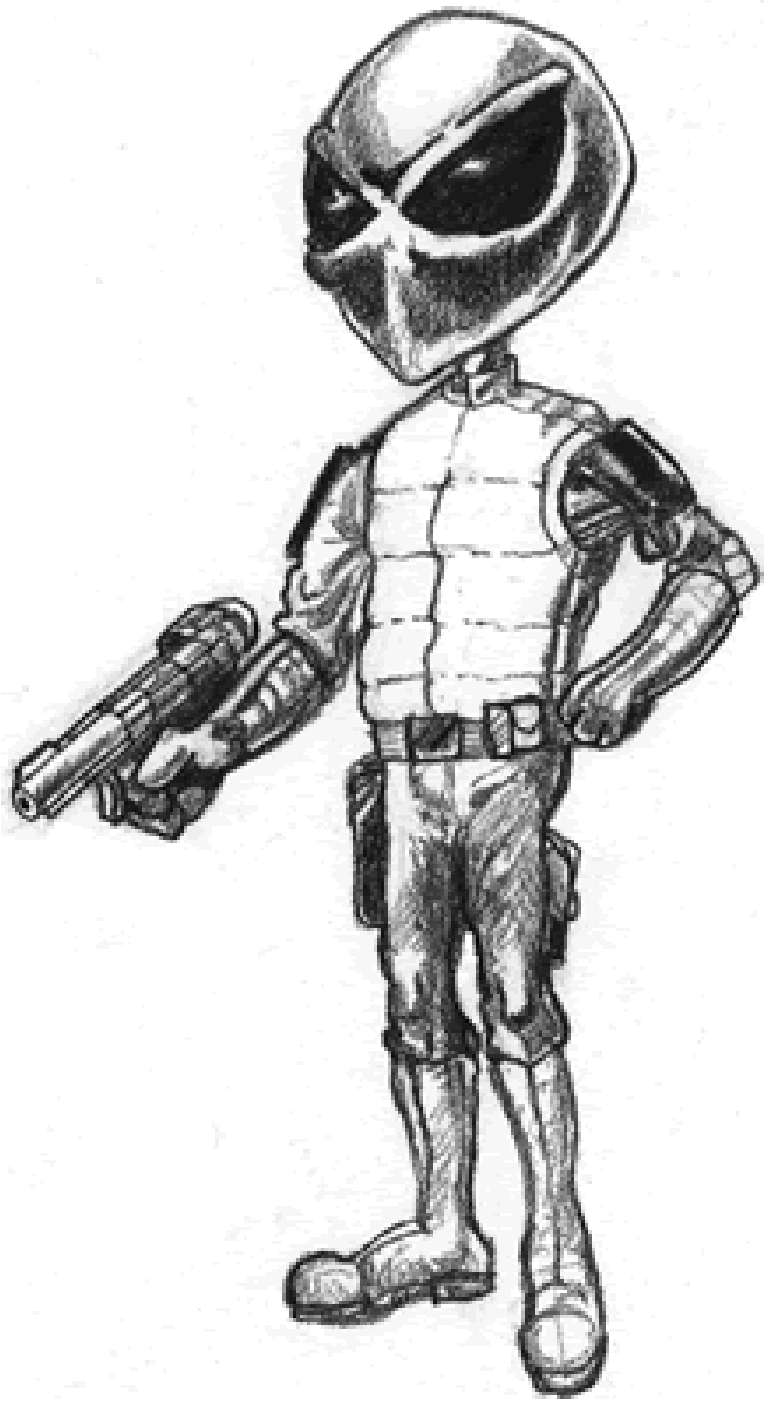
ESCANER DE VULNERABILIDADES WEB NIKTO

Fecha entrega: 11/03/2016

Autor: Pedro J. Ramos

1. **¿Qué es Nikto?**
2. **Versiones**
3. **Requisitos necesarios**
4. **Proceso de instalación**
5. **Archivo de configuración**
6. **Opciones de uso**
7. **Bibliografía**



1. ¿Qué es Nikto?

Nikto es un escáner de vulnerabilidades web gratuito de tipo Open Source, con licencia GPL. Entre sus principales características podemos destacar:

- Comprueba la existencia de elementos desfasados en un servidor.
- Tiene una configuración para usarse a través de un proxy.
- Puede exportar un informe en formato csv, html o xml.
- Permite escanear los puertos de un servidor a través de nmap.
- Se puede configurar el escaneado para centrarnos en lo que nos interesa excluyendo las vulnerabilidades que no nos interesan.
- Su desarrollo sigue en estatus activo
- Está escrito en lenguaje Perl
- Sólo disponible en inglés

2. Versiones

- Para Linux: Nikto 2.1.5
- Para Mac: MacNikto
- Para Windows: Wikto

3. Requisitos necesarios

Para su correcto funcionamiento nuestro sistema debe de contar con:

- Perl
- Openssl
- Libnet-ssleay-perl
- Nmap

4. Proceso de instalación

Si nuestro sistema no está dotado de los requisitos necesarios lo primero que debemos hacer es proceder a su instalación:

apt-get install perl

```
root@server-virtual-machine:/home/server# apt-get install perl
```

apt-get install libnet-ssleay-perl

```
root@server-virtual-machine:/home/server# apt-get install libnet-ssleay-perl
```

apt-get install openssl

```
root@server-virtual-machine:/home/server# apt-get install openssl
```

apt-get install nmap

```
root@server-virtual-machine:/home/server# apt-get install nmap
```

Descargamos y descomprimimos los archivos de nikto:

wget http://www.cirt.net/nikto/nikto-2.1.5.tar.gz

```
root@server-virtual-machine:/home/server# wget http://www.cirt.net/nikto/nikto-2
.1.5.tar.gz
--2016-03-01 13:27:41--  http://www.cirt.net/nikto/nikto-2.1.5.tar.gz
Resolviendo www.cirt.net (www.cirt.net)... 107.170.99.251
Conectando con www.cirt.net (www.cirt.net)[107.170.99.251]:80... conectado.
Petición HTTP enviada, esperando respuesta... 200 OK
Longitud: 371663 (363K) [application/x-gzip]
Grabando a: “nikto-2.1.5.tar.gz”

100%[======================================>] 371.663      450KB/s   en 0,8s

2016-03-01 13:27:42 (450 KB/s) - “nikto-2.1.5.tar.gz” guardado [371663/371663]
```

tar –xvf nikto-2.1.5.tar.gz

```
root@server-virtual-machine:/home/server# tar -xvf nikto-2.1.5.tar.gz
nikto-2.1.5/
nikto-2.1.5/docs/
nikto-2.1.5/docs/nikto.dtd
nikto-2.1.5/docs/CHANGES.txt
nikto-2.1.5/docs/nikto.1
nikto-2.1.5/docs/nikto_manual.html
nikto-2.1.5/docs/LICENSE.txt
nikto-2.1.5/databases/
nikto-2.1.5/databases/db_favicon
nikto-2.1.5/databases/db_subdomains
nikto-2.1.5/databases/db_outdated
nikto-2.1.5/databases/db_parked_strings
nikto-2.1.5/databases/db_multiple_index
nikto-2.1.5/databases/db_embedded
nikto-2.1.5/databases/db_content_search
nikto-2.1.5/databases/db_httpoptions
nikto-2.1.5/databases/db_dictionary
nikto-2.1.5/databases/db_404_strings
nikto-2.1.5/databases/db_headers
nikto-2.1.5/databases/db_realms
nikto-2.1.5/databases/db_tests
nikto-2.1.5/databases/db_variables
nikto-2.1.5/databases/db_server_msgs
nikto-2.1.5/plugins/
nikto-2.1.5/plugins/nikto_subdomain.plugin
nikto-2.1.5/plugins/nikto_clientaccesspolicy.plugin
```

5. Archivo de configuración

Si nos vamos a la carpeta de nikto podemos ver un fichero llamado *nikto.conf* en él se encuentran las configuraciones, y dentro hay diferentes variables:

```
root@server-virtual-machine:/home/server/nikto-2.1.5# cat nikto.conf
```

- SKIPPORTS=21 111 → Puertos que nunca van a ser escaneados
- DEFAULTHTTPVER=1.0 → Versión de HTTP que nikto usará
- UPDATES=no → Hace que se envíe información no sensible al servidor web de cirt.net, consultando la base de datos de nikto para verificar la versión del servidor web que se está escaneando. Si la versión es superior a la que está en el fichero db_outdated o ni si quiera está incluida, se enviarán estos datos al servidor oficial de nikto, para incluir estos datos en nuevas versiones.
- PROXYHOST=127.0.0.1 → Dirección IP del proxy
- PROXYPORT=8080 → Puerto del proxy

- PROXYUSER=proxyuserid → Usuario del proxy
- PROXYPASS=proxypassword → Contraseña del usuario del proxy
- STATIC-COOKIE=cookie1=cookie1value; cookie2=cookie2value;
- CHECKMETHODS=HEAD GET → Nikto intenta detectar objetivos, como servidores web mediante el envío de peticiones donde se busca una URL, gracias a métodos HTTP usados. A veces si el servidor web que se escanea no usa estos métodos, puede dar problemas en la búsqueda, en tal caso podemos dejar solo la opción HEAD
- RFIURL=http://cirt.net/rfiinc.txt? → Completa en un archivo de inclusión de archivos remotos. Este archivo debe contener una llamada a phpinfo (), donde Nikto buscará la salida de ese comando para determinar que el RFI tuvo éxito. Si se usa el archivo de cirt.net defecto, debe haber conectividad desde el servidor de destino a cirt.net
- USERAGENT=Mozilla/5.00 (Nikto/@VERSION) (Evasions:@EVASIONS) (Test:@TESTID) →Determina el user agent de nikto, una buena idea es cambiarlo, ya que la mayoría de los IDS, detectaría el escaneo con facilidad.

6. Opciones de uso

Cuando lanzamos nikto podemos especificar una serie de parámetros por ejemplo:

-maxtime n → Máximo tiempo de ejecución por host, expresado en n segundos (lo podemos cambiar a un valor más alto).

-config url→ Si tenemos otro archivo de configuración, le podemos indicar la ruta.

-Cgidirs all/none→ Escanea los directorios CGI. Con “all” escanea todos los directorios, o “none” para ninguno.

-Display

Muestra las diferentes salidas por pantalla de nikto:

1 – Mostrar las redirecciones. Esto mostrará todas las solicitudes que provocan una respuesta de “reorientar” desde el servidor.

2 – Mostrar las cookies recibidas. Esto mostrará todas las cookies que fueron enviados por el host remoto.

3 – Mostrar todas las respuestas 200/OK.

4 – Mostrar URLs que requieren autenticación.

D – Salida de depuración. Mostrar resultados de depuración, que muestra la salida detallada e información adicional, como el contenido variable.

E – Muestra todos los errores HTTP. Mostrar detalles de cualquier error HTTP encontrado.

P – Imprimir el progreso a STDOUT. Mostrar informe de estado para STDOUT durante la prueba (set interval en nikto.conf).

V – Salida detallada. Mostrar resultados detallados.

-evasion

Habilita la detección de intrusos mediante técnicas de evasión (se pueden usar varias opciones a la vez):

1 – Random URI encoding (non-UTF8)

2 – Directory self-reference (/./)

3 – Premature URL ending.

4 – Prepend long random string.

5 – Fake parameter.

6 – TAB as request spacer.

7 – Change the case of the URL.

8 – Use Windows directory separator (\).

-Format

Define el formato de salida, en un archivo concreto con las diferentes extensiones:

csv – una lista separada por Separado.

htm – un informe HTML.

msf – Ingresar a Metasploit.

txt – un informe de texto.

xml – un informe XML.

-mutate

Especifica una técnica de mutación. Una mutación se usa para combinar las pruebas o intentar adivinar valores. Estas técnicas pueden causar una enorme cantidad de acciones contra el objetivo. Cada número de referencia específica una acción:

1 – Análisis de todos los archivos con todos los directorios raíz. Esto toma cada prueba y la divide en una lista de archivos y directorios. Una lista de exploración es creada mediante la combinación de cada archivo en cada directorio.

2 – Adivina los nombres de archivo de contraseña. Toma una lista de nombres comunes de archivos de contraseñas como "passwd", "pasar", "contraseña" y extensiones de archivos como "txt", "pwd", "bak", etc., y genera una lista de archivos para comprobar.

3 – Enumera los nombres de usuario a través de Apache (/ ~ peticiones de tipo de usuario). Explota una mala configuración de UserDir de la configuración de Apache que permite descubrir a los nombres de usuario válidos.

4 – Enumera los nombres de usuario a través de cgiwrap (/ cgi-bin/cgiwrap/ ~ peticiones de tipo usuario). Explota un fallo en cgiwrap que permite adivinar los nombres de usuario válidos.

5 – Intento de forzar los nombres de subdominios. Tratará por fuerza bruta de conocer los nombres de dominio y se asumirá que el host dado (sin www) es el dominio principal.

6 – Descubrimiento de nombres de directorio por fuerza bruta. Esta es la única opción de mutar que requiere un archivo para ser ejecutado y se pasa como parámetro a la opción

-mutate-options → Se usará el fichero dado para tratar de adivinar los nombres de directorio. Las listas de los directorios comunes se pueden encontrar en el proyecto DirBuster OWASP.

-nocache → Desactivar la caché de respuesta

-nossl → No utilizar SSL para conectarse al servidor.

-no404 → Desactivar la comprobación 404 (archivo no encontrado)

-port → Se especifica el puerto a escanear, se pueden usar varios: 80,443.

-Plugins → Selecciona el plugin que se va a ejecutar: plugin-name [(parameter name [: parameter value], [other parameters])]

-root → Antepone el valor especificado al principio de cada solicitud. Esto es útil para aplicaciones de prueba o servidores web que tienen todos sus archivos bajo un directorio determinado.

-ssl → Sólo probar SSL en los puertos especificados..

-Single → Único modo de respuesta

-Tuning

Se puede utilizar para reducir el número de pruebas realizadas contra un objetivo. Los tipos de prueba pueden ser controlados a nivel individual, especificando su identificador con -T (-Tuningopción). Por ejemplo, usamos las opciones 3 y 4 de tuning:

nikto -h 192.168.1.2 -T 34

Si se pasa una "x" como parámetro entonces esto va a negar todas las pruebas de los tipos siguientes a la x. Por ejemplo se hace la opción 3 y 4 pero no la opción 7 ni la b:

nikto -h 192.168.0.1-T 34x7b

Las opciones de Tuning son:

0 – Subir Archivo. Exploits que permiten que un archivo sea cargado en el servidor de destino.

1 – Archivo interesante / Visto en los registros. Un archivo o un ataque desconocido que se ha visto en los registros del servidor web.

2 – Configuración errónea / Default File. Por defecto los archivos mal configurados o protegidos con una contraseña errónea.

3 – Divulgación de información. Un recurso que revela información sobre un objetivo. Esto podría ser una ruta de sistema de archivos o un nombre de cuenta.

4 – Inyección (XSS / Guión / HTML). Cualquier forma de inyección, incluyendo cross site scripting (XSS) o de contenido (HTML). Esto no incluye la inyección de comandos.

5 – Recuperación de archivos remotos – Inside Root Web. Recursos que permite a los usuarios remotos recuperar los archivos no autorizados en el directorio raíz del servidor web.

6 – Denegación de servicio. Recursos que permiten una denegación de servicio contra la aplicación de destino, el servidor web o el host.

7 – Recuperación de archivos remoto. Permite a los usuarios remotos recuperar archivos no autorizados desde cualquier punto.

8 – Ejecución de comandos / Shell remoto. Recursos permite al usuario ejecutar un comando del sistema o iniciar una shell.

9 – Inyección SQL. Cualquier tipo de ataque que permite inyección SQL.

a – Bypass de Autenticación. Permite al cliente acceder a un recurso que no tiene acceso.

b – Identificación de Software. Software o programas instalados podrían ser identificados positivamente.

c – la inclusión de origen remoto. El software permite la inclusión remota de código fuente.

x – Invertir Opciones de optimización. (excluye las opciones a partir de la "x")

-update Actualización de los plugins y bases de datos directamente desde cirt.net.

-vhost Especifica el encabezado de host que se enviará.

7. Bibliografía

- http://www.securityartwork.es/
- https://en.wikipedia.org
- https://cirt.net/nikto2-docs/ (Ayuda del propio Nikto)
- http://kalilinux.foroactivo.com/t44-manual-nikto-para-kali-linux (Manual nikto para Kali Linux)