PRÁCTICAS DE SEGURIDAD INFORMÁTICA PARA SITIOS WEB

Agencia de Gobierno Electrónico y Tecnologías de Información y Comunicación

Centro de Gestión de Incidentes Informáticos

Versión 2

AGETIC
Digitalizando Bolivia

ESTADO PLURINACIONAL DE BOLIVIA
MINISTERIO DE LA PRESIDENCIA

ÍNDICE

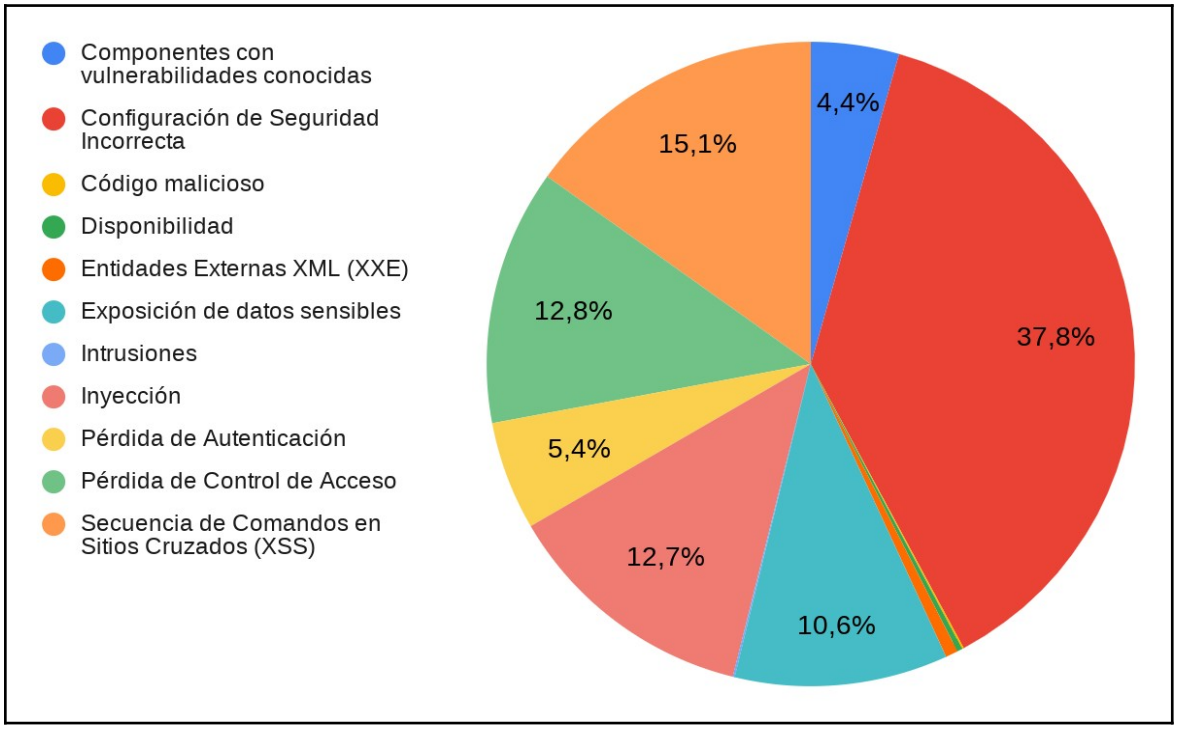
Introducción

La Agencia de Gobierno Electrónico y Tecnologías de Información y Comunicación tiene la función de evaluar la seguridad de los sistemas de información de las entidades públicas, comunicar y otorgar información acerca de incidentes y vulnerabilidades, además de realizar otras tareas orientadas a la mejora de la seguridad de la información de las entidades del sector público, funciones que están establecidas en Decreto Supremo 2514.

Además la norma establece obligaciones en materia de seguridad informática a las entidades del sector público sobre la notificación de incidentes y la solución de vulnerabilidades.

El Centro de Gestión de Incidentes Informáticos comunica y otorga información acerca de incidentes y vulnerabilidades identificadas mediante mecanismos de monitoreo, evaluaciones de seguridad y reportes de organismos de similar función. Los casos gestionados hasta el 2022 muestran un alto porcentaje relacionado a configuraciones de seguridad incorrectas como se puede apreciar en los siguientes gráficos.

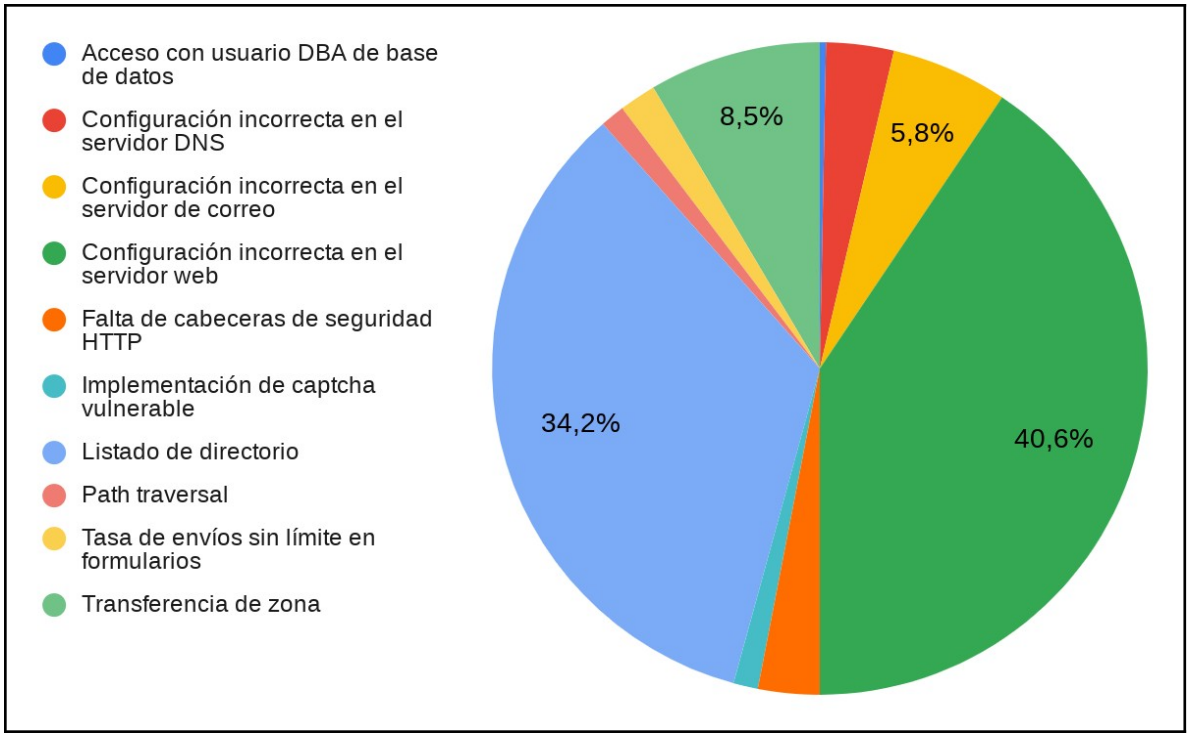
Categoría de vulnerabilidades gestionadas



Se evidencia un porcentaje alto de errores de configuración en el servidor web que derivaron en incidentes de seguridad, producto del uso de contraseñas débiles o por

defecto, software de servidor y aplicación desactualizada, uso de componentes con vulnerabilidades, entre otras.

Vulnerabilidades por configuración de seguridad incorrecta



Con el objetivo de prevenir y reducir la ocurrencia de incidentes informáticos en entornos web, se elabora el presente documento que expone prácticas de seguridad para sistemas de información en producción enfocado principalmente en sitios web. Las prácticas descritas no limitan la adopción e implementación de otras acciones orientadas a la seguridad de la información.

El documento cubre prácticas de seguridad del sistema operativo, software de servidor web, gestor de base de datos, sistemas de administración de contenidos y aplicaciones web en general.

1. Seguridad del sistema operativo

1.1. Instalar actualizaciones de seguridad

Verificar con regularidad si existen actualizaciones para el sistema operativo disponibles en sus repositorios oficiales, con especial atención a parches de seguridad de riesgo crítico o alto. En sistemas críticos se recomienda aplicar las actualizaciones en ambiente test para posteriormente aplicar en entorno de producción.

1.2. Controles de acceso y autenticación

Mantener el control de las cuentas de usuario habilitadas en el sistema, así como los privilegios de cada una, las altas y bajas de usuarios deben estar documentadas y autorizadas. El acceso a sistemas críticos debería estar autorizado por direcciones IP y MAC.

Las cuentas no deben ser compartidas por sus titulares, ya que elimina la responsabilidad y aumenta el riesgo de accesos y acciones no autorizadas en el sistema.

Utilizar un par de claves como método de autenticación en lugar de la contraseña, restringir la autenticación del usuario root y limitar intentos de inicio de sesión fallidas.

Las conexiones remotas deben realizarse por una red privada virtual y protocolos seguros como SSH.

1.3. Deshabilitar/desinstalar servicios innecesarios

Instalar únicamente software necesario para el despliegue de la aplicación, sitio o servicio web, puesto que la falta de control sobre el software instalado podría exponer a ataques y explotación de vulnerabilidades. También es recomendable deshabilitar protocolos, servicios para compartir archivos.

2. Seguridad del servidor web

2.1. Instalar actualizaciones de seguridad

Instalar con regularidad actualizaciones de seguridad, con especial atención de aquellas que impliquen la ejecución remota de código o cuenten con puntuación CRÍTICO/ALTO dentro de la valoración Common Vulnerability Scoring System (CVSS) y publicado en la base de Common Vulnerabilities and Exposures (CVE).

Cada proveedor del software de servidor web publica regularmente avisos de parches de seguridad para sus productos, al cual se recomienda suscribirse para estar al día con avisos de actualizaciones:

- Para apache: http://httpd.apache.org/lists.html#http-announce
- Para productos de Microsoft: https://msrc.microsoft.com/update-guide
- Para NGINX: http://nginx.org/en/security_advisories.html
- Para productos Oracle: https://www.oracle.com/security-alerts/

Es importante considerar las recomendaciones del proveedor respecto al procedimiento a seguir para aplicar correctamente las actualizaciones, para servicios críticos, es recomendable probar el parche en ambiente test y luego en producción.

2.2. Deshabilitar el listado de directorios

Esta configuración es propia del servidor Apache2 y viene por defecto en la instalación, y permite listar los directorios y archivos del sitio web, permitiendo navegar, descargar y visualizar el contenido de archivos desde el directorio raíz del sitio, estos podrían contener datos sensibles y confidenciales como ser: nombres de usuarios, contraseñas, parámetros de configuración. Incluso se ha identificado copias de respaldo en el mismo directorio.

Para deshabilitar el listado de directorios, consultar el numeral 2.1 del Anexo 1.

2.3. Quitar archivo info.php y deshabilitar la función phpinfo()

Aplicable para soluciones desarrolladas bajo el lenguaje de programación PHP, una práctica que se usa en entorno de desarrollo es visualizar las características de php en un archivo generalmente de nombre info.php u otro, a efectos de verificar la correcta instalación y configuración de PHP en el sistema. Sin embargo este archivo debe ser removido del ambiente de producción ya que devela información de las versiones, el sistema operativo y otros datos que podrían ser usados por un actor de amenaza.

Otra acción de asegurar la instalación de php que se ejecuta en el servidor web es deshabilitar la función phpinfo() del sistema, propiamente en el archivo de configuración "php.ini".

Consultar el numeral 2.2 del Anexo 1 para deshabilitar esta función.

2.4. Quitar/restringir directorios y archivos de control de versiones ".git", ".svn"

Una mala práctica en entornos de producción es exponer directorios ".git", ".svn" que contienen datos de configuración del control de versiones del código, repositorio. Entre algunos datos que se podrían exponer están nombres de usuarios, rutas, parámetros de configuración, historial de commits, entre otros.

Por ello es importante quitar o restringir su acceso. Para ver cómo restringir el acceso a estos directorios, consultar el numeral 2.3 del Anexo 1.

2.5. Restringir el acceso a archivos de configuración ".env", "printenv.pl"

El archivo .env es usado en proyectos para el almacenamiento de variables de entorno en ambiente development, test, production y otros. Contiene datos como nombres de usuario y contraseña para la conexión con la base de datos, tokens de acceso, cuentas de correo electrónico, direcciones IP, nombres de hosts y otros necesarios para la funcionalidad del software.

Los datos expuestos podrían ser usados para ataques específicos, con impactos sujetos a la sensibilidad de la información expuesta, por ello es importante restringir el acceso a estos y a archivos como ser "composer.json", "web.config" y "main.yml".

Consultar el numeral 2.4 del Anexo 1 para restringir el acceso a estos archivos.

2.6. Configurar el registro de logs de acceso y error del servidor web

Una de las configuraciones más importantes en el servidor web es el registro de logs, que permitirán registrar datos relativos a la funcionalidad del servidor, las solicitudes atendidas y toda información de actividad del servicio publicado, así como problemas que hayan podido ocurrir durante la operación del servicio.

Los registros son importantes para monitorear el rendimiento del servidor y en casos de incidentes, son un factor esencial para determinar las posibles causas, correlacionar eventos y otros datos. Cada software de servidor tiene configuraciones diferentes para el registro de logs.

El numeral 2.5 del Anexo 1 muestra la configuración de logs para el software de servidor web más usado, Apache2.

2.7. Deshabilitar el despliegue de errores

Las aplicaciones con depuración de errores habilitadas en entorno de producción podrían exponer variables de entorno, rutas, direcciones IP y código fuente producto de excepciones no controladas.

En aplicaciones desarrolladas en PHP es posible deshabilitar el despliegue de errores al usuario con ajustes en la configuración del lado del servidor. Para deshabilitar esta característica consultar el numeral 2.6 del Anexo 1.

2.8. Quitar contenido por defecto

Durante el despliegue de un sitio web se hace uso de distintos componentes, entre ellos, software del servidor web con directorios por defecto, manuales de instalación y uso; software de aplicación con funcionalidades de ejemplo, incluso la reutilización de código de terceros que luego de ser probada no se elimina.

Es importante eliminar todo archivo, código, módulo, complemento o componente que no se utilice, puesto que el mismo podría exponer información de versiones del software utilizado, puertas traseras, vulnerabilidades conocidas.

2.9. Realizar copias de seguridad

Otra de las acciones más importantes para una administración segura de un sitio, aplicación o servicio web es realizar copias de seguridad automatizadas de archivos funcionales y de configuración. La periodicidad estará en función de la criticidad de la información.

Es importante que las copias se resguarden en medios y ubicaciones distintas, las copias deben ser probadas regularmente a efectos de verificar que las mismas se generan y se restauran correctamente.

Uno de los errores más comunes es dejar las copias en el mismo servidor o peor aún, en el directorio raíz del sitio.

Las copias permitirán responder en tiempos y acciones ante incidentes de seguridad.

2.10. Deshabilitar la información de versión del servidor web y sistema operativo

La versión del software del servidor puede ser usada para verificar si el servidor tiene vulnerabilidades conocidas y si existen exploits públicos que podrían ser usados para comprometer el servidor, servicios y aplicaciones que se ejecutan en el sistema.

Quitar información de la versión del servidor web impedirá que se revele la versión del sistema operativo. Para deshabilitar la versión del servidor web Apache2 y Nginx consultar el numeral 2.7 del Anexo 1.

2.11. Habilitar cabeceras HTTP de seguridad

Agregar capas de seguridad al servidor web reducirá el riesgo de sufrir ataques, por ello es importante implementar directivas de seguridad en las cabeceras HTTP del servidor, entre las cabeceras más importantes se encuentran: Strict-Transport-Security (HSTS), Content-Security-Policy (CSP), X-Frame-Options y X-XSS-Protection.

Para configurar las cabeceras en el servidor web Apache2 consultar el numeral 2.8 del Anexo 1.

2.12. Instalar y configurar certificado SSL/TLS

Instalar y configurar el certificado de seguridad permitirá proteger la confidencialidad de datos de autenticación y otra información sensible que se intercambie entre el cliente y el servidor mediante el protocolo seguro HTTPS.

La alternativa para la implementación de certificados gratuitos la emite la autoridad de certificación Let's Encrypt.

Consultar el numeral 2.9 del Anexo 1 para la instalación y configuración del certificado de seguridad en Apache2.

3. Seguridad del sistema gestor de base de datos

3.1. Actualizaciones de seguridad

El sistema gestor de base de datos debe ser actualizado regularmente, con atención especial a parches de seguridad de severidad crítico y alto. La actualización debe seguir las recomendaciones y procedimientos establecidos por el proveedor.

En sistemas críticos la aplicación de parches debe ejecutarse primeramente en ambiente test y luego el pase a producción.

3.2. Control de acceso y autenticación

Limitar el acceso al servicio mediante listas de control de acceso, el servicio no debe ser publicado a internet, sólo usuarios y hosts autorizados deberían acceder al servicio. Del mismo modo a aplicaciones web que gestionan la administración de bases de datos como phpmyadmin, phppgadmin o similares.

Usar contraseñas fuertes y únicas para todos los usuarios, es importante mantener el registro, control de todos los usuarios con sus respectivos permisos de operación y autenticación, diferenciando si es local o remota.

3.3. Roles y privilegios

Establecer roles y privilegios a cada usuario a nivel del sistema gestor, bases de datos, tablas, vistas y otros. De ningún modo se debería establecer permisos de alto nivel a usuarios para la conexión con la aplicación o usar la cuenta de súper administrador predeterminado, puesto que un ataque exitoso a la aplicación podría hacer mal uso de los privilegios y comprometer todas las bases de datos.

3.4. Registro de logs

Es importante verificar que el registro de logs está habilitado, en función del sistema gestor de base de datos, los datos que podrían ser registrados como eventos son la creación de usuarios, bases de datos, tablas, vistas, funciones y otros datos de interés como la conexión origen, dirección IP, fecha y hora del evento.

Los registros de auditoría permitirán identificar, correlacionar, analizar actividad no autorizada junto con problemas de integridad de datos.

3.5. Copias de seguridad

La realización de copias de seguridad en bases de datos es crítico, ya que en función de la importancia de los datos, se debe establecer la frecuencia y redundancia de las copias.

Las copias deben resguardarse en medios y ubicaciones diferentes al sistema gestor de base de datos, mismas que deben probarse para verificar su restauración correcta. Entre

las alternativas libres está Backup Archiving Recovery Open Sourced - BAREOS que gestiona las copias de respaldo de manera automatizada.

4. Seguridad en aplicaciones web

4.1. Deshabilitar el modo depuración

Las aplicaciones desplegadas en entorno de producción deben contar con la configuración del modo depuración deshabilitado, puesto que un error o excepción no controlada causará que se muestre información detallada del problema, versiones usadas, rutas, direcciones IP, entre otros. Es importante que el personal de desarrollo verifique esta configuración antes de pasar el sitio, aplicación o servicio web a producción.

4.2. Registro de logs

Las aplicaciones desarrolladas deben contar con el registro de eventos, mismas que deben ser configuradas en entornos de producción para registrar acciones de operación efectuadas en el sistema, como ser: fecha y hora, acción, usuario, y otros datos relevantes y necesarios para correlacionar datos en investigación de incidentes en curso o pasados.

4.3. Generar copia estática

Esta es una medida de respuesta ante incidentes que tienen la finalidad de mantener la disponibilidad del sitio web de manera temporal. Generar una copia estática implica el uso de herramientas como “HTTrack” que obtiene archivos javascript, hojas de estilo, documentos y otros del servidor.

Para realizar la copia estática del sitio consultar el numeral 2.11 del Anexo 1.

5. Monitoreo de servicios y recursos

Monitorear el estado de los recursos hardware del servidor, así como el estado de servicios, permitirá prevenir y responder adecuadamente ante problemas o ataques al sitio o aplicación web. Se podría considerar más de un monitoreo para los servicios críticos, el primero instalado en infraestructura de la institución con una visibilidad mayor respecto al uso y estado de los servicios, puesto que podrían instalarse agentes en cada host a monitorear; el segundo monitoreo (respaldo) como servicio gratuito en la nube, con funcionalidades básicas que no requiere la instalación de agentes y basado en verificación de estados HTTP del servidor web.

Entre las alternativas libres para instalación en infraestructura de la institución están Zabbix para el monitoreo de redes, y goaccess para análisis de registros del servidor web Apache2 y Nginx por terminal o interfaz http.

6. Seguridad en sistema de administración de contenidos

Sitios web desarrollados bajo sistemas de administración de contenidos sufrieron ataques de defacement, malware, phishing, entre otros que se originaron en gran medida por la falta de actualización, uso de componentes con vulnerabilidades, contraseñas débiles .

En ese sentido, a continuación se brindan prácticas de seguridad que se deben implementar para reducir la superficie de exposición a ataques, junto a guías técnicas específicas para Sistemas de Administración de Contenidos ampliamente usados:

- Guía de seguridad Joomla - Anexo 2
- Guía de seguridad Drupal - Anexo 3
- Guía de seguridad Wordpress - Anexo 4

6.1. Desplegar ambiente de pruebas

Implementar un ambiente de pruebas o test, que será una réplica del sitio en producción, a efectos de probar las actualizaciones de seguridad y garantizar que la disponibilidad en ambiente de producción no se vea afectada.

6.2. Instalar actualizaciones de seguridad

Esta práctica es fundamental en sistemas de administración de contenidos, regularmente se publican actualizaciones de seguridad para el sistema base (core) y componentes (plugins). Un alto porcentaje de sitios comprometidos tienen como causa la falta de actualización del core y plugins.

Las actualizaciones deben realizarse en función a las notas, pasos y requisitos de instalación en ambiente test, en algunos CMS es posible habilitar actualizaciones automáticas. Se recomienda suscribirse a listas de avisos del CMS para estar al día con notificaciones de nuevas actualizaciones.

6.3. Usar plugins y componentes de fuentes confiables

Otro de los factores que inciden en el compromiso de sitios web es el uso de plugins y componentes de fuentes no confiables, es decir que no proceden del sitio web oficial del

CMS. Investigaciones que realizó el CGII en incidentes, identificó que estos componentes contenían puertas traseras.

Es importante usar plugins descargados desde los sitios web oficiales del CMS, algunos otorgan la verificación de seguridad para complementos. De ningún modo usar plugins de paga pirateados de fuente desconocida.

6.4. Realizar copias de seguridad

Algunos CMS facilitan la realización de copias de respaldo con herramientas integradas en la instalación o como complementos, en cualquier caso, al tratarse de sitios web es fundamental realizar copias de seguridad al menos una vez por día, tanto de archivos del sitio web como de la base de datos. Estas copias deben resguardarse en ubicaciones diferentes al servidor web.

Contar con copias de respaldo hará que el proceso de respuesta ante un incidente sea menos complejo, porque se podrá restaurar una copia no comprometida y sobre el mismo realizar acciones correctivas.

6.5. Controles de acceso y autenticación

Las credenciales de acceso no deben ser compartidas y mantener el control de qué usuarios tienen este permiso, evitar el uso de nombres de usuario “admin”, “administrador”. Revisar frecuentemente si existen usuarios nuevos, revisar sus roles y permisos.

Todas las cuentas de usuario deben usar contraseñas fuertes, algunos CMS implementan esta funcionalidad como obligatoria.

Los mensajes de error en la autenticación deben ser genéricos, no se debe revelar si el nombre o contraseña son incorrectas.

6.6. Protección contra ataques automatizados

El formulario de autenticación debe contar con la protección captcha, esto evitará ataques de fuerza bruta o diccionario que busque obtener nombres de usuario o contraseña válidos, limitar intentos de inicio de sesión fallidos.

6.7. Remover/deshabilitar contenido por defecto

Eliminar el contenido por defecto, como ser temas, plugins, páginas, directorios y archivos que no se usan. En general cada CMS viene con una guía de instalación que indica los ajustes de seguridad que se deben realizar luego de una instalación exitosa.

7. Alertas y avisos de seguridad

Se recomienda la suscripción a canales de avisos y alertas de seguridad genéricas o específicas, en función de la tecnología que se usa en la institución. El Centro de Gestión de Incidentes Informáticos emite alertas y avisos de seguridad respecto a nuevas amenazas, vulnerabilidades recientemente descubiertas, campañas de explotación de vulnerabilidades de tipo día cero, que son comunicadas a responsables de seguridad de la información y contactos técnicos de las entidades del sector público.

Las alertas y avisos tienen la finalidad de prevenir incidentes, siendo responsabilidad de la institución tomar acción frente al aviso o alerta si corresponde.

- CERT de Estados Unidos: https://www.cisa.gov/uscert/ncas/alerts
- SANS Risk: https://www.sans.org/newsletters/at-risk/
- CGII: https://www.cgii.gob.bo/es/alertas-de-seguridad

ANEXOS

Anexo	Código	Descripción del Anexo
1	ANEXO 1	Guía de seguridad para servidores web
2	ANEXO 2	Guía de seguridad Joomla
3	ANEXO 3	Guía de seguridad Drupal
4	ANEXO 4	Guía de seguridad Wordpress

CONTROL DE CAMBIOS

Versión	Fecha	Descripción de las modificaciones
1	12/01/2021	Versión nueva del documento.
2	05/01/2023	Modificación de la estructura y contenido: - Prácticas de seguridad sistema operativo, servidor web, gestor de base de datos, aplicación web, monitoreo de servicios y recursos, seguridad en sistemas de administración de contenidos, alertas y avisos de seguridad. - Se eliminó el Anexo 5.

ANEXO 1

Guía de seguridad para servidores web

1. Introducción

El software del servidor web es propenso a ataques producto de vulnerabilidades descubiertas o configuraciones por defecto que se hayan dejado con la instalación.

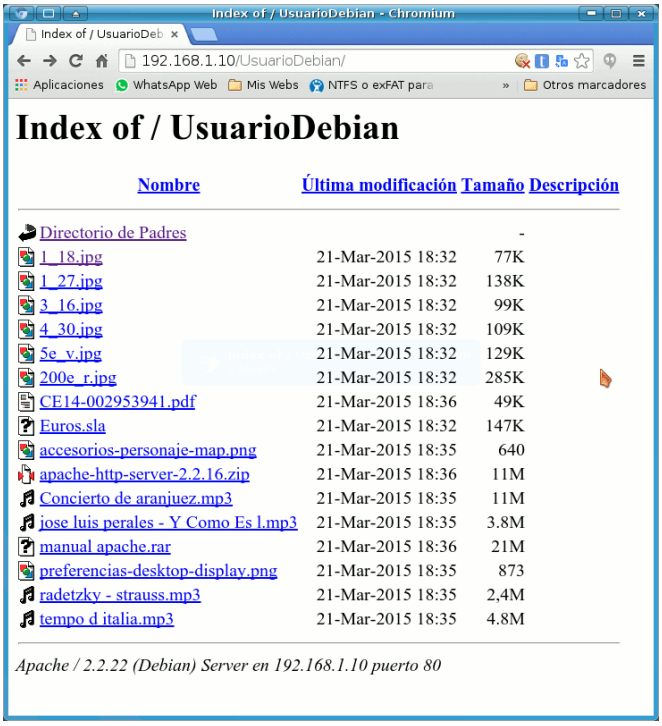
Por estas razones se presentan las prácticas de seguridad que debemos seguir para contar con un software de servidor web más seguro.

2. Prácticas de seguridad

Para solucionar las vulnerabilidades más comunes del software de servidor web, se recomienda implementar las siguientes prácticas de seguridad.

2.1. Deshabilitar el listado de directorios

Esta configuración es propia del servidor Apache2 y viene por defecto en la instalación, permite listar los directorios y archivos del sitio web, permitiendo navegar, descargar y visualizar el contenido de archivos desde el directorio raíz del sitio, que podrían contener datos sensibles o confidenciales como ser nombres de usuarios, contraseñas, copias de seguridad y archivos de configuración.



Las siguientes configuraciones fueron realizadas en Apache 2.4 mediante un usuario con privilegios sudo. Dependiendo del caso se puede elegir una de las siguientes opciones para deshabilitar el listado de directorio.

- Deshabilitar el módulo autoindex.
- Deshabilitar a través del archivo de configuración del sitio.
- Deshabilitar a través del archivo htaccess.

2.1.1. Deshabilitar el módulo autoindex

Deshabilitar la funcionalidad autoindex a nivel global:

sudo a2dismod autoindex

Después de ejecutar el comando, se mostrará el mensaje de advertencia el cual se tiene que responder con la siguiente frase:

To continue type in the phrase 'Yes, do as I say!' or retry by passing '-f': ***Yes, do as I say!***

Reiniciar el servidor.

sudo systemctl restart apache2.service

2.1.2. Deshabilitar a través del archivo de configuración del sitio

Este método deshabilita esta funcionalidad solo para el sitio en cuestión. Por ejemplo si se tiene el sitio www.sitio-de-prueba.com con el archivo de configuración (virtualhost) `sitio-de-prueba.conf`.

Agregar en el archivo la siguiente directiva:

```
<VirtualHost *:80>
.....
<Directory /var/www/sitio-de-prueba>
    Options -Indexes
</Directory>
```

.....

</VirtualHost>

Guardar y recargar la configuración.

sudo systemctl reload apache2.service

2.1.3. Deshabilitar a través del archivo .htaccess

Es una alternativa similar al archivo de configuración.

- Agregar en el archivo .htaccess del sitio respectivo:

  Options -Indexes

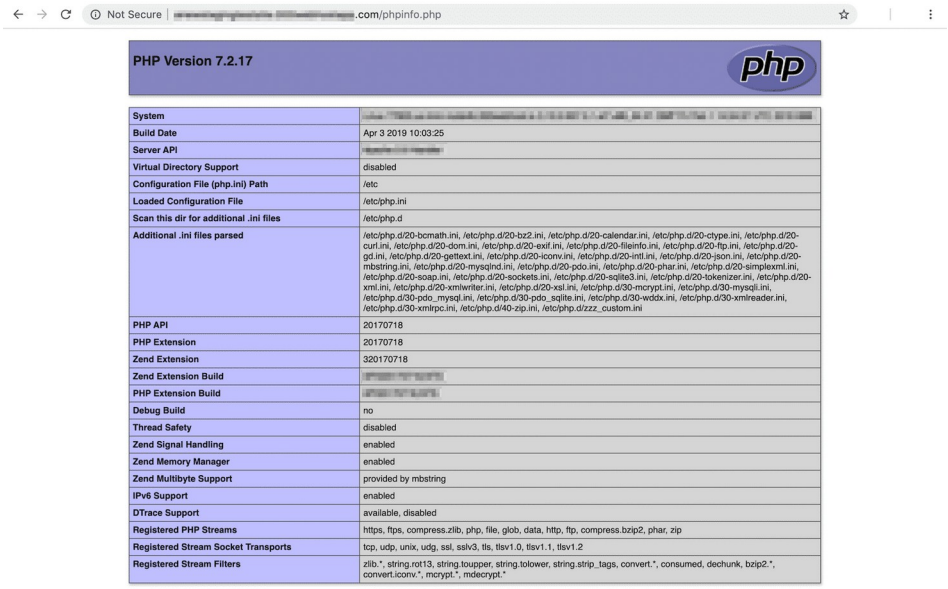
- Guardar el archivo y reiniciar el servidor.

  $ sudo systemctl restart apache2.service

El resultado de aplicar una de las configuraciones anteriores, será la restricción de listar archivos y carpetas.

2.2. Quitar archivo info.php y deshabilitar la función phpinfo()

Afecta a aplicaciones desarrolladas bajo el lenguaje de programación PHP, una práctica que se usa en entorno de desarrollo es imprimir las características de php en un archivo con nombre info.php a efectos de verificar la correcta instalación y configuración de PHP en el sistema. Sin embargo este archivo debe ser removido del ambiente de producción ya que devela información de las versiones, el sistema operativo y otros datos que podrían ser usados por un actor de amenaza.



Estas soluciones fueron probadas en PHP versión 7.2.

Acceder al archivo */etc/php/7.2/apache2/php.ini* y añadir el siguiente comando:

expose_php = Off

Una opción alternativa es deshabilitar mediante el comando “disable_funtions”, para ello se puede añadir “phpinfo”:

disable_functions = ..., phpinfo

2.3. Quitar/restringir directorios y archivos de control de versiones .git

Una mala práctica en entornos de producción es exponer directorios como “.git”, “.svn” que contienen datos de configuración del control de versiones del código. Entre algunos datos que se podrían exponer están nombres de usuarios, rutas, parámetros de configuración, historial de commits, entre otros datos. Por ello es importante quitar o restringir el acceso.

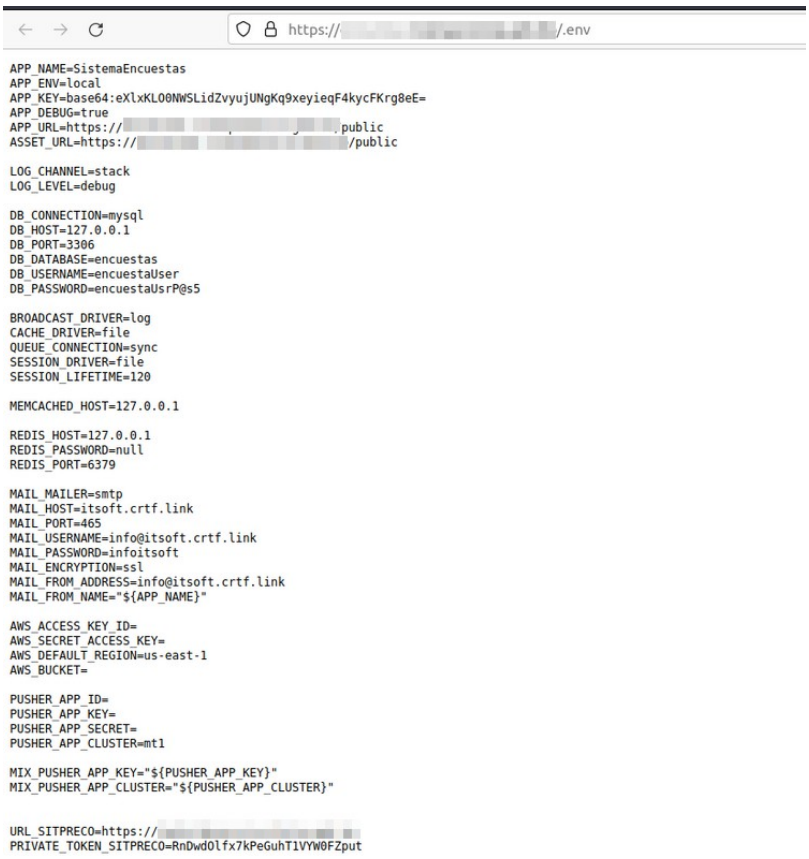

```
[core]
        repositoryformatversion = 0
        filemode = true
        bare = false
        logallrefupdates = true
[remote "origin"]
        fetch = +refs/heads/*:refs/remotes/origin/*
        url = https://gitlab.com/gcc2/archivo.git
[branch "master"]
        remote = origin
        merge = refs/heads/master
[branch "develop"]
        remote = origin
        merge = refs/heads/develop
```

Para restringir el acceso al archivo .git en apache, agregar la siguiente directiva en el archivo apache2.conf:

<Directory ~ "\.git">
Order allow,deny
Deny from all
</Directory>

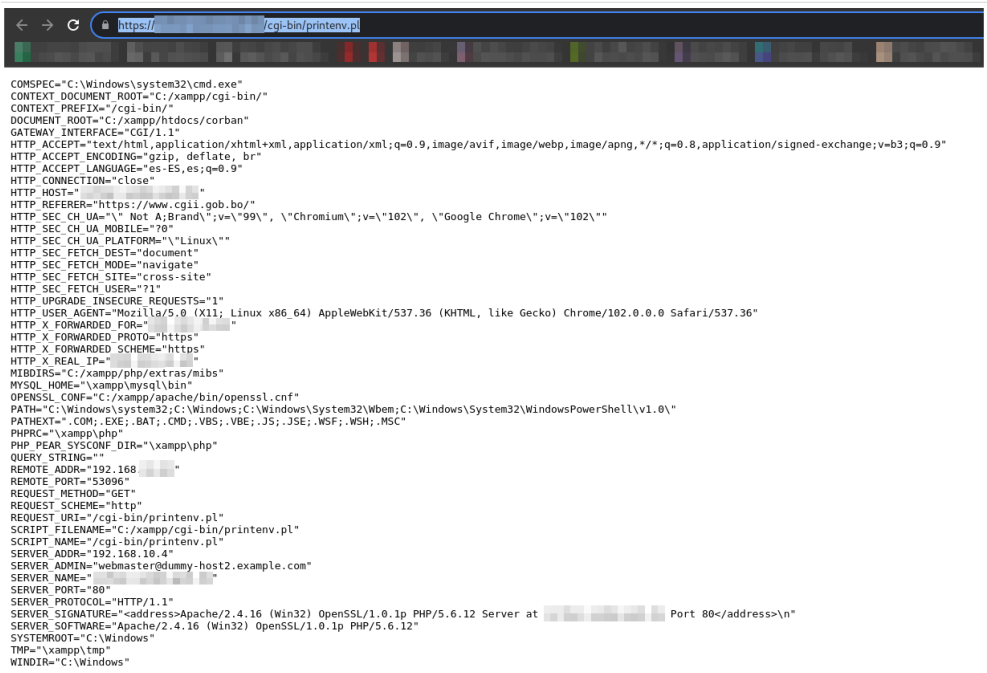
2.4. Restringir acceso a los archivos de configuración ".env" "printenv.pl"

El archivo .env es usado en proyectos para el almacenamiento de variables de entorno en ambiente development, test, production y otros. Contiene datos como nombres de usuario y contraseña para la conexión con la base de datos, tokens de acceso, cuentas de correo electrónico, direcciones IP, nombres de hosts y otros necesarios para la funcionalidad del software.



```
https://[redacted]/.env

APP_NAME=SistemaEncuestas
APP_ENV=local
APP_KEY=base64:eXlxKLO0NWSLidZvyujUNgKq9xeyieqF4kycFKrg8eE=
APP_DEBUG=true
APP_URL=https://[redacted].public
ASSET_URL=https://[redacted]/public

LOG_CHANNEL=stack
LOG_LEVEL=debug

DB_CONNECTION=mysql
DB_HOST=127.0.0.1
DB_PORT=3306
DB_DATABASE=encuestas
DB_USERNAME=encuestaUser
DB_PASSWORD=encuestaUsrP@s5

BROADCAST_DRIVER=log
CACHE_DRIVER=file
QUEUE_CONNECTION=sync
SESSION_DRIVER=file
SESSION_LIFETIME=120

MEMCACHED_HOST=127.0.0.1

REDIS_HOST=127.0.0.1
REDIS_PASSWORD=null
REDIS_PORT=6379

MAIL_MAILER=smtp
MAIL_HOST=itsoft.crtf.link
MAIL_PORT=465
MAIL_USERNAME=info@itsoft.crtf.link
MAIL_PASSWORD=infoitsoft
MAIL_ENCRYPTION=ssl
MAIL_FROM_ADDRESS=info@itsoft.crtf.link
MAIL_FROM_NAME="${APP_NAME}"

AWS_ACCESS_KEY_ID=
AWS_SECRET_ACCESS_KEY=
AWS_DEFAULT_REGION=us-east-1
AWS_BUCKET=

PUSHER_APP_ID=
PUSHER_APP_KEY=
PUSHER_APP_SECRET=
PUSHER_APP_CLUSTER=mt1

MIX_PUSHER_APP_KEY="${PUSHER_APP_KEY}"
MIX_PUSHER_APP_CLUSTER="${PUSHER_APP_CLUSTER}"


URL_SITPRECO=https://[redacted]
PRIVATE_TOKEN_SITPRECO=RnDwd0lfx7kPeGuhT1VYW0FZput
```

El archivo printenv.pl despliega las variables de entorno actuales.



```
https://[redacted]/cgi-bin/printenv.pl

COMSPEC="C:\Windows\system32\cmd.exe"
CONTEXT_DOCUMENT_ROOT="C:/xampp/cgi-bin/"
CONTEXT_PREFIX="/cgi-bin/"
DOCUMENT_ROOT="C:/xampp/htdocs/corban"
GATEWAY_INTERFACE="CGI/1.1"
HTTP_ACCEPT="text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,image/apng,*/*;q=0.8,application/signed-exchange;v=b3;q=0.9"
HTTP_ACCEPT_ENCODING="gzip, deflate, br"
HTTP_ACCEPT_LANGUAGE="es-ES,es;q=0.9"
HTTP_CONNECTION="close"
HTTP_HOST="[redacted]"
HTTP_REFERER="https://www.cgii.gob.bo/"
HTTP_SEC_CH_UA="\" Not A;Brand\";v=\"99\", \"Chromium\";v=\"102\", \"Google Chrome\";v=\"102\""
HTTP_SEC_CH_UA_MOBILE="?0"
HTTP_SEC_CH_UA_PLATFORM="\"Linux\""
HTTP_SEC_FETCH_DEST="document"
HTTP_SEC_FETCH_MODE="navigate"
HTTP_SEC_FETCH_SITE="cross-site"
HTTP_SEC_FETCH_USER="?1"
HTTP_UPGRADE_INSECURE_REQUESTS="1"
HTTP_USER_AGENT="Mozilla/5.0 (X11; Linux x86_64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/102.0.0.0 Safari/537.36"
HTTP_X_FORWARDED_FOR="[redacted]"
HTTP_X_FORWARDED_PROTO="https"
HTTP_X_FORWARDED_SCHEME="https"
HTTP_X_REAL_IP="[redacted]"
MIBDIRS="C:/xampp/php/extras/mibs"
MYSQL_HOME="\xampp\mysql\bin"
OPENSSL_CONF="C:/xampp/apache/bin/openssl.cnf"
PATH="C:\Windows\system32;C:\Windows;C:\Windows\System32\Wbem;C:\Windows\System32\WindowsPowerShell\v1.0\"
PATHEXT=".COM;.EXE;.BAT;.CMD;.VBS;.VBE;.JS;.JSE;.WSF;.WSH;.MSC"
PHPRC="\xampp\php"
PHP_PEAR_SYSCONF_DIR="\xampp\php"
QUERY_STRING=""
REMOTE_ADDR="192.168.[redacted]"
REMOTE_PORT="53096"
REQUEST_METHOD="GET"
REQUEST_SCHEME="http"
REQUEST_URI="/cgi-bin/printenv.pl"
SCRIPT_FILENAME="C:/xampp/cgi-bin/printenv.pl"
SCRIPT_NAME="/cgi-bin/printenv.pl"
SERVER_ADDR="192.168.10.4"
SERVER_ADMIN="webmaster@dummy-host2.example.com"
SERVER_NAME="[redacted]"
SERVER_PORT="80"
SERVER_PROTOCOL="HTTP/1.1"
SERVER_SIGNATURE="<address>Apache/2.4.16 (Win32) OpenSSL/1.0.1p PHP/5.6.12 Server at [redacted] Port 80</address>\n"
SERVER_SOFTWARE="Apache/2.4.16 (Win32) OpenSSL/1.0.1p PHP/5.6.12"
SYSTEMROOT="C:\Windows"
TMP="\xampp\tmp"
WINDIR="C:\Windows"
```

A continuación cambiar los permisos de los archivos .env y printenv.pl a 400 ó 440 para restringir el acceso a los usuarios públicos.

chmod 440 .env

Otra opción para restringir el acceso a los archivos .env y printenv.pl es agregar la siguientes siguientes líneas al archivo de configuración del sitio web o su respectivo .htaccess:

```
<Directory /cgi-bin>
    order deny,allow
    deny from all
</Directory>
```

También es posible utilizar la siguiente configuración:

```
<Directory /cgi-bin>
    Require all denied
</Directory>
```

Para el archivo .env:

```
<Files .env>
    Order allow,deny
    Deny from all
</Files>
```

2.5. Configurar el registro de logs de acceso y error del servidor web

2.5.1. Habilitar el registro de logs en apache

En el archivo de configuración de apache (/etc/apache2/sites-enabled/{subdominio}.conf), agregar las siguientes directivas:

ErrorLog ${APACHE_LOG_DIR}/{subdominio}-error.log

CustomLog ${APACHE_LOG_DIR}/{subdominio}-access.log combined

Se aconseja tener un error.log diferente por sitio web administrado.

2.5.2. Configurar el nivel de reporte de logs en apache

La configuración de loglevel permite seleccionar el nivel de detalle en el que se registrarán los logs, para ello se debe actualizar la variable "Loglevel" en el archivo apache2.conf (/etc/apache2/apache2.conf):

LogLevel info

Se puede seleccionar las siguientes opciones:

- warn: Advertencias
- info: Mensajes informativos.
- debug: Mensajes de depuración (producirá una gran cantidad de información).
- error: Errores producidos mientras se procesaba la solicitud.

2.5.3. Habilitar el registro de logs en PHP

Editar el archivo php.ini para habilitar el registro de logs:

log_errors = On;
error_log = /var/log/apache2/error_log;

2.5.4. Configurar el nivel de reporte de logs en PHP

Para configurar el nivel de reporte de logs en PHP se debe actualizar la variable "error_reporting" en el archivo php.ini:

error_reporting = E_ALL | E_STRICT;

El nivel de reporte puede tener los siguientes valores más comunes:

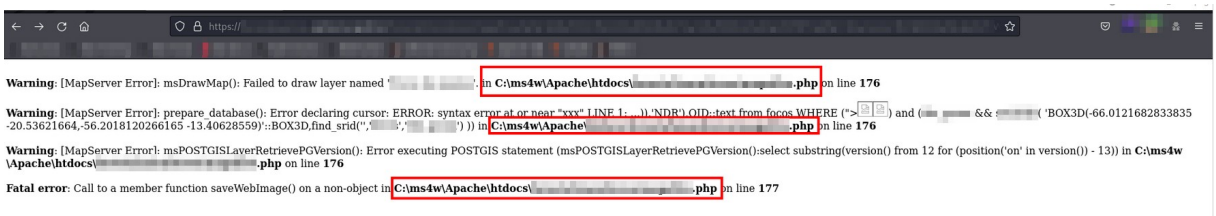
- E_ERROR: Errores fatales que no se pudieron recuperar, como un problema de asignación de memoria. Los mismos detienen la ejecución del programa.
- E_WARNING: Errores durante la ejecución del programa (no fatales).
- E_ALL: Todos los errores, advertencias y noticias (por defecto).
- E_STRICT: Habilita la sugerencia de cambios al código para mejorar la interoperabilidad y compatibilidad futura .
- E_PARSE: Errores de análisis en tiempo de ejecución.

- E_NOTICE: Avisos en tiempo de ejecución. Indica que la secuencia de comandos encontró algo que podría indicar un error.

Para ver más opciones, visitar la documentación oficial de PHP.

2.6. Deshabilitar el despliegue de errores

Es importante deshabilitar el despliegue de errores por excepciones no controladas en la aplicación web, debido a que cualquier mensaje de error mostrado al usuario final incluye no sólo la información del servidor, sino también un mensaje de excepción detallada que podría incluir variables de entorno, rutas, código fuente del recurso donde se produjo el error.



Para deshabilitar el despliegue de errores en PHP se debe editar el archivo php.ini cambiando a Off la directiva “display_errors”:

display_errors = Off;

2.7. Deshabilitar la información de versión del servidor web

Por defecto apache y nginx despliegan su versión utilizada cuando la solicitud a una página del sitio web responde con código de errores 40X.

Para evitar desplegar la versión de apache, se debe editar el archivo de configuración “/etc/apache2/apache2.conf” y agregar las siguientes directivas:

ServerTokens Prod

ServerSignature Off

El resultado será el siguiente:

En el caso de NGINX se debe editar el archivo nginx.conf actualizando la variable "server_tokens" con el valor "off":

```
http{
    ...
    server_tokens off;
    ...
}
```

Finalmente se debe reiniciar nginx:

systemctl restart nginx

2.8. Habilitar cabeceras HTTP de seguridad

Las cabeceras de seguridad agregan capas de seguridad adicionales al servidor web, para configurar las cabeceras de seguridad más importantes en Apache2 se debe agregar en el archivo de configuración del sitio web (por ejemplo: /etc/apache2/sites-enabled/example.conf):

```
<IfModule mod_headers.c>
    // Cabecera HTTP Strict Transport Security (HSTS)
    Header set Strict-Transport-Security 'max-age=31536000; includeSubDomains; preload'
    // Content Security Policy (CSP)
    Header always set Content-Security-Policy 'default-src 'self'; font-src *;img-src * data:; script-src *; style-src *;'
    //X-XSS-Proteccion
    Header set X-XSS-Protection '1; mode=block'
```

```
    //X-Frame-Options
    Header always set X-Frame-Options 'SAMEORIGIN'
</IfModule>
```

Seguido, habilitar el módulo "headers":

sudo a2enmod headers

Reiniciar Apache:

sudo systemctl restart apache2

2.9. Instalar y configurar certificado SSL/TLS

Lets Encrypt es un servicio que ofrece certificados SSL gratuitos a través de una API. Certbot es un cliente de Lets Encrypt, que tiene varias formas de validar el dominio, busca certificados y configura automáticamente Apache y Nginx.

Primero se deberá instalar Certbot:

sudo apt install certbot

Ejecutar Certbot:

sudo ufw allow 80

Ejecutar certbot en un servidor web temporal (--standalone) para obtener los certificados, en el ejemplo se usa el parámetro 'http', para https se utiliza tls-sni (--preferred-challenges), e introducir el nombre de dominio (-d):

sudo certbot certonly --stantalone --preferred-challenges http -d entidad.gob.bo

Para la configuración automática de Apache se puede usar:

sudo cerbot --apache -d entidad.gob.bo

Para configurar la aplicación, primero debe verificar los archivos relacionados al certificado:

sudo ls /etc/letsencrypt/live/entidad.gob.bo

Salida:

cert.pem chain.pem fullchain.pem privkey.pem README

2.9.1. Renovación automática del certificado

Los certificados de lets encrypt son válidos por 90 días, para hacer la renovación automáticamente se debe agregar un script a /etc/cron.d el cual se ejecutará 2 veces al día. Para completar la renovación es necesario aplicar los cambios editando el archivo:

sudo nano /etc/letsencrypt/renewal/entidad.gob.bo.conf

Agregar la opción renew_hook que permite realizar tareas posteriores a la obtención del certificado:

renew_hook = systemctl reload
<nombre del servicio>

Ejecutar certbot para comprobar que no haya errores:

sudo certbot renew --dry-run

2.9.2. Desabilitar protocolos inseguros

En el archivo de opciones de SSL, encontrará que los protocolos SSLv2 y SSLv3 ya están deshabilitados. Esto se debe a las inseguridades de estos protocolos y no deben utilizarse. Además, TLS v1.0 también es un protocolo heredado y no debe usarse. Sin embargo, aún es necesario para permitir que funcionen los navegadores web más antiguos. Habilítelo solo si es absolutamente necesario, para deshabilitar debe editar los archivos /etc/apache2/mods-enabled/ssl.conf, comentando y reemplazando:

SSLProtocol all -SSLv2 -SSLv3 -TLSv1 -TLSv1.1

Por:

SSLProtocol TLSv1.2

y en el archivo /etc/letsencrypt/options-ssl-apache.conf comentar y reemplazar:

SSLProtocol all -SSLv2 -SSLv3

Por:

SSLProtocol +all -SSLv2 -SSLv3 -TLSv1 -TLSv1.1

2.9.3. Deshabilitar la compresión ssl

En 2012, la vulnerabilidad del ataque CRIME demostró que la compresión TLS no se puede implementar de forma segura. Por lo tanto, no debe usarse y la única solución es deshabilitar la compresión TLS por completo. Verifique que lo siguiente esté en su archivo de configuración.

SSLCompression off

2.10. Proteger archivos de configuración

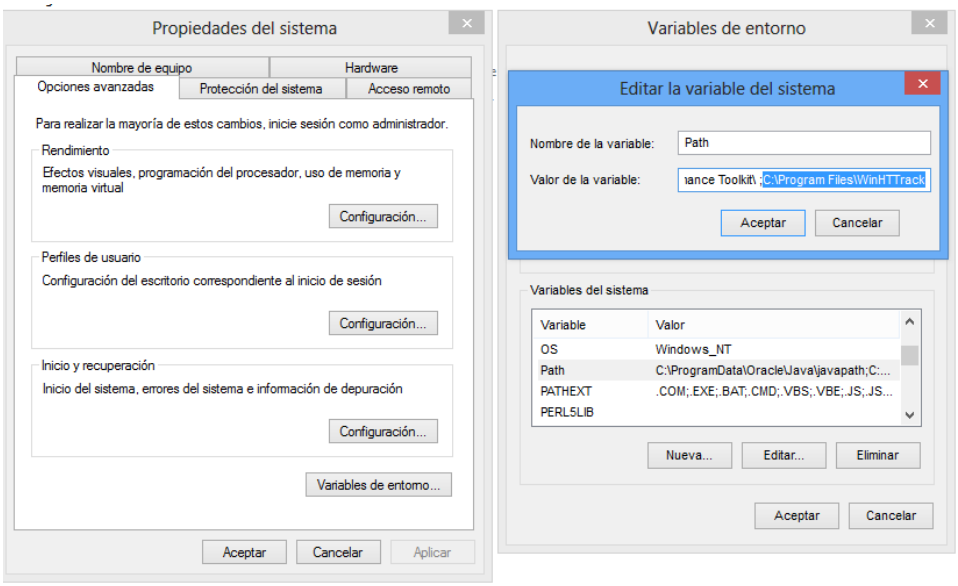
Es necesario que se configure correctamente los permisos de los archivos de configuración que contienen información sensible, de modo que se encuentre protegido contra accesos no autorizados, en ese sentido se recomienda considerar los siguientes permisos:

Archivos de configuración (composer.json, config.yml, etc) - 640
configuration.php, settings.php: 440 o 400
Otras carpetas - 755

2.11. Copia estática del sitio web

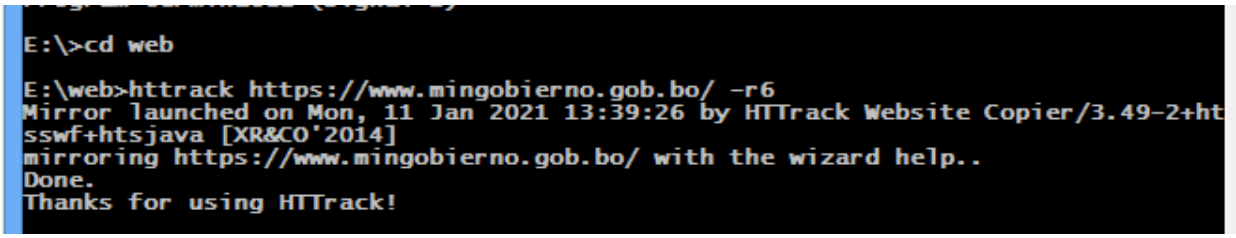
Se recomienda generar una copia estática reciente del sitio web para que pueda ser usada en caso de un incidente de seguridad. Para realizar esta tarea se establecen los siguientes pasos:

- Descargar Httrack https://www.httrack.com/page/2/en/index.html y después de instalar, modificar la variable de entorno PATH, agregar la ruta del Httrack "C:\ Program Files\WinHTTrack":



➢ Ejecutar el comando para crear la copia estática del sitio web:

httrack https://[dominio.gob.bo]/ -r6



Se recomienda realizar estas copias estáticas del sitio web cada vez que tenga un cambio significativo en su contenido.

CONTROL DE CAMBIOS

Versión	Fecha	Descripción de las modificaciones
1	12/01/2021	Versión nueva del documento.
2	03/01/2023	Se agregaron gráficos descriptivos de vulnerabilidades. Se agregó los siguientes subtítulos: - Quitar/Restringir Directorio y archivos de control de versiones .git. - Restringir el acceso a los archivos de configuración ".env" "printenv.pl". - Configurar el nivel de reporte de Logs apache y PHP. - Despliegue de errores. - Instalar y configurar certificado SSL/TLS. - Proteger archivos de configuración. - Copia estática del sitio web. - Habilitar cabeceras de seguridad. Se agregó la configuración para deshabilitar el despliegue de errores en NGINX.

ANEXO 2

Guía de seguridad Joomla

1. Introducción

Joomla es un sistema de gestión de contenido (CMS, Content Management System), que permite crear sitios web, su popularidad ha logrado que resulte muy atractivo para los actores maliciosos, con el fin de explotar vulnerabilidades.

2. Asegurando joomla

Para mitigar el riesgo de ataques a Joomla, se recomiendan las siguientes buenas prácticas de seguridad.

2.1. Verificar parches de seguridad

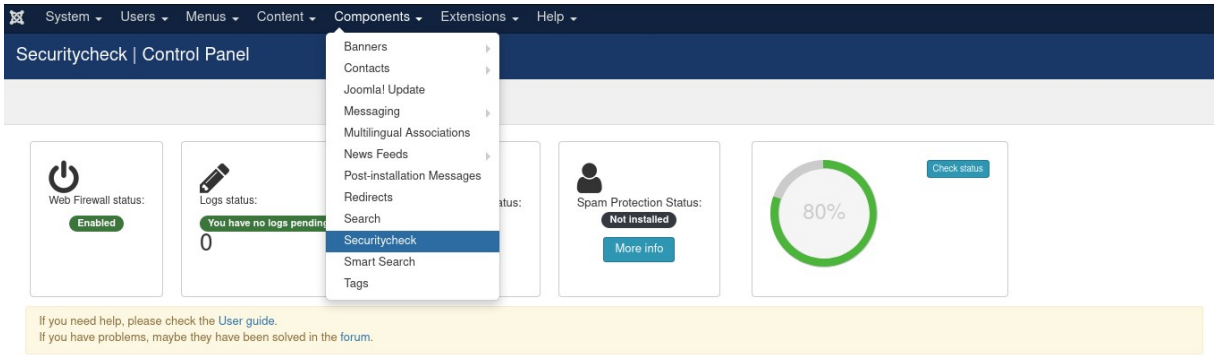
Comprobar regularmente si hay nuevos parches de seguridad disponibles para solucionar vulnerabilidades de seguridad e instalarlos, para ello existe el complemento "Plugin Securitycheck".

A continuación se describe los pasos para el uso del Plugin Securitycheck:

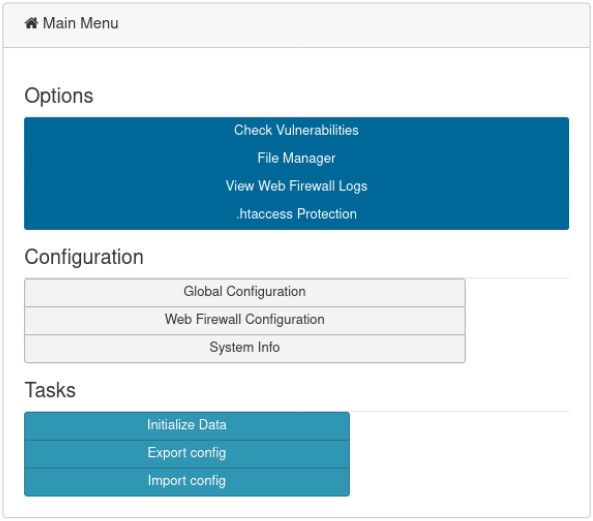
- Instalar el plugin siguiendo el enlace:

  https://extensions.joomla.org/extension/securitycheck/

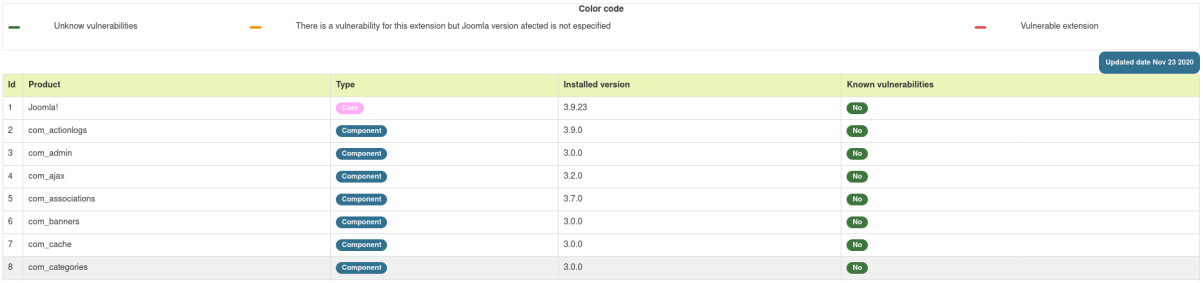
- Una vez instalado ir al menú Components→Securitycheck



- Verificar vulnerabilidades en los complementos.



- La columna de "Known vulnerabilities" de todos los complementos listados debe estar en estado "No".



Id	Product	Type	Installed version	Known vulnerabilities
1	Joomla!	Core	3.9.23	No
2	com_actionlogs	Component	3.9.0	No
3	com_admin	Component	3.0.0	No
4	com_ajax	Component	3.2.0	No
5	com_associations	Component	3.7.0	No
6	com_banners	Component	3.0.0	No
7	com_cache	Component	3.0.0	No
8	com_categories	Component	3.0.0	No

Se recomienda suscribirse a canales de seguridad oficiales de Joomla, por ejemplo:

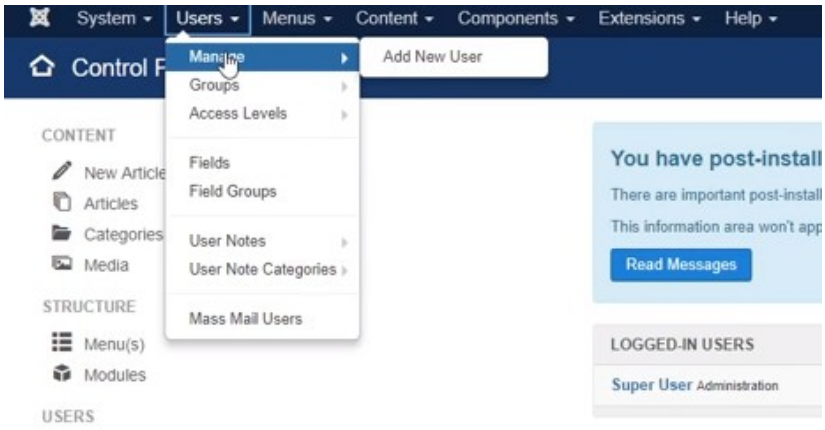
- https://docs.joomla.org/Security_hotfixes_for_Joomla_EOL_versions/es
- https://developer.joomla.org/security-centre.html

Siempre debe mantener actualizado Joomla a una versión con soporte.

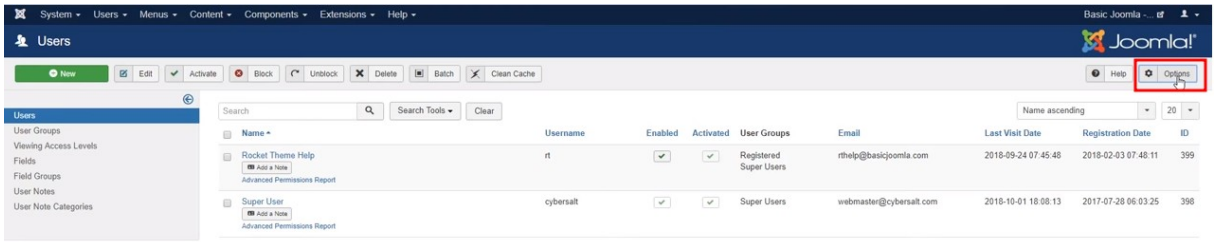
También se recomienda que se considere actualizar las tecnologías complementarias para el uso de Joomla como es php, mysql y el sistema operativo, tomando en cuenta que estas actualizaciones sean compatibles con la versión de Joomla que utiliza, aplicando estos cambios primero en un entorno de pruebas.

2.2. Asegurar nombre de usuario y contraseña

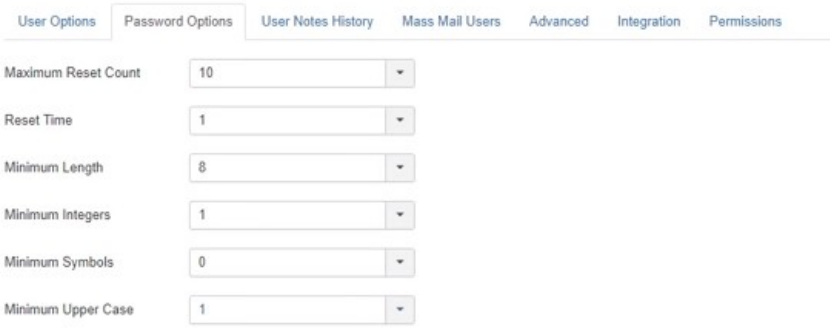
- No utilizar el nombre de usuario admin predeterminado.
- Utilizar una contraseña robusta, por ejemplo, que contenga mayúsculas, minúsculas, cifras y caracteres especiales.
- Configurar la robustez de la contraseña, para ello se debe ingresar a Users > Manage:



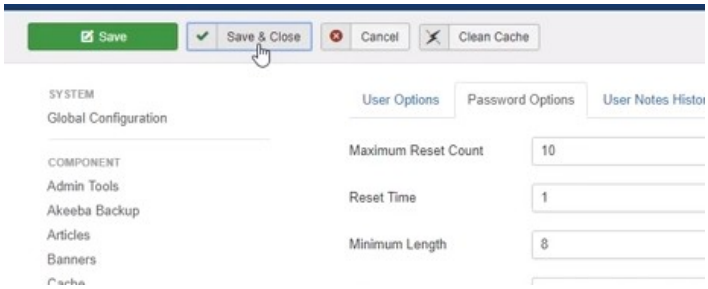
- Seleccionar Options:



- En password options establecer:



- Guardar y cerrar:



2.3. Proteger el archivo de configuración

Proteger el archivo configuration.php, que se encuentra en el directorio raíz de la instalación de Joomla con apache, para impedir que se pueda editar.

- Activar el módulo htaccess:

  $ sudo nano /etc/apache2/apache2.conf

- Buscar las Líneas:

  <Directory /var/www/>
  Options Indexes FollowSymLinks
  AllowOverride None
  Require all granted
  </Directory>

  Cambiar a:

  <Directory /var/www/>
  Options FollowSymLinks
  AllowOverride All
  Require all granted
  </Directory>

- Reiniciar servidor de apache:

  $ sudo service apache2 restart

- Añadir al archivo .htaccess:

```
<FilesMatch "configuration.php">
    Require all denied
</FilesMatch>
```

- Cambiar los permisos considerando:

Archivos PHP - 644
Archivos de configuración - 644
configuration.php: 440
Otras carpetas - 755

2.4. Proteger el acceso al panel de administrador

Por defecto el panel de administrador de Joomla se encuentra en la url "/administrator" de la página. Para evitar que personas no autorizadas intenten acceder al panel de administración seguir los siguientes pasos:

- Crear un directorio que únicamente conozcan los usuarios administradores del sitio web (debe recordar que el directorio miotroadm es solo un ejemplo).

```
$ sudo mkdir miotroadm
```

- Crear un archivo index.php para redireccionar al panel de administración, cambiar la cookie "admin_cookie_code" por una más larga y difícil de adivinar.

```
$ cd miotroadm
$ sudo nano index.php

<?php
    $admin_cookie_code='1254789654258'
    setcookie('JoomlaAdminSession',$admin_cookie_code,0,'/');
    header('Location: ../administrator/index.php');
?>
```

- Adicionar al principio del index.php del directorio "administrator" que solicite la cookie, caso contrario devolver al index.php

```
$ sudo nano administrator/index.php

if($_COOKIE['JoomlaAdminSession']!='1254789654258')
{
    setcookie('JoomlaAdminSession', null, -1, '/');
    header('Location: ../../index.php');
}
```

- Añadir al final en el index.php del panel de administración el siguiente comando para eliminar la cookie creada.

```
$ sudo nano index.php

if ($_COOKIE['JoomlaAdminSession']!='')
{
    setcookie('JoomlaAdminSession', null, -1, '/');
}
```

2.5. Ocultar la versión de Joomla

Deberá deshabilitar manualmente ingresando al panel de administración de joomla:

Site > Global Configuration

Establecer en "No" la opción "Show Joomla Version".

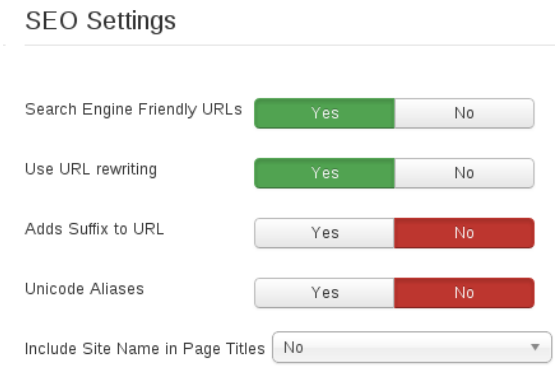
Adicionalmente, deberá eliminar la carpeta "installation" ubicada en el directorio raíz de la instalación de Joomla.

2.6. Activar search engine friendly (sef)

SEF permite hacer las URLs de Joomla más amistosas para el usuario y también dificulta a los escáneres automatizados encontrar información útil para efectuar ataques al sitio web.

Para activar SEF en Joomla debe acceder al panel de administración e ingresar a "Global Configuration".

Establecer la opción Search Engine Friendly URLs en "Yes":



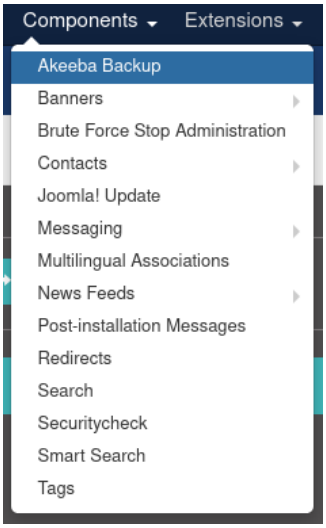
2.7. Realizar copia de seguridad

Para realizar la copia de seguridad de Joomla puede utilizar el plugin Akeeba Backup.

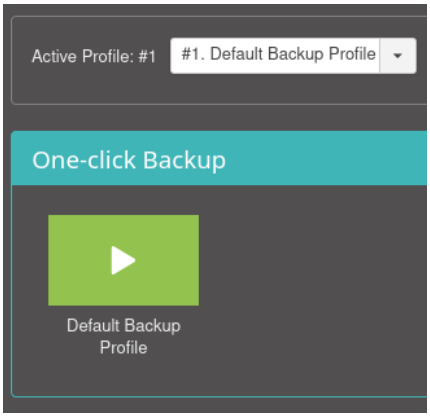
- Instalar el plugin siguiendo el enlace:

  Enlace: https://extensions.joomla.org/extension/akeeba-backup/

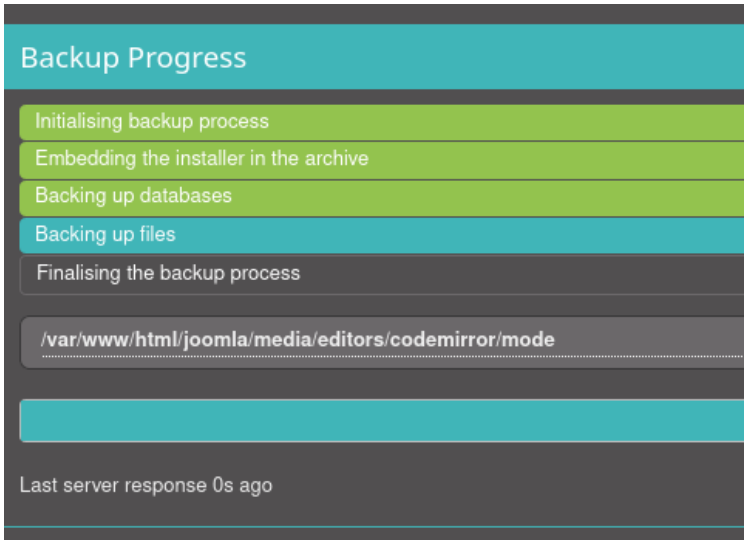
- Una vez instalado ir al menú Components → Akeeba Backup



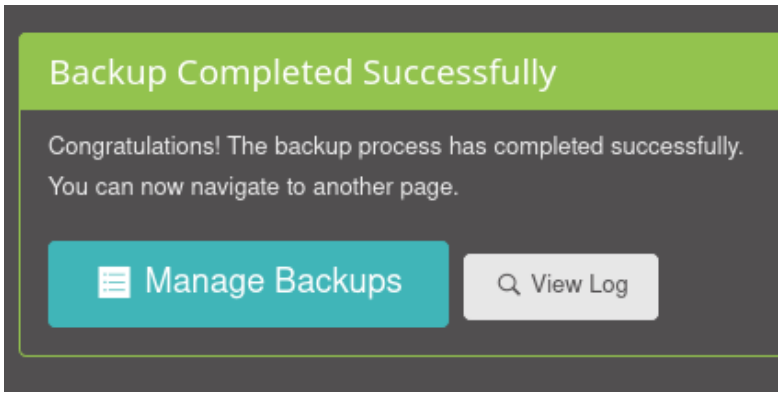
- Hacer clic en "Default Backup Profile" para sacar la copia de seguridad de Joomla.



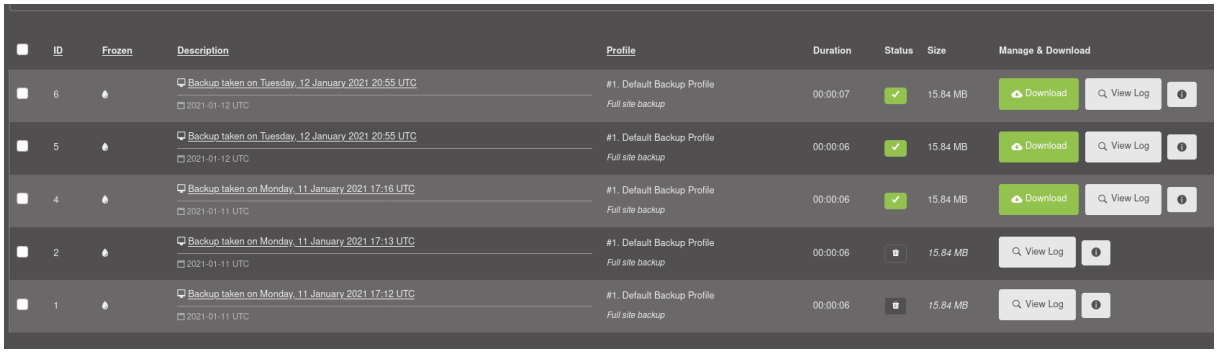
➢ Se despliega el proceso de backup.



➢ Los backups se muestran en Manage Backups.



➢ Se muestra el listado de backups generados.



- Establecer copias de respaldo cada cierto tiempo de acuerdo a las políticas de seguridad.

CONTROL DE CAMBIOS

Versión	Fecha	Descripción de las modificaciones
1	12/01/2021	Versión nueva del documento.
2	03/01/2023	Se agregó: ➢ Activar Search Engine Friendly (SEF) Se actualizó: - Asegurar nombre de usuario y contraseña. - Proteger el archivo de configuración.

ANEXO 3

Guía de seguridad Drupal

1. Introducción

Todo sistema de gestión de contenidos está propenso a ataques producto de vulnerabilidades descubiertas o configuraciones por defecto que se hayan dejado con la instalación. Por estas razones se presentan las prácticas de seguridad que debemos seguir para contar con un sitio web más seguro bajo Drupal.

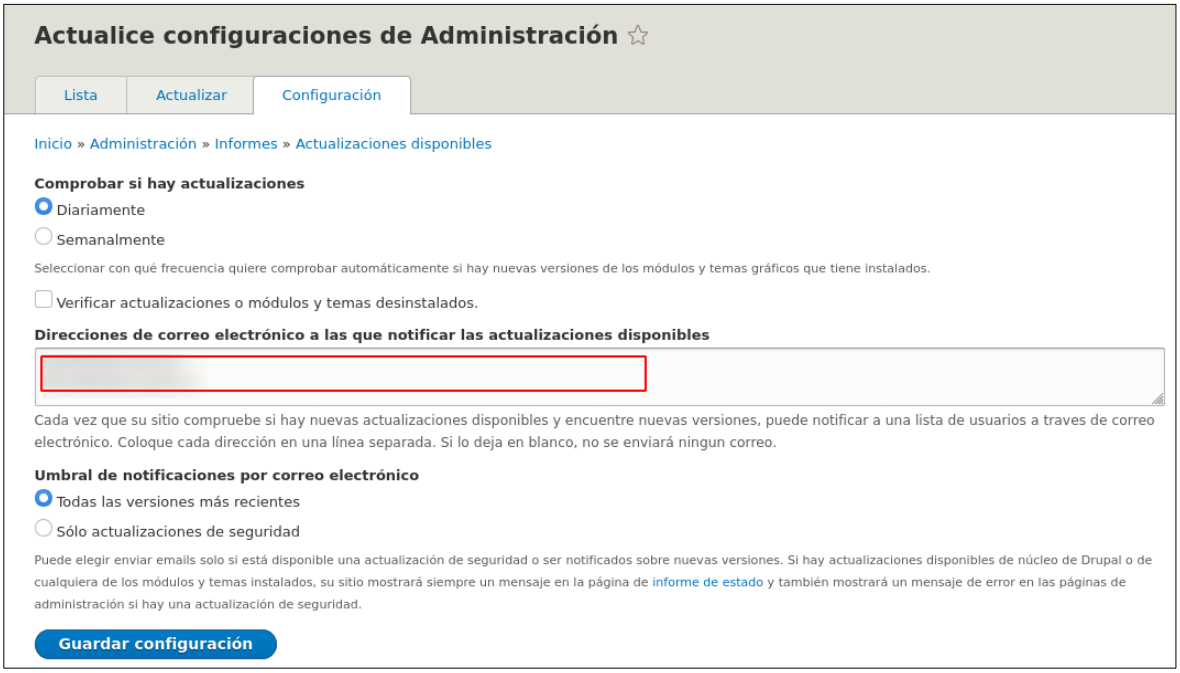
2. Asegurando drupal

Para mitigar el riesgo de ataques a Drupal, recomendamos aplicar las siguientes buenas prácticas de seguridad.

2.1. Activar notificaciones de actualizaciones de seguridad

Habilitar en la administración de Drupal notificaciones al correo electrónico sobre actualizaciones más recientes de seguridad de módulos contribuidos y del core.

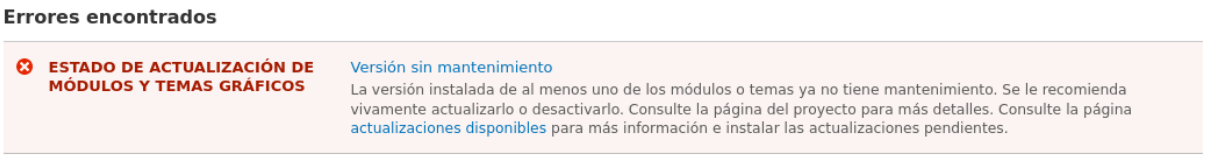
Administrar > Informes > Actualizaciones disponibles, pestaña "configuración"



Revisar de forma periódica el informe de estado de Drupal donde se muestra el estado de cada módulo y del core sobre actualizaciones funcionales y de seguridad. Aquellos

módulos que ya no cuentan con mantenimiento desde sus repositorios de origen se deben desactivar o migrar a uno con soporte.

Administración > Informes



Se recomienda seguir la cuenta de seguridad de Drupal en Twitter (@drupalsecurity) para estar al pendiente de las actualizaciones.

2.2. Instalar actualizaciones de seguridad

Como buena práctica de seguridad se debe tener un ambiente de pruebas donde se realicen las actualizaciones para verificar el correcto funcionamiento del sitio web antes de su pase a producción.

- Revisar el panel de actualizaciones disponibles.

- Replicar el ambiente de producción al ambiente de pruebas.
- Proceder con la actualización del core y módulos en el ambiente de pruebas.

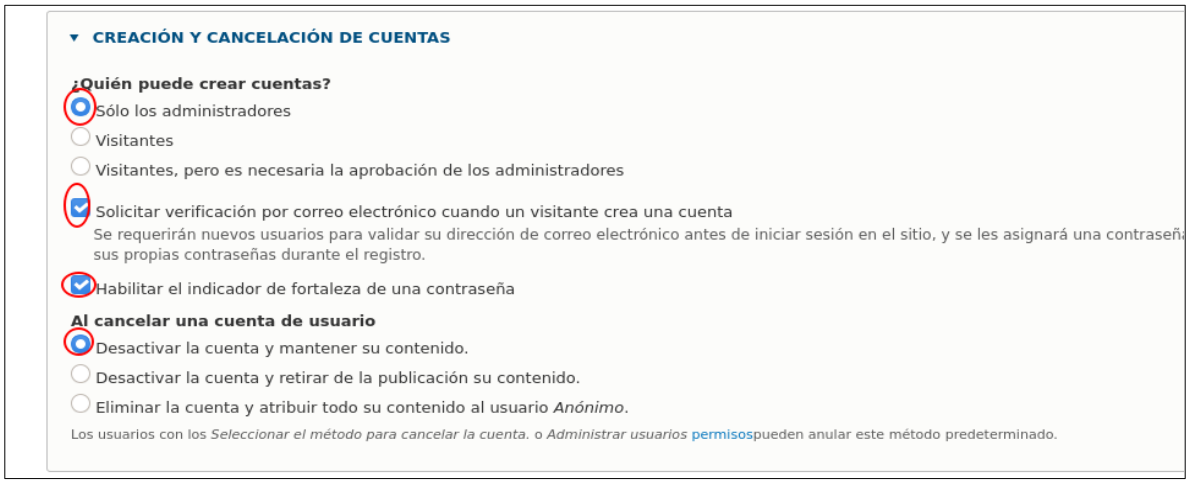
- Verificar que las actualizaciones se hayan aplicado correctamente.
- Realizar las pruebas funcionales.
- En caso de que todo esté correcto, replicar el ambiente de pruebas a producción en un horario donde no afecte a los usuarios.
- Drupal publica actualizaciones de seguridad todos los miércoles y pueden ser revisadas a través de la página:

  https://www.drupal.org/security

2.3. Gestionar usuarios

En Drupal se encuentra habilitada por defecto la opción de creación de cuentas por usuarios anónimos. Una buena práctica de seguridad es deshabilitar esta opción.

Administración > Configuración > Usuarios > Configuración de la cuenta



2.3.1. Configurar permisos

- Crear nuevos roles de acuerdo a las necesidades funcionales del sitio, estableciendo permisos específicos en módulos instalados.

  Administración > Usuarios > Permisos

PERMISO	USUARIO ANÓNIMO	USUARIO AUTENTICADO	ADMINISTRADOR
de contenido.			
Revertir todas las revisiones Para revertir una revisión, también necesita permiso para editar el elemento de contenido.	☐	☐	☑
Ver todas las revisiones Para ver una revisión, también necesita permiso para ver el elemento de contenido.	☐	☑	
Ver contenido publicado	☑	☑	
Ver el contenido propio sin publicar	☐	☑	
Path			
Administrar alias de URL	☐	☐	☑

2.4. Proteger formularios con captcha

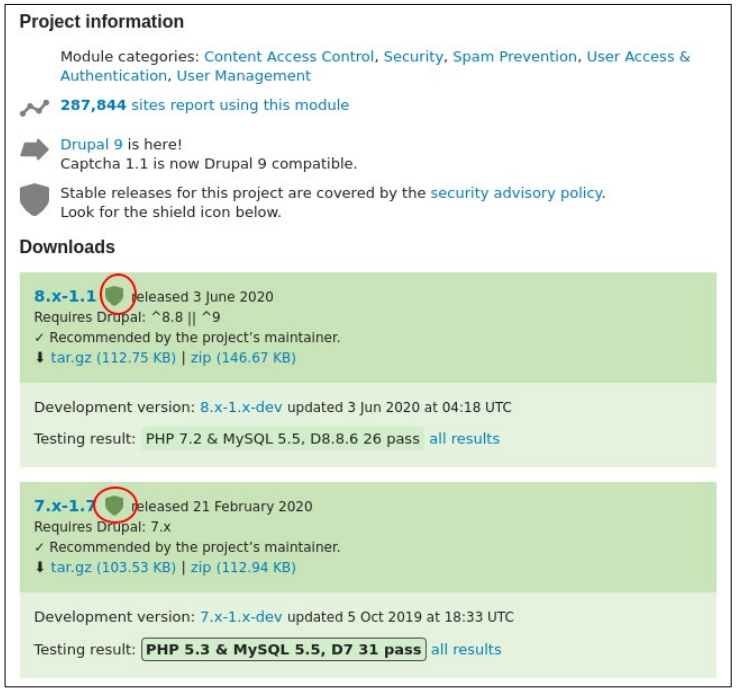
El captcha previene intentos automatizados de inicios de sesión y envíos masivos (spam) de datos a través de formularios.

- Instalar el módulo captcha (https://www.drupal.org/project/captcha) y habilitar.
- Entre las configuraciones se puede establecer el tipo de desafío del captcha matemático, imagen o personalizada con preguntas y respuestas pre establecidas.

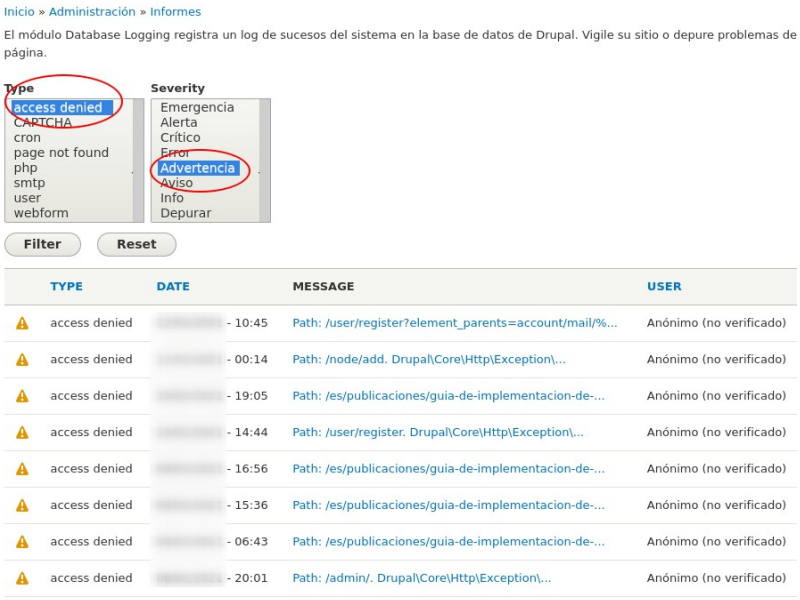
- Por ejemplo se habilita el captcha en el formulario de inicio de sesión, seleccionando el tipo de desafío.

2.5. Utilizar módulos con sello de seguridad

Sólo utilizar módulos que estén en fase estable y priorizar aquellos con el sello verde del Security Team de Drupal.



2.6. Revisar el registro reciente de mensajes

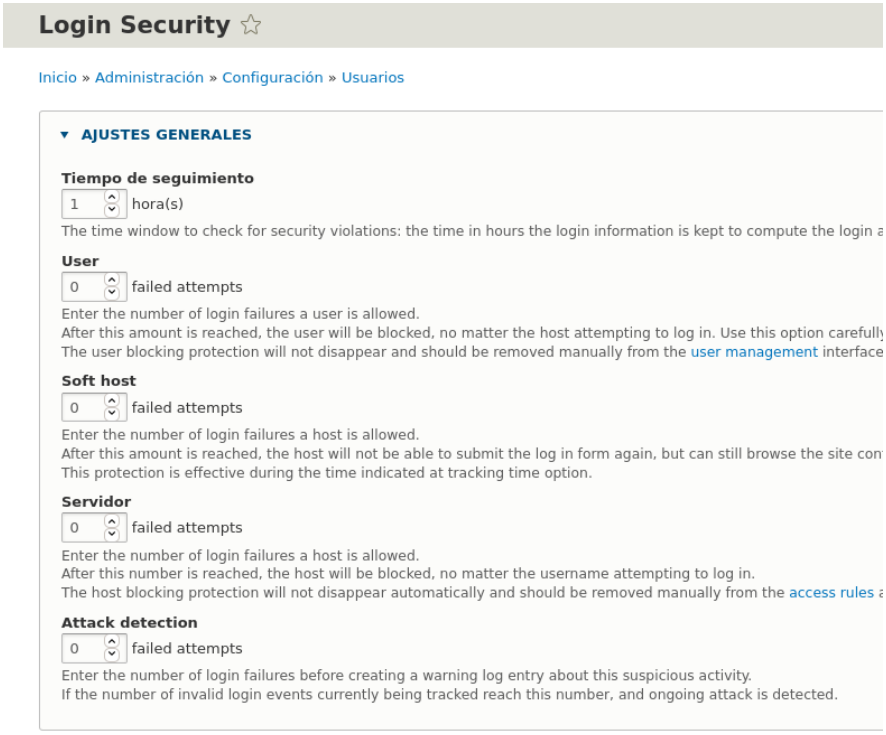


Revisar periódicamente el registro de mensajes de Drupal para identificar actividades sospechosas como intentos de inicio de sesión, errores de PHP, envío de formularios y otros datos importantes.

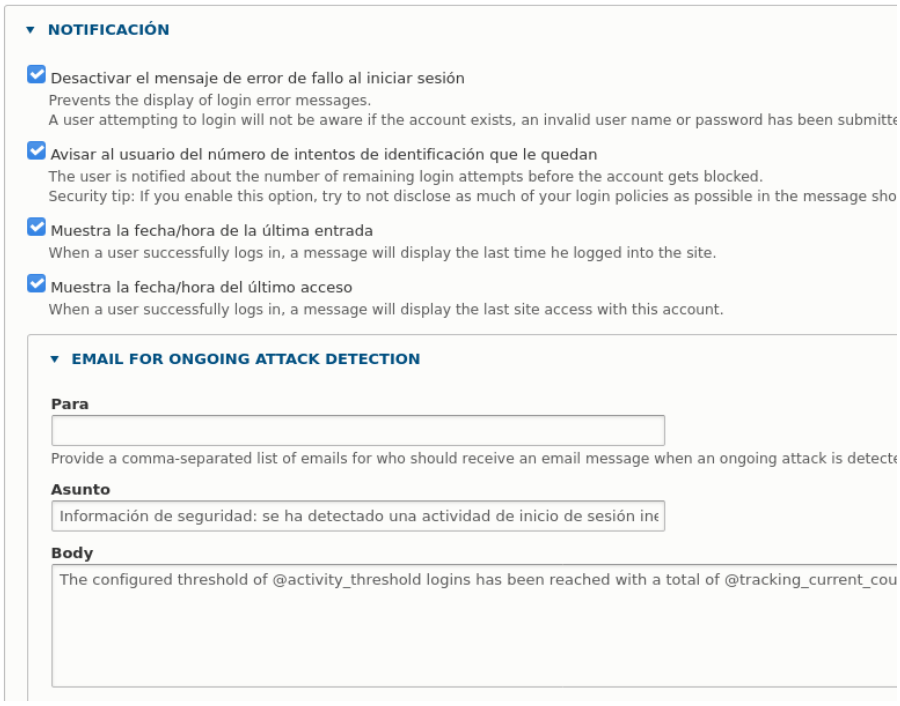
2.7. Módulo login security

El módulo Login Security permite configurar la forma en la que los usuarios se autentican en el sitio.

- Instalar el módulo login security (https://www.drupal.org/project/login_security) y habilitarlo simultáneamente con el módulo Ban que viene de forma predeterminada con el core de Drupal.
- Configurar en ajustes generales el número de intentos de inicio de sesión fallidos por usuario, host y también la detección de ataques.



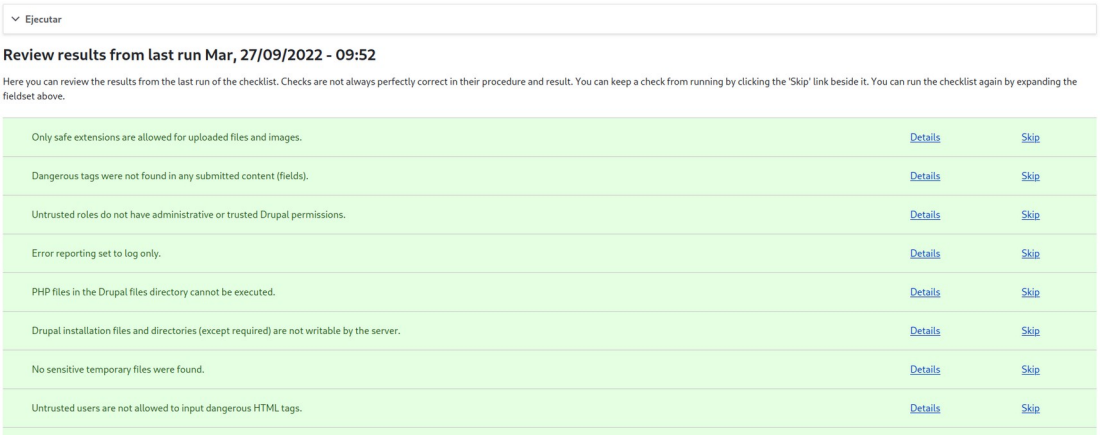
- Configurar notificaciones al correo electrónico, se recomienda deshabilitar los mensajes de error al iniciar sesión, avisar al usuario el número de intentos de inicio de sesión que le restan, mostrar la fecha y hora de último acceso.



2.8. Módulo security review

Este módulo prueba la configuración de Drupal en busca de vulnerabilidades de seguridad. Para aplicar este módulo se recomienda probar primero en el ambiente de pruebas.

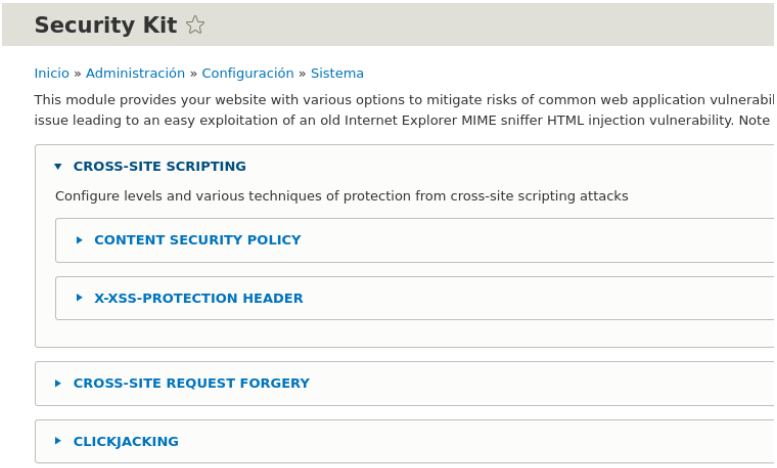
Instalar el módulo security review (https://www.drupal.org/project/security_review) y habilitarlo.



2.9. Módulo security kit

Security Kit permite proteger el sitio web de una amplia variedad de ataques como Cross-Site Scripting, Cross-Site Request Forgery, Clickjacking. Para aplicar este módulo se recomienda probar primero en el ambiente de pruebas.

- Instalar el módulo security kit (https://www.drupal.org/project/seckit) y habilitarlo.



- Configurar las opciones del módulo para mitigar riesgos de seguridad.

Si se desea ampliar las opciones de configuración de seguridad, se recomienda consultar los siguientes recursos:

- https://developer.mozilla.org/es/docs/Web/HTTP/CSP
- https://cheatsheetseries.owasp.org/cheatsheets/Cross-Site_Request_Forgery_Prevention_Cheat_Sheet.html
- https://developer.mozilla.org/es/docs/Web/HTTP/Headers/X-Frame-Options
- https://developer.mozilla.org/es/docs/Web/HTTP/Headers/Strict-Transport-Security

CONTROL DE CAMBIOS

Versión	Fecha	Descripción de las modificaciones
1	12/01/2021	Versión nueva del documento.
2	04/01/2023	Se actualizó: Instalar actualizaciones de seguridad Se agregó: Módulo Security Review

ANEXO 4

Guía de seguridad Wordpress

1. Introducción

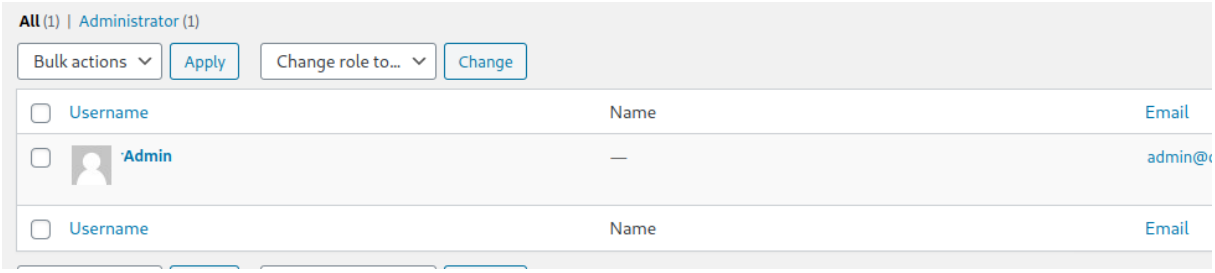
Wordpress es un sistema de gestión de contenido (CMS, Content Management System), que permite crear sitios web, su popularidad ha logrado que resulte muy atractivo para los "ciberatacantes", con el fin de explotar vulnerabilidad

2. Asegurando Wordpress

Para mitigar el riesgo de ataques a Wordpress, recomendamos las siguientes buenas prácticas de seguridad.

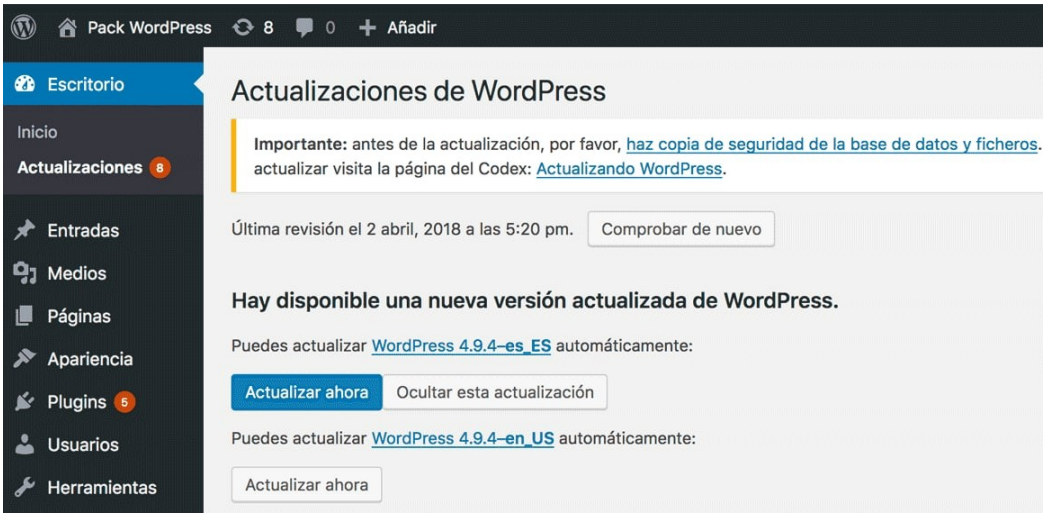
2.1. Configurar el control de acceso de usuarios

Ingresar al panel de administración de usuarios del sitio web: https://[mi-dominio.gob.bo]/wp-admin/users.php y eliminar a los usuarios administradores que no se usan actualmente:



2.2. Actualizar el core de Wordpress:

Ingresar a su sitio web https://[mi-dominio.gob.bo]/wp-admin/update-core.php y hacer click en el botón "Actualizar ahora".

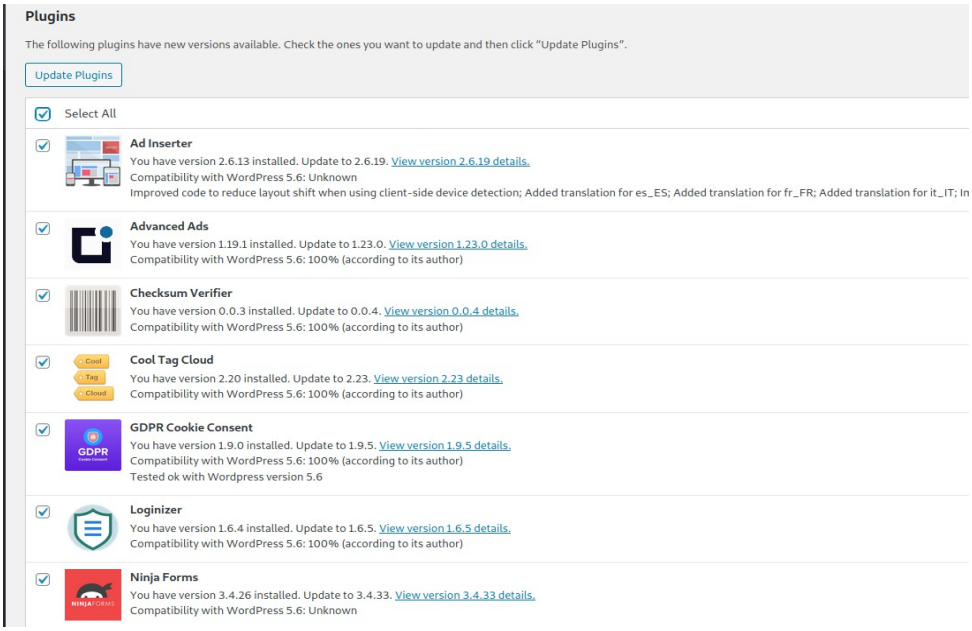


2.3. Actualizar los plugins de Wordpress

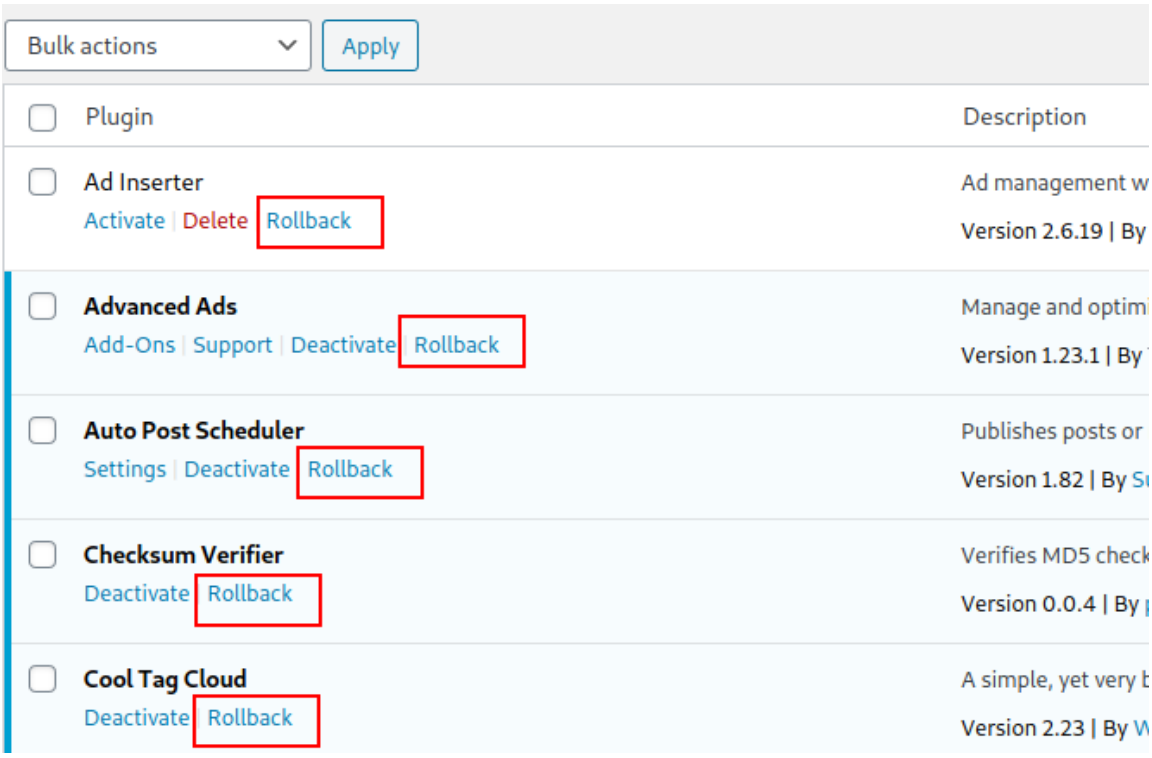
Previamente instalar y activar el plugin WP-Rollback (https://es.wordpress.org/plugins/wp-rollback/) que será útil en caso que la actualización de algún plugin no sea exitosa.

Realizar un backup de la base de datos.

Ingresar al sitio https://[mi-dominio.gob.bo]/wp-admin/update-core.php y seleccionar todos los plugins y presionar el botón "Actualizar plugins".



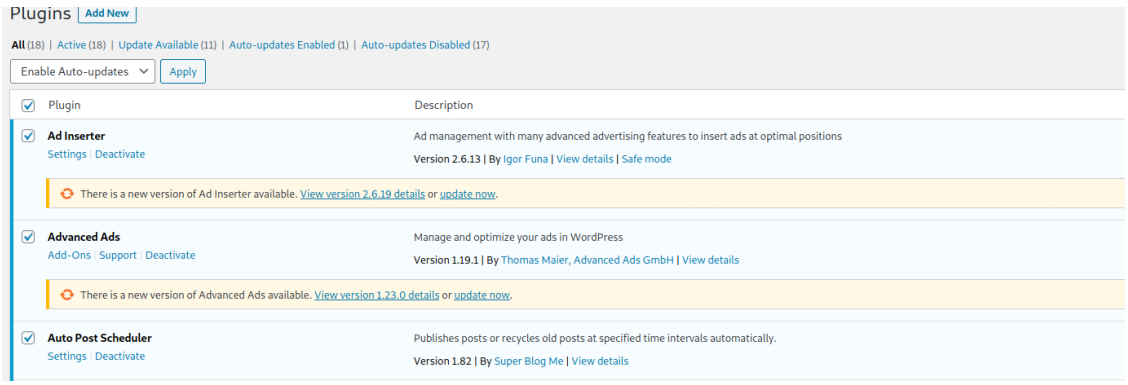
En caso de existir un error durante el proceso de actualización, realizar un ROLLBACK (https://[mi-dominio.gob.bo]/wp-admin/plugins.php).



2.4. Habilitar actualizaciones de seguridad automáticas

Ingresar al panel de administración https://[mi-dominio.gob.bo]/wp-admin/plugins.php y seleccionar todos los plugins.

Seleccionar la acción "habilitar actualizaciones automáticas" y ejecutar "Aplicar".



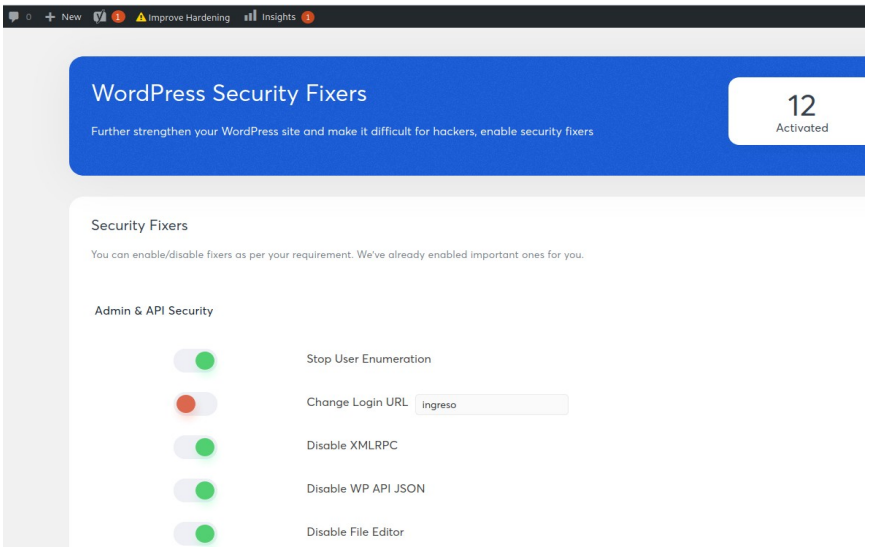
Borrar los themes no usados mediante la URL https://[mi-dominio.gob.bo]/wp-admin/themes.php

2.5. Instalar plugins de seguridad

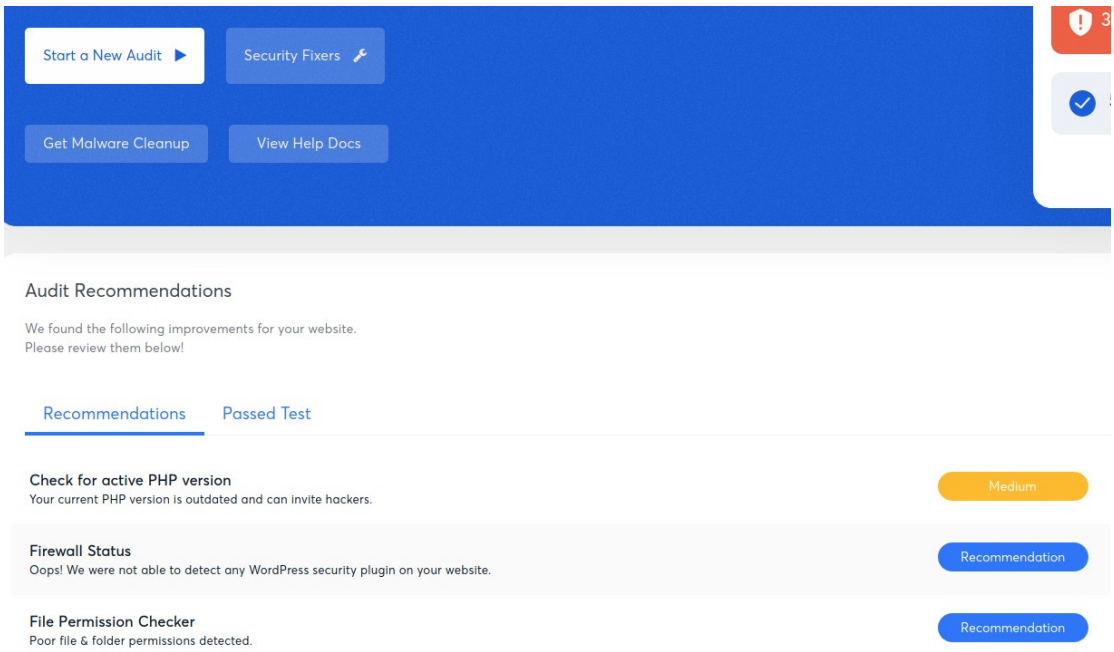
Instalar y habilitar el plugin "Logonizer" (https://wordpress.org/plugins/loginizer/) para proteger el sitio contra ataques de fuerza bruta.

Instalar y habilitar el plugin "WP Hardening" (https://wordpress.org/plugins/wp-security-hardening/).

En el panel de "WP hardening" https://[mi-dominio.gob.bo]/wp-admin/admin.php?page=wphwp_harden_fixers habilitar todas las opciones (12 en total) menos "Change Login URL":



En el panel de "WP hardening" https://[mi-dominio.gob.bo]/wp-admin/admin.php?page=wphwp_harden realizar la auditoría de seguridad y corregir todas las observaciones críticas (high), obteniendo como resultado final una pantalla similar a la siguiente:

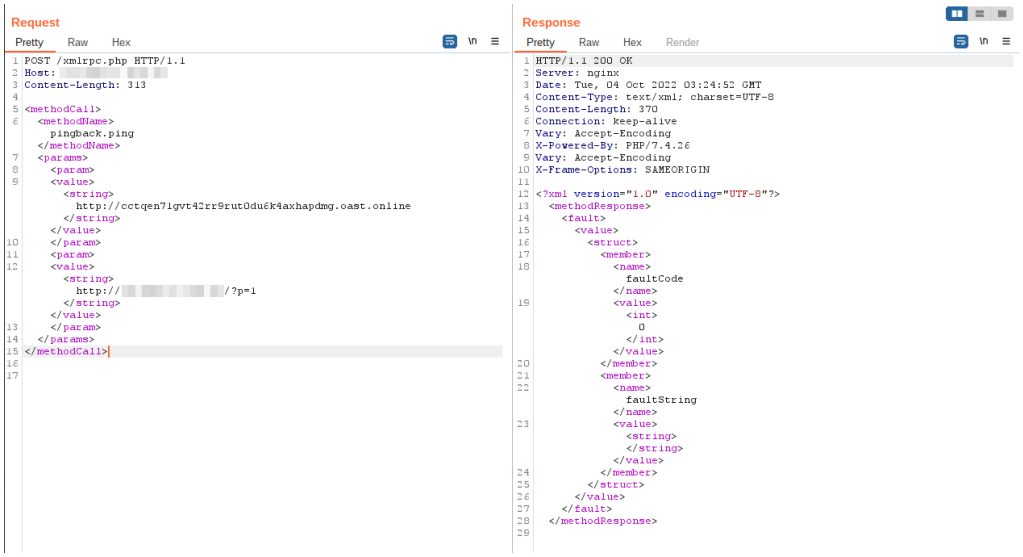


2.6. Deshabilitar XML-RPC

Wordpress tiene características para interactuar de forma remota con el sitio web, XML-RPC es una función de WordPress que permite la transmisión de datos con HTTP actuando como mecanismo de transporte y XML como mecanismo de codificación.

En la actualidad la funcionalidad del archivo xmlrpc.php ha disminuido considerablemente y su exposición y configuración insegura representa un riesgo en la seguridad del sitio web, provocando las siguientes vulnerabilidades:

- Ataques de fuerza bruta.
- Denegación Distribuida de Servicio.
- XMLRPC pingback.ping.



2.6.1. Restringir el acceso al archivo xmlrpc.php

Para restringir el acceso al archivo xmlrpc.php en apache2, se debe editar el archivo de configuración del sitio web o mediante el archivo .htaccess agregando las siguientes líneas:

<files xmlrpc.php>
order allow,deny
deny from all
</files>

Después reiniciar apache:

systemctl restart apache2

2.6.2. Deshabilitar la función XML-RPC

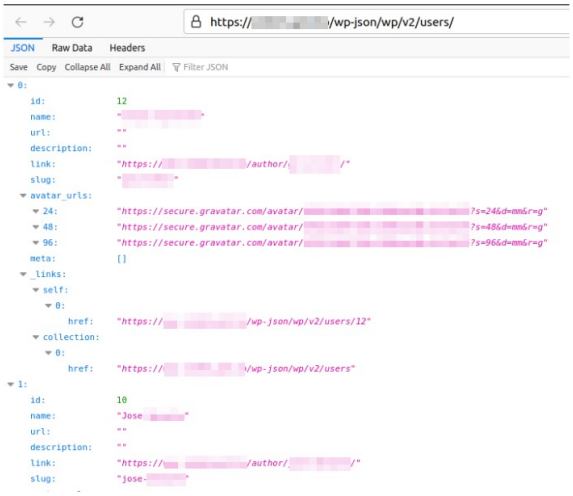
Si se determina que la función XML-RPC no es necesaria, se recomienda deshabilitarla, para ello se debe editar el archivo wp-config.php y agregar la siguiente línea:

add_filter('xmlrpc_enabled', '__return_false');

Otra opción para deshabilitar la función XML-RPC es instalando y activando el complemento “Disable XML-RPC”.

2.7. Enumeración de usuarios

Wordpress a través de su REST API puede exponer datos sensibles, como es el caso de los usuarios del sistema, si el sitio no está configurado de forma segura.



Para solucionar esta vulnerabilidad se debe actualizar a la versión 4.7.1 o posterior de Wordpress.

Una opción, en caso de no hacer uso del API REST de Wordpress, es deshabilitarlo, para ello se debe agregar las siguientes líneas en el archivo de configuración de wordpress wp-config.php:

add_filter('rest_enabled', '_return_false');
add_filter('rest_jsonp_enabled', '_return_false');

CONTROL DE CAMBIOS

Versión	Fecha	Descripción de las modificaciones
1	12/01/2021	Versión nueva del documento.
2	03/01/2023	Se agregó los siguientes subtítulos: ➢ Deshabilitar la función XML-RPC ➢ Enumeración de usuarios