
12. Hackeando Servidores Web
Julio Javier Iglesias Pérez
C|EH Julio Iglesias Pérez

Servidores Web

Apache e IIS tienen la mayoría de websites hosteados en todo el mundo.

Internet Information Services

IIS es seguro pero hay que asegurarse de deshabilitar los módulos que no vayan a ser utilizados

Desfiguración de un Sitio Web

Ocurre cuando un intruso maliciosamente altera la apariencia de un sitio web, esto expone a los visitantes a ciertas propagandas.

¿Por qué los servidores Web son comprometidos?

- Falta de políticas de seguridad apropiadas, y mantenimiento.
- Mala configuración en los servidores web, sistemas operativos y redes.
- Errores en el software, S.O. y aplicaciones web.
- Instalación del servidor con las opciones por defecto.
- Defectos de seguridad sin parchar en el servidor, S.O. y aplicaciones.

¿Por qué los servidores Web son comprometidos?

- Archivos por defecto, respaldados o simples innecesarios.
- Permisos de archivos y directorios inapropiados.
- Servicios innecesarios habilitados, incluyendo administración de contenido y administración remota.
- Cuentas con sus contraseñas por defecto.

¿Por qué los servidores Web son comprometidos?

- Funciones administrativas o depuradas que están habilitadas o accesibles.
- Certificados SSL y opciones de encriptación mal configurados.
- Uso de certificados auto firmados o certificados por defecto.
- Autenticación inapropiada con sistemas externos.
- Conflictos de seguridad con facilidad de caso de uso de negocio.

Impacto de ataques de Servidor Web

- Cuentas de usuario comprometidas.
- Manipulación de datos.
- Ataques secundarios desde el sitio web.
- Desfiguración del sitio web.
- Robo de datos.
- Acceso root a otras aplicaciones o servidores.

Amenazas al Servidor Web

Mala Configuración: Se refiere a una debilidad en la configuración en la infraestructura web que puede ser explotada para realizar varios ataques como transversalidad de directorio, intrusión al servidor y robo de datos. Una vez detectados, estos problemas pueden ser fácilmente explotados y como resultado habrá un compromiso total de un sitio Web.

Amenazas al Servidor Web

Ejemplos de malas configuraciones:

- Funciones de administración remota.
- Servicios innecesarios habilitados.
- Mala configuración de Certificados SSL por defecto.
- Depuración detallada/mensajes de error.
- Usuarios por defecto o anónimos/contraseñas.
- Configuración simple, y archivos script

Ejemplo

httpd.conf en un servidor Apache

<Location /server-status>
SetHandler server-status
</Location>

Esta configuración permite a cualquiera ver la página de estado del servidor que contiene información detallada acerca del usuario actual en el servidor Web, incluyendo información acerca de los hosts actuales y solicitudes que están siendo procesadas

Ejemplo

Archivo php.ini

display_error = On

log_errors = On

error_log = On

ignore_repeated_errors = Off

Esta configuración da mensajes de error detallados

Ataques transversales al directorio

Permite a los atacantes acceder a directorios restringidos y ejecutar comandos fuera del directorio raíz del servidor Web. Los atacantes pueden utilizar métodos de error y de ensayo para navegar fuera del directorio raíz y acceder a información sensible en el sistema

http://server.com/scripts/..%5c../Windows/System32/cmd.exe?/c+dir+c:\

Ataque HTTP Response Splitting

Implica agregar datos de respuesta en el encabezado dentro de un campo así el servidor divide la respuesta en dos respuestas. Un atacante pasa datos maliciosos a una aplicación vulnerable y la aplicación incluye los datos en una respuesta HTTP en el encabezado. El atacante puede controlar la primera respuesta para redirigir al usuario a un sitio web malicioso mientras las otras respuestas serán descartadas por el navegador.

Ejemplo

Código del servidor:

```
String author =
request.getParameter(AUTHOR_PARAM);
Cookie cookie = new
Cookie("author", author);
cookie.setMaxAge(cookieExpiration);
response.addCookie(cookie);
```

```
Input = Jason
HTTP/1.1 200 OK
...
Set-Cookie: author=Jason
...
Input = JasonTheHacker\r\nHTTP/1.1
200 OK\r\n
```

Primera respuesta

```
Set-Cookie:
author=JasonTheHacker
HTTP/1.1 200 OK
...
```

Segunda respuesta

```
HTTP/1.1 200 OK
...
```

Ataque Web Cache Poisoning

Un atacante fuerza a la cache del servidor web que limpie su contenido caché actual y envía su propia petición realizada, la cual será almacenada en caché.

HTTP Response Hijacking

La víctima enviará una respuesta de fraccionamiento al servidor. El servidor realiza la primera respuesta. El cliente solicita el servicio http://www.juggybank.com/account?id=214

Luego el servidor envía la respuesta a la solicitud del atacante. El atacante solicita para http://www.juggybank.com/index.html

El atacante obtiene la respuesta de la solicitud de la víctima.

Ataque SSH de fuerza bruta

Los protocolos SSH son utilizados para cifrar un túnel SSH entre dos hosts para poder transferir datos sin encriptación en una red insegura. Los atacantes pueden hacer fuerza bruta a las credenciales login SSH para obtener acceso no autorizado al túnel SSH. Los túneles SSH puede ser utilizados para transferir malwares y otros exploits a las víctimas sin ser detectados.

Ataque Man-in-the-Middle

Permite al atacante obtener información sensible Interceptando y Alterando las comunicaciones entre el usuario final y los servidores Web. Un atacante actúa como un proxy de manera tal que las comunicaciones entre el cliente y el servidor Web se harán a través de él.

Passwords crackin de los servidores Web

Se utilizan los mismos métodos de crackeo de siempre.

Un atacante intenta explotar una debilidad para hackear los passwords bien elegidos.

Muchos intentos de hacking comienzan con crackeo de contraseñas y prueban al servidor Web que son un usuario válido.

Los atacantes utilizan distintos métodos como ingeniería social, spoofing, phishing, caballos de troya, virus, wiretapping, keyloggers, etc.

Passwords crackin de los servidores Web

Un Atacante apunta sobre todo para:

- Crackear formularios web de autenticación.
- Túneles SSH
- Servidores FTP
- Servidores SMTP
- Web shares

Las contraseñas más comunes son: root, administrator, admin, demo, test, guest, qwerty, nombres de mascotas, etc.

Técnicas de crackeo de contraseñas para los servidores WEB

1. Adivinando
2. Ataques de diccionario
3. Ataques híbridos.
4. Ataques de fuerza bruta

Pueden ser realizados manualmente o utilizando herramientas automatizadas como: Caín & Abel, Brutus, TCH Hydra, etc.

Ataques a una aplicación WEB

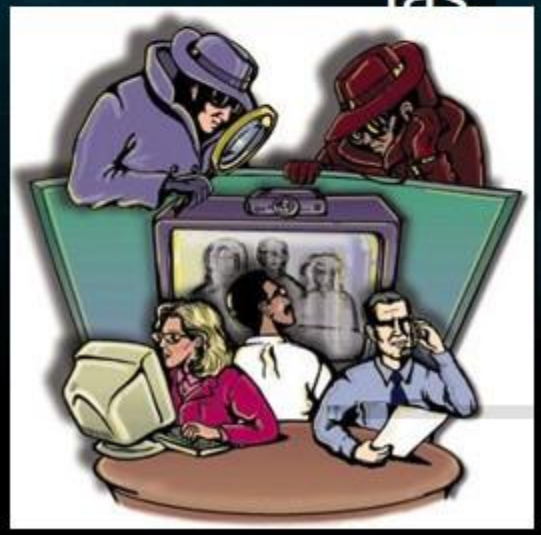
- Un input no validado.
- Manipulación de un formulario o parámetro.
- Directorio transversal.
- Ataques SQL Injection.
- Ataques de inyección de comandos.
- Ataques de inyección de archivos.
- Ataques Cross-Site scripting (XSS).
- Ataque Cross-Site Request Forgery.
- Ataque DoS.
- Ataques Buffer Overflow.

Metodología de ataque a Servidor Web

- Information Gathering.
- WebServer footprinting.
- Mirror Website.
- Vulnerability Scanning.
- Session Hijacking.
- Hacking Webserver Passwords.

Information Gathering

- Implica recolectar información acerca de la compañía objetivo. Los atacantes buscan en Internet, newsgroups, tablas de anuncios, etc. información sobre la compañía. Los atacantes utilizan herramientas como whois, tracerout, active whois, etc. y consultas a las bases de datos para obtener sobre el nombre dominio, IP, etc.



Webserver Footprinting

Obtener información importante sobre niveles del sistema como detalles de cuentas, S.O., etc. y versiones de software, nombres de servidores, esquema de base de datos, etc. Hacer telnet a un servidor web para hacer footprint y obtener información como nombres de servidores, tipos de servidor, S.O., aplicaciones corriendo, etc. Utilizar herramientas como ID Serve, httprecon Netcraft para realizar footprinting.

Reflejando o duplicando un Sitio Web

Se realiza esta acción para ver la estructura del directorio, de archivos, links externos, etc. Buscan comentarios en el código HTML para hacer las actividades footprinting más eficientes. Utilizar herramientas como HTTrack, Web Copier , BlackWidow, etc.

Escaneo de vulnerabilidades en los Servidores Web.

Realizar un escaneo para encontrar vulnerabilidades en la red y determinar si el sistema puede ser explotado. Utilizar escaners como HP WebInspect, Nessus, Paros proxy, etc. para encontrar hosts, servicios y vulnerabilidades. Olfatear el tráfico de la red para encontrar sistemas activos, servicios de red, aplicaciones y vulnerabilidades presentes. Testear la infraestructura del Servidor Web para ver si hay alguna mala configuración, contenido obsoleto, y vulnerabilidades conocidas.

Ataque Session Hijacking al Servidor Web

Sniffear un ID de sesión válido para obtener acceso no autorizado al servidor Web y curiosear los datos. Utilizar técnicas de sesión Hijacking como session fixation, session sidejacking, Cross-site scripting, etc. para capturar cookies de sesiones válidas e IDs. Utilizar herramientas como Burp Suite Hamster, Firesheel, etc. para realizar una sesión hijacking automatizada.

Ataque hacking web Server Passwords

Ataque hacking web Server Passwords

Utilizar técnicas de craqueo de contraseñas como fuerza bruta, diccionario, password guessing, etc. Utilizar herramientas como Brutus, THC-Hydra, etc.

Herramientas de ataque a Servidores Web

Metasploit: Abusa de vulnerabilidades conocidas y aprovecha las contraseñas débiles vía Telnet, SSH, HTTP y SNM.

Modulo Exploit de Metasploit

Es el módulo básico en Metasploit utilizado para encapsular un exploit utilizando que usuarios se dirigen en muchas plataformas con un exploit simple. Este módulo viene con campos simplificados de meta información. Utilizando la característica Mixins, los usuarios también pueden modificar el comportamiento de un exploit dinámicamente. ataques de fuerza bruta e intentos pasivos de exploits.

Pasos para explotar un sistema utilizando Metasploit Framework

- Configurar Active Exploit.
- Verificar las Opciones de Exploit.
- Seleccionar un objetivo.
- Seleccionar el Payload.
- Ejecutar el Exploit.

El módulo Payload de Metasploit

1. Establece un canal de comunicación entre Metasploit y el host víctima.
2. Combina código arbitrario que es ejecutado como resultado de una explotación correcta.
3. Para generar payloads, primero se debe seleccionar el payload utilizando el comando:

use Windows/shell_reverse_tcp

Módulo auxiliar de Metasploit

Puede ser utilizado para realizar arbitrario, un frente de acción como port scanning, DoS, incluso fuzzing. Para ejecutar el módulo auxiliar, se puede utilizar el comando "run" o utilizar el comando "exploit".

Módulo NOPS de Metasploit

Genera instrucciones no operacionales utilizadas para bloquear los buffers. Utilizar el comando "generate" para generar un trineo NOP en un campo arbitrario y mostrarlo en un formato dado.

ej:

use x86/opty2

generate -t c 50

Herramienta de ataque Web Wfetch

Permite al atacante personalizar totalmente una solicitud HTTP y enviarla al servidor web para ver la solicitud HTTP prima y responder datos. Permite al atacante testear el rendimiento de sitios Web que contienen nuevos elementos como ASP o protocolos Wireless.

Herramienta de crackeo de contraseñas Brutus

Soporta, HTTP, POOP3, FTP, SMB, Telnet, IMAP, NNTP y muchos otros protocolos de autenticación. Incluye un motor de autenticación múltiple y puede hacer hasta 60 conexiones simultáneas. Soporta el no uso de nombre de usuario, nombres de usuario múltiples, listas de contraseñas, listas combo (usuarios/contraseñas) y modos configurables de fuerza bruta.

Herramienta de cracko de contraseñas TCH-Hydra

Es un crackeador rápido de logon de red. Soporta TELNET, FTP, HTTP, HTTPS, HTTP-PROXY, SMB, SMFBT, MS-SQL, MYSQL, REXEC, etc.

Contramedidas

Contramedida Parches y Actualizaciones

- Escanear vulnerabilidades existentes, parcharlas y actualizar el software del servidor regularmente. Antes de aplicar cualquier service pack, hotfix o parche de seguridad, leer la revisión de toda la documentación relevante.
- Aplicar todas las actualizaciones sin tener en cuenta su tipo en base a "según sea necesario". Testear los service packs y hotfixes en un ambiente no productivo antes de implementarla en la producción.

Contramedidas

- Asegurarse que los SP, Hotfixes y niveles de parches de seguridad sean consistentes en Todos los controladores de dominio DC. Asegurarse que los cortes de servidores son programados y que haya backups y discos de reparación de emergencias disponibles.
- Tener un plan de "marcha atrás" que permita al sistema y la empresa volver a un estado original, antes de que la implementación fallida. Programar periódicamente actualizaciones como parte de las operaciones de mantenimiento y nunca intentar tener mas de un SP atrás.

Contramedida Protocolos

- Bloquear puertos innecesarios, tráfico ICMP, y protocolos innecesarios como NetBIOS y SMB.
- Endurecer la pila TCP/IP y consistentemente aplicar los últimos parches y actualizaciones en el software del sistema.

Contramedida Protocolos

- Si se utilizan protocolos no seguros como Telnet, POP3, SMTP, FTP, tomar las medidas para proveer autenticación y comunicación seguras, por ejemplo utilizando directivas IPSec.
- Si el acceso remoto es necesario, asegurarse que la conexión remota está asegurada de manera apropiada utilizando protocolos de encriptación y túnel.
- Deshabilitar WebDAV si no esta siendo utilizado por ninguna aplicación o mantenerlo seguro si es que es necesario.

Contramedidas Cuentas

1. Remover todos los módulos y extensiones de aplicaciones.
2. Deshabilitar las cuentas por defecto creadas durante la instalación de un S.O.
3. Cuando se creé un nuevo directorio raíz Web, dar los permisos NTFS (mínimos si es posible) al usuario anónimo en el servidor IIS a ser utilizado.
4. Eliminar las bases de datos de usuarios y procedimientos almacenados y seguir siempre el mínimo privilegio para la aplicación de bases de datos para defenderse contra SQL query poisoning.

Contramedidas Cuentas

5. Utilizar permisos WEB, permisos NTFS y mecanismos de control de acceso .NET incluyendo la autorización URL.

6. Implementar directivas de contraseña fuertes para ralentizar los ataques de diccionario y fuerza bruta y los logs de auditoria y alertas de falla.

7. Ejecutar procesos utilizando cuentas con menos privilegios, servicios con mínimos privilegios y cuentas de usuarios.

Contramedida Archivos y Directorios

- Eliminar archivos innecesarios .jar
- Eliminar configuración sensible entre el código byte.
- Impedir el mapeo de directorios virtuales entre dos servidores distintos, o sobre la red.
- Monitorear y revisar frecuentemente todos los logs de servicios de red, logs de acceso a sitio web, los de base de dados (ej: SQL Server,MySQL, Orable) y logs de S.O.

Contramedida Archivos y Directorios

- Deshabilitar el listado de directorio.
- Eliminar la presencia de archivos NO web como archivos, archivos backup, archivos de texto, etc.
- Deshabilitar ciertos tipos de archivos creando un mapeo de recursos.
- Asegurarse de que las aplicaciones web y los scripts estén en particiones separadas del disco del S.O., sistema, logs y otros archivos del sistema

¿Cómo defenderse contra ataques al servidor Web?

- - Puertos: Auditar los puertos en el servidor regularmente para asegurarse que los servicios inseguros o no necesarios están inactivos en el Servidor Web. Limitar tráfico de entrada al puerto 80 para HTTP y puerto 443 para HTTPS. Cifrar o restringir el tráfico intranet.
- - Configuración del equipo: Asegurarse que los recursos protegidos están mapeados en HttpForbiddenHandler y los HttpModules no utilizados sean removidos. Asegurarse que el seguimiento (tracing) esté deshabilitado <trace enable="false"/> y la compilaciones de depuración estén apagados.

¿Cómo defenderse contra ataques al servidor Web?

- Certificados del Servidor: Asegurarse que los rangos de datos de los certificados sean válidos y los certificados sean utilizados por el propósito pretendido. Asegurarse que el certificado no ha sido revocado y que la public key del certificado sea válida, todo se encamine a una root authority confiada.

¿Cómo defenderse contra ataques al servidor Web?

- Seguridad de código de acceso: Implementar prácticas de código seguro para impedir ataques de revelación y validación. Restringir opciones de directivas de código de acceso seguro para asegurarse que el código descargado de internet o intranet no tenga permisos de ejecución. Configurar IIS para rechazar URLs con "../" para prevenir transversión de ruta, bloquear comandos del sistema y utilidades con Listas de control de acceso (ACLs) e instalar nuevos parches y actualizaciones.

¿Cómo defenderse contra ataques de servidor Web?

- -Registry: Aplicar ACLs y bloquear la administración remota del registro. Asegurar la SAM (Stand-alone Servers Only).
- -Shares: Remover todos los archivos compartidos innecesarios, recursos compartidos administrativos por defecto si no son requeridas. Asegurar los recursos compartidos con permisos NTFS restringidos.
- IIS Metabase: Asegurarse que la seguridad está configurada apropiadamente y acceder al archivo metabase y restringir con permisos NTFS más duros. Restringir banner information retornado por IIS.

¿Cómo defenderse contra ataques de servidor Web?

- Auditing and Logging: Habilitar un nivel mínimo de auditoría en su servidor Web y permisos NTFS para proteger los archivos log.
- Script mappings: Remover todos los IIS script mappings por extensiones de archivo opcionales para impedir la explotación de cualquier error en las extensiones ISAPI que manipulan estos tipos de archivos.

¿Cómo defenderse contra ataques de servidor Web?

- Sites and Virtual Directories: Realojar los sitios y directorios virtuales a particiones que no se encuentre el sistema y utilizar permisos IIS Web para restringir el acceso.
- ISAPI Filters: Remover todos los filtros ISAP del servidor Web.

¿Cómo defenderse contra ataques de servidor Web?

- Crear URL mappings para los servidores internos cautelosamente.
- Si un servidor de base de datos como SQL Server será utilizado como DB back-end, instalarlo en un servidor separado.
- Utilizar herramientas de seguridad previstas con el software del servidor Web y escáner para automatizar y facilitar el proceso de asegurar un servidor Web.

¿Cómo defenderse contra ataques de servidor Web?

- - Utilizar del lado del servidor Session ID tracking y combinar conexiones con marcas de tiempo, direcciones IP, etc.
- - No instalar IIS en un DC.
- - No utilizar equipos dedicados como Servidores Web.
- - Filtrar las solicitudes de tráfico de entrada.
- - Realizar protección física al servidor Web en un ambiente seguro.

¿Cómo defenderse contra ataques de servidor Web?

- No configurar cuentas anónimas separadas para cada aplicación, si es que se tienen varias aplicaciones Web.
- Limitar la funcionalidad del servidor con el fin de soportar las tecnologías que serán utilizadas.
- No permitir a nadie iniciar sesión localmente solo al administrador.
- No conectar el IIS a Internet hasta que esté endurecido.

¿Cómo defenderse contra HTTP Response splitting y Web CachePoisoning?

- Server Admin:

1. Utilizar la última versión de software para el servidor.
2. Regularmente actualizar el S.O. el servidor web.
3. Ejecutar escaneo de vulnerabilidades.

¿Cómo defenderse contra HTTP Response splitting y Web CachePoisoning?

- -Desarrolladores de aplicaciones:

1. Restringir el acceso a la aplicación a los únicos.
2. Impedir el retorno (%d or \r) y línea de alimentación (%0a ir \n)
3. Cumplir con las especificaciones RFC 2616 para HTTP/1.1

¿Cómo defenderse contra HTTP Response splitting y Web CachePoisoning?

- Servidores Proxy:

1. Impedir el ingreso de conexiones compartidas TCP entre clientes distintos.
2. Utilizar conexiones TCP distintas entre el proxy para hosts virtuales distintos.
3. Implementar correctamente "maintain request host header"

Administración de Parches

Parches y Hotfixes

Un parche es una pequeña pieza de software para corregir problemas, vulnerabilidades de seguridad y errores.

¿Qué es la administración de parches?

Es el proceso utilizado para asegurar que los parches apropiados son instalados en un sistema para ayudar a corregir vulnerabilidades conocidas.

¿Qué es la administración de parches?

Proceso de administración automatizada de administración de parches:

1. Detect: Utilizar herramientas para detectar parches de seguridad que faltan
2. Assess (evaluar): Evaluar problemas y su severidad mitigando los factores que pueden influenciar en su decisión.
3. Acquire: Descargar el parche para testearlo.

¿Qué es la administración de parches?

4. Test: Instalar el parche primero en un equipo de pruebas para verificar las consecuencias de la actualización.
5. Deploy: Implementar el parche a los equipos y asegurarse que las aplicaciones no se vean afectadas.
6. Maintain: Suscribirse para obtener notificaciones sobre vulnerabilidades reportadas.

Identificando fuentes apropiadas para parches y actualizaciones

Primero hacer un plan de administración de parches que se ajuste al ambiente operacional y objetivos del negocio. Encontrar actualizaciones y parches apropiados en los sitios oficiales de las aplicaciones y del vendedor del S.O. El camino recomendado de hacer un seguimiento a los problemas relevantes para parchar proactivamente es registrándose en los sitios mencionados para recibir alertas.

Instalación de un parche

Los usuarios pueden acceder e instalar los parches de seguridad vía WWW. Los parches pueden ser instalados de dos maneras: Manualmente y automáticamente.

Implementación y verificación de un parche de seguridad

Antes de instalar cualquier parche verificar la fuente. Utilizar el programa de administración de parches apropiado para validar las versiones de archivos y las sumas de comprobación antes de implementar los parches de seguridad. La herramienta de administración de parches debe ser capaz de monitorear los sistemas parchados. El equipo de administración de parches debe revisar regularmente actualizaciones y parches.

Herramientas de parches de seguridad

- Microsoft Baseline Security Analyzer
- Altiris
- Etc.

Herramientas de Seguridad de Servidor Web

- Sandcat: Aplicación de escaneo de vulnerabilidades web remota multi proceso. Mapea toda la estructura Web. También revisa SQL Injection, XSS, File inclusión, etc. Automatiza el proceso de revisar el código de la aplicación web.
- Wikto: Es un escáner de servidores Web para Windows.

Herramientas de Seguridad de Servidor Web

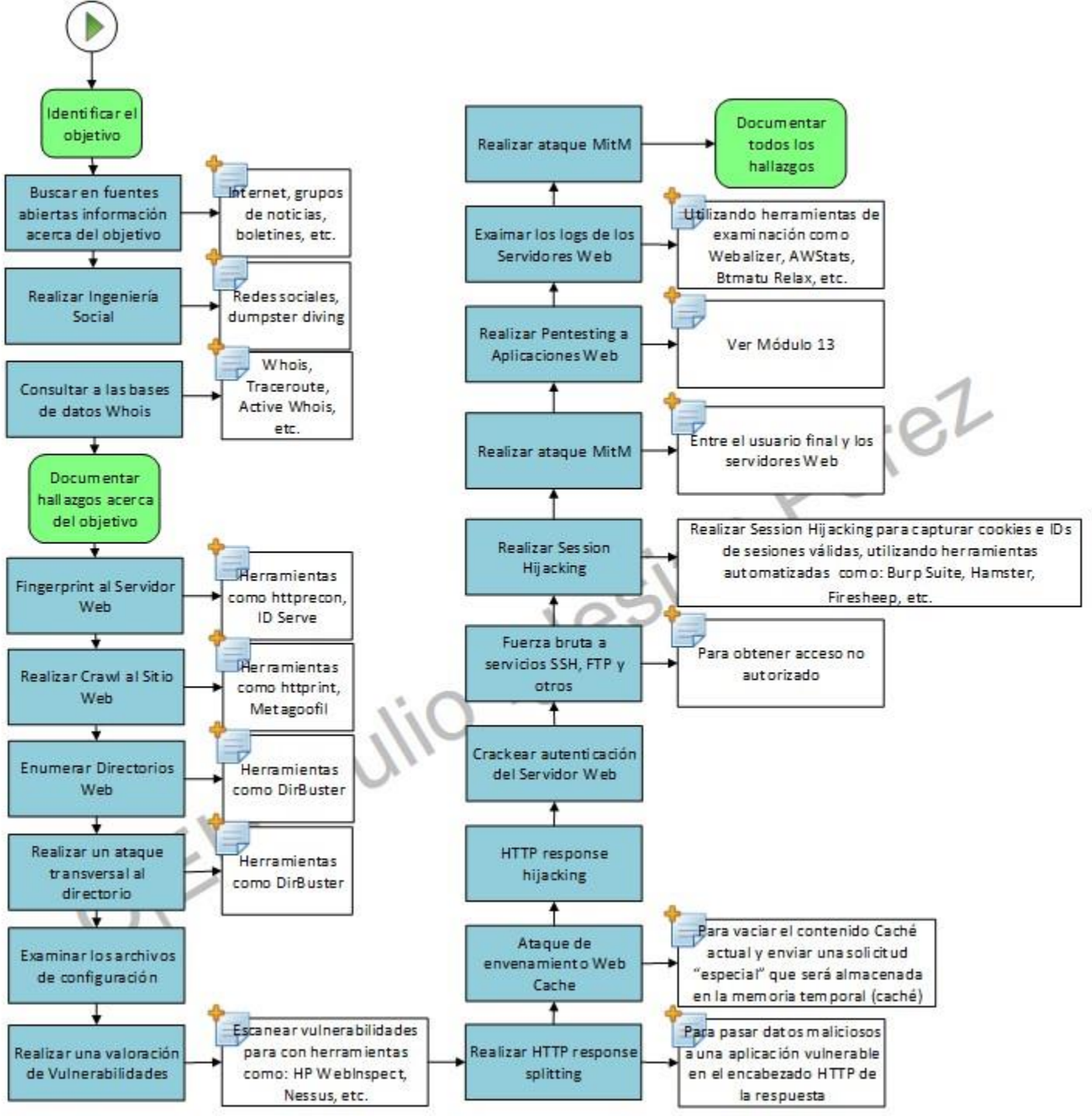
Herramienta de monitoreo de infección Malware: Hackalert: Servicio basado en la nube que provee en tiempo real identificaciones y alarmas para accionamiento de descargas y amenazas malware zero day en sitios y anuncios Online. Identifica el malware antes de que el sitio se ponga como malicioso.

Test de Intrusión a Servidores Web

- Es utilizado para identificar, analizar y reportar vulnerabilidades como debilidad de autenticación, errores de configuración, protocolo relacionado a las vulnerabilidades, etc. en un servidor web.
- La mejor manera de realizar un pen test es realizando una serie de tests metódicos y repetitivos en búsqueda de las distintas vulnerabilidades de aplicaciones Web.

¿Por qué Pen testing a servidores Web?

- Identificación de infraestructura Web.
- Verificación de vulnerabilidades.
- Remediación de vulnerabilidades.


Identificar el objetivo
Buscar en fuentes abiertas información acerca del objetivo
Internet, grupos de noticias, boletines, etc.
Realizar Ingeniería Social
Redes sociales, dumpster diving
Consultar a las bases de datos Whois
Whois, Traceroute, Active Whois, etc.
Documentar hallazgos acerca del objetivo
Fingerprint al Servidor Web
Herramientas como httprecon, ID Serve
Realizar Crawl al Sitio Web
Herramientas como httprint, Metagoofil
Enumerar Directorios Web
Herramientas como DirBuster
Realizar un ataque transversal al directorio
Herramientas como DirBuster
Examinar los archivos de configuración
Realizar una valoración de Vulnerabilidades
Escanear vulnerabilidades para con herramientas como: HP WebInspect, Nessus, etc.
Realizar HTTP response splitting
Para pasar datos maliciosos a una aplicación vulnerable en el encabezado HTTP de la respuesta
Ataque de envenamiento Web Cache
Para vaciar el contenido Caché actual y enviar una solicitud "especial" que será almacenada en la memoria temporal (caché)
HTTP response hijacking
Crackear autenticación del Servidor Web
Fuerza bruta a servicios SSH, FTP y otros
Para obtener acceso no autorizado
Realizar Session Hijacking
Realizar Session Hijacking para capturar cookies e IDs de sesiones válidas, utilizando herramientas automatizadas como: Burp Suite, Hamster, Firesheep, etc.
Realizar ataque MitM
Entre el usuario final y los servidores Web
Realizar Pentesting a Aplicaciones Web
Ver Módulo 13
Examar los logs de los Servidores Web
Utilizando herramientas de examinación como Webalizer, AWStats, Btmatu Relax, etc.
Realizar ataque MitM
Documentar todos los hallazgos


¡Muchas Gracias!
C|EH Julio Iglesias Perez