SARAC

Reporte Técnico

Evaluación Inicial del Estado de Amenazas en Ciberseguridad

DOCUMENTO CONFIDENCIAL

Fecha
27 de septiembre de 2024

CIBER-TEC Security and Defense Solutions

UT GESTIÓN TECNOLÓGICA SSTC

Tabla de Contenidos

Cronología del Proyecto

Septiembre 18 de 2024	Kick-off del proyecto
Septiembre 25 de 2024	Reunión técnica
Septiembre 27 de 2024	Reporte finalizado
Septiembre 30 de 2024	Reporte enviado

Resumen Ejecutivo

Este informe presenta un análisis detallado del estado actual de la seguridad de la información y la ciberseguridad en la organización, identificando las principales vulnerabilidades, riesgos y vectores de ataque que afectan su infraestructura tecnológica. A través de un proceso riguroso de evaluación de riesgos y auditorías técnicas, se han detectado vulnerabilidades críticas en varios sistemas clave, lo que representa un riesgo significativo para la confidencialidad, integridad y disponibilidad de los datos. Se han priorizado las acciones inmediatas necesarias para mitigar estos riesgos, incluyendo la aplicación de parches de seguridad, el endurecimiento de sistemas, y la implementación de medidas de control de acceso más estrictas.

Durante el proceso de evaluación se identificaron un total de 49 vulnerabilidades distribuidas en 7 críticas, 18 altas y 24 medias. La metodología caja negra fue clave para entender la postura actual de la organización frente a un posible ataque en un entorno de producción sobre su infraestructura expuesta a internet. El estudio de anticipación de amenazas determino que existe un grupo llamado APT-C-36 que podría intentar comprometer la organización mediante técnicas especificas que son detalladas en este informe junto con las contramedidas necesarias para evitar la materialización e impacto de un ataque relacionado. Adicionalmente, la identificación de 54 credenciales filtradas en múltiples fuentes durante los últimos 9 meses permitió detectar posibles fallos en las políticas de contraseñas en la organización y usuarios que han estado comprometidos en algún transcurso del tiempo mencionado. Finalmente se identificaron 2 repositorios de código públicos que contiene información sensible relacionada a la organización.

Este informe recopila el detalle técnico de todos los hallazgos durante la evaluación inicial de seguridad para la Unidad Nacional de Protección UNP en el periodo de ejecución mencionado en la cronología del proyecto. Se recomienda validar los sistemas de seguridad que pudieran estar implicados en la detección de todas las acciones realizas con el objetivo de identificar puntos a fortalecer frente a este tipo de escenarios de reconocimiento sobre la organización.

Alcance de la Evaluación

El objetivo principal del **análisis de vulnerabilidades** fue identificar, evaluar y priorizar los riesgos presentes en la superficie de ataque externa de la organización, compuesta por dominios, IPs públicas, y aplicaciones críticas. Este análisis detallado se enfocó en detectar fallos de seguridad que pudieran ser explotados por actores maliciosos para comprometer la integridad, confidencialidad y disponibilidad de los sistemas y datos de la organización. Los hallazgos del análisis proporcionarán la base para implementar medidas de mitigación que refuercen la postura de seguridad cibernética de la organización.

Tabla de Activos Evaluados

DIRECCIÓN IP	HOSTNAME	DESCRIPCIÓN	ESTADO
20.62.113.96	soadoc.unp.gov.co/share/	Alfresco - Gestor de Contenidos Apache Solr WildFly	VULNERABLE
170.254.230.199	yunalesca.unp.gov.co	DNS Server	SEGURO
170.254.230.202	correo.unp.gov.co	Servidor de Correo	SEGURO
170.254.230.203	No identificado	Hikvision Web Server (1450)	SEGURO
170.254.230.205	sip.unp.gov.co	No identificado	OFFLINE
170.254.230.206	webconf.unp.gov.co	No identificado	OFFLINE
170.254.230.207	av.unp.gov.co	No identificado	OFFLINE
186.28.255.105	correo.unp.gov.co	Anterior IP del hostname	OFFLINE
186.31.104.65	No identificado	SSH (830)	SEGURO
186.31.104.67	No identificado	SSH (830)	SEGURO
186.154.254.161	No identificado	SSH (830)	SEGURO
186.154.254.162	No identificado	SSH (830)	SEGURO
190.145.207.66	No identificado	Bloqueo de Página	SEGURO
190.145.207.67	deming.unp.gov.co	Suite Visión Empresarial	SEGURO
190.145.207.68	No identificado	Palo Alto Networks	SEGURO
190.145.207.69	yunalesca.unp.gov.co	DNS Server	VULNERABLE
190.145.207.70	hannah.unp.gov.co	DNS Server	VULNERABLE
190.145.207.71	intranet.unp.gov	Intranet UNP	SEGURO
190.145.207.72	philomena.unp.gov.co	Microsoft Exchange	VULNERABLE
190.145.207.73	paco.unp.gov.co	Plataforma de Aprendizaje PACO	VULNERABLE
190.145.207.74	dialin.unp.gov.co lyncdiscover.unp.gov.co meet.unp.gov.co oliver.unp.gov.co PRTG	Plataforma PRTG	SEGURO
190.145.207.78	prueba.unp.gov.co ser.unp.gov.co	Pruebas SER	SEGURO
190.145.207.79	pqrs.unp.gov.co solicitudes.unp.gov.co	PQRS Solicitudes	SEGURO
190.145.207.80	www.unp.gov.co	Dominio Principal	VULNERABLE
190.145.207.81	socrates.unp.gov.co	Pagina Bloqueada	SEGURO
190.145.207.82	izanami.unp.gov.co	Pagina Bloqueada	SEGURO
190.145.207.84	mesadeservicios.unp.gov.co	Mesa de Servicios	SEGURO
190.145.207.85	Pagina Bloqueada	Pagina Bloqueada	SEGURO
190.145.207.86	Pagina Bloqueada	Pagina Bloqueada	SEGURO
190.145.207.88	No identificado	GEDOC - Servicios Digitales	VULNERABLE
190.145.207.89	csa.unp.gov.co	Ivanti®Cloud Services Appliance	SEGURO
190.145.207.90	No identificado	GEDOC - Servicios Digitales	VULNERABLE
190.145.207.91	No identificado	SGDEA - DOCUFILE	VULNERABLE
190.145.207.92	unpradio.unp.gov.co	Radio UNP	VULNERABLE

Metodología Utilizada

El análisis se llevó a cabo utilizando una combinación de técnicas **automatizadas** y **manuales** que permitieron una evaluación exhaustiva de las vulnerabilidades presentes en los sistemas. Este enfoque híbrido permitió detectar tanto vulnerabilidades conocidas mediante escaneos automatizados como configuraciones inseguras más complejas a través de validaciones manuales. La metodología aplicada incluyó las siguientes fases:

1. **Escaneo Automatizado de Vulnerabilidades**:
    - Se utilizaron herramientas avanzadas de escaneo para identificar vulnerabilidades en las aplicaciones web y direcciones IP. Los escáneres automatizados permitieron la detección rápida de vulnerabilidades conocidas, incluyendo puertos abiertos, configuraciones por defecto, y versiones de software desactualizadas.
    - Se realizó un análisis de la seguridad de los servicios expuestos, identificando posibles vectores de ataque, como protocolos inseguros o falta de cifrado adecuado en las comunicaciones.
2. **Validación Manual de Resultados**:
    - Tras los escaneos iniciales, un equipo de analistas de seguridad llevó a cabo **validaciones manuales** para confirmar los resultados obtenidos y verificar posibles falsos positivos. Durante esta fase se evaluaron aspectos más profundos de las aplicaciones y sistemas, como configuraciones inseguras, credenciales predeterminadas y errores en la implementación de controles de seguridad.
    - La validación manual es crucial para detectar fallas más complejas que las herramientas automatizadas no siempre identifican, como fallos en la lógica de autenticación o en la gestión de sesiones.

Importancia de la Metodología Híbrida

La metodología utilizada es esencial para obtener una visión completa y precisa del estado de seguridad de la organización. El uso de escaneos automatizados asegura que el análisis cubra una amplia gama de vulnerabilidades de forma eficiente, mientras que la validación manual garantiza que los resultados sean fiables y relevantes, permitiendo así priorizar las acciones correctivas según su nivel de criticidad.

Análisis de Vulnerabilidades

7 Críticas | **18 Altas** | **24 Medias** | **0 Bajas**

Descripción Técnica de las Vulnerabilidades

HOST	20.62.113.96/solr		
ID-001	Panel de Administración Expuesto en Apache Solr	**CVSS v3: 9.8**	**CRÍTICA**

El **panel de administración de Apache Solr**, accesible a través de la ruta /solr/, expone funciones críticas que podrían ser aprovechadas por un atacante si no se protegen adecuadamente. En las versiones vulnerables, como la 6.6.5, este panel permite a los usuarios realizar acciones administrativas sin la autenticación adecuada, lo que compromete seriamente la seguridad de la plataforma.

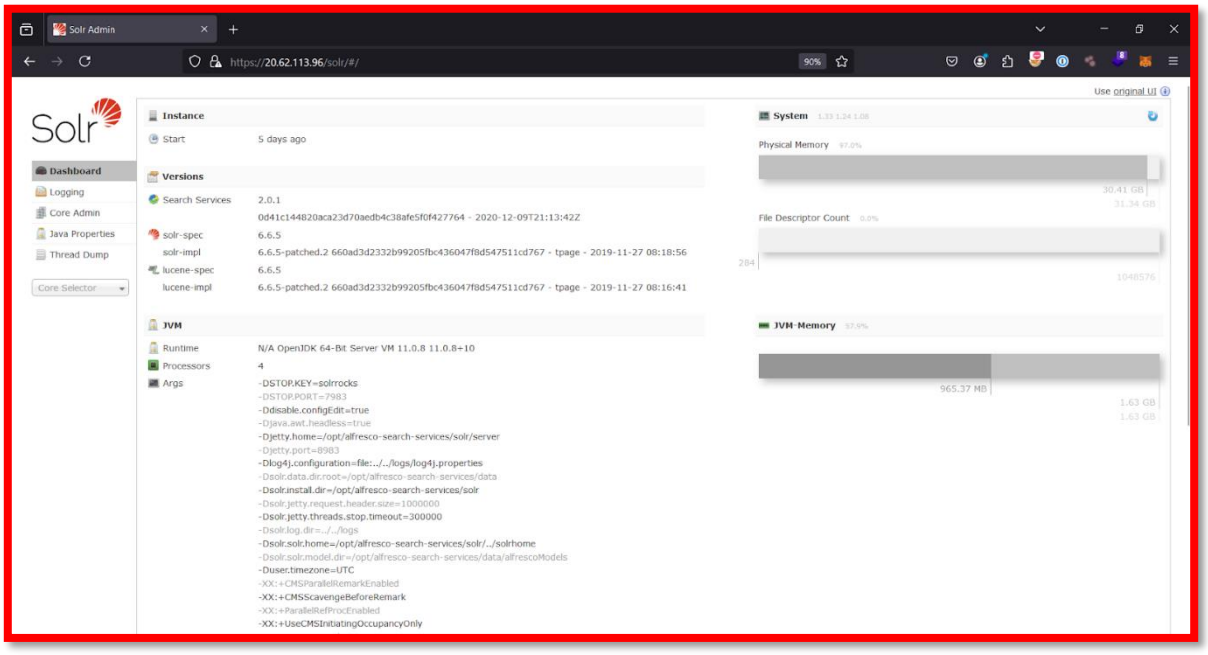


Imagen 1. Panel de Administración de Apache Solr

Es posible visualizar las versiones de los servicios en ejecución dentro de la plataforma. Al realizar una validación de vulnerabilidades asociadas se identifica un CVE crítico en la versión de Solr 6.6.5 que se detalla en la siguiente sección.

Imagen 2. Identificación de versiones vulnerables

Los núcleos en Solr son índices independientes que se pueden añadir y gestionar desde el panel de administración. Un atacante con acceso no autorizado al panel puede **añadir más núcleos** al sistema, lo que les permitiría **indexar información adicional**, y posiblemente, inyectar contenido malicioso para futuras consultas.

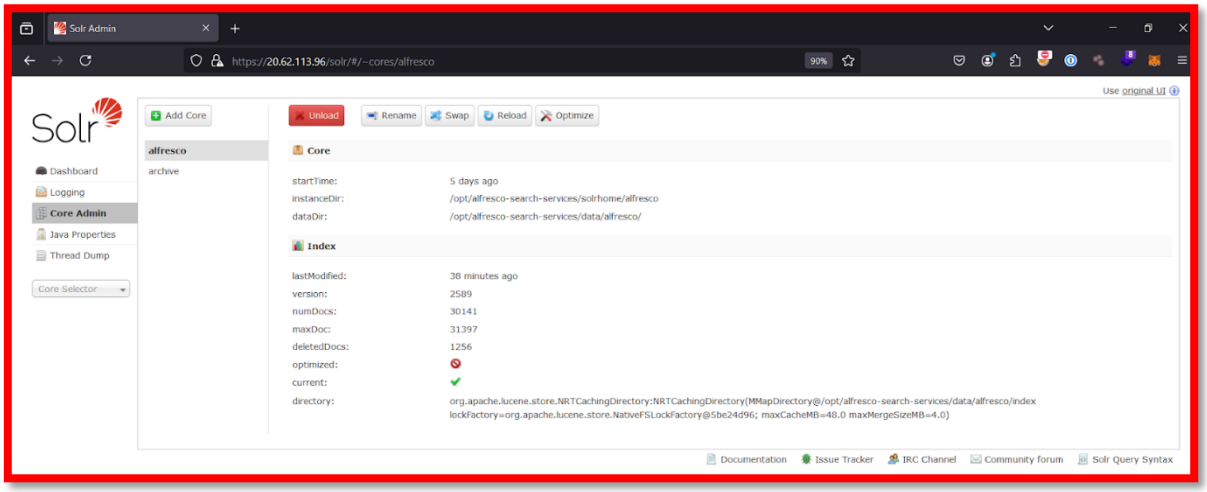


Imagen 3. Sección para añadir o modificar los núcleos en la plataforma

Solicitud para obtener información sobre los núcleos creados

El atacante puede realizar solicitudes al sistema para obtener una lista de los núcleos creados, lo que expone **rutas internas** del servidor, como ubicaciones de almacenamiento de los índices. Esto puede ofrecer información sobre la arquitectura interna del sistema, permitiendo al atacante mapear mejor la infraestructura para futuros ataques.

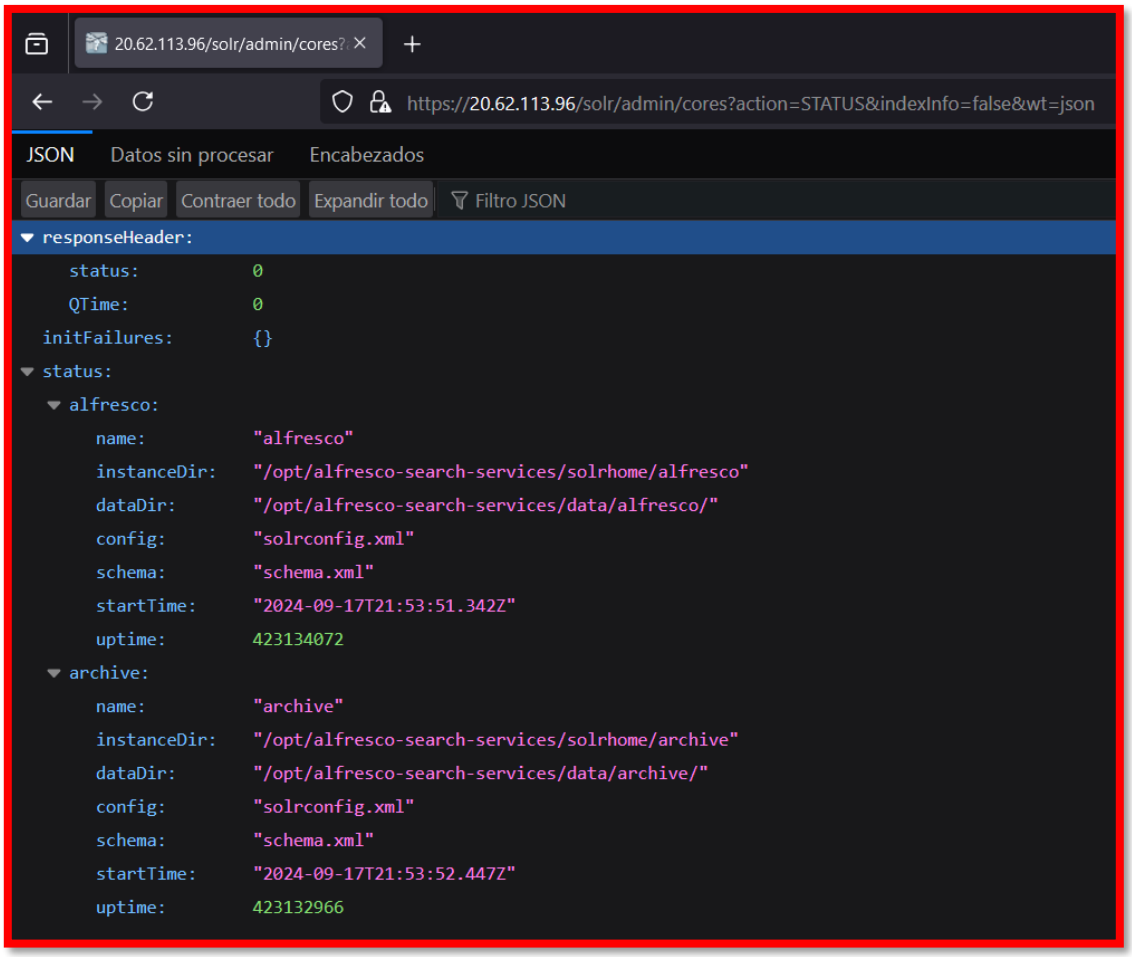


Imagen 4. Solicitud exitosa en el Endpoint para obtener información de los núcleos

Al acceder al panel de administración, un atacante puede realizar **modificaciones críticas** en el sistema con permisos de administrador, como cambiar configuraciones en los archivos solrconfig.xml o schema.xml, que controlan el comportamiento del sistema de indexación. Estas modificaciones podrían ser utilizadas para deshabilitar medidas de seguridad, redirigir datos sensibles, o incluso permitir la ejecución de código malicioso en el sistema.

HOST	20.62.113.96/solr		
ID-002	RCE en Apache Solr (CVE-2019-0192)	**CVSS v3: 9.8**	**CRÍTICA**

La vulnerabilidad **CVE-2019-0192** afecta a las versiones de **Apache Solr** desde la 5.0.0 hasta la 6.6.5. Se trata de una vulnerabilidad crítica relacionada con la deserialización insegura de datos no confiables, que permite la **ejecución remota de código (RCE)**.

Detalles Técnicos

El vector de ataque aprovecha la **Config API** de Solr, que permite configurar el servidor **JMX** a través de una solicitud HTTP POST. Un atacante remoto, sin autenticación, puede enviar una solicitud maliciosa apuntando al servidor a un **RMI (Remote Method Invocation)** malicioso. Al explotar la deserialización insegura de objetos Java, el atacante puede ejecutar código arbitrario en el sistema afectado.

La explotación de esta vulnerabilidad permite al atacante tomar control total del servidor Solr, comprometiendo la integridad y disponibilidad del sistema. Los sistemas afectados pueden ser utilizados para instalar malware, robar datos o interrumpir servicios críticos.

Mecanismo de Explotación

El atacante utiliza la **Config API** para redirigir la configuración del JMX hacia un servidor RMI malicioso, lo que desencadena la ejecución de código remoto al procesar objetos Java deserializados de forma insegura. Esto abre la posibilidad de manipular el servidor Solr y ejecutar cualquier comando en el sistema.

Versiones Afectadas

- **Apache Solr**: 5.0.0 a 5.5.5 y 6.0.0 a 6.6.5

Recomendaciones de Mitigación

1. **Actualizar Apache Solr**: Se recomienda actualizar a la versión **7.0 o superior**, que incluye un parche que corrige esta vulnerabilidad.
2. **Configurar controles de acceso**: Limitar el acceso al **Config API** de Solr a través de listas de control de acceso y reglas de firewall.
3. **Deshabilitar JMX** si no es necesario para reducir la superficie de ataque.
4. **Monitorear el tráfico de red** en busca de solicitudes POST maliciosas hacia el servidor JMX o intentos de explotación

Referencias

- https://www.clouddefense.ai/cve/2019/CVE-2019-0192

Mitigación y Recomendaciones

Apache Solr

1. **Actualizar a la versión 7.0 o superior**:
    - La vulnerabilidad que afecta las versiones anteriores a la 7.0 de Apache Solr debe ser corregida a través de una actualización a la versión **7.0 o superior**, la cual contiene los parches de seguridad necesarios para resolver problemas críticos como la **CVE-2019-0192**.
2. **Deshabilitar el acceso público al panel de administración**:
    - Limitar el acceso al panel de administración de Apache Solr a través de **controles de acceso basados en IP**, **VPNs**, o **firewalls**. Solo personal autorizado debe poder acceder al panel de administración para evitar que actores maliciosos aprovechen las funciones administrativas expuestas.

HOST	https://soadoc.unp.gov.co/share/		
ID-003	Credenciales por Defecto en la Plataforma Alfresco Community Edition	**CVSS v3: 9.8**	**CRÍTICA**

La vulnerabilidad en **Alfresco Community Edition** se debe a la utilización de **credenciales por defecto** en la **página de inicio de sesión**. Esto permite que un atacante acceda a la plataforma sin necesidad de comprometer credenciales legítimas, lo que da acceso a información interna de la organización. En este caso, se ha detectado acceso a carpetas que contienen **ficheros internos** y **documentos confidenciales de personas protegidas**.

Riesgo Principal de Exposición

Acceso a la plataforma de Alfresco con credenciales por defecto:

La vulnerabilidad surge cuando no se cambian las credenciales por defecto después de la instalación de Alfresco. Un atacante puede utilizar la contraseña admin/admin para acceder a la **página de inicio de sesión**, ganando privilegios administrativos.

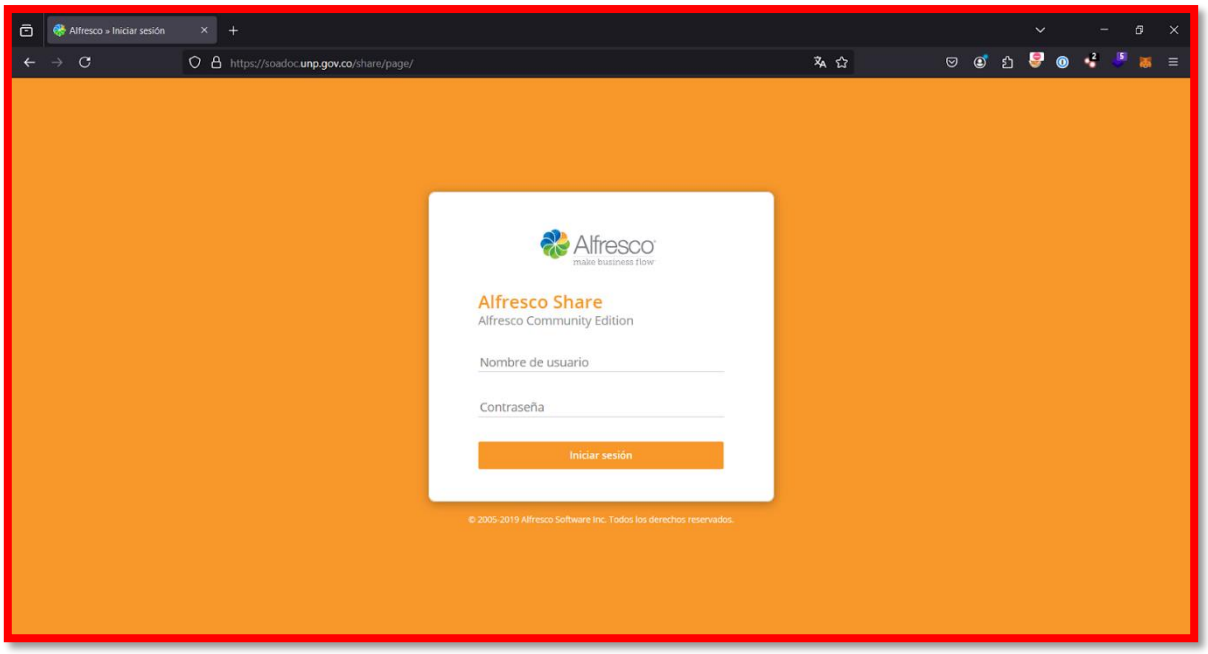


Imagen 5. Panel de Acceso con Credenciales por Defecto

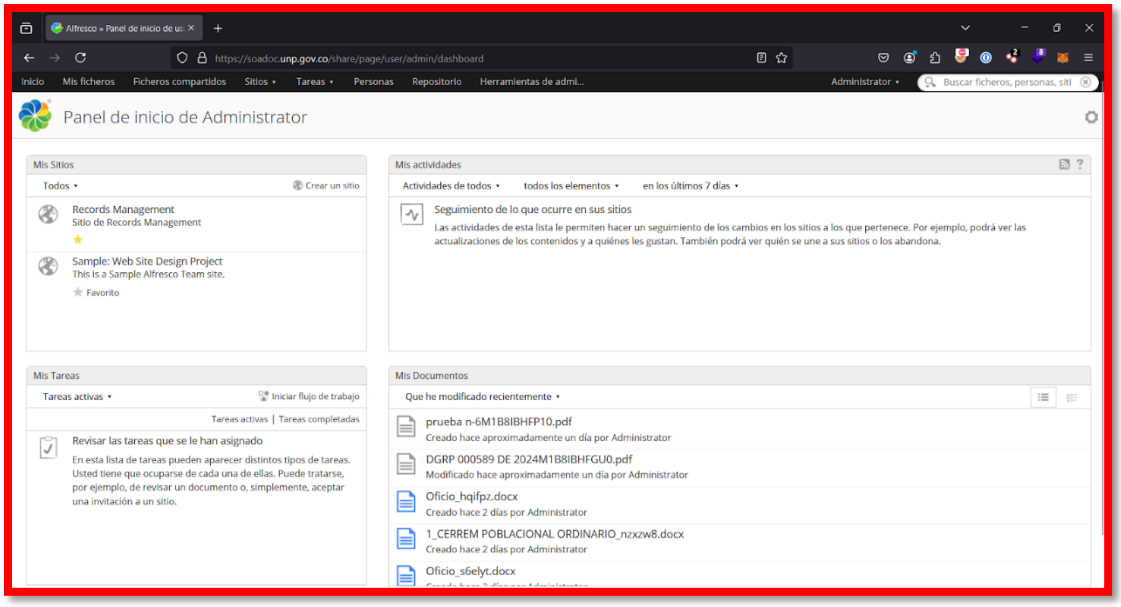


Imagen 6. Acceso como Administrador

Acceso a carpetas internas

Una vez dentro de la plataforma, el atacante puede navegar a través de las carpetas almacenadas en Alfresco, obteniendo **acceso a documentos confidenciales** relacionados con las operaciones internas de la organización, incluyendo ficheros sensibles sobre personas protegidas.

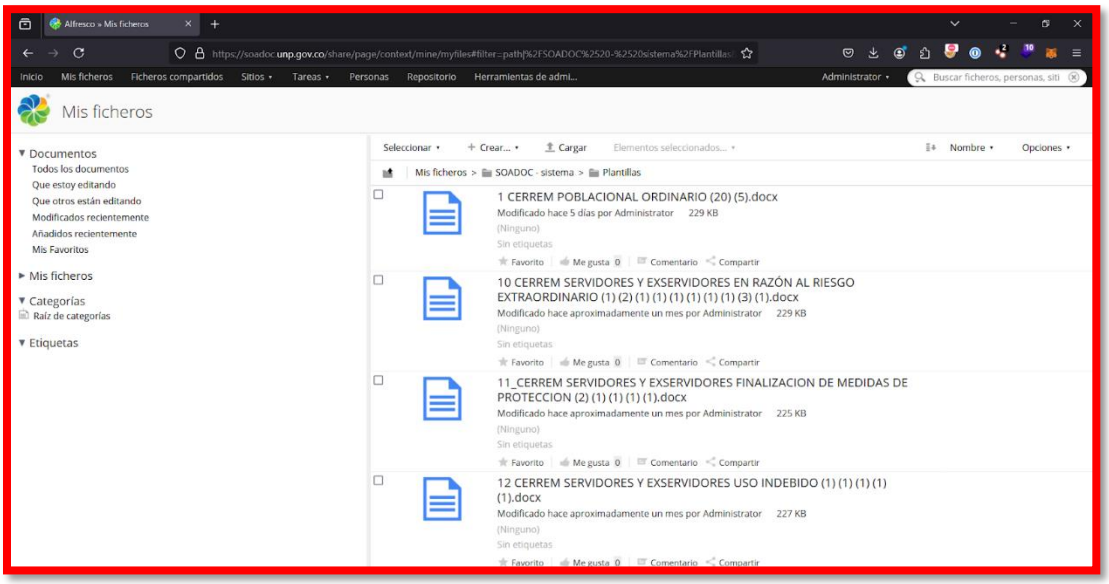
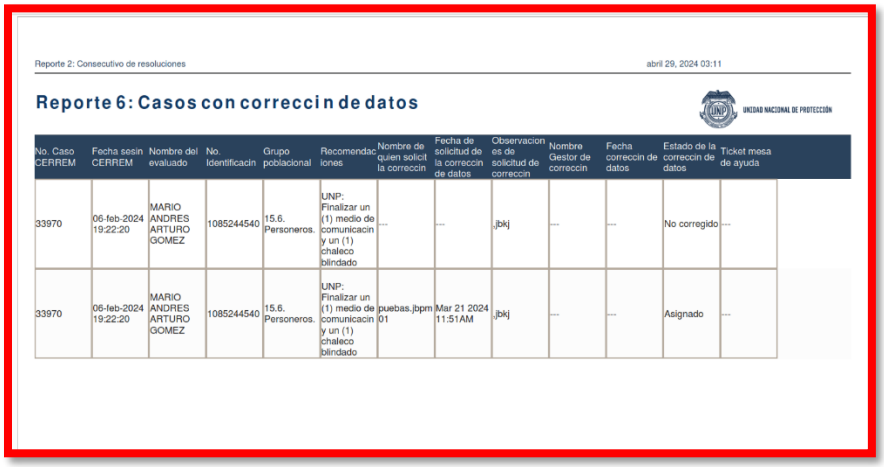


Imagen 7. Ficheros Disponibles

Acceso a documentos con información clasificada

La falta de medidas de seguridad adecuadas puede exponer **documentos confidenciales** relacionados con la seguridad de los individuos protegidos por la UNP. Esta información puede ser filtrada o utilizada para ataques más sofisticados.


Reporte 2: Consecutivo de resoluciones — abril 29, 2024 03:11

Reporte 6: Casos con correcci n de datos

No. Caso CERREM	Fecha sesin CERREM	Nombre del evaluado	No. Identificacin	Grupo poblacional	Recomendaciones	Nombre de quien solicit la correccin	Fecha de solicitud de la correccin de datos	Observaciones de solicitud de correccin	Nombre Gestor de correccin	Fecha correccin de datos	Estado de la correccin de datos	Ticket mesa de ayuda
33970	06-feb-2024 19:22:20	MARIO ANDRES ARTURO GOMEZ	1085244540	15.6. Personeros.	UNP: Finalizar un (1) medio de comunicacin y un (1) chaleco blindado	---	---	jbkj	---	---	No corregido	---
33970	06-feb-2024 19:22:20	MARIO ANDRES ARTURO GOMEZ	1085244540	15.6. Personeros.	UNP: Finalizar un (1) medio de comunicacin y un (1) chaleco blindado	puebas.jbpm 01	Mar 21 2024 11:51AM	jbkj	---	---	Asignado	---

Imagen 8. Fichero de Ejemplo con Información de Protegidos

En los archivos analizados dentro de la plataforma **Alfresco Community Edition**, aunque se identificaron varios **archivos de prueba** que no contenían información sensible, también se encontraron **carpetas estructuradas** por áreas de la organización. Dentro de estas carpetas, se localizaron varios **documentos** que contienen **información aparentemente confidencial**, incluidos datos relacionados con las actividades internas de la organización y sus operaciones. La exposición de estos documentos supone un riesgo significativo para la seguridad de la organización, dada la naturaleza sensible de los datos involucrados.

Alfresco

1. **Cambiar inmediatamente las credenciales por defecto y utilizar contraseñas seguras**:
    - Acceder al portal de **Alfresco Community Edition** y cambiar **todas las credenciales por defecto**, como admin/admin. Es fundamental que las nuevas contraseñas sean **robustas** y cumplan con políticas de seguridad, como longitud mínima, uso de caracteres especiales, y cambios regulares.
2. **Implementar autenticación multifactor (MFA)**:
    - Configurar **autenticación multifactor (MFA)** para todas las cuentas de usuarios con acceso administrativo o a documentos confidenciales en la plataforma. Esto añade una capa adicional de seguridad, haciendo más difícil que un atacante acceda incluso si compromete una contraseña.

HOST	190.145.207.72		
ID-004	Versión De Microsoft Exchange 2016 Obsoleta (CVE-2022-41080 y CVE-2022-41082)	**CVSS v3: 10.0**	**CRÍTICA**

Esta vulnerabilidad identifica que la instalación de **Microsoft Exchange Server 2016 (v15.1.2308)** en el host remoto ya no está soportada por el proveedor. La falta de soporte implica que no se recibirán nuevos parches de seguridad, lo que expone al servidor a vulnerabilidades conocidas y desconocidas que no serán corregidas por Microsoft. Los atacantes pueden aprovechar vulnerabilidades no parchadas para comprometer el servidor y, por lo tanto, la infraestructura de correo de la organización.

Las vulnerabilidades **CVE-2022-41080** y **CVE-2022-41082** son fallos críticos que afectan a **Microsoft Exchange Server**. Estas vulnerabilidades están relacionadas con la exposición de servidores a un ataque de **Server-Side Request Forgery (SSRF)** en **Outlook Web Access (OWA)** y permiten la ejecución remota de código (RCE) a través de PowerShell. Los atacantes pueden explotar estos errores al enviar solicitudes maliciosas, aprovechando una falta de validación adecuada en OWA, para obtener acceso a PowerShell y ejecutar comandos arbitrarios en el servidor.

1. **CVE-2022-41080**: Este fallo permite el ataque SSRF mediante OWA, que puede ser utilizado para invocar solicitudes HTTP internas no autorizadas.
2. **CVE-2022-41082**: Esta vulnerabilidad permite a los atacantes ejecutar código arbitrario en el servidor Exchange a través del acceso a PowerShell cuando **CVE-2022-41080** ha sido explotado previamente.

Referencias

- https://nsfocusglobal.com/exchange-server-owassrf-vulnerability-cve-2022-41080-cve-2022-41082-alert/
- https://www.rapid7.com/blog/post/2022/12/21/cve-2022-41080-cve-2022-41082-rapid7-observed-exploitation-of-owassrf-in-exchange-for-rce/
- https://unit42.paloaltonetworks.com/threat-brief-owassrf/
- https://campus.barracuda.com/product/webapplicationfirewall/doc/98211382/zero-day-microsoft-exchange-server-critical-vulnerabilities-owassrf-and-proxynotshell

Impacto

El impacto de estas vulnerabilidades es extremadamente crítico, ya que permite a un atacante remoto sin autenticación ejecutar código arbitrario en el servidor Exchange afectado. Esto puede llevar a:

- **Compromiso total del servidor**: Los atacantes pueden tomar control completo del servidor afectado, permitiendo la ejecución de Ransomware, la modificación de correos electrónicos y la explotación de servicios internos.
- **Fugas de datos**: Acceso a correos electrónicos y datos sensibles, con la posibilidad de realizar movimientos laterales dentro de la infraestructura.
- **Disrupción de servicios**: Los ataques pueden deshabilitar el acceso a correos electrónicos y otros servicios críticos proporcionados por el servidor Exchange.

Mitigaciones técnicas

Aplicar parches de seguridad:

Microsoft lanzó actualizaciones de seguridad críticas en noviembre de 2022 que corrigen ambas vulnerabilidades. Se recomienda aplicar el parche **KB5019758**. Para instalar el parche:

- Dirígete al Centro de actualizaciones de Microsoft y descarga el último parche de seguridad para tu versión de Exchange Server.
- Instala el parche ejecutando el archivo descargado como administrador y sigue los pasos del instalador.
- Una vez instalado el parche, reinicia el servidor para completar la aplicación de la actualización.

Nota: Ejecuta el script **Exchange Server Health Checker** para verificar que todos los parches y actualizaciones se hayan instalado correctamente y que no haya problemas pendientes

Referencias

- https://techcommunity.microsoft.com/t5/exchange-team-blog/released-november-2022-exchange-server-security-updates/ba-p/3669045

Bloquear el acceso a PowerShell remoto (Mitigación temporal)

Desactivar el acceso a PowerShell remoto para usuarios que no sean administradores es una medida temporal para mitigar el riesgo de explotación:

- Abre PowerShell en modo administrador.
- Elimina los permisos para usuarios no administradores.
- Desactiva el acceso remoto de PowerShell para usuarios no administradores ejecutando el siguiente comando:

```
Set-PSSessionConfiguration -Name Microsoft.PowerShell -ShowSecurityDescriptorUI -Force
```

Configurar reglas de reescritura de URL en IIS:

Si no puedes aplicar los parches de inmediato, puedes implementar mitigaciones temporales en el servidor **IIS**. Sigue estos pasos para bloquear solicitudes maliciosas:

- Abre el Administrador de IIS en el servidor.
- Navega hasta el sitio que ejecuta Exchange.
- En el menú del sitio, selecciona **Reescritura de URL (URL Rewrite)**.
- Agrega una nueva regla de bloqueo con los siguientes detalles:
  - Patrón: .*autodiscover\.json.*Powershell.*
  - Tipo de acción: **Abort Request** (Abortar la solicitud).
- Guarda la configuración y reinicia el servidor IIS para aplicar los cambios.

Monitoreo activo de PowerShell y actividad sospechosa:

Monitorear la actividad de PowerShell es crucial para identificar posibles intentos de explotación de estas vulnerabilidades. Sigue estos pasos para configurar el monitoreo:

- Utiliza el EDR para registrar eventos de PowerShell. Configura reglas específicas para alertar sobre actividades sospechosas en PowerShell, especialmente en procesos que involucren w3wp.exe o powershell.exe.
- Configura alertas automáticas que se activen cuando se ejecuten comandos críticos en el servidor sin autorización.

HOST	190.145.207.69, 190.145.207.70		
ID-005	(SIGRed) Ejecución Remota De Código (RCE) en Servidor DNS Microsoft (CVE-2020-1350)	**CVSS v3: 10.0**	**CRÍTICA**

La vulnerabilidad **CVE-2020-1350**, conocida como **SIGRed**, es una vulnerabilidad crítica de **Ejecución Remota de Código (RCE)** que afecta a los servidores **Microsoft DNS** desde **Windows Server 2003 hasta 2019**. La vulnerabilidad se encuentra en la manera en que los servidores DNS de Windows procesan solicitudes de DNS, específicamente en la gestión incorrecta de los registros **SIG** (Signature), que puede llevar a un desbordamiento de búfer en el heap.

Un atacante remoto puede enviar una respuesta DNS maliciosa que supere los **64 KB** y, de este modo, provocar un desbordamiento de búfer, lo que permitiría la ejecución de código arbitrario con privilegios del **Sistema Local**. Dado que la vulnerabilidad es "wormable", significa que puede propagarse sin intervención del usuario a otros sistemas vulnerables dentro de la red, lo que la convierte en una amenaza altamente crítica.

Impacto

- **Ejecución Remota de Código**: Un atacante exitoso puede ejecutar código arbitrario en el contexto de **SYSTEM**, lo que otorga control total sobre el servidor afectado.
- **Propagación en la red**: Como la vulnerabilidad es wormable, permite a los atacantes propagar malware o ransomware dentro de una red corporativa, explotando múltiples sistemas sin intervención del usuario.
- **Interrupción de servicios**: Los servidores DNS afectados pueden dejar de funcionar correctamente o ser utilizados para lanzar ataques más amplios, afectando servicios esenciales.

Mitigaciones técnicas

Aplicar el parche de seguridad: Microsoft lanzó una actualización de seguridad en julio de 2020 para corregir esta vulnerabilidad. Es **imperativo** que los administradores apliquen el parche lo antes posible.

Instrucciones para aplicar el parche:

- Descargue e instale el parche **KB4569509** correspondiente a la versión de Windows Server desde Microsoft.
- Una vez descargado, ejecute el instalador como **Administrador**.
- Reinicie el servidor para asegurar que las actualizaciones se apliquen correctamente.

Nota: Los servidores con actualizaciones automáticas activadas ya deberían haber recibido esta actualización

Implementar la solución temporal basada en el Registro

En situaciones donde no es posible aplicar el parche de inmediato, Microsoft ha proporcionado una solución temporal para reducir el riesgo de explotación. Esta mitigación limita el tamaño de los paquetes de respuesta DNS TCP, lo que ayuda a mitigar la explotación del desbordamiento de búfer.

- **Instrucciones para aplicar la solución temporal**:
    - Abra el **Editor de Registro** como administrador.
    - Navegue a la clave del registro:

HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\DNS\Parameters

- Agregue o modifique el valor **DWORD** llamado ***TcpReceivePacketSize*** y configure el valor en **0xFF00** (en hexadecimal), que es 65280 en decimal.
- Guarde los cambios y reinicie el servicio DNS con los siguientes comandos en el símbolo del sistema elevado (net stop dns && net start dns)

Importante: Este cambio puede afectar las respuestas DNS que excedan los 65,280 bytes, por lo que algunas consultas DNS podrían no ser resueltas. Para más información consultar la guía oficial de Microsoft en la referencia.

Referencias

- https://support.microsoft.com/es-es/topic/kb4569509-orientaci%C3%B3n-para-la-vulnerabilidad-cve-2020-1350-del-servidor-dns-6bdf3ae7-1961-2d25-7244-cce61b056569

Monitoreo y validación de la solución temporal: Después de aplicar la solución temporal, es recomendable habilitar el registro de diagnóstico para identificar posibles efectos secundarios. Debe capturar un conjunto de muestras de tráfico DNS y revisar si el límite de tamaño está afectando negativamente a su entorno.

- Utilice herramientas como **Wireshark** o **DNS Logging** para supervisar las respuestas DNS y verificar si alguna respuesta supera el nuevo límite establecido

Eliminar la solución temporal después del parche: Una vez que el parche de seguridad haya sido aplicado, elimine el cambio en el registro para evitar problemas futuros con el tamaño de los paquetes DNS.

- Navegue nuevamente a la clave del registro y elimine el valor TcpReceivePacketSize o restablezca su valor a **0xFFFF**.
- Reinicie el servicio DNS para que los cambios surtan efecto.

Existen exploits públicos disponibles

Existen **exploits públicos** para la vulnerabilidad **CVE-2020-1350**, lo que aumenta significativamente el riesgo de explotación en entornos no parcheados. Se han publicado pruebas de concepto (PoCs) que muestran cómo los atacantes pueden explotar esta vulnerabilidad para obtener ejecución remota de código. Entre los exploits más conocidos se encuentran:

- Exploit en GitHub por chompie1337
- Exploit en GitHub por tinkersec

La disponibilidad de estos exploits significa que **cualquier atacante con habilidades moderadas** puede aprovechar esta vulnerabilidad si no se ha aplicado el parche. Esto aumenta la urgencia de aplicar las medidas de mitigación para proteger los servidores afectados .

Referencias

- https://datafarm-cybersecurity.medium.com/exploiting-sigred-cve-2020-1350-on-windows-server-2012-2016-2019-80dd88594228

HOST	20.62.113.96/solr		
ID-006	Ejecución Remota De Código (RCE) a Través De La API De ConfigSet (CVE-2020-13957)	**CVSS v3: 9.4**	**CRÍTICA**

La vulnerabilidad **CVE-2020-13957** afecta a **Apache Solr** en las versiones 6.6.0 a 6.6.6, 7.0.0 a 7.7.3 y 8.0.0 a 8.6.2. La vulnerabilidad permite que un atacante ejecute código de manera remota mediante la API **ConfigSet**. Esto sucede porque los mecanismos de autenticación y autorización que protegen la **API de ConfigSet** pueden ser eludidos al combinar acciones de **UPLOAD** y **CREATE**.

El problema radica en que cuando se sube un ConfigSet sin autenticación, el sistema lo trata como **no confiable**. Sin embargo, al crear un nuevo ConfigSet a partir de uno subido previamente mediante la operación **CREATE**, este nuevo conjunto de configuración es tratado incorrectamente como confiable, lo que permite la ejecución de funciones peligrosas que podrían ser utilizadas para ejecutar código malicioso en el servidor Solr

Referencias

- https://nsfocusglobal.com/apache-solr-configset-api-upload-function-vulnerability-cve-2020-13957-threat-alert/
- https://www.tenable.com/cve/CVE-2020-13957
- https://nvd.nist.gov/vuln/detail/CVE-2020-13957

Impacto:

- **Ejecución Remota de Código**: Un atacante que explote esta vulnerabilidad puede obtener control total del servidor Apache Solr, ejecutando comandos arbitrarios y comprometiendo la seguridad del sistema.
- **Falta de autenticación adecuada**: Al no haber un control de autenticación y autorización correcto, los atacantes pueden manipular los archivos de configuración, lo que les permite modificar o crear nuevos ConfigSets que incluyen parámetros inseguros.
- **Riesgo para la infraestructura de búsqueda**: Dado que Apache Solr es utilizado en grandes infraestructuras de búsqueda, una explotación exitosa podría comprometer la integridad de los datos y servicios dependientes del motor de búsqueda.

Mitigaciones técnicas

Actualizar Apache Solr a una versión segura: La actualización a **Apache Solr 8.6.3** o posterior corrige esta vulnerabilidad. Los administradores deben descargar la versión segura desde el sitio oficial de Solr.

Instrucciones de actualización:

- Descargue la última versión de Apache Solr.
- Realice una copia de seguridad de la configuración y datos actuales.
- Siga los pasos indicados en la guía de actualización de Apache Solr.
- Reinicie el servidor y valide que la versión instalada es la correcta.

Deshabilitar la función de upload en ConfigSet API: Si no es posible aplicar la actualización de inmediato, puede mitigar la vulnerabilidad deshabilitando la funcionalidad de **UPLOAD** en la API de ConfigSet.

Instrucciones para deshabilitar la funcionalidad:

- Edite el archivo de configuración de Apache Solr y agregue la opción de tiempo de ejecución *-Dconfigset.upload.enabled=false*.
- Guarde los cambios y reinicie el servicio Solr para aplicar la nueva configuración.

Habilitar autenticación y autorización: Implementar mecanismos de autenticación y autorización robustos es fundamental para proteger la API ConfigSet y otros puntos de entrada críticos. Apache Solr ofrece múltiples opciones de autenticación y autorización que deben habilitarse para controlar el acceso a la API y evitar que usuarios no autorizados realicen cambios peligrosos.

Instrucciones para habilitar la autenticación:

- Configure un **plugin de autenticación** siguiendo las guías oficiales de Apache Solr.
- Aplique reglas de autorización estrictas para restringir qué usuarios tienen acceso a modificar ConfigSets.

Aplicar reglas de firewall: Limitar el acceso a la API de Solr, incluidas las interfaces administrativas, únicamente a direcciones IP confiables. Esto reducirá la exposición del sistema a posibles atacantes externos que intenten explotar esta vulnerabilidad.

Instrucciones para configurar reglas de firewall:

- Utilice su firewall existente para restringir el acceso al puerto 8983 (puerto predeterminado de Solr) únicamente a IPs internas o de confianza.
- Asegúrese de que cualquier intento de acceso no autorizado sea bloqueado y monitoreado a través de registros de red.

Exploits públicos disponibles

Existen exploits públicos que demuestran cómo un atacante puede aprovechar la vulnerabilidad **CVE-2020-13957** para obtener acceso no autorizado y ejecutar código arbitrario en el servidor Solr afectado. Se recomienda aplicar las mitigaciones lo antes posible para evitar la explotación de estos ataques.

- Prueba de concepto (PoC) en GitHub

HOST	20.62.113.96/solr		
ID-007	Ejecución Remota De Código a Través De VelocityResponseWriter (CVE-2019-17558)	**CVSS v3: 7.5**	**ALTA**

CVE-2019-17558 afecta a **Apache Solr** en versiones desde la **5.0.0 hasta la 8.3.1**. La vulnerabilidad reside en el uso de la función **VelocityResponseWriter**, un componente del motor de plantillas Velocity, que permite a los usuarios generar respuestas personalizadas en Solr. Los atacantes pueden aprovechar esta vulnerabilidad enviando solicitudes HTTP maliciosas que activan el cargador de recursos de Velocity, permitiendo la ejecución de plantillas arbitrarias y, por lo tanto, la **ejecución remota de código (RCE)**.

El atacante puede explotar esta vulnerabilidad manipulando la configuración de Solr a través de la API de configuración y activando el parámetro **params.resource.loader.enabled** en **solrconfig.xml**, lo que les permite ejecutar plantillas de Velocity con código malicioso.

Impacto:

- **Ejecución Remota de Código (RCE)**: Un atacante que explote esta vulnerabilidad puede ejecutar código arbitrario en el servidor afectado, comprometiendo el sistema y obteniendo acceso completo a la infraestructura.
- **Fuga de datos y compromiso total**: Los atacantes pueden ejecutar comandos que comprometan la integridad de los datos en los sistemas basados en Solr.
- **Ataques a gran escala**: Dado que Solr se utiliza en entornos empresariales críticos, la explotación de esta vulnerabilidad podría llevar a la caída de servicios de búsqueda esenciales o a su uso como vector para ataques adicionales.

Mitigaciones técnicas

Actualizar a la versión segura de Apache Solr: La vulnerabilidad **CVE-2019-17558** se corrigió en **Apache Solr 8.3.1** y se reforzó en **Solr 8.4**, donde se eliminó por completo el cargador de recursos de Velocity. La recomendación principal es actualizar a esta versión o una posterior.

Instrucciones de actualización:

- Descargue la versión 8.4 o superior desde el sitio oficial de Apache Solr.
- Realice una copia de seguridad de su configuración y datos antes de proceder con la actualización.
- Siga las instrucciones de actualización proporcionadas por Apache Solr.
- Reinicie el servidor después de completar la actualización para asegurarse de que los cambios surtan efecto.

Deshabilitar la funcionalidad de VelocityResponseWriter: Si no puede actualizar de inmediato, se recomienda deshabilitar el cargador de recursos de Velocity para mitigar el riesgo de explotación.

Instrucciones para deshabilitar Velocity:

- Acceda a **solrconfig.xml** y asegúrese de que el parámetro params.resource.loader.enabled esté configurado en false.
- Reinicie Apache Solr para que los cambios surtan efecto.

Implementar controles de acceso fuertes en la API: Limitar el acceso a la **API de configuración** de Apache Solr solo a usuarios autenticados y autorizados es crucial para evitar manipulaciones maliciosas.

- Configure la autenticación y autorización siguiendo la guía oficial.
- Use firewalls o controles de acceso basados en IP para restringir el acceso al puerto de administración de Solr (generalmente 8983).
- Utilice herramientas de monitoreo para revisar el tráfico hacia la API de Solr y verifique si se están enviando solicitudes inusuales a **solrconfig.xml**.

Exploits públicos disponibles:

Existen exploits públicos para **CVE-2019-17558** que demuestran cómo aprovechar esta vulnerabilidad para obtener ejecución remota de código. Se han publicado pruebas de concepto (PoC) y herramientas que permiten a atacantes con habilidades moderadas comprometer servidores Solr vulnerables.

- Exploit de prueba de concepto en GitHub
- Exploit en Rapid7

HOST	20.62.113.96/solr		
ID-008	Divulgación de Información en Apache Solr (CVE-2020-13941)	**CVSS v3: 8.8**	**ALTA**

CVE-2020-13941 afecta a **Apache Solr** en versiones anteriores a la **8.6.0**. La vulnerabilidad se encuentra en el **Replication Handler**, que permite ejecutar comandos como **backup**, **restore**, y **deleteBackup** sin la validación adecuada del parámetro de ubicación. Esto posibilita que un atacante no autenticado pueda leer o escribir en cualquier ubicación accesible por el usuario **solr**, lo que podría resultar en la divulgación de información sensible o en la manipulación de datos almacenados.

El problema se agrava porque los atacantes pueden utilizar rutas SMB maliciosas, lo que podría conducir a la exfiltración de datos sensibles como los **hashes NTLM** o permitir ataques de tipo **SMB relay**.

Impacto:

- **Divulgación de Información**: Un atacante puede obtener acceso no autorizado a archivos sensibles almacenados en el servidor Solr, lo que podría incluir datos críticos del sistema.
- **Manipulación de datos**: La falta de validación permite a los atacantes escribir en ubicaciones del sistema, comprometiendo la integridad de los datos.
- **Posibilidad de ataques SMB relay**: En sistemas configurados incorrectamente, los atacantes pueden utilizar esta vulnerabilidad para realizar ataques más avanzados de **relé SMB**, que podrían llevar a la **impersonación de usuarios** en los recursos de red.

Mitigaciones técnicas

Actualizar a Apache Solr 8.6.0 o posterior: El parche para corregir esta vulnerabilidad está disponible en la versión **8.6.0** de Apache Solr. Es fundamental actualizar el software para cerrar esta brecha de seguridad.

Restringir el acceso al Replication Handler: Limite el acceso al **Replication Handler** para que solo los usuarios autenticados y autorizados puedan ejecutar comandos críticos como backup y restore.

Instrucciones para restringir el acceso:

- Modifique el archivo de configuración de Solr (solrconfig.xml) para establecer reglas de control de acceso en las rutas sensibles del Replication Handler.
- Aplique autenticación básica o basada en tokens para estos endpoints.

Referencias

- https://security.snyk.io/vuln/SNYK-JAVA-ORGAPACHESOLR-598793
- https://nvd.nist.gov/vuln/detail/CVE-2020-13941

HOST	190.145.207.88/sys/es/neoclassic/login/login 190.145.207.90/sys/es/neoclassic/login/login 190.145.207.91/login		
ID-009	Apache HTTP Server 2.4.37 Desactualizado	**CVSS v3: 9.8**	**ALTA**

La versión **Apache HTTP Server 2.4.37** presenta múltiples vulnerabilidades críticas que permiten la ejecución remota de código (RCE), divulgación de información, desbordamiento de búfer, y falsificación de solicitudes del lado del servidor (SSRF). Estas vulnerabilidades exponen a los servidores a ataques que comprometen su integridad y disponibilidad, especialmente en entornos donde el servidor gestiona tráfico web crítico.

Entre las vulnerabilidades más destacadas corregidas en versiones posteriores se encuentran:

1. **CVE-2019-0211**: Vulnerabilidad de **escalación de privilegios** que permite que procesos con privilegios bajos ejecuten código con privilegios de root.
2. **CVE-2019-0217**: Bypass de control de acceso en **mod_auth_digest**, lo que permite a un atacante autenticar incorrectamente bajo otra identidad.
3. **CVE-2021-44790**: Desbordamiento de búfer en **mod_lua**, que permite la ejecución de código arbitrario mediante peticiones maliciosas.

Impacto

- **Ejecución remota de código (RCE)**: Las vulnerabilidades permiten que un atacante ejecute código arbitrario en el servidor, lo que podría llevar al control total de la máquina.
- **Escalación de privilegios**: Un atacante con acceso limitado podría aumentar sus privilegios y ejecutar comandos con permisos de administrador.
- **Fuga de información**: Los atacantes podrían acceder a información sensible almacenada en el servidor.
- **Denegación de servicio (DoS)**: Las vulnerabilidades pueden ser explotadas para interrumpir la disponibilidad del servidor, afectando el rendimiento de servicios críticos.

Mitigaciones técnicas

Actualizar a la versión 2.4.62 o posterior: Las versiones más recientes de Apache HTTP Server corrigen las vulnerabilidades mencionadas. Se recomienda actualizar a la versión 2.4.62, que incluye parches de seguridad críticos.

Configurar control de acceso a módulos: Restringir el acceso a módulos críticos, como **mod_lua**, y deshabilitar funciones innecesarias puede reducir la superficie de ataque.

HOST	unpradio.unp.gov.co		
ID-010	Apache HTTP Server - Múltiples Vulnerabilidades Críticas (CVE-2021-44790 - CVE-2021-44224)	**CVSS v3: 9.0**	**ALTA**

La versión Apache HTTP Server 2.4.52 es vulnerable a múltiples fallos de seguridad que afectan tanto a la integridad como a la confidencialidad del servidor. Entre los problemas más graves, se encuentra un desbordamiento de búfer en el módulo mod_lua, que podría permitir la ejecución de código arbitrario en el servidor afectado. Además, una vulnerabilidad en mod_proxy permite la explotación de una falsificación de solicitudes del servidor (SSRF), lo que podría permitir que los atacantes redirijan peticiones a servidores arbitrarios, causando una fuga de información o comprometiendo servicios internos.

CVE-2021-44790 (Desbordamiento de búfer en mod_lua): Esta vulnerabilidad, calificada como **crítica** con un puntaje CVSS de **9.8**, se encuentra en el módulo **mod_lua** de Apache HTTP Server. Un atacante remoto puede enviar un cuerpo de solicitud HTTP especialmente diseñado que provoca un **desbordamiento de búfer** en el parser de contenido multipart. Esto podría permitir la ejecución de código arbitrario, lo que lleva a un control total del servidor afectado si no se aplican medidas correctivas.

CVE-2021-44224 (Falsificación de Solicitudes del Servidor - SSRF): Esta vulnerabilidad, calificada como **alta** con un puntaje CVSS de **8.2**, afecta a la configuración de **mod_proxy**. Un atacante podría enviar una URI maliciosa a un servidor configurado como proxy (con ProxyRequests on), lo que podría causar una **falsificación de solicitudes del servidor (SSRF)** o una **desreferencia de puntero nulo**, lo que puede llevar a un **ataque de denegación de servicio (DoS)** o permitir la redirección de solicitudes hacia sockets de Unix previamente definidos.

Referencias

https://www.cybersecurity-help.cz/vdb/apache_foundation/apache_http_server/2.4.52/

Impacto

- **CVE-2021-44790**: Un atacante exitoso puede ejecutar código arbitrario en el servidor, comprometiendo completamente su integridad y exponiendo datos sensibles. El desbordamiento de búfer puede explotarse para obtener acceso privilegiado y ejecutar acciones maliciosas.
- **CVE-2021-44224**: Esta vulnerabilidad permite a los atacantes enviar peticiones maliciosas que pueden redirigir tráfico hacia servidores arbitrarios, provocando filtraciones de información o interrumpiendo servicios críticos, comprometiendo tanto la confidencialidad como la disponibilidad del sistema.

Mitigaciones técnicas

Actualizar Apache HTTP Server a la versión 2.4.52 o posterior: La solución definitiva para ambas vulnerabilidades es actualizar Apache HTTP Server a la versión **2.4.52** o superior.

Deshabilitar el módulo mod_lua (si no es necesario): Para entornos que no utilizan **mod_lua**, desactivar este módulo puede mitigar el riesgo de explotación del desbordamiento de búfer.

Configurar adecuadamente mod_proxy: Si utiliza **mod_proxy**, asegúrese de configurar correctamente las directivas de proxy para evitar la explotación de SSRF.

HOST	unpradio.unp.gov.co		
ID-011	Contributor+ Path Traversal en El Bloque Template-Part (CVE-2024-36232)	**CVSS v3: 7.2**	**ALTA**

La vulnerabilidad **CVE-2021-44790** afecta a **Apache HTTP Server** en las versiones desde **2.4.0 hasta 2.4.51**. Se trata de un **desbordamiento de búfer** en el módulo **mod_lua**, el cual se activa cuando el servidor procesa contenido multipart mediante el método r:parsebody() en los scripts Lua. Un atacante podría enviar una solicitud cuidadosamente manipulada, lo que causaría el desbordamiento de búfer y podría derivar en la **ejecución remota de código (RCE)** en el servidor afectado. Aunque no se ha informado de exploits activos en el momento de su publicación, el riesgo de explotación sigue presente debido a la criticidad de la vulnerabilidad.

Además, la vulnerabilidad **CVE-2021-44224** afecta al módulo **mod_proxy** cuando el servidor está configurado como proxy de reenvío. Esta falla permite una **falsificación de solicitudes del lado del servidor (SSRF)**, lo que podría causar un compromiso de los servicios internos, incluida la redirección de solicitudes a otros servidores arbitrarios o la denegación de servicio.

Impacto

- **Ejecución remota de código (RCE)**: A través de la explotación del desbordamiento de búfer en **mod_lua**, un atacante podría ejecutar código arbitrario en el servidor afectado.
- **SSRF**: Con **mod_proxy**, un atacante puede redirigir peticiones a servidores arbitrarios, lo que potencialmente expone información interna o compromete servicios esenciales.
- **Fuga de información**: Los atacantes podrían manipular los datos o exponer información confidencial, poniendo en riesgo la integridad del sistema.

Mitigaciones técnicas

Actualizar a Apache HTTP Server 2.4.52 o posterior: La solución más efectiva es actualizar a la versión **2.4.52**, que corrige tanto el desbordamiento de búfer en **mod_lua** como la vulnerabilidad SSRF en **mod_proxy**.

HOST	unp.gov.co		
ID-012	Inclusión De Archivos Locales (LFI) Autenticado a Través Del Parámetro 'Layout_name' (CVE-2024-5709)	**CVSS v3: 8.8**	**ALTA**

CVE-2024-5709 afecta a **WPBakery Visual Composer** en versiones hasta la **7.7**. Esta vulnerabilidad permite a usuarios autenticados con permisos de "Autor" explotar el parámetro layout_name para realizar un ataque de **Inclusión de Archivos Locales (LFI)**. Este ataque se basa en la falta de validación del parámetro, lo que permite la inclusión de archivos arbitrarios del servidor.

Un atacante puede aprovechar esta vulnerabilidad para incluir archivos del sistema que contengan código ejecutable o información sensible. Si el archivo incluido es un script PHP, el atacante podría ejecutar código en el servidor, comprometiendo completamente el sistema.

Impacto:

- **Ejecución de código arbitrario**: El atacante puede incluir y ejecutar archivos PHP locales, lo que le otorga control sobre el servidor y le permite ejecutar comandos arbitrarios.
- **Exposición de información sensible**: El atacante podría acceder a archivos de configuración, credenciales y otros datos sensibles almacenados en el servidor.
- **Escalamiento de privilegios**: Aunque la vulnerabilidad requiere que el atacante esté autenticado como Autor, la explotación exitosa podría otorgar acceso a recursos de mayor privilegio.

Mitigaciones técnicas

Actualizar a la versión 7.8 o superior: La vulnerabilidad ha sido corregida en la versión **7.8** de WPBakery Visual Composer. La solución más recomendada es actualizar el plugin a esta versión o una más reciente.

HOST	unp.gov.co		
ID-013	Ejecución Remota de Código (RCE) a través de deserialización insegura (CVE-2023-6528)	**CVSS v3: 8.8**	**ALTA**

El plugin **Slider Revolution** para WordPress, en versiones anteriores a la 6.6.19, permite a los usuarios con el rol de **Autor** deserializar contenido arbitrario al importar sliders. Esta vulnerabilidad abre la puerta a un **ataque de Ejecución Remota de Código (RCE)**, lo que permite a un atacante ejecutar código malicioso en el servidor afectado. La deserialización insegura es un problema crítico que ocurre cuando los datos de entrada no son verificados de manera adecuada antes de ser procesados por el sistema.

Impacto

Si se explota con éxito, la vulnerabilidad puede comprometer completamente el sistema, permitiendo a los atacantes realizar acciones maliciosas como:

- **Ejecución de comandos arbitrarios**.
- **Acceso a datos confidenciales**.
- **Modificación o eliminación de archivos críticos del servidor**. Este tipo de ataque podría llevar al **control total del servidor**, comprometiendo la integridad, confidencialidad y disponibilidad de los datos y servicios.

Mitigaciones técnicas

Actualización inmediata del plugin:

- Actualizar Slider Revolution a la versión **6.6.19** o posterior, donde se ha corregido la vulnerabilidad.

Referencias

- https://wpscan.com/vulnerability/36ced447-84ea-4162-80d2-6df226cb53cb/
- https://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2023-6528

HOST	unp.gov.co		
ID-014	Carga Arbitraria De Archivos (CVE-2023-47784)	**CVSS v3: 7.2**	**ALTA**

La vulnerabilidad **CVE-2023-47784** afecta a la versión **6.6.15 y anteriores** del plugin **Slider Revolution** para WordPress, desarrollado por **ThemePunch OHG**. Este fallo de seguridad permite la **carga de archivos sin restricciones de tipo peligroso**, lo que podría dar lugar a la **ejecución de código remoto** en el servidor comprometido.

El problema se debe a la **validación insuficiente** en el manejo de archivos cargados. Los atacantes con permisos de contribuidor o superior pueden cargar archivos con extensiones peligrosas, como **.php**, que pueden ser ejecutados en el servidor, permitiendo que un atacante tome control del sitio o acceda a información confidencial.

Impacto

- **Ejecución remota de código**: Un atacante podría cargar un archivo PHP malicioso y ejecutar comandos en el servidor, comprometiendo todo el sitio web.
- **Pérdida de confidencialidad**: Acceso no autorizado a información confidencial almacenada en el servidor.
- **Modificación o destrucción de datos**: Un atacante podría modificar archivos del servidor, lo que podría resultar en la corrupción de los datos o la eliminación de información crítica.

Mitigaciones técnicas

Actualizar a Slider Revolution 6.6.16 o posterior

Realice un escaneo exhaustivo del directorio wp-content/uploads para detectar y eliminar archivos maliciosos que hayan podido ser cargados previamente.

Aplique restricciones adicionales en el tipo de archivos permitidos en las cargas de archivos del plugin. Esto se puede lograr utilizando reglas de firewall o configurando ModSecurity para bloquear solicitudes maliciosas.

Asegúrese de que solo usuarios de confianza tengan permisos de contribuidor o superiores en su sitio de WordPress.

HOST	unp.gov.co		
ID-015	Remote Code Execution (RCE) (CVE-2023-2359)	**CVSS v3: 8.8**	**ALTA**

La vulnerabilidad **CVE-2023-2359** afecta al plugin de WordPress **Slider Revolution** en versiones anteriores a **6.6.12**. El fallo ocurre porque el plugin no verifica correctamente los archivos de imagen durante la importación, permitiendo la subida de archivos arbitrarios. En ciertas configuraciones de servidor, este defecto puede ser aprovechado para ejecutar código de forma remota (RCE). Aunque la funcionalidad de importación está diseñada para administradores, algunos sitios pueden tener configuraciones que permitan a **autores o editores** utilizar esta característica, lo que amplía el riesgo de explotación.

Impacto

- **Ejecución Remota de Código (RCE)**: Un atacante podría subir un archivo malicioso a través de la funcionalidad de importación de imágenes, obteniendo acceso a ejecutar comandos en el servidor web.
- **Compromiso del servidor**: En configuraciones vulnerables, un atacante podría tomar el control del servidor, comprometer sitios web o realizar movimientos laterales en la red.
- **Confidencialidad, Integridad y Disponibilidad**: La vulnerabilidad tiene un impacto alto en la confidencialidad, integridad y disponibilidad de los datos en el servidor afectado, con un CVSS de **8.8 (Alta)**.

Mitigaciones técnicas

Actualizar a la versión 6.6.13 o superior: El problema fue corregido en la versión **6.6.13**. Se recomienda encarecidamente actualizar el plugin a la última versión disponible.

Deshabilitar la ejecución de archivos PHP en el directorio de uploads: Para mitigar la explotación de esta vulnerabilidad, se recomienda deshabilitar la ejecución de archivos PHP en el directorio de cargas (wp-content/uploads).

Restringir el acceso a la funcionalidad de importación: Asegúrese de que solo los administradores puedan acceder a la funcionalidad de importación de archivos. Revise la configuración del plugin y las capacidades asignadas a otros roles como autores o editores.

HOST	unp.gov.co		
ID-016	Stored Cross-Site Scripting (XSS) a Través Del Callback De Slider (CVE-2024-0611)	**CVSS v3: 5.9**	**MEDIA**

CVE-2024-0611 es una vulnerabilidad de tipo **Stored Cross-Site Scripting (XSS)** que afecta al plugin de WordPress **Master Slider – Responsive Touch Slider**, en todas las versiones hasta la **3.9.5**. Esta vulnerabilidad se origina en la función **slides callback** del plugin, la cual no valida ni escapa adecuadamente las entradas del usuario. Los atacantes con permisos de **Editor+** pueden explotar esta vulnerabilidad para inyectar scripts web maliciosos en las páginas que contienen sliders. Estos scripts se ejecutan cada vez que un usuario accede a las páginas afectadas, lo que puede llevar a la ejecución de código no autorizado en el navegador de la víctima.

Impacto

- **Ejecución de scripts maliciosos**: Los atacantes pueden ejecutar scripts arbitrarios en los navegadores de los usuarios que visitan las páginas afectadas. Estos scripts pueden robar cookies, credenciales o realizar acciones maliciosas en nombre del usuario.
- **Fuga de información**: Un atacante puede acceder a información sensible almacenada en el navegador de la víctima, como sesiones autenticadas.
- **Compromiso de la integridad de la web**: La presencia de scripts maliciosos puede dañar la reputación del sitio web y comprometer la confianza de los usuarios.

Mitigaciones técnicas

Actualizar a la versión segura del plugin: El equipo de desarrollo de **Master Slider** ha lanzado parches de seguridad que corrigen esta vulnerabilidad. Es fundamental actualizar el plugin a la versión **3.9.6** o posterior.

Revisar y corregir los permisos de usuario: Limitar el acceso de usuarios de nivel **Editor+** a funciones críticas del sitio es una buena práctica de seguridad. Asegúrese de que solo usuarios confiables tengan estos permisos.

HOST	unp.gov.co		
ID-017	Falta De Autorización ([CVE-2024-34444](CVE-2024-34444))	**CVSS v3: 6.4**	**MEDIA**

CVE-2024-34444 es una vulnerabilidad de **Falta de Autorización** que afecta al plugin **Slider Revolution** de WordPress en versiones anteriores a la **6.7.0**. La vulnerabilidad se debe a que no se realiza un chequeo adecuado de capacidades en la función **init_rest_api**, lo que permite a atacantes no autenticados modificar datos de los sliders sin los permisos correspondientes. Esta falta de validación adecuada permite que usuarios no autenticados accedan y alteren información crítica en los sliders.

El problema es especialmente grave porque los sliders a menudo se utilizan para mostrar contenido dinámico en sitios web, y la alteración de su información podría llevar a la inyección de código malicioso o a la modificación de contenido de manera no autorizada.

Impacto

- **Modificación no autorizada de datos**: Un atacante puede modificar el contenido del sitio web alterando los sliders, lo que podría llevar a la desfiguración del sitio o la inyección de código malicioso.
- **Riesgo para la integridad del contenido**: La vulnerabilidad permite a usuarios no autenticados manipular los sliders de WordPress, lo que podría tener un impacto significativo en la presentación del contenido del sitio web.
- **Posibilidad de explotación adicional**: Si un atacante logra inyectar scripts maliciosos a través de los sliders, podría comprometer a los usuarios del sitio web o realizar ataques más complejos como el robo de credenciales.

Mitigaciones técnicas

Actualizar a la versión 6.7.0 o posterior: La actualización a **Slider Revolution 6.7.0** corrige esta vulnerabilidad. Es crucial que los administradores del sitio actualicen el plugin a la última versión disponible.

Habilitar controles de acceso: Implemente controles de acceso adicionales para proteger las funciones del plugin, limitando el acceso a usuarios con privilegios específicos.

Monitorear cambios en el contenido del slider: Monitoree de manera activa los cambios realizados en los sliders para detectar cualquier actividad inusual o no autorizada.

HOST	unp.gov.co		
ID-018	Cross-Site Scripting (XSS) Autenticado a Través Del Atributo "VC Single Image Link" (CVE-2024-5265)	**CVSS v3: 6.4**	**MEDIA**

La vulnerabilidad **CVE-2024-5265** permite un ataque de **Cross-Site Scripting (XSS) autenticado** en el constructor de páginas **WPBakery Page Builder**. Afecta a la función relacionada con el atributo **"VC Single Image Link"** en el manejo de imágenes. Los atacantes autenticados, como contribuyentes con permisos mínimos, pueden inyectar código malicioso en los atributos del enlace de la imagen. Esto se debe a una falta de validación y saneamiento adecuados del contenido proporcionado por el usuario.

Un atacante puede aprovechar esta vulnerabilidad para inyectar código JavaScript malicioso que se ejecuta en el navegador de otros usuarios que visualicen el contenido afectado, lo que podría llevar a la **exfiltración de cookies**, secuestro de sesiones, o redirecciones a sitios maliciosos.

Impacto:

- **Ejecución de código en el navegador**: Los atacantes pueden ejecutar código JavaScript en el navegador de los usuarios que visualicen el contenido modificado, lo que podría comprometer su sesión de usuario.
- **Robo de credenciales o cookies**: Las víctimas pueden verse afectadas por la exfiltración de información sensible, como cookies de sesión, que podría ser utilizada para obtener acceso no autorizado.
- **Modificación de la página web**: Un atacante puede redirigir a los usuarios a sitios web maliciosos o modificar la apariencia del sitio para realizar ataques de phishing.

Mitigaciones técnicas

Actualizar a la versión segura de WPBakery: La vulnerabilidad ha sido corregida en la versión **7.7** de WPBakery Page Builder. Se recomienda actualizar el plugin lo antes posible para evitar la explotación de esta vulnerabilidad.

HOST	unp.gov.co		
ID-019	Cross-Site Scripting (XSS) Almacenado (CVE-2024-34443)	**CVSS v3: 5.9**	**MEDIA**

La **CVE-2024-34443** es una vulnerabilidad de **Cross-Site Scripting (XSS) Almacenado** en el plugin **Slider Revolution** de WordPress, antes de la versión 6.7.11. Esta vulnerabilidad ocurre debido a una inadecuada neutralización de entradas durante la generación de páginas web. Un atacante puede inyectar scripts maliciosos que se almacenan en el servidor y se ejecutan cuando los usuarios legítimos visitan la página afectada. Esta vulnerabilidad puede ser explotada remotamente si se envía contenido malicioso que luego es ejecutado en los navegadores de los usuarios que acceden a las páginas afectadas. El XSS almacenado es particularmente peligroso ya que la carga útil maliciosa persiste en el servidor.

Impacto

Una explotación exitosa de esta vulnerabilidad permite que un atacante ejecute scripts arbitrarios en el navegador de la víctima. Esto podría comprometer la **confidencialidad** (por ejemplo, robo de cookies o credenciales), afectar la **integridad** de las interacciones del usuario en el sitio web (modificando contenido o comportamiento), y en algunos casos, llevar a ataques de suplantación de identidad (phishing). Aunque el impacto en la **disponibilidad** es bajo, el riesgo de comprometer datos personales y realizar acciones no autorizadas en nombre del usuario es significativo.

Mitigaciones técnicas

1. **Actualización**: Actualiza el plugin Slider Revolution a la versión 6.7.11 o superior, donde esta vulnerabilidad ha sido parcheada.
2. **Escapes y sanitización**: Asegúrate de que cualquier entrada de usuario sea correctamente escapada antes de ser procesada o almacenada, para evitar la inyección de código malicioso.
3. **Configuración de seguridad adicional**: Implementa políticas de seguridad como Content Security Policy (CSP) para mitigar la ejecución de scripts maliciosos y utiliza técnicas de seguridad adicionales como el escaneo regular de vulnerabilidades en los plugins.

HOST	unp.gov.co		
ID-020	Cross-Site Scripting (XSS) Almacenado a Través Del Shortcode "Ms_layer" (CVE-2024-4375)	**CVSS v3: 4.4**	**MEDIA**

La vulnerabilidad **CVE-2024-4375** es una vulnerabilidad de **Cross-Site Scripting (XSS) Almacenado** que afecta al plugin **Master Slider – Responsive Touch Slider** de WordPress, en todas las versiones hasta la 3.9.10. El problema reside en el manejo del shortcode **ms_layer**, específicamente en el atributo css_id suministrado por el usuario. Debido a una sanitización insuficiente de entradas y la falta de escape adecuado en la salida, los atacantes autenticados con permisos de nivel **contribuyente** o superiores pueden inyectar scripts arbitrarios en las páginas web. Estos scripts se ejecutan cuando otros usuarios acceden a las páginas afectadas.

Impacto

Una explotación exitosa permite a un atacante inyectar scripts maliciosos que se ejecutarán en el navegador de los usuarios que visiten las páginas comprometidas, comprometiendo potencialmente la **confidencialidad** (robo de cookies, credenciales, etc.) y la **integridad** (modificación de contenido). Aunque la **disponibilidad** no se ve directamente afectada, la ejecución de scripts maliciosos puede llevar a otras consecuencias, como redirecciones no deseadas o ataques de suplantación

Mitigaciones técnicas detalladas:

1. **Actualización del plugin**: Actualiza el plugin **Master Slider** a la versión más reciente superior a la 3.9.10, donde se ha solucionado este problema.
2. **Revisión de permisos**: Limita los permisos de los usuarios con acceso de contribuyente para minimizar el riesgo de inyección de código.
3. **Política de seguridad de contenido (CSP)**: Implementa una política de seguridad de contenido en tu sitio web para mitigar la ejecución de scripts no autorizados.

HOST	unp.gov.co		
ID-021	Cross-Site Scripting (XSS) Reflejado (CVE-2024-37222)	**CVSS v3: 7.1**	**MEDIA**

La vulnerabilidad **CVE-2024-37222** es una vulnerabilidad de **Cross-Site Scripting (XSS) Reflejado** que afecta al plugin **Master Slider** de WordPress en todas las versiones hasta la 3.9.10. El problema surge porque el plugin no maneja correctamente la entrada de ciertos parámetros del usuario, lo que permite que atacantes inyecten scripts maliciosos a través de URL manipuladas. Cuando un usuario legítimo hace clic en uno de estos enlaces maliciosos, el script se ejecuta en su navegador sin necesidad de almacenamiento persistente en el servidor.

Impacto

Una explotación exitosa de esta vulnerabilidad permite a los atacantes ejecutar scripts arbitrarios en el navegador de la víctima, comprometiendo la **confidencialidad** (robo de cookies o datos de sesión), la **integridad** (alteración de contenido visualizado) y, en menor medida, la **disponibilidad** (redirigiendo a los usuarios a sitios maliciosos). Este tipo de XSS es particularmente peligroso cuando los usuarios son inducidos a hacer clic en enlaces de confianza que han sido modificados.

Mitigaciones técnicas

1. **Actualización del plugin**: Actualiza el plugin **Master Slider** a la versión más reciente disponible superior a la 3.9.10, donde se ha corregido esta vulnerabilidad.
2. **Validación de entradas**: Implementa un filtrado adecuado en todas las entradas del usuario para evitar que se inyecten caracteres o scripts no deseados.

Referencias

https://www.wordfence.com/threat-intel/vulnerabilities/wordpress-plugins/master-slider/master-slider-3910-reflected-cross-site-scripting

HOST	unp.gov.co		
ID-022	Cross-Site Scripting (XSS) Almacenado a Través Del Parámetro Htmltag (CVE-2024-4092)	**CVSS v3: 6.4**	**MEDIA**

La vulnerabilidad **CVE-2024-4092** es una vulnerabilidad de **Cross-Site Scripting (XSS) Almacenado** que afecta al plugin **Slider Revolution** de WordPress en todas las versiones hasta la 6.7.7. El problema radica en el parámetro htmltag, el cual no es adecuadamente sanitizado ni escapado antes de su uso. Esto permite a atacantes autenticados (con permisos de autor o superiores) inyectar scripts maliciosos en las páginas del sitio web, que luego se ejecutan cuando un usuario accede a esas páginas comprometidas. Aunque por defecto esta vulnerabilidad puede ser explotada solo por administradores, la configuración del plugin puede extender el uso del parámetro htmltag a otros usuarios como autores

Impacto

Una explotación exitosa permitiría a un atacante inyectar y ejecutar código JavaScript malicioso, afectando la **confidencialidad** de los usuarios (robo de cookies o credenciales), comprometiendo la **integridad** de los datos mostrados en la página y, en menor grado, impactando la **disponibilidad** del sitio (por ejemplo, redirigiendo a usuarios a sitios maliciosos)

Mitigaciones técnicas detalladas:

1. **Actualización del plugin**: Actualiza a la versión 6.7.8 o posterior, donde esta vulnerabilidad ha sido corregida.
2. **Validación de entradas**: Implementa una correcta validación de todas las entradas del usuario, asegurando que parámetros como htmltag sean sanitizados antes de su procesamiento.

Referencias

https://wpscan.com/vulnerability/50dd9668-c263-4bfd-8a94-00a3ad9f656d/

https://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2024-4092

https://www.wordfence.com/threat-intel/vulnerabilities/wordpress-plugins/revslider/slider-revolution-677-authenticated-author-stored-cross-site-scripting-via-htmltag-parameter

HOST	unp.gov.co		
ID-023	Cross-Site Scripting (XSS) Almacenado Autenticado a Través Del Atributo De Etiqueta "Custom Heading" (CVE-2024-1842)	**CVSS v3: 6.4**	**MEDIA**

La vulnerabilidad **CVE-2024-1842** afecta al plugin **WPBakery Visual Composer** de WordPress en todas las versiones hasta la 7.5. El problema se presenta debido a una falta de sanitización y escape de las entradas del usuario en el atributo de la etiqueta "Custom Heading". Esto permite que un atacante autenticado con permisos de **contribuyente** o superior inyecte scripts maliciosos en las páginas web. Dichos scripts se ejecutan cuando los usuarios acceden a las páginas comprometidas.

Impacto

Una explotación exitosa de esta vulnerabilidad permitiría a un atacante ejecutar scripts arbitrarios, afectando la **confidencialidad** (por ejemplo, robo de cookies y credenciales) y la **integridad** de la información. Además, aunque el impacto en la **disponibilidad** es menor, puede llevar a la manipulación de contenido o redirecciones maliciosas

Mitigaciones técnicas

1. **Actualización del plugin**: Se debe actualizar a la versión **7.6** o superior del plugin, donde el problema ha sido corregido.

Referencias

https://wpscan.com/vulnerability/8ebfad34-7b46-4783-9fad-c96ab4f4c737/

https://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2024-1842

https://www.wordfence.com/threat-intel/vulnerabilities/wordpress-plugins/js_composer-2/wpbakery-visual-composer-75-authenticated-contributor-stored-cross-site-scripting-via-custom-heading-tag-attribute

HOST	unp.gov.co		
ID-024	Cross-Site Scripting (XSS) Almacenado Autenticado a Través Del Atributo De Etiqueta Del Título Del Post (CVE-2024-1841)	**CVSS v3: 6.4**	**MEDIA**

La vulnerabilidad **CVE-2024-1841** es un caso de **Cross-Site Scripting (XSS) Almacenado** que afecta al plugin **WPBakery Visual Composer** en WordPress, en todas las versiones hasta la 7.5. El problema reside en el atributo de la etiqueta del título del post, que no es adecuadamente sanitizado o escapado. Esto permite que un atacante autenticado con permisos de contribuyente o superior inyecte scripts maliciosos, que se ejecutarán cada vez que un usuario acceda a la página comprometida. Este tipo de ataque aprovecha la falta de validación de las entradas del usuario en el título del post.

Impacto

Una explotación exitosa permitiría a un atacante ejecutar scripts arbitrarios, afectando la **confidencialidad** (por ejemplo, robo de cookies o credenciales) y la **integridad** del sitio (alteración del contenido visualizado por otros usuarios). No tiene un impacto directo en la **disponibilidad**, pero podría utilizarse para redirigir a los usuarios a sitios maliciosos o ejecutar otras acciones indeseadas

Mitigaciones técnicas

1. **Actualizar el plugin**: Se debe actualizar a la versión 7.6 o posterior, donde se ha corregido esta vulnerabilidad.

Referencias

https://www.wordfence.com/threat-intel/vulnerabilities/wordpress-plugins/js_composer-2/wpbakery-visual-composer-75-authenticated-contributor-stored-cross-site-scripting-via-post-title-tag-attribute

https://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2024-1841

https://wpscan.com/vulnerability/787a71f8-1179-4442-9441-87fbe83a7e67/

HOST	unp.gov.co		
ID-025	Cross-Site Request Forgery (CSRF) Para La Eliminación De Sliders (CVE-2024-6490)	**CVSS v3: 6.5**	**MEDIA**

La vulnerabilidad **CVE-2024-6490** es una vulnerabilidad de **Cross-Site Request Forgery (CSRF)** que afecta al plugin **Master Slider – Responsive Touch Slider** en todas las versiones hasta la 3.9.10. Esta vulnerabilidad permite que un atacante no autenticado manipule solicitudes en nombre de un usuario víctima, como la eliminación de sliders del plugin sin el consentimiento del usuario. Esto es posible explotando la confianza que un sitio web tiene en el navegador del usuario, mediante la ejecución de acciones maliciosas cuando un administrador visita una página con código malicioso.

Impacto

Una explotación exitosa permite a un atacante borrar todos los sliders de un sitio web WordPress que utilice el plugin Master Slider, lo que podría afectar gravemente la **integridad** del contenido del sitio web. Esto puede resultar en la pérdida de elementos visuales importantes para la experiencia del usuario, lo que podría tener implicaciones financieras y de reputación para los propietarios del sitio.

Mitigaciones técnicas

1. **Actualización del plugin**: Se recomienda actualizar a la versión más reciente del plugin, superior a la 3.9.10, donde el problema ha sido corregido.
2. **Implementar nonces**: Los desarrolladores de plugins deben implementar **nonces** en los formularios y solicitudes AJAX para evitar que las acciones sean ejecutadas sin la autenticación adecuada del usuario.

Referencias

https://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2024-6490

https://research.cleantalk.org/cve-2024-6490/

https://www.wordfence.com/threat-intel/vulnerabilities/wordpress-plugins/master-slider/master-slider-responsive-touch-slider-395-cross-site-request-forgery-via-process-bulk-action

HOST	unp.gov.co		
ID-026	Cross-Site Scripting (XSS) Autenticado a Través Del Parámetro 'Link' (CVE-2024-5708)	**CVSS v3: 6.4**	**MEDIA**

La vulnerabilidad **CVE-2024-5708** es un caso de **Cross-Site Scripting (XSS) Almacenado** que afecta al plugin **WPBakery Visual Composer** de WordPress en todas las versiones hasta la 7.7. El problema se origina debido a la sanitización insuficiente del parámetro 'link', lo que permite que atacantes autenticados con permisos de **Autor** o superiores puedan inyectar scripts maliciosos en las páginas. Estos scripts se ejecutan automáticamente cuando otros usuarios acceden a las páginas afectadas, comprometiendo la seguridad del sitio

Impacto

Una explotación exitosa de esta vulnerabilidad permite a un atacante ejecutar scripts arbitrarios que podrían comprometer la **confidencialidad** de los usuarios (robando cookies o credenciales) y alterar la **integridad** del contenido del sitio web. Aunque no afecta directamente la **disponibilidad**, el riesgo de manipulación de contenido es significativo

Mitigaciones técnicas

1. **Actualizar el plugin**: Actualiza el plugin a la versión **7.8** o superior, donde esta vulnerabilidad ha sido corregida.
2. **Sanitización adecuada de entradas**: Asegúrate de implementar una correcta validación y escape de las entradas de usuario, especialmente en parámetros sensibles como 'link'.

Referencias

https://wpscan.com/vulnerability/992e5d47-e290-420a-adf8-f552a929e51d/

https://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2024-5708

https://www.wordfence.com/threat-intel/vulnerabilities/wordpress-plugins/js_composer-2/wpbakery-77-authenticated-author-stored-cross-site-scripting

HOST	unp.gov.co		
ID-027	Cross-Site Scripting (XSS) Autenticado a Través Del Atributo "Post Author" (CVE-2024-1840)	**CVSS v3: 6.4**	**MEDIA**

La vulnerabilidad **CVE-2024-1840** es una vulnerabilidad de **Cross-Site Scripting (XSS) Almacenado** que afecta al plugin **WPBakery Visual Composer** en versiones hasta la 7.5. El problema reside en el atributo "Post Author" que no se escapa ni se sanitiza correctamente, lo que permite a los atacantes autenticados con permisos de **Contribuyente** o superiores inyectar scripts maliciosos en las páginas web. Estos scripts se ejecutan automáticamente cuando los usuarios acceden a las páginas comprometidas.

Impacto

Una explotación exitosa permite a un atacante ejecutar scripts arbitrarios que comprometen la **confidencialidad** (por ejemplo, robando cookies o credenciales) y la **integridad** del sitio web (modificación de contenido). Aunque no afecta directamente la **disponibilidad**, este tipo de ataque podría usarse para llevar a cabo otras acciones maliciosas o redirigir a los usuarios a sitios maliciosos.

Mitigaciones técnicas

1. **Actualizar el plugin**: Se debe actualizar el plugin a la versión **7.6** o superior, donde se ha corregido esta vulnerabilidad.

Referencias

https://wpscan.com/vulnerability/b41c2343-3be4-4bd9-ae5d-69ae96ba23ae/

https://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2024-1840

https://www.wordfence.com/threat-intel/vulnerabilities/wordpress-plugins/js_composer-2/wpbakery-visual-composer-75-authenticated-contributor-stored-cross-site-scripting-via-button-onclick-attribute

HOST	unp.gov.co		
ID-028	Cross-Site Scripting (XSS) Almacenado (CVE-2024-1449)	**CVSS v3: 5.4**	**MEDIA**

La vulnerabilidad **CVE-2024-1449** es una vulnerabilidad de **Cross-Site Scripting (XSS) Almacenado** que afecta al plugin **Master Slider – Responsive Touch Slider** de WordPress en todas las versiones hasta la 3.9.5. Esta vulnerabilidad ocurre debido a una insuficiente sanitización de entradas y escape de atributos proporcionados por el usuario en el shortcode ms_slide. Un atacante autenticado con permisos de **contribuyente** o superiores puede inyectar scripts maliciosos en las páginas web, que se ejecutan automáticamente cuando otros usuarios acceden a las páginas afectadas.

Impacto

Una explotación exitosa permite que un atacante ejecute scripts arbitrarios, comprometiendo la **confidencialidad** (por ejemplo, robo de cookies o credenciales) y la **integridad** del contenido del sitio web. Aunque no afecta directamente la **disponibilidad**, podría redirigir a los usuarios a sitios maliciosos o ejecutar otras acciones no autorizadas.

Mitigaciones técnicas

1. **Actualizar el plugin**: Se debe actualizar el plugin a la versión **3.9.6** o superior, donde se ha corregido esta vulnerabilidad.

Referencias

https://wpscan.com/vulnerability/a6bc043f-291d-4f69-b7b6-da3ef8401f6e/

https://www.wordfence.com/threat-intel/vulnerabilities/wordpress-plugins/master-slider/master-slider-responsive-touch-slider-395-authenticated-contributor-stored-cross-site-scripting

HOST	unp.gov.co		
ID-029	Cross-Site Scripting (XSS) Almacenado a Través De Los Parámetros Elementor Wrapperid Y Zindex (CVE-2024-4637)	**CVSS v3: 6.4**	**MEDIA**

La vulnerabilidad **CVE-2024-4637** es un caso de **Cross-Site Scripting (XSS) Almacenado** que afecta al plugin **Slider Revolution** de WordPress, en todas las versiones hasta la 6.7.10. El problema surge por una insuficiente sanitización y escape de los atributos proporcionados por el usuario, específicamente en los parámetros wrapperid y zindex de **Elementor**. Esto permite que un atacante autenticado con permisos de **contribuyente** o superiores inyecte scripts maliciosos en las páginas web, los cuales se ejecutarán automáticamente cuando otros usuarios accedan a las páginas comprometidas.

Impacto

Una explotación exitosa permite que un atacante ejecute scripts arbitrarios que comprometen la **confidencialidad** (por ejemplo, robo de cookies o credenciales) y la **integridad** del contenido del sitio web. Aunque el impacto directo sobre la **disponibilidad** es limitado, la ejecución de scripts maliciosos podría permitir redirecciones o acciones no deseadas para los usuarios que visiten la página afectada.

Mitigaciones técnicas

1. **Actualizar el plugin**: Actualiza a la versión **6.7.11** o superior de **Slider Revolution**, donde esta vulnerabilidad ha sido corregida.
2. **Sanitización de entradas**: Asegúrate de implementar una validación adecuada y un escape de las entradas del usuario, en especial los parámetros wrapperid y zindex.

Referencias

https://wpscan.com/vulnerability/32b1ee26-525a-4b1e-bdf0-881f0d161788/

https://www.wordfence.com/threat-intel/vulnerabilities/wordpress-plugins/revslider/slider-revolution-6710-authenticated-contributor-stored-cross-site-scripting-via-elementor-wrapperid-and-zindex

HOST	unp.gov.co		
ID-030	Cross-Site Scripting (XSS) Almacenado a Través De Los Atributos Class, Id, Y Title De Add Layer (CVE-2024-4581)	**CVSS v3: 5.4**	**MEDIA**

La vulnerabilidad **CVE-2024-4581** es un caso de **Cross-Site Scripting (XSS) Almacenado** que afecta al plugin **Slider Revolution** de WordPress en todas las versiones hasta la 6.7.11. El problema surge debido a la insuficiente sanitización y escape de las entradas de usuario en los atributos class, id y title del widget **Add Layer**. Esto permite a los atacantes autenticados con permisos de **autor** o superiores inyectar scripts maliciosos, que se ejecutarán automáticamente cuando los usuarios accedan a las páginas afectadas. Para que esta vulnerabilidad se explote, el administrador debe otorgar permisos de creación de sliders a usuarios de nivel autor.

Impacto

Una explotación exitosa de esta vulnerabilidad permitiría a los atacantes ejecutar scripts arbitrarios, comprometiendo la **confidencialidad** (robo de cookies o credenciales) y la **integridad** del contenido del sitio. Aunque el impacto en la **disponibilidad** es limitado, los atacantes pueden manipular o redirigir contenido en las páginas afectadas, lo que aumenta el riesgo de daño.

Mitigaciones técnicas

1. **Actualizar el plugin**: Se debe actualizar a la versión **6.7.12** o superior, donde se ha corregido esta vulnerabilidad.
2. **Validación de entradas**: Asegúrate de que los atributos proporcionados por los usuarios, como class, id y title, sean correctamente validados y escapados.

Referencias

https://wpscan.com/vulnerability/4a436977-1295-40e8-9957-bda1b0a3e6d6/

https://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2024-4581

HOST	unpradio.unp.gov.co		
ID-031	Contributor+ Stored Cross-Site Scripting (XSS) en HTML API (CVE-2024-6307)	**CVSS v3: 6.4**	**MEDIA**

La vulnerabilidad **CVE-2024-6307** afecta a **WordPress Core** en versiones anteriores a la 6.5.5. Es una vulnerabilidad de **Cross-Site Scripting (XSS) Almacenado**, explotable a través de la API HTML. El problema surge debido a una sanitización insuficiente de las entradas del usuario y una falta de escape adecuado de URLs, lo que permite a atacantes autenticados con permisos de **contribuyente** o superiores inyectar scripts maliciosos. Estos scripts se ejecutan automáticamente cuando otros usuarios acceden a las páginas comprometidas.

Impacto

Una explotación exitosa permite a los atacantes inyectar scripts arbitrarios, afectando la **confidencialidad** (por ejemplo, robo de cookies o credenciales) y la **integridad** del contenido de las páginas. Aunque no afecta directamente la **disponibilidad**, puede ser utilizado para redirigir a los usuarios a sitios maliciosos o manipular la información visible en la página.

Mitigaciones técnicas

1. **Actualizar WordPress**: Actualiza a la versión 6.5.5 o superior, donde la vulnerabilidad ha sido corregida.
2. **Sanitización de entradas**: Asegúrate de que todas las entradas de usuario y URLs se validen y escapen adecuadamente antes de ser procesadas por la API HTML.

Referencias

https://wpscan.com/vulnerability/2c63f136-4c1f-4093-9a8c-5e51f19eae28/
https://wordpress.org/news/2024/06/wordpress-6-5-5/

HOST	unpradio.unp.gov.co		
ID-032	Contribuidor+ Descarga Arbitraria De SVG (CVE-2024-37437)	**CVSS v3: 6.5**	**MEDIA**

La vulnerabilidad **CVE-2024-37437** es una vulnerabilidad de **Descarga Arbitraria de Archivos SVG** que afecta al plugin **Elementor Website Builder** de WordPress en todas las versiones hasta la 3.22.1. Esta vulnerabilidad ocurre debido a la falta de restricciones adecuadas en las rutas de los archivos (Path Traversal), lo que permite a atacantes autenticados con permisos de **contribuyente** o superiores descargar archivos SVG arbitrarios. Esta falla expone a los sitios a posibles riesgos relacionados con el manejo inseguro de archivos SVG.

Impacto

Una explotación exitosa permitiría a un atacante descargar archivos SVG sin las restricciones adecuadas, lo que podría permitir la exposición de información confidencial o el uso de archivos maliciosos en ataques adicionales. Esta vulnerabilidad afecta la **confidencialidad** del sistema.

Mitigaciones técnicas

1. **Actualizar el plugin**: Se debe actualizar a la versión **3.22.2** o superior, donde se ha corregido esta vulnerabilidad.
2. **Restringir acceso**: Revisar los permisos otorgados a los usuarios con acceso de contribuyente o superior y restringir el acceso a la descarga de archivos a los roles que realmente lo necesiten.

Referencias

https://wpscan.com/vulnerability/e6d56be1-9a2a-426f-88ca-1ffa773622c1/

HOST	unpradio.unp.gov.co		
ID-033	Contribuidor+ XSS Almacenado en Parámetros De URL en Múltiples Widgets (CVE-2024-5416)	**CVSS v3: 5.4**	**MEDIA**

La vulnerabilidad **CVE-2024-5416** afecta al plugin **Elementor Website Builder** de WordPress, en todas las versiones hasta la 3.23.4. Esta es una vulnerabilidad de **Cross-Site Scripting (XSS) Almacenado** que permite a atacantes autenticados, con permisos de **contribuyente** o superiores, inyectar scripts maliciosos en múltiples widgets a través de los parámetros de URL. Esta falla ocurre debido a una sanitización y escape insuficiente de las entradas proporcionadas por el usuario. Los scripts maliciosos se ejecutan automáticamente cuando otros usuarios acceden a las páginas comprometidas.

Impacto:

Una explotación exitosa de esta vulnerabilidad compromete la **confidencialidad** (por ejemplo, robo de cookies y credenciales) y la **integridad** del sitio web (manipulación de contenido). No afecta directamente la **disponibilidad**, pero puede usarse para redirigir a usuarios o ejecutar acciones no autorizadas.

Mitigaciones técnicas

1. **Actualización del plugin**: Se recomienda actualizar a la versión **3.24.0** o superior, donde la vulnerabilidad ha sido corregida.

Referencias

https://wpscan.com/vulnerability/5200943b-5e07-4342-a090-f78435e30d30/

HOST	unpradio.unp.gov.co		
ID-034	Contributor+ Stored XSS en El Bloque Template-Part (CVE-2024-6305)	**CVSS v3: 6.1**	**MEDIA**

La vulnerabilidad **CVE-2024-6305** es un caso de **Cross-Site Scripting (XSS) Almacenado** que afecta a **WordPress Core** en todas las versiones hasta la 6.5.5. Este problema surge en el bloque **Template Part**, donde el atributo tagName no es correctamente sanitizado ni escapado, lo que permite que los atacantes autenticados con permisos de **contribuyente** o superiores inyecten scripts maliciosos en las páginas. Estos scripts se ejecutan cuando otros usuarios acceden a las páginas comprometidas.

Impacto

Una explotación exitosa de esta vulnerabilidad permite a un atacante comprometer la **confidencialidad** (robo de cookies o credenciales) y la **integridad** del sitio web (alteración del contenido). Aunque el impacto en la **disponibilidad** es limitado, esta vulnerabilidad puede usarse para redirigir a los usuarios o realizar acciones no deseadas en las páginas afectadas.

Mitigaciones técnicas

1. **Actualización de WordPress**: Se recomienda actualizar a la versión **6.5.5** o superior, donde esta vulnerabilidad ha sido corregida.

Referencias

https://wpscan.com/vulnerability/7c448f6d-4531-4757-bff0-be9e3220bbbb/

HOST	unpradio.unp.gov.co		
ID-035	Cross-Site Scripting (XSS) Almacenado Autenticado (Contributor+) a Través Del Atributo Align (CVE-2024-8267)	**CVSS v3: 6.4**	**MEDIA**

La vulnerabilidad **CVE-2024-8267** afecta al plugin **Radio Player – Live Shoutcast, Icecast and Any Audio Stream Player** de WordPress en todas las versiones hasta la 2.0.78. Se trata de un caso de **Cross-Site Scripting (XSS) Almacenado** a través del atributo align dentro del bloque **wp** en el editor Gutenberg. Debido a una insuficiente sanitización y escape de las entradas proporcionadas por el usuario, un atacante con permisos de **contribuyente** o superior puede inyectar scripts maliciosos que se ejecutan cuando otros usuarios acceden a las páginas afectadas.

Impacto

Una explotación exitosa permitiría a los atacantes ejecutar scripts arbitrarios en el navegador de los usuarios, comprometiendo la **confidencialidad** (robando cookies o credenciales) y la **integridad** (modificando el contenido). Aunque no afecta la **disponibilidad**, este tipo de ataque podría usarse para redirigir a los usuarios o llevar a cabo otras acciones no deseadas.

Mitigaciones técnicas

1. **Actualizar el plugin**: Se recomienda actualizar a la versión 2.0.79 o superior, donde el problema ha sido solucionado.

Referencias

https://wpscan.com/vulnerability/b631ac2c-5e6e-4b60-ad5f-542a5e85523f/

HOST	unpradio.unp.gov.co		
ID-036	Contribuidor+ XSS Almacenado en DOM (CVE-2024-4619)	**CVSS v3: 6.4**	**MEDIA**

La vulnerabilidad **CVE-2024-4619** es un caso de **Cross-Site Scripting (XSS) Almacenado basado en DOM**, que afecta al plugin **Elementor Website Builder** en versiones hasta la 3.21.5. Esta vulnerabilidad surge debido a la sanitización y escape insuficiente del parámetro hover_animation. Los atacantes autenticados con permisos de **contribuyente** o superiores pueden inyectar scripts maliciosos en las páginas, que se ejecutarán automáticamente cuando un usuario acceda a una página comprometida.

Impacto

Una explotación exitosa de esta vulnerabilidad permite a los atacantes ejecutar scripts arbitrarios en los navegadores de los usuarios, comprometiendo la **confidencialidad** (robo de cookies o credenciales) y la **integridad** del sitio web (alteración del contenido). No tiene un impacto directo en la **disponibilidad**, pero el riesgo de redirigir a los usuarios a sitios maliciosos es alto.

Mitigaciones técnicas

1. **Actualizar el plugin**: Actualiza a la versión **3.21.6** o superior, donde se ha corregido esta vulnerabilidad.
2. **Sanitización de entradas**: Implementar una validación y escape adecuados de todas las entradas de usuario, especialmente en el parámetro hover_animation.

Referencias

https://wpscan.com/vulnerability/8b8f30d6-bd11-4155-bfd2-3ac15248382b/

HOST	unpradio.unp.gov.co		
ID-037	Cross-Site Scripting (XSS) Almacenado Autenticado (Contribuidor+) (CVE-2024-4984)	**CVSS v3: 6.4**	**MEDIA**

La vulnerabilidad **CVE-2024-4984** es un caso de **Cross-Site Scripting (XSS) Almacenado Autenticado**, que afecta al plugin **Yoast SEO** de WordPress en todas las versiones hasta la 22.6. El problema surge debido a una sanitización insuficiente de la entrada en el meta de autor display_name. Esto permite que los atacantes autenticados, con permisos de **contribuyente** o superiores, inyecten scripts maliciosos en las páginas del sitio web. Estos scripts se ejecutan cuando los usuarios acceden a las páginas comprometidas, lo que puede llevar al robo de información confidencial o a la modificación no autorizada del contenido.

Impacto

El impacto incluye la posibilidad de robar cookies o credenciales de los usuarios, comprometiendo la **confidencialidad** y la **integridad** del sitio web. No afecta directamente la **disponibilidad**, pero el riesgo de que se utilicen estos scripts para redirigir a los usuarios o manipular la información es alto.

Mitigaciones técnicas

1. **Actualizar el plugin**: Se debe actualizar a la versión **22.7** o posterior, donde se ha corregido esta vulnerabilidad.

Referencias

https://wpscan.com/vulnerability/467936e2-fe82-4cdc-afec-6782afee3e4e/

https://www.wordfence.com/threat-intel/vulnerabilities/wordpress-plugins/wordpress-seo/yoast-seo-206-authenticated-contributor-stored-cross-site-scripting

HOST	https://190.145.207.91/login		
ID-038	Versión PHP Desactualizada Con Múltiples Vulnerabilidades De Seguridad (CVE-2023-0567, CVE-2023-0568, CVE-2023-0662, CVE-2022-31626, CVE-2022-31625, CVE-2023-23921)	**CVSS v3: 7.0**	**MEDIA**

Las vulnerabilidades en versiones desactualizadas de **PHP** abarcan varios problemas críticos. La **CVE-2023-0567** afecta la función password_verify(), permitiendo que hashes inválidos de Blowfish sean aceptados como válidos, lo que podría comprometer la autenticación de las aplicaciones. La **CVE-2023-0568** se relaciona con un desbordamiento de buffer en la resolución de rutas, lo que puede causar acceso no autorizado o la modificación de datos. Además, la **CVE-2023-0662** puede resultar en una denegación de servicio debido a la sobrecarga del sistema mediante la carga de formularios HTTP con demasiadas partes.

Otras vulnerabilidades, como la **CVE-2022-31626** y la **CVE-2023-23921**, permiten la ejecución de código arbitrario y acceso no autorizado, afectando la integridad y confidencialidad del sistema. Todas estas fallas de seguridad se han corregido en la versión **PHP 8.2.3** y posteriores, por lo que se recomienda actualizar inmediatamente para mitigar los riesgos asociados.

Impacto

1. **Confidencialidad**: Riesgo de acceso no autorizado a datos sensibles debido a vulnerabilidades en la validación y autenticación.
2. **Integridad**: Posibilidad de modificar datos críticos sin autorización, comprometiendo el estado del sistema o de las aplicaciones.
3. **Disponibilidad**: Vulnerabilidad de Denial of Service (DoS) debido a la explotación de recursos.

Mitigaciones Técnicas

1. **Actualizar PHP**: Se recomienda actualizar a **PHP 8.2.3** o superior, donde todas estas vulnerabilidades han sido corregidas

HOST	paco.unp.gov.co		
ID-039	Cross-Site Scripting (XSS) Almacenado Autenticado (CVE-2024-4892)	**CVSS v3: 6.4**	**MEDIA**

La vulnerabilidad **CVE-2024-4892** afecta al plugin **BuddyPress** de WordPress en todas las versiones hasta la 12.4.1. Se trata de un **Cross-Site Scripting (XSS) Almacenado Autenticado** que ocurre debido a la insuficiente sanitización del parámetro display_name. Un atacante con permisos de **suscriptor** o superior puede inyectar scripts maliciosos que se almacenan en la base de datos y se ejecutan cada vez que un usuario legítimo accede a la página comprometida. Esto permite la ejecución de código no autorizado, comprometiendo la integridad y confidencialidad del sitio afectado.

Impacto

El impacto de esta vulnerabilidad es moderado, con un puntaje CVSS de 6.4. Afecta principalmente la integridad y la confidencialidad del sitio, ya que permite la modificación del contenido y la posible captura de datos sensibles como cookies o credenciales de usuarios. No impacta directamente la disponibilidad del sitio.

Mitigaciones técnicas

La recomendación principal es actualizar el plugin **BuddyPress** a la versión 12.5.1 o superior, donde esta vulnerabilidad ha sido corregida. Además, se debe revisar cualquier entrada de usuario para evitar futuras inyecciones de scripts maliciosos

Referencias

https://www.wordfence.com/threat-intel/vulnerabilities/id/113c154d-94a0-41da-a5ed-d9b2617e1c2c

https://wpscan.com/vulnerability/d927baf0-0797-4b9c-b170-c06baf4e9080/

Vulnerabilidad asociada

CVE-2024-3974

HOST	paco.unp.gov.co		
ID-040	Inyección SQL via Rtmedia_gallery Shortcode en rtMedia para WordPress, BuddyPress y bbPress (Contributor+) (CVE-2024-3293)	**CVSS v3: 8.8**	**ALTA**

La vulnerabilidad **CVE-2024-3293** afecta al plugin **rtMedia for WordPress, BuddyPress y bbPress**, en todas las versiones hasta la 4.6.18. Esta vulnerabilidad es un caso de **SQL Injection Autenticado** que se puede explotar a través del shortcode rtmedia_gallery. Un atacante con permisos de **contribuyente** o superior puede inyectar consultas SQL maliciosas en el sitio web debido a la falta de escape adecuado y preparación en las consultas SQL existentes. Esto les permite extraer información sensible directamente desde la base de datos.

Impacto

El impacto es crítico, con un puntaje CVSS de 8.8, lo que permite a los atacantes interactuar directamente con la base de datos, comprometiendo la **confidencialidad** e **integridad** del sistema, como el robo de datos sensibles. No requiere interacción del usuario y puede ser explotado de manera remota.

Mitigaciones técnicas

Se recomienda actualizar a la versión **4.6.19** o superior del plugin, donde este problema ha sido corregido. Además, es importante realizar auditorías periódicas de seguridad para detectar este tipo de vulnerabilidades

Referencias

https://www.wordfence.com/threat-intel/vulnerabilities/wordpress-plugins/buddypress-media/rtmedia-for-wordpress-buddypress-and-bbpress-4618-authenticated-contributor-sql-injection-via-rtmedia-gallery-shortcode

https://wpscan.com/vulnerability/3cc1fb8d-f013-4a93-8724-426cd7481724/

HOST	paco.unp.gov.co		
ID-041	Eliminación arbitraria de archivos y deserialización PHAR autenticada (Contributor+) (CVE-2024-24934)	**CVSS v3: 8.5**	**ALTA**

La vulnerabilidad **CVE-2024-24934** afecta al plugin **Elementor Website Builder** de WordPress en todas las versiones hasta la 3.19.0. Se trata de un caso de **eliminación arbitraria de archivos** y **deserialización PHAR** autenticada. Esta vulnerabilidad es explotada mediante el parámetro tmp_name, que no cuenta con una validación adecuada de las rutas de archivo (Path Traversal). Esto permite a atacantes autenticados con permisos de **contribuyente** o superiores eliminar archivos arbitrarios y ejecutar código malicioso a través de la deserialización de objetos PHP, lo que potencialmente puede conducir a la **ejecución remota de código**.

Impacto

El impacto de esta vulnerabilidad es alto, con un puntaje CVSS de **8.8**. La explotación exitosa permite a los atacantes comprometer la **confidencialidad**, **integridad** y **disponibilidad** del sistema afectado. Los atacantes pueden eliminar archivos cruciales o cargar objetos maliciosos que les otorguen control sobre el servidor.

Mitigaciones técnicas

1. **Actualizar el plugin**: Se recomienda actualizar a la versión **3.19.1** o superior del plugin Elementor, donde esta vulnerabilidad ha sido corregida.
2. **Validación de rutas**: Implementar controles adicionales en las entradas relacionadas con la gestión de archivos, asegurándose de que las rutas sean seguras y validadas correctamente.

Vulnerabilidad asociada

- CVE-2024-2117
- CVE-2024-4619
- CVE-2024-37437
- CVE-2024-5416

HOST	paco.unp.gov.co		
ID-042	Cross-Site Scripting (XSS) Reflejado (CVE-2024-35656)	**CVSS v3: 7.1**	**ALTA**

La vulnerabilidad **CVE-2024-35656** es un caso de **Cross-Site Scripting (XSS) Reflejado** que afecta al plugin **Elementor Pro** en todas las versiones hasta la 3.21.2. Esta vulnerabilidad ocurre debido a una insuficiente sanitización de entradas y escape de salidas, lo que permite a atacantes no autenticados inyectar scripts maliciosos en las páginas web. El ataque se puede explotar cuando un usuario es engañado para realizar una acción, como hacer clic en un enlace malicioso. El impacto se centra en la **confidencialidad** y la **integridad** de los datos, con un puntaje CVSS de 7.1, clasificándolo como de severidad alta.

Impacto

El ataque compromete la **confidencialidad** (posibilitando el robo de cookies y datos) y la **integridad** (modificación del contenido), aunque no afecta directamente la **disponibilidad** del sitio.

Mitigaciones técnicas

Se recomienda actualizar a la versión **3.21.3** o superior, donde esta vulnerabilidad ha sido corregida. Además, es esencial garantizar una correcta validación de entradas en todas las aplicaciones que utilicen el plugin

Vulnerabilidad asociada

- CVE-2024-4107
- CVE-2024-1521
- CVE-2024-2120
- CVE-2024-2121
- CVE-2024-2781
- CVE-2024-1364
- CVE-2024-23523

HOST	paco.unp.gov.co		
ID-043	Cross-Site Scripting (XSS) Almacenado en Plugin Essential Addons for Elementor (CVE-2024-2650)	**CVSS v3: 7.0**	**ALTA**

La vulnerabilidad **CVE-2024-2650** afecta al plugin **Essential Addons for Elementor** en todas las versiones hasta la 5.9.10. Esta es una vulnerabilidad de **Cross-Site Scripting (XSS) Almacenado** que puede ser explotada a través del parámetro alignment en el widget **Woo Product Carousel**. El problema radica en la sanitización insuficiente de entradas y en la falta de escape de salidas, lo que permite a los atacantes autenticados con permisos de **contribuyente** o superior inyectar scripts maliciosos que se ejecutarán cuando otros usuarios accedan a las páginas afectadas.

Impacto

Este XSS almacenado puede comprometer la **confidencialidad** y la **integridad** del sitio web al permitir que los atacantes roben información sensible o manipulen contenido. Aunque el impacto en la **disponibilidad** es bajo, el riesgo de ataque sigue siendo significativo.

Mitigaciones técnicas

Se recomienda actualizar el plugin a la versión **5.9.12** o superior, donde el problema ha sido corregido. También se debe reforzar la validación de entradas en los widgets del plugin para prevenir futuros ataques.

Vulnerabilidad asociada

- CVE-2024-4624
- CVE-2024-5189
- CVE-2024-4003
- CVE-2024-5188
- CVE-2024-4449
- CVE-2024-4448
- CVE-2024-4275
- CVE-2024-1172
- CVE-2024-34764
- CVE-2024-5073
- CVE-2024-39649
- CVE-2024-2623
- CVE-2024-7092
- CVE-2024-8440
- CVE-2024-8742
- CVE-2024-1536
- CVE-2024-3333
- CVE-2024-1171

- CVE-2024-1537
- CVE-2024-3728
- CVE-2024-1236
- CVE-2024-3733
- CVE-2024-4156
- CVE-2024-1276

HOST	paco.unp.gov.co		
ID-044	Inyección de objetos PHP autenticado en Plugin Essential Addons for Elementor (CVE-2024-3018)	**CVSS v3: 8.8**	**ALTA**

La vulnerabilidad **CVE-2024-3018** afecta al plugin **Essential Addons for Elementor** en todas las versiones hasta la 5.9.13. Esta es una vulnerabilidad de **inyección de objetos PHP** autenticada, que ocurre debido a la deserialización de entradas no confiables desde el atributo error_resetpassword en el widget **Login | Register Form** (que está deshabilitado por defecto). Esto permite que atacantes con permisos de **autor** o superiores inyecten objetos PHP, lo que podría llevar a la ejecución de código, eliminación de archivos arbitrarios o la extracción de datos sensibles si se combina con una cadena de POP (Property Oriented Programming) en otro plugin o tema.

Impacto

El impacto es crítico, con un puntaje CVSS de 8.8. La explotación exitosa de esta vulnerabilidad puede comprometer la **confidencialidad**, **integridad** y **disponibilidad** del sistema afectado, ya que un atacante puede ejecutar código o manipular archivos importantes en el servidor.

Mitigaciones técnicas

Se recomienda actualizar el plugin a la versión **5.9.14** o superior, donde esta vulnerabilidad ha sido corregida. Además, es importante revisar y deshabilitar cualquier widget innecesario que pueda representar un riesgo en la deserialización de entradas.

Referencias

https://wpscan.com/vulnerability/1f720bb4-0018-4d46-83bb-034cb4d5c372/

https://github.com/advisories/GHSA-qqr7-5v2h-fcpq

https://nvd.nist.gov/vuln/detail/CVE-2024-3018

HOST	paco.unp.gov.co		
ID-045	Inyección SQL Autenticada (Contribuyente+) a Través de Shortcode (CVE-2024-1799)	**CVSS v3: 8.8**	**ALTA**

La vulnerabilidad **CVE-2024-1799** afecta al plugin **GamiPress** en todas las versiones hasta la 6.8.6. Se trata de una vulnerabilidad de **inyección SQL autenticada**, explotada a través del atributo achievement_types en el shortcode gamipress_earnings. Debido a la falta de escape adecuado de los parámetros proporcionados por el usuario y la preparación insuficiente de las consultas SQL existentes, un atacante con permisos de **contribuyente** o superiores puede inyectar consultas SQL adicionales. Esto permite a los atacantes acceder a información sensible almacenada en la base de datos del sitio WordPress.

Impacto

La severidad de esta vulnerabilidad es alta, con un puntaje CVSS de 8.8, ya que compromete la **confidencialidad**, **integridad** y **disponibilidad** de la información. Un atacante puede extraer datos confidenciales y modificar información crítica del sitio afectado.

Mitigaciones técnicas

Se recomienda actualizar el plugin a la versión **6.8.7** o superior, donde el problema ha sido corregido. También es importante revisar la validación de parámetros y reforzar las medidas de seguridad en la ejecución de consultas SQL.

Referencias

https://securityvulnerability.io/vulnerability/CVE-2024-1799

https://www.recordedfuture.com/vulnerability-database/CVE-2024-1799

Vulnerabilidad asociada

CVE-2024-30455

CVE-2024-2783

CVE-2024-2505

HOST	paco.unp.gov.co		
ID-046	Multiples Vulnerabilidades enElementor Header & Footer Builder	**CVSS v3: 7.0**	**ALTA**

El plugin **Elementor Header & Footer Builder** ha experimentado múltiples vulnerabilidades de **Cross-Site Scripting (XSS) Almacenado Autenticado** en varias versiones, afectando a sitios web que utilizan este plugin. Estas vulnerabilidades permiten a atacantes autenticados con permisos de **contribuyente** o superiores inyectar scripts maliciosos en diferentes partes del sitio, como los widgets de título del sitio o funciones relacionadas con atributos SVG. Los scripts inyectados se ejecutan cuando los usuarios acceden a las páginas comprometidas, lo que expone a riesgos de robo de datos o manipulación no autorizada del contenido.

Impacto

El impacto de estas vulnerabilidades es significativo, comprometiendo la **confidencialidad** y **integridad** de los datos de los usuarios. Un atacante podría modificar el contenido de la página o robar información sensible como cookies o credenciales, lo que puede llevar a una escalada de privilegios o robo de identidad.

Mitigaciones técnicas

La solución recomendada es actualizar el plugin **Elementor Header & Footer Builder** a la **última versión disponible** que corrija estas vulnerabilidades. Las versiones afectadas incluyen aquellas anteriores a la **1.6.29**,

Vulnerabilidad asociada

- CVE-2024-1237
- CVE-2024-4634
- CVE-2024-2619
- CVE-2024-2618
- CVE-2024-5757
- CVE-2024-33933

HOST	paco.unp.gov.co		
ID-047	Exposición de Información Sensible Via API de LearnDash LMS (CVE-2024-1210, CVE-2024-1209, CVE-2024-1208)	**CVSS v3: 8.0**	**ALTA**

El plugin **LearnDash LMS** en versiones anteriores a la 4.10.2 es vulnerable a la **exposición de información sensible** a través de su API en múltiples CVEs (CVE-2024-1208, CVE-2024-1209, CVE-2024-1210). Estas vulnerabilidades permiten que atacantes no autenticados accedan a preguntas de cuestionarios y otros contenidos del sistema de gestión de aprendizaje, lo que compromete la **confidencialidad** de la información.

Los atacantes pueden aprovechar las API habilitadas por defecto, como /wp/v2/ y /ldlms/v1/, para obtener acceso a los cuestionarios y las preguntas sin estar inscritos en los cursos asociados. Esto puede llevar a la exposición de datos educativos y reducir la efectividad de los cuestionarios como mecanismo de evaluación

Impacto

El principal riesgo es la exposición de preguntas de exámenes y cuestionarios. Esta información puede ser accedida sin autenticación, lo que compromete la validez de las evaluaciones y la equidad del proceso educativo. Los estudiantes podrían obtener respuestas antes de presentar los exámenes, disminuyendo la credibilidad del sistema.

URLs Vulnerables

https://paco.unp.gov.co/wp-json/wp/v2/sfwd-question

https://paco.unp.gov.co/wp-json/ldlms/v1/sfwd-quiz

Mitigaciones técnicas

1. Se recomienda actualizar a la versión **4.10.3** o superior del plugin, donde se ha corregido esta vulnerabilidad.

Referencias

https://github.com/karlemilnikka/CVE-2024-1208-and-CVE-2024-1210

HOST	paco.unp.gov.co		
ID-048	Cross-Site Scripting (XSS) Almacenado no Autenticado (CVE-2024-6931)	**CVSS v3: 7.5**	**ALTA**

La vulnerabilidad **CVE-2024-6931** afecta al plugin **The Events Calendar** en versiones hasta la **6.6.3**. Se trata de un caso de **Cross-Site Scripting (XSS) Almacenado no Autenticado**, que se explota a través del campo RSVP name. Esto permite que atacantes no autenticados inyecten scripts maliciosos en las páginas del sitio web, los cuales se ejecutan cuando los usuarios legítimos acceden a las páginas comprometidas. La falla se debe a una insuficiente sanitización de las entradas de usuario y la falta de escape adecuado en la salida.

Impacto

El impacto es severo, con un puntaje CVSS de **7.2**. Este tipo de vulnerabilidad compromete principalmente la **confidencialidad** y la **integridad** de los datos, ya que los scripts inyectados pueden robar información sensible, redirigir usuarios a sitios maliciosos o alterar el contenido de la página.

Mitigaciones técnicas

La solución es actualizar a la versión **6.6.4** o superior del plugin, donde se ha corregido este problema. Se recomienda aplicar esta actualización lo antes posible para evitar posibles explotaciones

Vulnerabilidad asociada

- CVE-2024-4180
- CVE-2024-1295
- CVE-2024-37518
- CVE-2024-1295
- CVE-2024-8493

HOST	paco.unp.gov.co		
ID-049	Inyección SQL no autenticada (CVE-2024-8275)	**CVSS v3: 9.8**	**CRÍTICO**

La vulnerabilidad **CVE-2024-8275** en el plugin **The Events Calendar** de WordPress permite la explotación de una **inyección SQL no autenticada**. Esto ocurre debido a una insuficiente sanitización en el parámetro 'order' dentro de la función tribe_has_next_event(), lo que permite a los atacantes agregar consultas SQL maliciosas y extraer información sensible de la base de datos. Esta falla afecta a todas las versiones del plugin hasta la **6.6.4** y es crítica, con un puntaje CVSS de **9.8**.

El ataque no requiere autenticación y puede ser realizado remotamente, lo que incrementa considerablemente el riesgo. Solo los sitios que han implementado manualmente la función tribe_has_next_event() están expuestos a este ataque.

Impacto

Un atacante puede utilizar esta vulnerabilidad para obtener acceso a datos sensibles del sistema, como credenciales, datos personales, o información almacenada en la base de datos, comprometiendo tanto la **confidencialidad** como la **integridad** del sitio afectado. Esto podría llevar a una escalada de privilegios o incluso a la toma de control del sistema.

Mitigaciones técnicas

La solución recomendada es actualizar el plugin **The Events Calendar** a la versión **6.6.4.1** o superior, donde la vulnerabilidad ha sido corregida. También es aconsejable revisar el código personalizado que involucra la función tribe_has_next_event() para asegurar que no se sigan utilizando consultas SQL no seguras.

Exploit Publico Disponible

https://github.com/p33d/CVE-2024-8275

Estudio de Anticipación Cibernética

Grupo APT Blind Eagle (APT-C-36)

Resumen Ejecutivo

APT-C-36 (Blind Eagle) es un grupo de amenazas persistentes avanzadas (APT) enfocado en espionaje y cibercrimen, principalmente en Colombia. Su relevancia para la **UNP** radica en sus ataques dirigidos a instituciones gubernamentales colombianas, incluyendo sectores de seguridad y defensa. El grupo utiliza técnicas avanzadas como Spear-phishing y herramientas de acceso remoto (RATs) para comprometer sistemas críticos y robar información confidencial. Las amenazas identificadas incluyen acceso no autorizado a datos sensibles, control remoto de sistemas y persistencia prolongada dentro de las redes gubernamentales.

Aunque la atribución precisa es compleja debido a las tácticas de ofuscación y desinformación empleadas, se cree que APT-C-36 tiene su origen en actores internos de la región o cuenta con colaboradores locales. Su principal motivación es el espionaje cibernético para obtener información sensible que pueda ser utilizada con fines políticos, económicos o estratégicos.

Objetivos Principales

- **Entidades Gubernamentales**: Ministerios, agencias de seguridad y organismos relacionados con la defensa y política exterior.
- **Instituciones Financieras**: Bancos, cooperativas y organizaciones relacionadas con el sector financiero.
- **Empresas de Telecomunicaciones**: Proveedores de servicios de Internet y comunicaciones móviles.
- **Organizaciones No Gubernamentales (ONGs)**: Especialmente aquellas involucradas en derechos humanos y asuntos sociales.

Técnicas y Procedimientos Comunes

- **Spear Phishing Personalizado**: Envío de correos electrónicos altamente personalizados que contienen archivos adjuntos maliciosos o enlaces a sitios web comprometidos.
- **Uso de Malware Personalizado**: Desarrollo y despliegue de troyanos de acceso remoto (RATs) y otras formas de malware diseñadas para evadir soluciones antivirus convencionales.
- **Exfiltración de Datos**: Transferencia de información sensible a servidores controlados por el atacante utilizando técnicas de encriptación para evitar la detección.

- **Movimiento Lateral**: Una vez dentro de una red, utilizan credenciales robadas para expandir su acceso y comprometer múltiples sistemas y cuentas de usuario.
- **Persistencia**: Implementación de Backdoors y otros mecanismos para mantener el acceso a largo plazo en las redes comprometidas.

Herramientas y Recursos Utilizados

1. **Servidores de Comando y Control (C2)**: Infraestructura robusta para gestionar malware y coordinar operaciones de ataque.
2. **Técnicas de Ofuscación y Evasión**: Uso de empaquetadores y cifrado para ocultar el código malicioso y dificultar su análisis.
3. **Recolección de Credenciales**: Empleo de keyloggers y herramientas de scraping para capturar nombres de usuario y contraseñas.

Casos Destacados

- **Campaña de 2019**: Ataques dirigidos a ministerios gubernamentales utilizando documentos de Microsoft Office con macros maliciosas.
- **Operación de 2020**: Compromiso de varias instituciones financieras mediante correos electrónicos falsificados que simulaban provenir de entidades reguladoras.

Riesgo para la UNP

La UNP, por su naturaleza y responsabilidades, es un objetivo atractivo para APT-C-36. La información manejada por la UNP sobre personas protegidas, operativos y protocolos de seguridad es de alto valor para actores maliciosos que buscan desestabilizar estructuras gubernamentales o aprovecharse de información confidencial.

Es imperativo que la UNP fortalezca sus defensas cibernéticas y adopte una postura proactiva en la detección y mitigación de amenazas asociadas con APT-C-36. Esto incluye la capacitación del personal en reconocimiento de intentos de phishing, implementación de soluciones avanzadas de seguridad y establecimiento de protocolos de respuesta a incidentes.

Tácticas, Técnicas y Procedimientos (TTPs) del Grupo APT-C-36

1. Spear-Phishing con Archivos Adjuntos Maliciosos

APT-C-36 utiliza correos electrónicos dirigidos a empleados de instituciones clave. En la UNP, un correo aparentemente legítimo de una agencia gubernamental podría contener un archivo adjunto protegido con contraseña. Al abrirlo, se ejecutaría un backdoor como *Imminent Monitor*, comprometiendo la seguridad de los sistemas de la organización y otorgando a los atacantes acceso a información confidencial o incluso datos de protección de altos funcionarios.

Ejecución Técnica: El archivo adjunto suele estar en formato MHTML o Word, y una vez abierto, pide a la víctima que habilite macros o ingrese la contraseña proporcionada en el cuerpo del correo. Al habilitarse, las macros ejecutan un código que descarga y ejecuta un payload, generalmente un Remote Access Trojan (RAT) como *Imminent Monitor*, que conecta el sistema infectado a un servidor de comando y control (C2) externo. Desde este punto, los atacantes pueden:

- Capturar información confidencial.
- Controlar remotamente el dispositivo.
- Escalar privilegios en la red.

2. Uso de RATs (Remote Access Trojans)

Los Remote Access Trojans (RATs) son herramientas maliciosas que permiten a los atacantes tomar el control remoto de un sistema infectado, proporcionando acceso total al dispositivo comprometido. Una vez que el RAT se instala en la máquina, el atacante puede ejecutar una variedad de acciones maliciosas sin el conocimiento del usuario.

En el contexto de la **UNP**, el uso de RATs como *QuasarRAT* y *AsyncRAT* representa una amenaza crítica, ya que permiten a los atacantes controlar remotamente las máquinas comprometidas, acceder a información sensible y monitorear operaciones de seguridad. Los riesgos principales incluyen la vigilancia no autorizada de comunicaciones, la manipulación de datos protegidos y el control a largo plazo de sistemas clave. Esto podría afectar directamente la capacidad de la UNP para llevar a cabo sus operaciones de manera segura, ya que los atacantes tendrían la posibilidad de extraer información crítica o modificar datos importantes relacionados con la protección de altos funcionarios.

Capacidades de los RATs:

- **Ejecución Remota de Comandos**: Los RATs permiten que los atacantes ejecuten comandos en el sistema comprometido, lo que puede incluir la instalación de malware adicional o la manipulación de archivos y procesos.
- **Registro de Pulsaciones de Teclas (Keylogging)**: Con esta función, los atacantes pueden capturar las pulsaciones de teclas del usuario, lo que les permite robar credenciales, información financiera y otra información sensible.
- **Captura de Pantallas y Datos Sensibles**: Los RATs permiten la captura de pantallas en tiempo real, brindando a los atacantes acceso visual a información crítica.
- **Movimiento Lateral en la Red**: Los RATs pueden propagarse dentro de la red, infectando más dispositivos y aumentando el control de los atacantes sobre la infraestructura de la organización.

Una vez que un RAT como *QuasarRAT* o *AsyncRAT* se instala, establece una conexión con un servidor de comando y control (C2), lo que permite a los atacantes operar de manera encubierta. Este servidor C2 es responsable de enviar instrucciones al RAT, que a su vez ejecuta diversas acciones como la recopilación de datos o la instalación de otras herramientas maliciosas. En muchos casos, los RATs también deshabilitan medidas de seguridad en los sistemas infectados, como antivirus o firewalls, lo que facilita su operación sin ser detectados.

3. Persistencia mediante Tareas Programadas Enmascaradas

APT-C-36 utiliza tareas programadas para garantizar la persistencia de su malware. En un entorno como el de la UNP, esto aseguraría que el malware siga activo, incluso después de intentos de desinfección o reinicios del sistema. Las tareas maliciosas están disfrazadas de actualizaciones legítimas, lo que dificulta su identificación y eliminación.

APT-C-36 crea tareas programadas maliciosas inmediatamente después de comprometer un sistema, disfrazándolas como tareas de mantenimiento del sistema o actualizaciones legítimas. Estas tareas se configuran para ejecutarse de manera periódica, activando scripts que conectan con un servidor de comando y control (C2), lo que permite a los atacantes mantener su acceso remoto y persistente en la máquina infectada.

Ciclo de Persistencia

1. **Compromiso inicial**: Tras una infección (por ejemplo, mediante Spear-phishing), el malware instala una tarea programada maliciosa.
2. **Disfraz**: La tarea se disfraza de proceso legítimo.
3. **Ejecución automática**: La tarea se ejecuta en intervalos regulares o en eventos específicos (como el inicio del sistema), permitiendo que el malware vuelva a activarse tras reinicios o intentos de eliminación.
4. **Conexión a C2**: Cada vez que se ejecuta la tarea, se establece una nueva conexión con el servidor C2 del atacante.

Impacto en la UNP

Para la UNP, el uso de tareas programadas maliciosas puede permitir que APT-C-36 mantenga un acceso continuo a sus sistemas sin ser detectados por largos periodos. Incluso después de eliminar una infección visible, la tarea programada puede reactivar el malware, comprometiendo la capacidad de la UNP para asegurar sus operaciones. Esto pone en riesgo la integridad de datos sensibles y la seguridad de las personas protegidas por la organización.

4. Evasión de Detección mediante Ofuscación de Código y Modificación de Servicios

APT-C-36 implementa para ocultar sus actividades técnicas de ofuscación de código y modificaciones en los servicios de seguridad del sistema. Estas capacidades permiten que el malware evite ser detectado por soluciones antivirus y firewalls, permaneciendo en el sistema durante períodos prolongados sin levantar alertas.

Técnicas Utilizadas:

- **Obfuscación del Código**: APT-C-36 emplea herramientas como *ConfuserEx* para ocultar el código del malware, haciéndolo más difícil de analizar y detectar por software de seguridad.
- **Modificación de Servicios de Seguridad**: Desactivan o modifican servicios críticos como Windows Defender o el Firewall de Windows. Esto permite que el malware opere sin interferencias.

Impacto en la UNP

Para la UNP, estas técnicas significan que APT-C-36 podría permanecer oculto en los sistemas comprometidos, incluso después de que se tomen medidas de seguridad iniciales. La ofuscación del código y la desactivación de servicios como el antivirus permiten que los atacantes realicen actividades maliciosas como la exfiltración de datos o el control remoto de sistemas sin ser detectados por largos períodos. Esto expone a la organización a un riesgo considerable de fugas de información crítica y compromete la efectividad de las defensas de seguridad implementadas.

5. Exfiltración de Datos a través de Plataformas Públicas (Pastebin, Discord)

APT-C-36 emplea plataformas públicas como **Pastebin** y **Discord** para alojar fragmentos de código malicioso y cargas útiles, permitiendo a sus RATs (QuasarRAT, AsyncRAT) acceder a estos recursos en tiempo real. La información exfiltrada desde las víctimas es enviada a estos servicios, lo que facilita una comunicación constante entre el malware y el servidor de comando y control (C2).

APT-C-36 aloja scripts en **Pastebin** y archivos de carga en **Discord**, ya que estas plataformas no generan sospechas inmediatas. Usan URLs públicas para entregar Payloads o modificar malware dinámicamente.

Impacto en la UNP

El uso de plataformas públicas permite a los atacantes exfiltrar información crítica sin ser detectados. En el caso de la UNP, los atacantes podrían extraer datos confidenciales relacionados con las operaciones de seguridad o sobre las personas protegidas. Esto compromete no solo la seguridad operativa sino también la integridad de las comunicaciones internas, exponiendo información clasificada a actores externos. Las plataformas públicas también permiten a los atacantes actualizar y redistribuir sus herramientas sin intervención directa, complicando la detección y mitigación por parte de los equipos de seguridad de la UNP.

Este método de exfiltración es altamente efectivo porque estas plataformas no suelen estar bajo un monitoreo riguroso por parte de las soluciones de seguridad estándar, lo que permite que los atacantes operen con bajo riesgo de ser detectados.

Contramedidas Técnicas

Ataques de Spear-Phishing

Defensa Sugerida: Filtros avanzados de correo electrónico y Sandboxing

- **DMARC, DKIM, SPF**: Estas tecnologías autentican correos y verifican que los mensajes provengan de remitentes autorizados. DMARC asegura que los correos no sean falsificados, mientras que DKIM y SPF validan el dominio y las direcciones IP de envío.
- **Gateway Anti-Phishing**: Estas soluciones permiten analizar los correos electrónicos en busca de comportamientos sospechosos basados en su estructura, adjuntos y configuraciones del servidor del correo remitente.

Implementación en la UNP:

- **Capacitación Recurrente**: Incluir módulos educativos trimestrales sobre identificación de phishing para todo el personal de la UNP. Las capacitaciones deben ser prácticas y personalizadas, simulando ataques dirigidos que el personal pueda enfrentar en su entorno real de trabajo. Esto no solo refuerza el conocimiento teórico, sino que también mejora la capacidad de respuesta ante posibles amenazas.
- **Configuración de Soluciones Anti-Phishing con Sandboxing**:
    1. **Integración de DMARC, DKIM, y SPF**: Verificar que todos los dominios bajo el control de la UNP cuenten con configuraciones correctas de DMARC, DKIM y SPF. Esto permite que solo correos legítimos lleguen a los empleados, reduciendo el riesgo de phishing. Además, configurar políticas estrictas de rechazo de correos no autenticados.
    2. **Soluciones Anti-Phishing en Gateways de Correo**: Implementar firewalls especializados en correos electrónicos que incorporen detección avanzada de amenazas y sandboxing como ***Proofpoint***. Estas soluciones deben integrarse con las infraestructuras de correo existentes y ser capaces de escanear tanto enlaces como archivos adjuntos para identificar potenciales amenazas.
    3. **Pruebas Periódicas de Phishing**: Realizar campañas simuladas de phishing internas para evaluar la efectividad de las capacitaciones y las tecnologías implementadas. Esto proporciona datos sobre la tasa de éxito de los ataques simulados y ayuda a ajustar las estrategias de defensa.

Ataques con RATs

Defensa Sugerida:

1. **Garantizar la Instalación Completa del EDR**: Asegurar que el EDR esté presente y funcionando correctamente en todas las estaciones de trabajo. El EDR debe monitorizar constantemente el tráfico, procesos y conexiones sospechosas para detectar intentos de acceso remoto o comportamientos maliciosos, como el control de un servidor C2 (Command & Control).
2. **Monitoreo Constante**: Configurar el EDR para generar alertas automáticas basadas en patrones de comportamiento, detectando conexiones inusuales y tráfico persistente a direcciones IP no reconocidas, característico de RATs.
3. **Ejercicios de Threat Hunting**: Realizar búsquedas activas de amenazas, enfocándose en la detección de comportamientos anómalos, tráfico sospechoso, procesos ocultos o maliciosos, y signos de movimiento lateral en la red.

Implementación en la UNP:

1. **Garantizar la Instalación Completa del EDR**:
    - **Auditorías Periódicas**: Realizar auditorías internas para confirmar que todas las estaciones de trabajo están protegidas por el EDR. Implementar procesos automáticos que alerten cuando un Endpoint no esté protegido adecuadamente.
2. **Monitoreo Constante**:
    - **Análisis de Tráfico**: Revisar regularmente los registros de red para detectar patrones anómalos. Las alertas generadas por el EDR deben ser investigadas y respondidas de inmediato.
3. **Ejercicios de Threat Hunting**:
    - **Calendario de Ejercicios**: Realizar ejercicios de Threat Hunting cada trimestre, revisando los logs del EDR para identificar actividades maliciosas que hayan pasado desapercibidas.
    - **Revisión de Indicadores de Compromiso (IoCs)**: Utilizar los datos recolectados por el EDR para buscar indicadores de compromiso específicos de APT-C-36, como intentos de conexión a servidores C2 o presencia de RATs conocidos.

Persistencia mediante Tareas Programadas

Defensa Sugerida

1. **Monitoreo de Tareas Programadas**: Implementar un sistema de monitoreo que audite y registre cambios en las tareas programadas en todos los Endpoints, verificando la creación de tareas maliciosas.
2. **Restricción de Creación de Tareas**: Limitar la capacidad de crear tareas programadas solo a usuarios con permisos elevados para evitar la persistencia del malware.

Implementación en la UNP:

1. **Monitoreo de Tareas Programadas**:
    - **Auditorías Automáticas**: Configurar el EDR para monitorear la creación y modificación de tareas programadas. Establecer alertas cuando se detecten tareas creadas fuera de políticas establecidas o tareas que imiten servicios legítimos como "Google Update" o "Windows Update".
    - **Revisión Manual**: Implementar revisiones periódicas en sistemas críticos para detectar cualquier tarea que no esté documentada como parte del mantenimiento rutinario.
2. **Restricción de Creación de Tareas**:
    - **Permisos Limitados**: Modificar las políticas de seguridad en la UNP para que solo administradores y usuarios con privilegios elevados puedan crear o modificar tareas programadas.
    - **Políticas de Grupo (GPO)**: Usar directivas de grupo (GPO) para establecer reglas que limiten la creación de tareas a roles específicos, minimizando el riesgo de que los atacantes mantengan persistencia mediante estas tareas.

Evasión de Detección mediante Ofuscación de Código

Defensa Sugerida:

1. **Análisis Dinámico de Comportamiento**: Fortalecer los mecanismos de sandboxing dinámico para observar el comportamiento del código en lugar de depender únicamente de firmas estáticas. Esto ayuda a detectar la ofuscación cuando el código se ejecuta.

Implementación en la UNP:

1. **Análisis de Entropía en Código**:
    - **Monitoreo de Ejecutables Comprimidos o Encriptados**: Añadir reglas específicas en el SIEM que detecten cuando se ejecutan archivos comprimidos o encriptados sospechosamente, ya que esto podría ser un indicio de ofuscación.
2. **Análisis Dinámico de Comportamiento**:
    - **Ampliar el Uso de Sandboxing Dinámico**: Utilizar el componente del laboratorio de análisis de Malware de SARAC para realizar análisis de comportamiento de archivos sospechosos. Esta técnica es clave para identificar actividades de ofuscación que solo son visibles en tiempo de ejecución, como la descodificación dinámica de Payloads.
    - **Monitoreo de Indicadores Basados en Comportamiento**: Desarrollar perfiles basados en comportamiento en el EDR y SIEM para identificar patrones que el código ofuscado podría tratar de ocultar, como la inyección de procesos, el uso inusual de memoria o la manipulación de servicios de seguridad.

Exfiltración de Datos a través de Plataformas Públicas

Defensa Sugerida:

1. **Monitorización de Tráfico hacia Servicios Públicos**: Implementar controles estrictos que detecten y bloqueen el tráfico saliente hacia plataformas como Pastebin y Discord, donde los atacantes suelen exfiltrar datos.
2. **Uso de Soluciones DLP (Data Loss Prevention)**: Implementar sistemas avanzados de prevención de pérdida de datos que identifiquen patrones de exfiltración basados en tipos de datos sensibles y rutas de salida no autorizadas.

Implementación en la UNP:

1. **Monitorización de Tráfico hacia Servicios Públicos**:
   - **Bloqueo de Plataformas Públicas**: Configurar el firewall para bloquear el acceso a plataformas como Pastebin, Discord, y otros servicios no autorizados para la transmisión de datos. Asegurar que todo el tráfico a estos sitios sea registrado y auditado.
   - **Alertas Basadas en Análisis de Tráfico**: Programar el SIEM para generar alertas cuando detecte intentos de conexión a estos servicios desde sistemas internos, con un enfoque especial en los servidores sensibles que manejan información crítica.
2. **Uso de Soluciones DLP**:
   - **Implementación de DLP**: Configurar políticas de DLP para monitorear los intentos de transferencia de archivos o datos clasificados desde estaciones de trabajo de la UNP hacia sitios públicos no autorizados. Asegurarse de que los documentos que contienen datos sensibles estén marcados para que el sistema DLP pueda detectarlos si se intentan transferir de forma no segura.
   - **Análisis de Comportamiento de Exfiltración**: Establecer reglas específicas para detectar patrones de comportamiento que podrían indicar un intento de exfiltración (como grandes volúmenes de datos que salen de la red o múltiples intentos de subir archivos a servicios en la nube).

Identificación de Exposición y Filtraciones de Datos

Repositorios de Código Públicos

En esta sección, se ha identificado una exposición significativa de información sensible en dos repositorios públicos de GitHub asociados con aplicaciones internas de la organización. Se ha detectado la publicación de usuarios, contraseñas, IP internas y configuraciones críticas de las aplicaciones. Esto representa un riesgo grave para la seguridad de la organización, ya que permite a los actores maliciosos acceder a la infraestructura interna, explotando vulnerabilidades y comprometiendo la integridad de los sistemas. Es imperativo realizar acciones inmediatas para mitigar estos riesgos.

Resumen de Hallazgos

La organización tiene dos repositorios públicos de GitHub que contienen información confidencial y código relacionado con desarrollos internos. Estos repositorios incluyen configuraciones de bases de datos, credenciales de usuarios, y una dirección IP interna (172.16.20.121), entre otros detalles sensibles. Los repositorios identificados son:

1. **Repositorio 1: Ecosistema Sesp - SicpApp**: Contiene el código completo de una aplicación interna de gestión. Incluye configuraciones de bases de datos, rutas de API, y otros elementos críticos que podrían ser aprovechados por un atacante.
2. **Repositorio 2: Luis Sarmiento - Reuniones**: Contiene configuraciones para una aplicación de reuniones internas, con credenciales y una IP interna que podrían ser utilizadas para ataques dirigidos.

Componentes Identificados

- **Exposición de credenciales**: Se han encontrado usuarios y contraseñas directamente en los archivos del repositorio. Estas credenciales podrían ser utilizadas por atacantes para acceder a bases de datos y otros sistemas internos.
- **Exposición de una IP interna**: La IP (172.16.20.121) permite a los atacantes identificar y localizar potenciales puntos de entrada en la red corporativa.
- **Código completo de las aplicaciones**: Al estar publicado el código fuente completo, los atacantes pueden analizar las estructuras internas, entender cómo interactúan los servicios y explotar cualquier vulnerabilidad no parcheada.

Evidencias de Repositorios Públicos

Repositorio 1: Ecosistema Sesp - SicpApp

Esta imagen muestra la página principal del repositorio **"sicpApp"** en GitHub, administrado por el usuario **Ecosistema-sesp**. El repositorio contiene diversas carpetas que incluyen BaseDatos, src, android, ios, entre otras, lo que indica que es un proyecto relacionado con una aplicación multiplataforma. Este repositorio está público, lo que expone la posibilidad de que usuarios no autorizados accedan al código y a cualquier información confidencial presente.

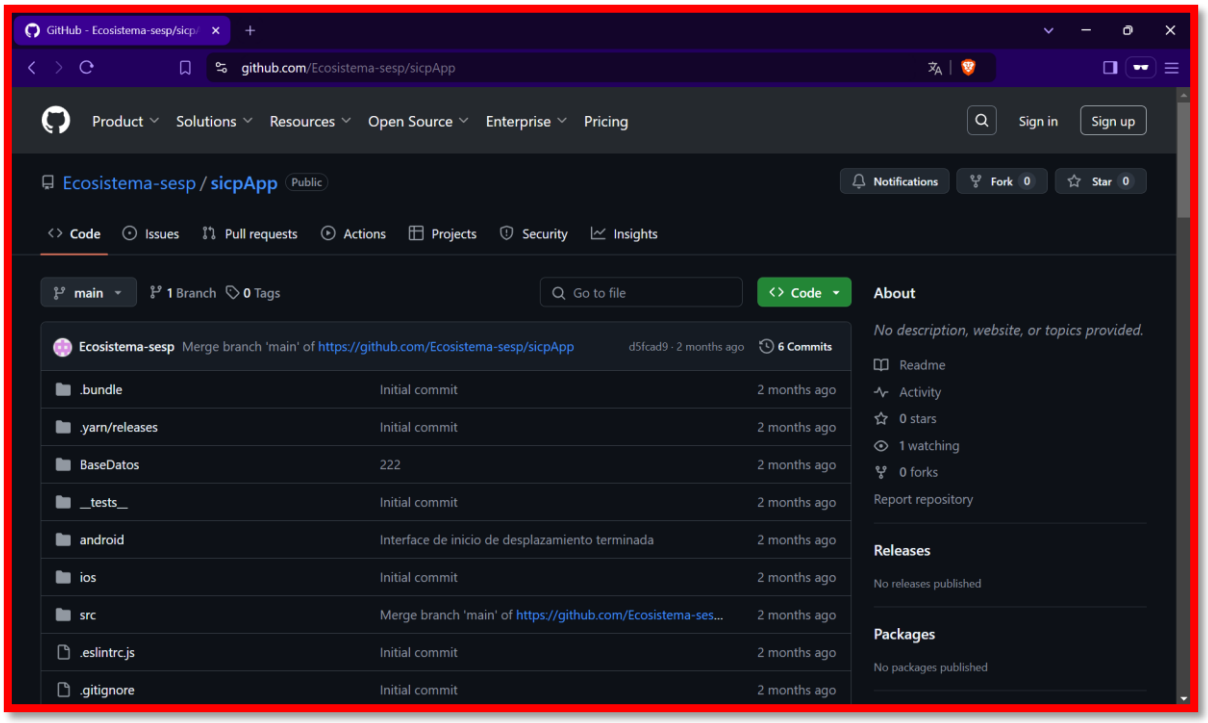


Imagen 9. Repositorio sicApp Identificado en GitHub

En esta imagen se puede observar el archivo Pasajero.tsx del repositorio **SicpApp**. Dentro del código, se realiza una llamada HTTP POST hacia la URL: http://ecosistemasesp.unp.gov.co/sicp/api/pasajero/, lo que sugiere una API expuesta públicamente perteneciente a la organización. Esto representa un riesgo considerable, ya que permite a los atacantes estudiar cómo interactuar con la API, buscar posibles vulnerabilidades en los Endpoints, o explotar las rutas expuestas.

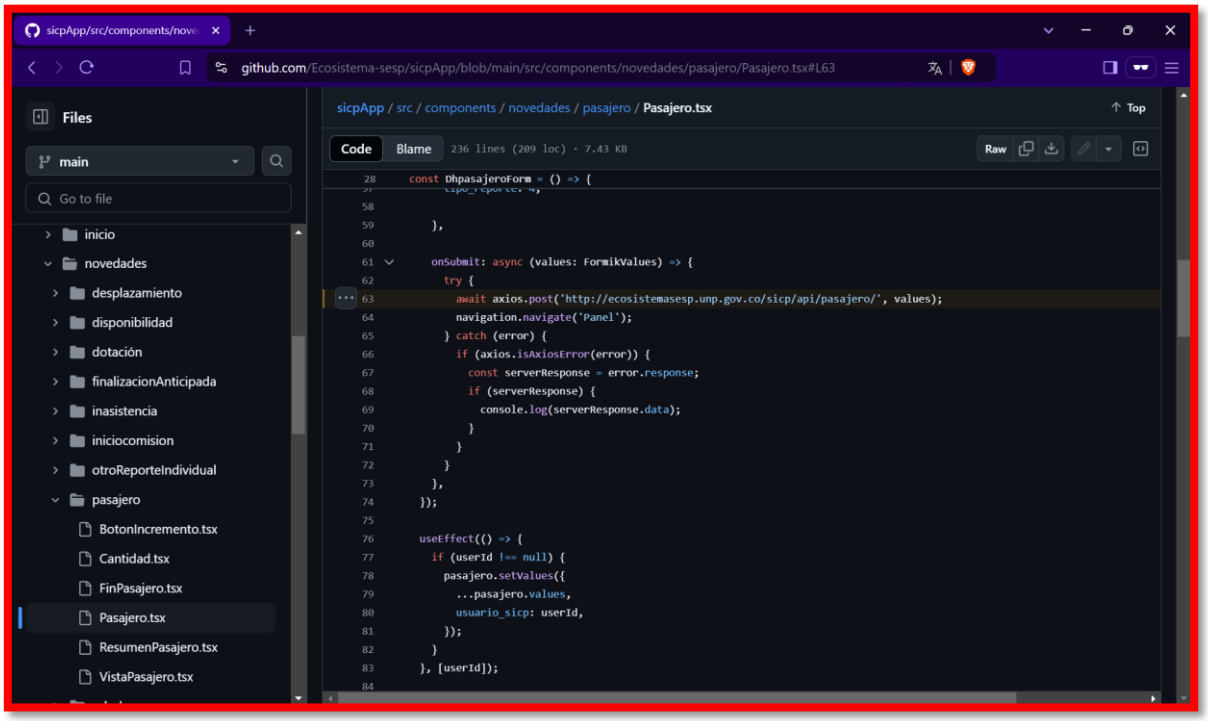


Imagen 10. Acceso a Código con Rutas de la Aplicación en Desarrollo

Repositorio 2: Luis Sarmiento - Reuniones

Esta imagen muestra la página principal del repositorio público **"reuniones"**, perteneciente a Luis Sarmiento. En ella se destacan las diferentes carpetas del proyecto y su estructura, que incluyen directorios como ajax, config, classes, y más.

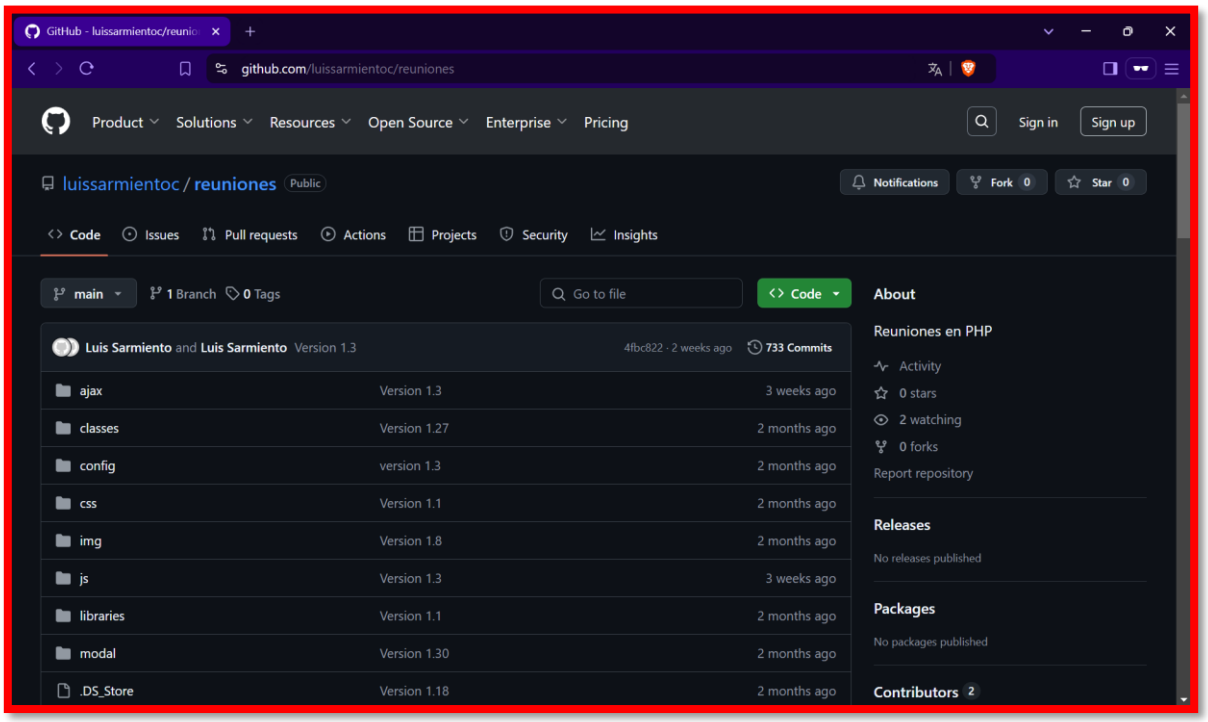


Imagen 11. Repositorio Público con Proyecto *Reuniones*

Esta imagen presenta el archivo db.php, el cual contiene credenciales de conexión a una base de datos PostgreSQL. Las credenciales expuestas incluyen el nombre de usuario, la contraseña (Odiseas125psql), la dirección IP interna (172.16.20.121), y el nombre de la base de datos (unp_reuniones). La exposición de esta información es crítica, ya que podría permitir a atacantes acceder directamente a la base de datos.

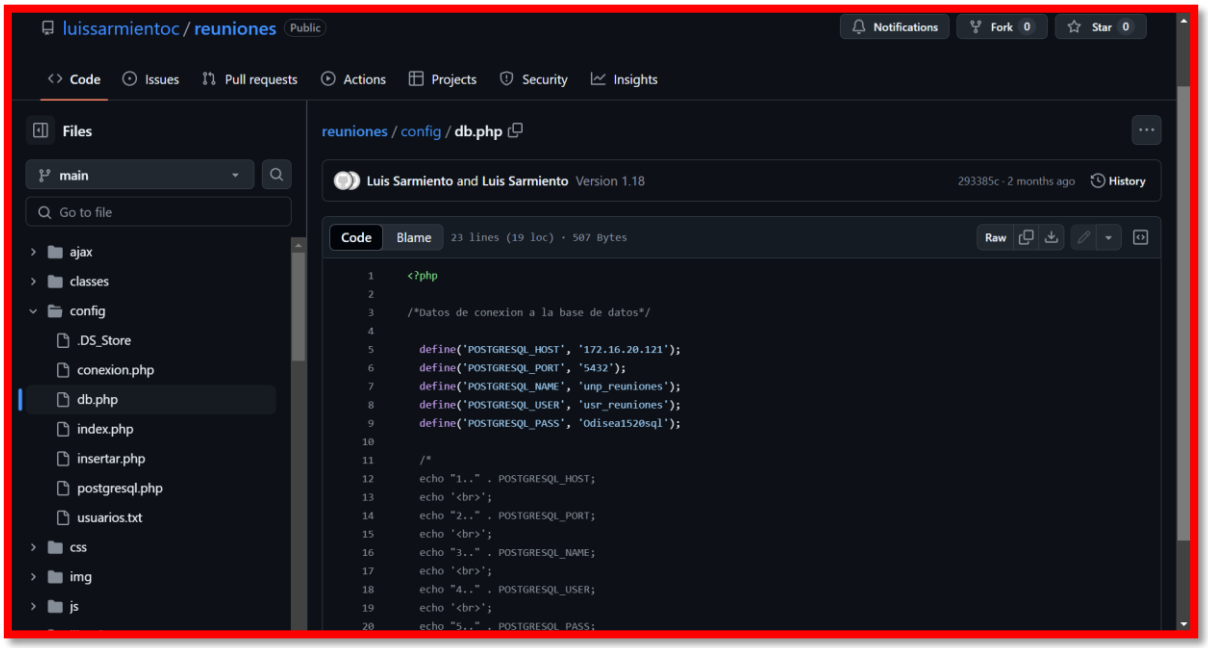


Imagen 12. Archivo db.php con credenciales expuestas

Ruta del archivo

https://github.com/luissarmientoc/reuniones/blob/main/config/db.php

Esta imagen muestra el archivo insertar.php, donde también se exponen las mismas credenciales de la base de datos encontradas en el archivo db.php. Además, se muestran nombres de usuario (robert) y contraseñas (Ilidias52psql) dentro del código, lo que evidencia una mala gestión de credenciales en el código fuente. Esta información puede ser explotada fácilmente por atacantes.

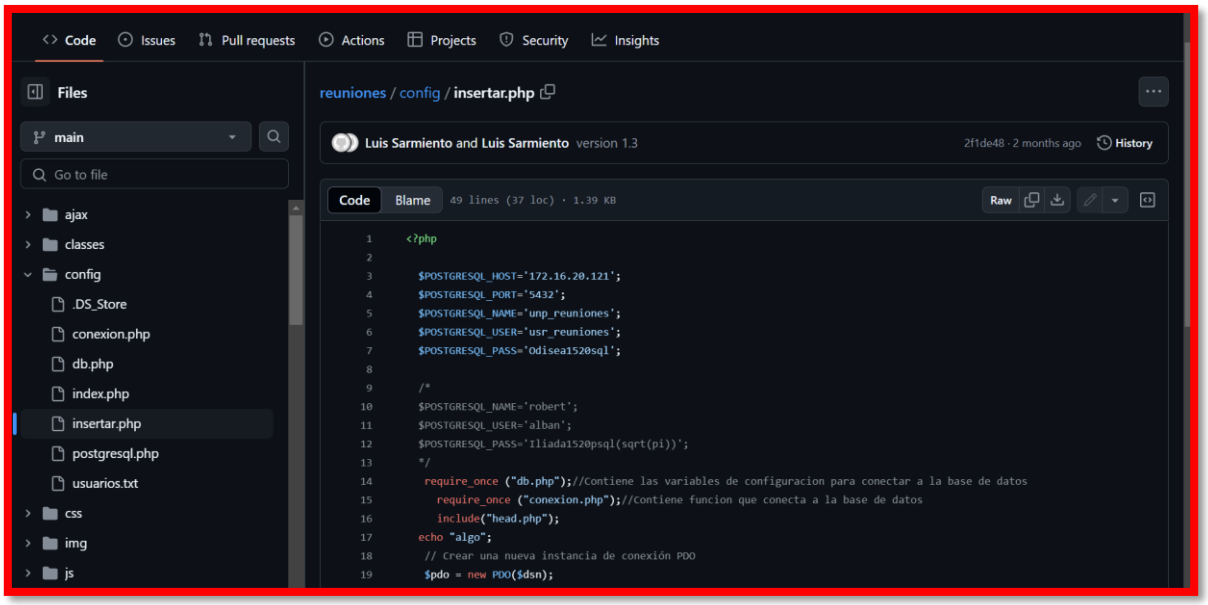


Imagen 13. Credenciales expuestas en archivo insertar.php

Ruta del archivo

https://github.com/luissarmientoc/reuniones/blob/main/config/insertar.php

La imagen muestra el archivo usuarios.txt, el cual contiene una lista de usuarios junto con sus nombres, apellidos, correos electrónicos, y nombres de usuario. Entre los correos expuestos, se encuentran algunos con el dominio oficial (@unp.gov.co), lo que podría comprometer la privacidad de los empleados y facilitar ataques de spear-phishing o suplantación de identidad.

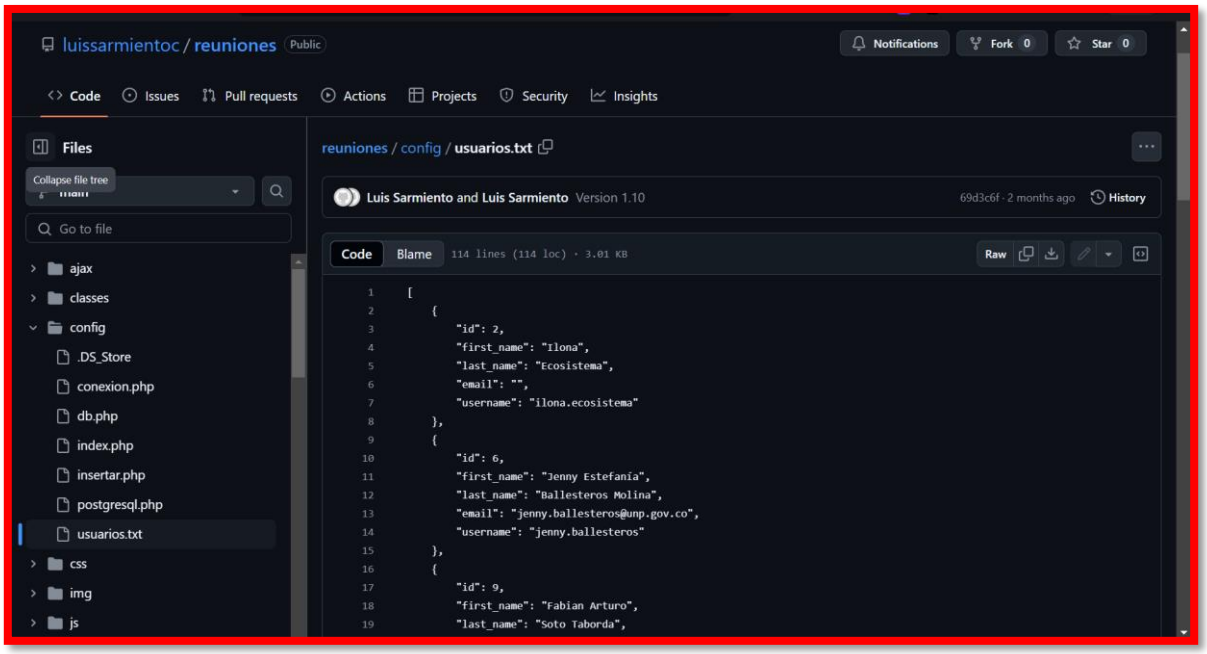


Imagen 14. Usuarios identificados en archivo usuarios.txt

Ruta del archivo
https://github.com/luissarmientoc/reuniones/blob/main/config/usuarios.txt

Riesgos Asociados

1. **Acceso no autorizado a sistemas**: La publicación de credenciales y configuraciones puede facilitar el acceso directo a las bases de datos y otros servicios, comprometiendo la confidencialidad e integridad de la información.
2. **Ataques dirigidos**: Con la información de las IPs y rutas internas, un atacante podría orquestar ataques dirigidos hacia los sistemas de la organización, como intentos de escalada de privilegios o ejecución de código malicioso en entornos sensibles.
3. **Explotación de vulnerabilidades**: Al tener acceso al código fuente completo, un atacante puede analizar posibles vulnerabilidades en las aplicaciones, como inyecciones de SQL, debilidades en la autenticación o mal manejo de excepciones.

Recomendaciones

1. **Cambio inmediato de credenciales expuestas**: Todas las contraseñas y usuarios comprometidos deben ser reemplazados de inmediato. Se recomienda también implementar autenticación multifactor (MFA) para mitigar riesgos futuros.
2. **Repositorios privados**: Todos los repositorios que contengan información sensible deben ser convertidos en privados, garantizando que solo los usuarios autorizados tengan acceso. Además, se deben auditar los cambios históricos en los repositorios para identificar posibles exposiciones previas.
3. **Auditorías de código y revisiones de seguridad**: Es crucial establecer un proceso continuo de revisión del código fuente para detectar posibles vulnerabilidades antes de que se publiquen en entornos de producción. El uso de herramientas de análisis estático y dinámico ayudará a identificar problemas de seguridad de forma temprana.
4. **Capacitación en seguridad**: Es fundamental proporcionar capacitación continua a los desarrolladores y personal técnico sobre las mejores prácticas de seguridad, incluyendo el manejo de credenciales, el uso de repositorios y el cifrado de datos sensibles. Las contraseñas y variables críticas deben almacenarse en entornos protegidos, como archivos .env o sistemas de gestión de secretos.

Credenciales Filtradas en Internet

El objetivo de este análisis es identificar las credenciales de empleados de la organización que han sido filtradas en internet a lo largo de los últimos 9 meses en diversos foros de hacking y grupos de Telegram. Este proceso busca no solo alertar sobre las posibles vulnerabilidades de seguridad asociadas con estas filtraciones, sino también permitir a la organización tomar medidas proactivas para mitigar el riesgo de futuros compromisos. El análisis se centra en entender el origen de las filtraciones, determinar su impacto potencial y establecer un marco sólido para la política de contraseñas. A partir de este análisis, se pretende diseñar e implementar una estrategia integral de monitoreo de usuarios, reforzar las políticas de contraseñas y establecer mecanismos preventivos que limiten futuras exposiciones de datos confidenciales, garantizando así una mejor protección de la organización frente a ataques avanzados.

Metodología

El análisis se realizó mediante un monitoreo exhaustivo de foros de hacking y grupos de Telegram donde se comparten habitualmente credenciales filtradas. Se utilizaron motores especializados en detección de filtraciones, los cuales escanean continuamente estos sitios y plataformas. Estos motores permiten identificar las credenciales asociadas a la organización y generar alertas automáticas cuando se detectan datos sensibles. Una vez recolectada la información, se procedió a un análisis detallado de las fuentes de la filtración, enfocándose en cómo los actores maliciosos obtuvieron las credenciales, para entender mejor los métodos de exfiltración utilizados.

Hallazgos

Se identificaron 54 credenciales filtradas, provenientes principalmente de infecciones con **RATs (Remote Access Trojans)**. Los RATs permiten a los atacantes capturar credenciales directamente desde los dispositivos comprometidos, lo que representa un peligro grave, ya que este método es más sigiloso y difícil de detectar que ataques de fuerza bruta o phishing. La obtención de estas credenciales a través de RATs sugiere que en algún momento los usuarios afectados fueron víctimas de infecciones en sus dispositivos, lo que significa que no solo sus credenciales estaban comprometidas, sino también el entorno en el que operaban. En varios casos, se encontraron múltiples credenciales pertenecientes a los mismos usuarios, lo que sugiere que estos individuos han sido repetidamente atacados, probablemente por la reutilización de contraseñas o falta de prácticas seguras de seguridad.

Tabla de Credenciales Identificadas

*Análisis de los últimos 9 meses

Usuario	Contraseña
victor.martinez	Plat22.Vic
santander.negrete.co	Monteria*2023**
santa.negrete.co	MaGXv29+HkD7Gph
santander.negrete.co	MiEstudio.91
wilmar.arias	Dios9899**
wilmar.arias	Unp123456
angela.torres	Adela1030*
diego.guarin	Unp123456**
luz.canchica	Proteccion2018
nubia.diaz	Unp2020**
nubia.diaz	Unp20202020**
omar.paredes	SERunp20202018
dulfary.restrepo	Unp9514**
wilmar.arias	Dulwil9514**
dulfary.restrepo	Dios9899**
dulfary.restrepo	Unp123456
diego.guarin	Colombia2020
yarlenny.vargas	Dani1811
leidy.lmbachi	Unp2023*
fredy.valencia	Security57
fredy.valencia	Ejercitos2022*
fredy.valencia.co	rOKEFELER23A
fredy.valencia.co	Saturno66
fredy.valencia.co	Neptuno#11
fredy.valencia.co	nEPTUNO#11
manuel.pinzon	#St4rTr3k9
jennifer.arenas.co	Enero4242*
jennifer.arenas	Enero2020*
jennifer.arenas.co	Yo1073172624.
jennifer.arenas.co	Jennifer.Arenas18
jennifer.arenas.co	95120808177
u0902226.co	Jennifer.Arenas18
nohora.gutierrez	Colombia1010*
nohora.gutierrez	Junio1026
nohora.gutierrez	Junio3128
Jose.perdomo.co	MEunp2023*@12
kattiarenas1995	besosdeballena
santa.negrete.co	miscompras#20
brayan.aguas	Estudios2021*
leida.rodriguez.co	ANALISTA25*

juan.cardozo	C@rdozo1303
carlos.ruano	Patoyuyis42435.
rolando.montero	Superacion2415
carlos.mina	Abril2020*
jairo.amezquita	Jaas201801*
josefina.pimiento	Escolta102020*
desplazamiento.zona1	Colombia2018
andres.aguilar@unp.gov.co	Caro2012.*
martin.pinzon	Pacaylala29
Martin.pinzon	Minino329*
martin.pinzon	MEPCunp22*
martin.pinzon	Mike329*
angie.cano	Jero0718*
Ivan.vides	Es2022.*

Impacto

El hecho de que las credenciales filtradas provengan de infecciones con RATs presenta un riesgo significativo para la organización. Este tipo de ataque no solo implica la exposición de las contraseñas, sino que también sugiere que los sistemas o dispositivos de los usuarios comprometidos estuvieron, o podrían estar, completamente bajo el control de un actor malicioso. Los RATs permiten el monitoreo continuo de las actividades de un usuario, lo que incluye la captura de teclas (keylogging), el control remoto y la exfiltración de datos adicionales que van más allá de las credenciales, incluyendo información confidencial de la organización. Esto es especialmente preocupante para la UNP, ya que los dispositivos comprometidos podrían haber sido utilizados para acceder a sistemas críticos y datos sensibles, poniendo en peligro la seguridad de la organización y sus operaciones. La presencia de múltiples credenciales filtradas para ciertos usuarios indica una debilidad significativa en la gestión de contraseñas y, posiblemente, una falta de conciencia o control sobre los dispositivos afectados, lo que aumenta el riesgo de futuros compromisos si no se abordan de manera adecuada.

Recomendaciones

1. **Revisión Exhaustiva de Dispositivos Comprometidos**: Se debe realizar una inspección profunda de los dispositivos asociados a las cuentas filtradas, especialmente en los casos donde se encontraron múltiples credenciales. Esto incluye análisis forense de los Endpoints para detectar infecciones de RATs activas o pasadas, asegurando que los sistemas estén completamente limpios y fuera de peligro.
2. **Política de Contraseñas Robusta**: Implementar políticas estrictas de cambio de contraseñas que prohíban la reutilización de contraseñas y exijan el uso de contraseñas fuertes y únicas para cada servicio. Se recomienda habilitar la autenticación multifactor (MFA) en todas las cuentas críticas para dificultar aún más el acceso no autorizado en caso de futuras filtraciones.
3. **Monitoreo y Detección Continua**: Fortalecer el monitoreo de credenciales filtradas mediante herramientas de vigilancia continua de la dark web, deep web, y foros de hacking. Implementar alertas que informen automáticamente cuando credenciales asociadas a la organización se detecten en plataformas públicas o privadas.
4. **Capacitación de Concienciación en Seguridad**: Proveer capacitaciones periódicas a los empleados sobre la importancia de la higiene de seguridad digital, incluyendo la gestión de contraseñas, la detección de posibles infecciones de malware, y la importancia de reportar cualquier actividad sospechosa en sus dispositivos.
5. **Implementación de un Sistema DLP (Data Loss Prevention)**: Implementar un sistema de prevención de pérdida de datos que detecte y bloquee intentos de exfiltración de información confidencial. Esto puede prevenir que los atacantes extraigan información adicional en caso de que algún dispositivo sea comprometido nuevamente.
6. **Fortalecimiento del Monitoreo de Red y Endpoint**: Asegurarse de que el EDR (Endpoint Detection and Response) y los sistemas de monitoreo de red estén optimizados para detectar actividades relacionadas con RATs, como conexiones no autorizadas a servidores de comando y control (C2), comportamientos anómalos, o actividad sospechosa de exfiltración de datos.

Principales Vectores de Ataque Identificados

Vector	Probabilidad	Impacto
Vulnerabilidades Críticas (ID-001-ID-006, ID-049)	Alta	Crítico
Ataques vía Spear-Phishing APT-C-36	Media	Crítico
Credenciales filtradas en Internet	Media	Crítico
Credenciales filtradas en repositorios de GitHub	Media	Crítico

Ruta de Trabajo

Gestión de Vulnerabilidades

1. Corrección de vulnerabilidades identificadas en los sistemas con mayor impacto de acuerdo con la priorización de impacto.
2. Revisión del proceso para gestión de vulnerabilidades.
3. Revisión del proceso de *actualización de componentes tecnológicos* de la organización para garantizar la correcta aplicación de parches de seguridad a los sistemas actuales en la infraestructura tecnológica.
4. Establecer KPIs para medir la eficiencia interna del proceso de gestión de vulnerabilidades.

Protección de Endpoints

1. Despliegue completo del EDR en todos los Endpoints de la organización
2. Verificación de servicios y/o puertos innecesarios en equipos de usuarios finales.
3. Generar un reporte de las alertas en categoría crítica y alta de los últimos 30 días en los equipos que cuenten con el EDR.

Seguridad en Aplicaciones Web

1. Validar la existencia o creación de un proceso para auditoria técnica de requerimientos para desarrollo de aplicaciones web por parte de terceros.
2. Validar aplicaciones en arquitecturas Cloud (AWS, Azure, GC, etc) para incluirlas dentro de los procesos de auditorías de seguridad.
3. Generar un reporte de las alertas generadas por el WAF en los últimos 30 días y realizar el despliegue completo sobre todas las aplicaciones web (si aplica).

Respuesta a Incidentes y Backups

1. Validar las políticas de backup en la organización sobre los sistemas misionales para determinar su eficiencia actual de acuerdo con la criticidad de los datos.
2. Realizar un simulacro de restauración para comprobar la eficiencia de la política actual.
3. Revisión del plan de respuesta a incidentes informáticos frente a los principales escenarios de ataque expuestos en este informe técnico para determinar el nivel de preparación teórico.

Recomendaciones de Implementación

Seguridad de la Información

1. Implementar una política de contraseñas estricta mediante el uso de gestores de contraseñas para fortalecer el acceso a las identidades digitales corporativas y evitar el uso de contraseñas genéricas
2. Implementación de una plataforma de tipo DLP para monitorear el flujo de información y/o datos sensibles en la organización.
3. Implementación de una plataforma automatizada para la realización de copias de seguridad sobre los sistemas categorizados como críticos.
4. Definir políticas claras de retención de datos, asegurando que se conserven el tiempo necesario para cumplir las normativas aplicables o necesidades operacionales.