Wordpress comprometido

Revisión Online

Verificar que no tenga instalado malware:

- https://www.virustotal.com/#/home/upload
- https://sitecheck.sucuri.net/

Verificar con herramientas de google el grado de compromiso del sitio:

- https://transparencyreport.google.com/safe-browsing/overview
- https://www.youtube.com/watch?v=lG5lOix9b9k

Es necesario acceder a la consola de búsqueda de google:

- https://search.google.com/search-console/welcome

Para verificar enlaces comprometidos, obtener el contenido en línea, recuperar posible código malicioso.

Coordinación con el RSI

- Solicitar al RSI que no debe eliminar ningún archivo ni realice cambio s en la base de datos.
- Si el sitio web sufrió un defacement solicitar al RSI que mueva los archivos de la instalación wordpress a otra ruta que no sea accesible desde el servidor web. Ej. /homve/[USUARIO] y que establezca un sitio web temporal mientras dure la investigación.
- Solicitar los archivos de la instalación de wordpress.
- Solicitar el envió del backup de la base de datos.
- Solicitar el envió de los logs del servidor web.

Revisión Offline

1. Comprobar la integridad de los archivos WordPress (Core y Plugins)

- Descargar la versión de wordpress que se tiene instalada de https://wordpress.org/download/releases/

```
diff -r wordpress_Instalado wordpress_descargado
```

- De manera similar proceder con los plugins y themes.
- Alternativamente se puede usar:

```
wp core verify-checksums --allow-root
```

2. Buscando patrones de cadena maliciosos:

2.1. Buscar webshells/backdoors en el sistema de archivos

2.1.1. Usando GREP

Guardar en el archivo pattern.txt

```
eval($_REQUEST
eval($_GET
eval($_POST
eval(
$GLOBALS[
$strrev('dedoce
base64_decode
str_replace
preg_replace
gzinflate
$f53[
hacked
FilesMan
\x73\x74\x72\x5f
str_rot13
Location:
google.com
```

```
bing.com
```

Ejecutar el comando:

```
fgrep -rf pattern.txt folderInstalacion | egrep -iv "\.js|\.css|\.po|\.html|Binary" > resultado.txt
```

Eliminar los falsos positivos.

Otros patrones sospechosos (ejecutar dentro del directorio principal):

```
grep -r '\\x' * | egrep -iv "\.gif|\.js|\.png|\#|entities"
grep -r '\\057' * | egrep -iv "\.gif|\.js|\.png|\#|binary|Crypto|zip|case|elseif|ChaCha|\\x00|pie|escaper"
```

2.1.2. Usando la herramienta webshell-scanner-client

Descargar e instalar webshell-scanner-client

```
wget https://github.com/baidu-security/webshell-scanner-client/releases/download/v1.0/webdir-linux32.bin
sudo mv webdir-linux32.bin /usr/bin/webshell-scanner-client
sudo chmod a+x /usr/bin/webshell-scanner-client
```

Escanear un archivo

```
webshell-scanner-client.bin archivo.php-
Solicitar al RSI que mueva los archivos de la instalacion wordpress a otra ruta que no sea accesible desde el servidor web.
Ej: /home/[USUARIO]
```

2.1.3. Usando la herramienta webshell-scan

Descargar e instalar webshell-scan:

```
git clone https://github.com/tstillz/webshell-scan
go build main.go
sudo mv main /usr/bin/webshell-scan
```

Escanear en busca de webshells con extensión en php:

```
webshell-scan -dir . -exts php
```

2.1.4. Usando findbot

Instalación:

```
wget https://raw.githubusercontent.com/wellr00t3d/findbot.pl/master/findbot.pl
chmod a+x findbot.pl
sudo mv findbot.pl /usr/bin
```

Buscar archivos maliciosos:

```
findbot.pl directorio
```

Verificando archivos de wordpress

- .htaccess
- wp-config.php
- Revisar los archivos functions.php

```
find . -iname "functions.php"
```

- Verificar que archivos se han modificado recientemente:

```
find . -type f -printf "%-.22T+ %M %n %-8u %-8g %8s %Tx %.8TX %p\n" | sort | cut -f 2- -d ' '
```

Escanear el servidor en busca de códigos maliciosos en los archivos

Buscar código malicioso en archivos y carpetas de WordPress:

```
find wp-includes -iname "*.php"
find wp-content/uploads -name "*.php" -print
```

Escanear la base de datos en busca de códigos maliciosos

- Descargar un back de la base de datos "backup.sql"
- Guardar en el archivo pattern.txt

```
<script>
eval
base64_decode
gzinflate
preg_replace
str_rot13
```

- Ejecutar el comando:

```
fgrep -rf pattern.txt backup.sql > resultado.txt
```

- Eliminar los falsos positivos.

Detección de cuentas de administrador falsas

Encuentre y elimine nuevos usuarios administradores o cuentas FTP que no ha creado.

Restauración del stitio:

- Eliminar los backdoors identificados y restaurar el sitio web con los archivos antiguos.
- Borrar themes no usados.
- Borrar plugins no usados.
- Actualizar el core, theme y plugins.
- Instalar el plugin WP Hardening (https://wordpress.org/plugins/wp-security-hardening/)
- Cambiar la contraseña de los usuarios de :
  - WordPress
  - Base de datos
  - Hosting/Cpanel

Herramientas

https://malwaredecoder.com

Anexos

Instalar wp-cli y checksum:

```
curl -O https://raw.githubusercontent.com/wp-cli/builds/gh-pages/phar/wp-cli.phar
chmod +x wp-cli.phar
sudo mv wp-cli.phar /usr/local/bin/wp
wp package install git@github.com:wp-cli/checksum-command.git --allow-root
```

Revision #3
Created 6 marzo 2023 10:08:05 by Vladimir Urquiola
Updated 3 abril 2025 12:30:23 by Franz Rojas