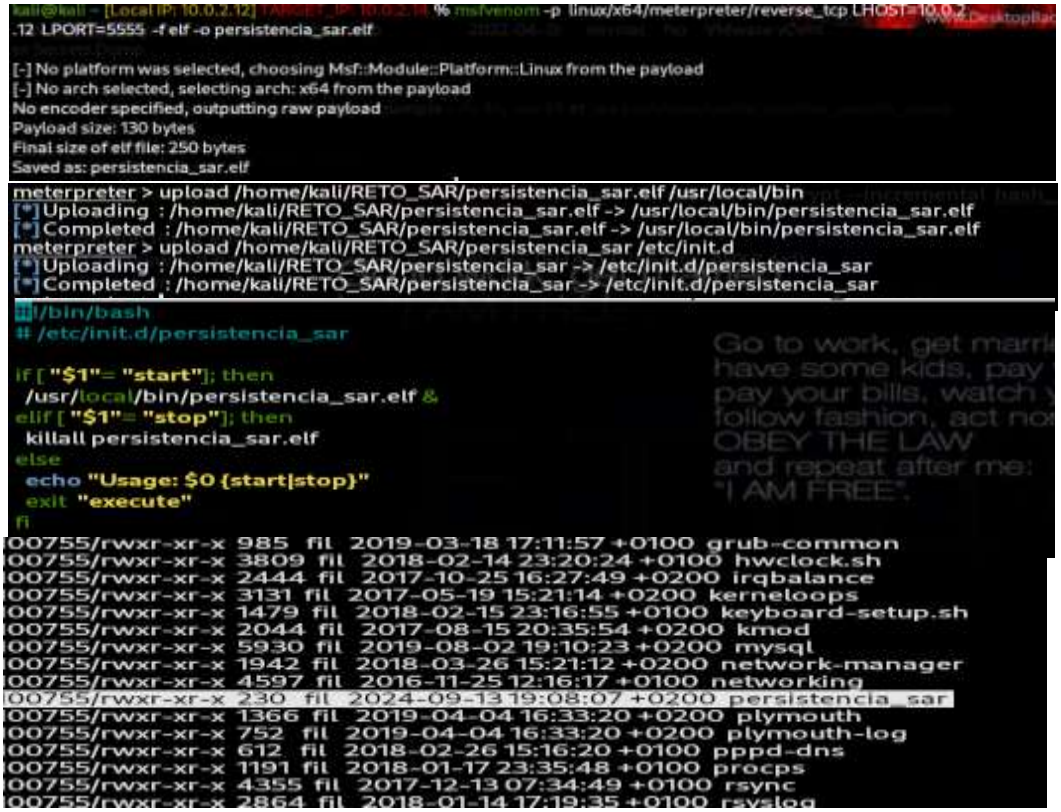
- A través de la shell básica anteriormente explotada mediante un script de python para sar2HTML y cambiando a reverse shell con la opción que viene descrita, accediendo a *"/sarDATA/uPLOAD"* encontrando en esa carpeta el payload subido, dándole permisos adecuados y ejecutando el script "*./pwnkit.elf*", 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.

2. Vulnerabilidad XSS Reflejado:

   - Se ha encontrado una vulnerabilidad XSS reflejado en el sistema a través del parámetro *"plot"* de la URL en sar2HTML[10], 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:

```
<script>
    // Entrada del usuario maliciosa
    var userInput = "<script>alert('XSS!')</script>";

    // Función de escapada para sanitizar el HTML
    function sanitize(input) {
      var element = document.createElement('div');
      // Convierte el texto a formato seguro
      element.innerText = input;
      // Devuelve el texto sanitizado
      return element.innerHTML;
    }
    // Salida sanitizada
    document.getElementById('output').innerHTML = sanitize(userInput);
</script>
```

3. 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.

4. Explotación de la Vulnerabilidad CVE-2021-4034 (PwnKit):

   - Se ha explotado la vulnerabilidad de la aplicación pkexec, herramienta del paquete Polkit[11], 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.

[10] http://10.0.2.14/sar2HTML/index.php?plot=NEW

[11] paquete de software que permite gestionar permisos y privilegios en sistemas basados en Linux.

- 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.

5. 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/robots.txt