



06 de agosto de 2021
Ficha N° 15 NIKTO
CSIRT DE GOBIERNO

Comando de la semana “NIKTO”

I. CONTEXTO

Este documento, denominado, en esta oportunidad, “NIKTO”, tiene como objetivo ilustrar sobre una herramienta que puede ser de utilidad para el lector, a objeto de ir potenciando las capacidades locales de autochequeo, detección simple de vulnerabilidades que están expuestas a internet en sus sitios o sistemas web y, a su vez, la obtención de una verificación de la subsanación de aquellas que se les han sido reportadas, facilitando la interacción con el CSIRT de Gobierno. El objetivo no es reemplazar una auditoria de código o evaluación de vulnerabilidades, sino que establecer capacidades básicas de chequeo y obtención de información de manera rápida para temas específicos, como por ejemplo la verificación de la subsanación de alertas o vulnerabilidades reportadas por “CSIRT GOB CL”. Todas estas herramientas al contar con la posibilidad de ser usadas desde una línea de comando permiten en algún grado la integración dentro de script o lenguajes de automatización o programación como PERL, AWK, Shell Scripting¹, Expect, Python, C, C#, C++, Golang, JavaScript, PowerShell, Ruby, Java, PHP, Elixir, Elm, Go, Dart, DLang, Pony, TypeScript, Kotlin, Nim, OCaml, Q#², Reason, Rust (RustyBuer), Swift entre otros con miras a automatizar estas actividades y concentrar el tiempo de los especialistas en el análisis de los datos para encontrar los problemas relevantes y descartar los falsos positivos.

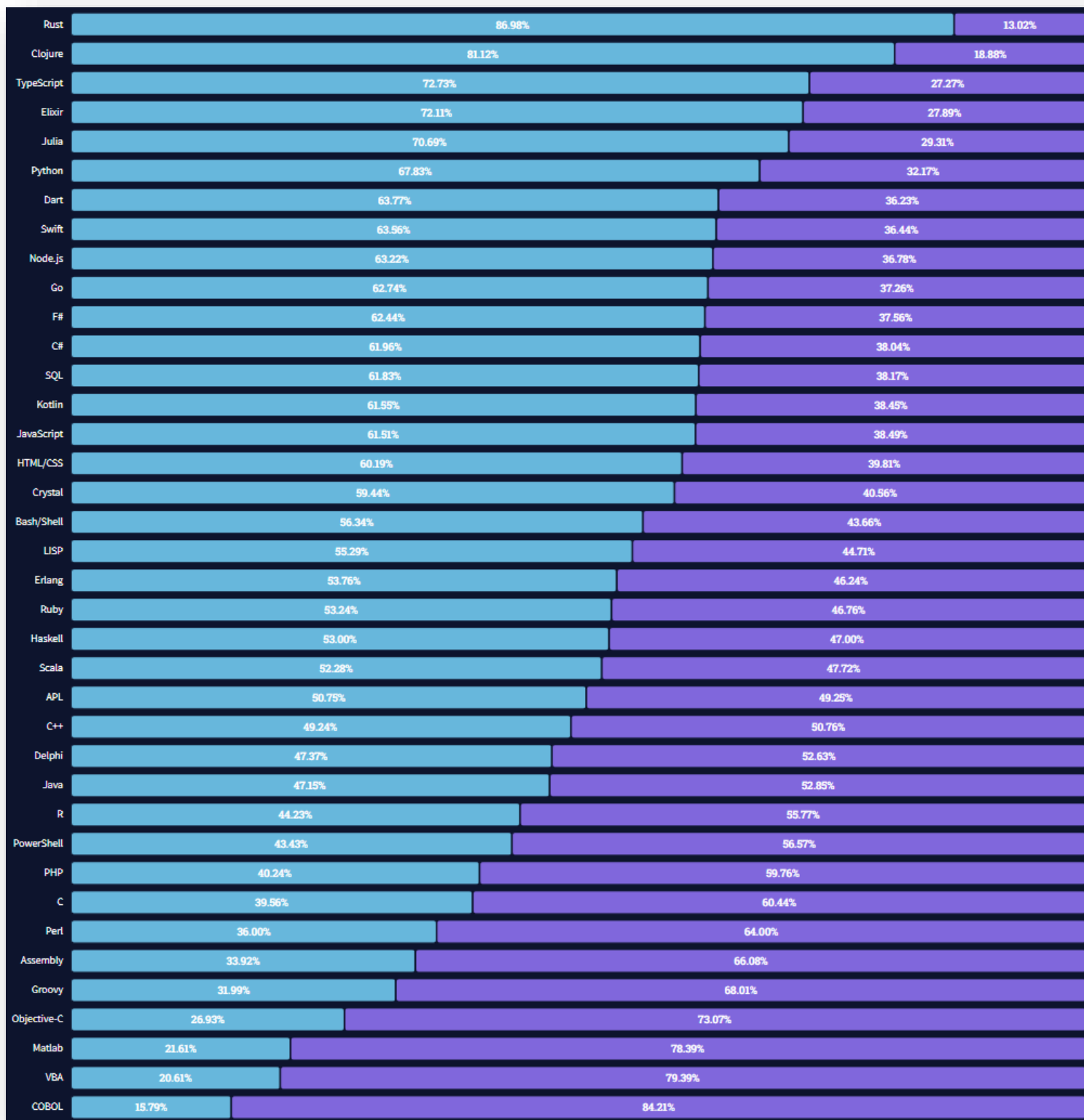
Es importante que conozca al menos lo básico de los lenguajes más nuevos o no convencionales, pues se ha detectado que los desarrolladores de malware van incorporándolos como estrategia de ofuscación, para dificultar la detección y análisis que proveen las soluciones de seguridad.

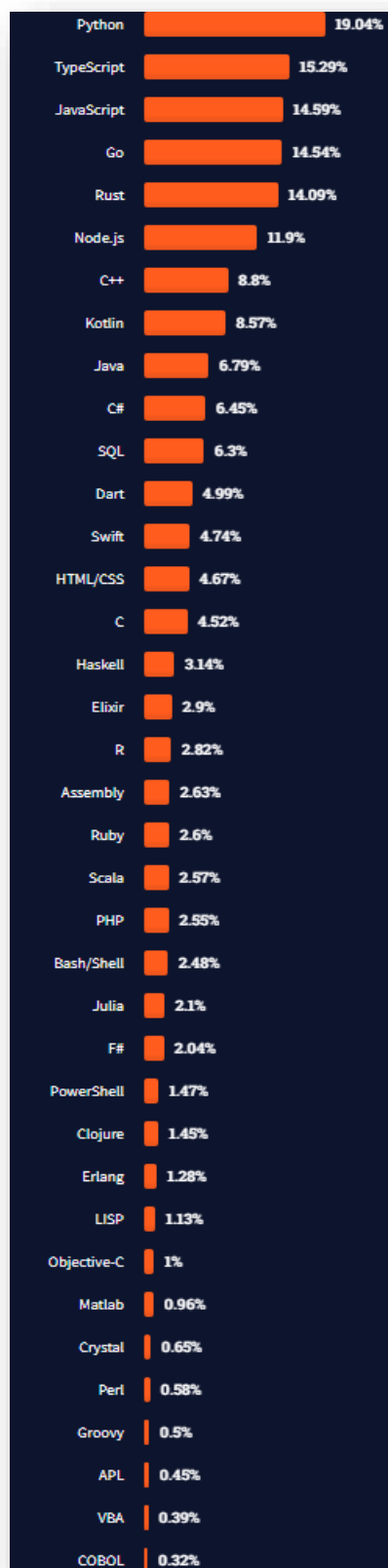
Solo a modo de curiosidad se comparte un gráfico en el que se muestra el resultado de una encuesta entre muchos desarrolladores, dejando ver que lenguajes son más queridos/temidos (primer gráfico) y luego cuales son los más preferidos³ (segundo gráfico).

¹ <https://scis.uohyd.ac.in/~apcs/itw/UNIXProgrammingEnvironment.pdf>

² <https://github.com/Microsoft/QuantumKatas/>

³ <https://insights.stackoverflow.com/survey/2021#most-loved-dreaded-and-wanted-language-love-dread>







II. INTRODUCCIÓN

Una de las tareas regulares que en ciberseguridad se realizan es la verificación de los sitios o sistemas que están expuestos a Internet. Volvemos a reforzar la revisión del componente web de los sitios o sistemas que se compone básicamente por un web Server, un sitio o sistema codificado en alguno de los lenguajes aptos para este ambiente digital y probablemente alguna base de datos.

Algunos de los web server más usados son por ejemplo:

- Nginx Web Server⁴
- Apache HTTP Server⁵
- LiteSpeed Web Server⁶
- Lighttpd Web Server⁷
- Caddy⁸
- Apache Tomcat⁹
- Node.js¹⁰
- OpenLiteSpeed¹¹
- Hiawatha Web Server¹²
- Cherokee¹³
- H2O¹⁴
- IIS¹⁵
- OpenResty¹⁶
- Quark¹⁷

⁴ <https://www.nginx.com/>

⁵ <https://httpd.apache.org/>

⁶ <https://www.litespeedtech.com/products/litespeed-web-server>

⁷ <https://www.lighttpd.net/>

⁸ <https://caddyserver.com/>

⁹ <https://tomcat.apache.org/>

¹⁰ <https://nodejs.org/en/>

¹¹ <https://openlitespeed.org/>

¹² <https://www.hiawatha-webserver.org/>

¹³ <https://cherokee-project.com/>

¹⁴ <https://h2o.example.net/>

¹⁵ <https://www.iis.net/>

¹⁶ <https://openresty.org/>

¹⁷ <https://tools.suckless.org/quark/>



¿Qué es NIKTO?

Nikto es un escáner de servidor web de código abierto (GPL) que realiza pruebas exhaustivas contra servidores web para varios elementos, incluidos más de 6700 archivos / programas potencialmente peligrosos, verifica versiones desactualizadas de más de 1250 servidores y problemas específicos de la versión en más de 270 servidores. También comprueba los elementos de configuración del servidor, como la presencia de varios archivos de índice, las opciones del servidor HTTP e intentará identificar los servidores web y el software instalados. Los elementos de escaneo y los complementos se actualizan con frecuencia y se pueden actualizar automáticamente.

Nikto no está diseñado como una herramienta sigilosa. Probará un servidor web en el menor tiempo posible y es obvio en los archivos de registro o en un IPS / IDS. Sin embargo, hay soporte para los métodos anti-IDS de LibWhisker en caso de que quiera probarlo (o probar su sistema IDS).

No todos los cheques son un problema de seguridad, aunque la mayoría lo son. Hay algunos elementos que son verificaciones de tipo "solo información" que buscan cosas que pueden no tener una falla de seguridad, pero que el webmaster o el ingeniero de seguridad pueden no saber que están presentes en el servidor. Estos elementos suelen estar debidamente marcados en la información impresa. También hay algunas comprobaciones de elementos desconocidos que se han analizado en los archivos de registro.

Características:

- Estas son algunas de las características principales de Nikto. Consulte la documentación para obtener una lista completa de funciones y cómo utilizarlas.
- Soporte SSL (Unix con OpenSSL o quizás Windows con Perl / NetSSL de ActiveState)
- Soporte completo de proxy HTTP
- Comprobaciones de componentes de servidor obsoletos
- Guarde informes en texto sin formato, XML, HTML, NBE o CSV
- Motor de plantillas para personalizar fácilmente los informes
- Escanee varios puertos en un servidor o varios servidores a través del archivo de entrada (incluida la salida nmap)
- Técnicas de codificación IDS de LibWhisker
- Se actualiza fácilmente a través de la línea de comandos
- Identifica el software instalado a través de encabezados, favicons y archivos





- Autenticación de host con Basic y NTLM
- Adivinar subdominio
- Enumeración de nombre de usuario de Apache y cgiwrap
- Técnicas de mutación para "pescar" contenido en servidores web
- Ajuste de escaneo para incluir o excluir clases enteras de verificaciones de vulnerabilidad
- Adivina las credenciales para los reinos de autorización (incluidas muchas combinaciones de ID / pw predeterminadas)
- La adivinación de autorización maneja cualquier directorio, no solo el directorio raíz
- Reducción mejorada de falsos positivos a través de varios métodos: encabezados, contenido de la página y hash de contenido
- Informa que se han visto encabezados "inusuales"
- Estado interactivo, pausa y cambios en la configuración de verbosidad
- Guarde la solicitud / respuesta completa para las pruebas positivas
- Reproducir solicitudes positivas guardadas
- Tiempo máximo de ejecución por objetivo
- Pausa automática en un momento específico
- Verificaciones de sitios de "estacionamiento" comunes
- Iniciar sesión en Metasploit
- Documentación completa

NOTA IMPORTANTE 1: Dado que es relevante un buen manejo de los comandos básicos de Linux, tanto para posteriores manejos de los datos o archivos como para usos de la información resultante de la ejecución de los comandos, es que el comité editorial decidió que se incluya en esta edición y en las subsiguientes un anexo de comandos Linux que son de utilidad para moverse en este sistema operativo. Se sugiere dominarlos todos para facilitar el acceso y manipulación de la información. En futuras ediciones se irán incorporando nociones más avanzadas sobre el uso de estos comandos para procesamiento de archivos, procesos, y de sus usos en scripting.

Vea anexo I: Comandos básicos de Linux

NOTA IMPORTANTE 2: Dado que un altísimo porcentaje de los equipos de usuarios y servidores operando en un entorno Windows, el comité editorial ha decidido ir incorporando “tips” para este entorno computacional.

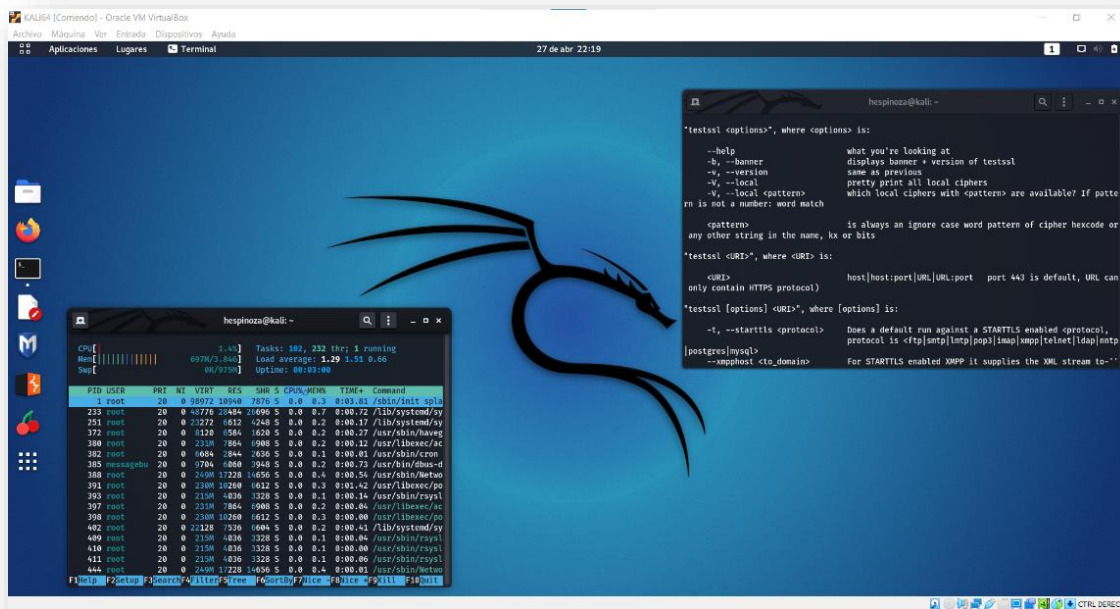
Vea anexo II: Comandos o aplicativos básicos para Windows: TCPView



III. PASO A PASO

PASO 1: Un entorno adecuado para trabajar.

Primero debe contar con una distribución de Kali¹⁸ Linux funcionando ya sea en una máquina física o en una máquina virtual^{19,20}.



Instalación de Kali Linux

La instalación de Kali Linux (arranque único) en su computadora es un proceso sencillo. Esta guía cubrirá la instalación básica (que se puede realizar en una máquina virtual invitada o sobre un equipo entero), con la opción de cifrar la partición. En ocasiones, es posible que tenga datos confidenciales que preferiría cifrar con Full Disk Encryption (FDE). Durante el proceso de instalación, puede iniciar una instalación cifrada LVM en el disco duro o en las unidades USB.

Primero, necesitará hardware de computadora compatible. Kali Linux es compatible con plataformas amd64 (x86_64 / 64-Bit) e i386 (x86 / 32-Bit). Siempre que sea posible, el fabricante recomienda utilizar las imágenes amd64. Los requisitos de hardware son mínimos como se enumeran en la

¹⁸ <https://www.kali.org/downloads/>

¹⁹

https://my.vmware.com/en/web/vmware/downloads/info/slug/desktop_end_user_computing/vmware_workstation_player/16_0

²⁰ <https://www.virtualbox.org/wiki/Downloads>



sección siguiente, aunque un mejor hardware naturalmente proporcionará un mejor rendimiento. Debería poder usar Kali Linux en hardware más nuevo con UEFI y sistemas más antiguos con BIOS.

Las imágenes i386, de forma predeterminada, utilizan un kernel PAE, por lo que puede ejecutarlas en sistemas con más de 4 GB de RAM.

En el ejemplo que se menciona más adelante, se instalará Kali Linux en una nueva máquina virtual invitada, sin ningún sistema operativo existente preinstalado.

Requisitos del sistema

Los requisitos de instalación para Kali Linux variarán según lo que le gustaría instalar y su configuración. Para conocer los requisitos del sistema:





En el extremo inferior, puede configurar Kali Linux como un servidor Secure Shell (SSH) básico sin escritorio, utilizando tan solo 128 MB de RAM (se recomiendan 512 MB) y 2 GB de espacio en disco.

En el extremo superior, si opta por instalar el escritorio Xfce4 predeterminado y el kali-linux-default metapaquete, realmente debería apuntar a al menos 2 GB de RAM y 20 GB de espacio en disco.

Cuando se utilizan aplicaciones que consumen muchos recursos, como Burp Suite, recomiendan al menos 8 GB de RAM (¡e incluso más si se trata de una aplicación web grande!) O utilizar programas simultáneos al mismo tiempo.

Requisitos previos de instalación²¹

Esta la guía se harán las siguientes suposiciones al instalar Kali Linux:

-  Usando la imagen del instalador de amd64.
-  Unidad de CD / DVD / soporte de arranque USB.
-  Disco único para instalar.
-  Conectado a una red (con DHCP y DNS habilitados) que tiene acceso a Internet saliente.

Preparación para la instalación




-  Descargue Kali Linux²² (el fabricante recomienda²³ la imagen marcada como Instalador).

²¹ Dependiendo del tipo de instalación que seleccione, se pueden borrar todos los datos existentes en el disco duro, así que haga una copia de seguridad de la información importante del dispositivo en un medio externo.

²² <https://www.kali.org/docs/introduction/download-official-kali-linux-images/>

²³ <https://www.kali.org/docs/introduction/what-image-to-download/#which-image-to-choose>



-  Grabe²⁴ la ISO de Kali Linux en un DVD o una imagen de Kali Linux Live en una unidad USB. (Si no puede, consulte la instalación en red²⁵ de Kali Linux).
-  Realice una copia de seguridad de la información importante del dispositivo en un medio externo.
-  Asegúrese de que su computadora esté configurada para arrancar desde CD / DVD / USB en su BIOS / UEFI.

Un vez que tiene preparado todos los materiales y el entorno para comenzar la instalación siga los pasos indicados en la sección “Kali Linux Installation Procedure” del siguiente enlace:

<https://www.kali.org/docs/installation/hard-disk-install/>



²⁴ <https://www.kali.org/docs/usb/live-usb-install-with-windows/>

²⁵ <https://www.kali.org/docs/installation/network-pxe/>



PASO 2: Instalar el comando.

Una vez que se cuenta con este sistema operativo de manera funcional podemos instalar los comandos; algunos ya vienen preinstalados en la distribución KALI²⁶, pero si no fuere así puede instalarlos con los siguientes comandos, **previamente tomando privilegios de usuario “root”**:

```
# apt update && apt full-upgrade
```

```
# apt install nikto
```

```
# apt search ^nikto
```

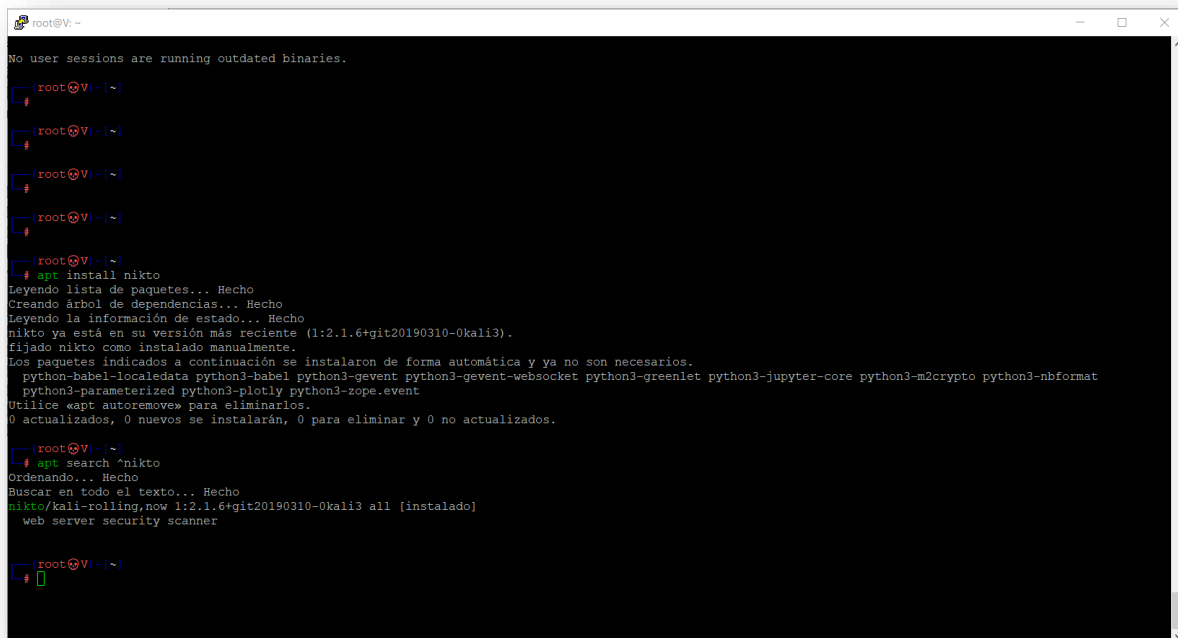
```
Ordenando... Hecho
```

```
Buscar en todo el texto... Hecho
```

```
nikto/kali-rolling,now 1:2.1.6+git20190310-0kali3 all [instalado]
```

```
web server security scanner
```

Nota: el símbolo “^” indica que busque los patrones que comienza por lo indicado en la línea de comando.



```
root@V: ~  
No user sessions are running outdated binaries.  
[root@V: ~]  
[root@V: ~]  
#  
[root@V: ~]  
#  
[root@V: ~]  
#  
[root@V: ~]  
# apt install nikto  
Leyendo lista de paquetes... Hecho  
Creando árbol de dependencias... Hecho  
Leyendo la información de estado... Hecho  
nikto ya está en su versión más reciente (1:2.1.6+git20190310-0kali3).  
fijado nikto como instalado manualmente.  
Los paquetes indicados a continuación se instalaron de forma automática y ya no son necesarios.  
python3-babel-localizedata python3-babel python3-gevent python3-gevent-websocket python3-greenlet python3-jupyter-core python3-m2crypto python3-nbformat  
python3-parameterized python3-plotly python3-zope.event  
Utilice «apt autoremove» para eliminarlos.  
0 actualizados, 0 nuevos se instalarán, 0 para eliminar y 0 no actualizados.  
[root@V: ~]  
# apt search ^nikto  
Ordenando... Hecho  
Buscar en todo el texto... Hecho  
nikto/kali-rolling,now 1:2.1.6+git20190310-0kali3 all [instalado]  
web server security scanner  
[root@V: ~]  
#
```

²⁶ <https://www.offensive-security.com/kali-linux-vm-vmware-virtualbox-image-download/>



PASO3: Verificar su instalación.

Una vez que se ha instalado podemos verificar y explorar las múltiples opciones que ofrece para su ejecución:

En una consola de su KALI, dentro del directorio donde quedó instalada la aplicación, ejecute el comando para que muestre la ayuda: “nikto -h”.

```
root@V: ~  
(root@V) - [~]  
# nikto -h  
Option host requires an argument  
  
-config+      Use this config file  
-Display+     Turn on/off display outputs  
-dbcheck      check database and other key files for syntax errors  
-Format+      save file (-o) format  
-Help         Extended help information  
-host+        target host/URL  
-id+          Host authentication to use, format is id:pass or id:pass:realm  
-list-plugins List all available plugins  
-output+      Write output to this file  
-noss1        Disables using SSL  
-no404        Disables 404 checks  
-Plugins+     List of plugins to run (default: ALL)  
-port+        Port to use (default 80)  
-root+        Prepend root value to all requests, format is /directory  
-ssl          Force ssl mode on port  
-Tuning+      Scan tuning  
-timeout+     Timeout for requests (default 10 seconds)  
-update       Update databases and plugins from CIRT.net  
-Version      Print plugin and database versions  
-vhost+       Virtual host (for Host header)  
              + requires a value  
  
Note: This is the short help output. Use -H for full help text.  
  
(root@V) - [~]  
#
```

Debiéramos, entonces, lograr desplegar todas las opciones y parámetros de ejecución, junto a su explicación en la consola con el comando “nikto -H”.



```
# nikto -H

Options:
  -ask+                Whether to ask about submitting updates
                        yes   Ask about each (default)
                        no    Don't ask, don't send
                        auto  Don't ask, just send
  -Cgidirs+            Scan these CGI dirs: "none", "all", or values like
"/cgi/ /cgi-a/"
  -config+             Use this config file
  -Display+            Turn on/off display outputs:
                        1     Show redirects
                        2     Show cookies received
                        3     Show all 200/OK responses
                        4     Show URLs which require authentication
                        D     Debug output
                        E     Display all HTTP errors
                        P     Print progress to STDOUT
                        S     Scrub output of IPs and hostnames
                        V     Verbose output
  -dbcheck             Check database and other key files for syntax errors
  -evasion+            Encoding technique:
                        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 (\)
                        A     Use a carriage return (0x0d) as a request
spacer
                        B     Use binary value 0x0b as a request spacer
  -Format+            Save file (-o) format:
                        csv   Comma-separated-value
                        json  JSON Format
                        htm   HTML Format
                        nbe   Nessus NBE format
                        sql   Generic SQL (see docs for schema)
                        txt   Plain text
                        xml   XML Format
                        (if not specified the format will be taken from
the file extension passed to -output)
  -Help               Extended help information
  -host+              Target host/URL
  -404code            Ignore these HTTP codes as negative responses (always).
Format is "302,301".
  -404string          Ignore this string in response body content as negative
response (always). Can be a regular expression.
  -id+                Host authentication to use, format is id:pass or
id:pass:realm
  -key+               Client certificate key file
  -list-plugins        List all available plugins, perform no testing
  -maxtime+           Maximum testing time per host (e.g., 1h, 60m, 3600s)
  -mutate+            Guess additional file names:
```



```

1      Test all files with all root directories
2      Guess for password file names
3      Enumerate user names via Apache (/~user type
requests)
4      Enumerate user names via cgiwrap (/cgi-
bin/cgiwrap/~user type requests)
5      Attempt to brute force sub-domain names,
assume that the host name is the parent domain
6      Attempt to guess directory names from the
supplied dictionary file
-mutate-options    Provide information for mutates
-nointeractive     Disables interactive features
-nolookup          Disables DNS lookups
-nossl             Disables the use of SSL
-no404             Disables nikto attempting to guess a 404 page
-Option           Over-ride an option in nikto.conf, can be issued
multiple times
-output+          Write output to this file ('.' for auto-name)
-Pause+           Pause between tests (seconds, integer or float)
-Plugins+         List of plugins to run (default: ALL)
-port+            Port to use (default 80)
-RSAcert+         Client certificate file
-root+            Prepend root value to all requests, format is /directory
-Save             Save positive responses to this directory ('.' for
auto-name)
-ssl              Force ssl mode on port
-Tuning+          Scan tuning:
1      Interesting File / Seen in logs
2      Misconfiguration / Default File
3      Information Disclosure
4      Injection (XSS/Script/HTML)
5      Remote File Retrieval - Inside Web Root
6      Denial of Service
7      Remote File Retrieval - Server Wide
8      Command Execution / Remote Shell
9      SQL Injection
0      File Upload
a      Authentication Bypass
b      Software Identification
c      Remote Source Inclusion
d      Webservice
e      Administrative Console
x      Reverse Tuning Options (i.e., include all
except specified)
-timeout+         Timeout for requests (default 10 seconds)
-Userdbs          Load only user databases, not the standard databases
all      Disable standard dbs and load only user dbs
tests    Disable only db_tests and load udb_tests
-useragent        Over-rides the default useragent
-until            Run until the specified time or duration
-update           Update databases and plugins from CIRT.net
-url+            Target host/URL (alias of -host)
-useproxy         Use the proxy defined in nikto.conf, or argument
http://server:port
-Version          Print plugin and database versions
-vhost+          Virtual host (for Host header)
+ requires a value

```



Paso 4: Ponerlo en marcha para verificar nuestra infraestructura.

Un ejemplo de ejecución básica para nuestros primeros pasos:

Probaremos el comando OSCANNER con nuestro KALI en un ataque un sitio web determinado:

EJEMPLO 1 NIKTO

Analizamos el webserver "192.168.0.102" y generamos un reporte "report.html"

```
# nikto -Display 1234EP -o report.html -Format htm -Tuning 123bde -host 192.168.0.102
- Nikto v2.1.6
```

```
-----
+ IP de destino: 192.168.0.102
+ Nombre de host de destino: 192.168.0.102
+ Puerto de destino: 80
+ Hora de inicio: 2018-03-23 10:49:04 (GMT0)
```

```
-----
+ Servidor: Apache / 2.2.22 (Ubuntu)
+ El servidor filtra inodos a través de ETags, encabezado encontrado con el archivo /, inodo: 287, tamaño: 11832, mtime : Viernes 2 de febrero 15:27:56 2018
+ El encabezado de opciones de X-Frame anti-clickjacking no está presente.
+ El encabezado X-XSS-Protection no está definido. Este encabezado puede sugerirle al agente de usuario que se proteja contra algunas formas de XSS
+. El encabezado X-Content-Type-Options no está configurado. Esto podría permitir al agente de usuario representar el contenido del sitio de una manera diferente al tipo MIME
+ No se encontraron directorios CGI (use '-C all' para forzar la verificación de todos los directorios posibles)
+ "robots.txt" contiene 1 entrada que debe visualizarse manualmente.
+ Se encontró un encabezado poco común 'tcn', con contenido: list
+ Apache mod_negotiation está habilitado con MultiViews, lo que permite a los atacantes usar nombres de archivos de fuerza bruta fácilmente. Véase http://www.wisec.it/sectou.php?id=4698ebdc59d15. Se encontraron las siguientes alternativas para 'index': index.html
+ Apache / 2.2.22 parece estar desactualizado (el actual es al menos Apache / 2.4.12). Apache 2.0.65 (versión final) y 2.2.29 también están actualizados.
+ Métodos HTTP permitidos: GET, HEAD, POST, OPTIONS
+ 371 solicitudes: 0 error (s) y 9 elementos informados en el host remoto
+ Hora de finalización: 2018-03-23 10:50:44 (GMT0) (100 segundos)
```

```
-----
+ 1 host (s) probado (s)
root @ kali: ~ #
```

Luego el reporte se puede visualizar con un browser como Firefox u otro:

```
root @ kali: ~ # firefox report.html
```



Otro ejemplo y como se vería en un browser el archivo report.html:

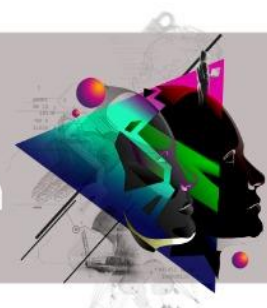
EJEMPLO 2 NIKTO

Analizamos el webserver "192.168.0.102" y generamos un reporte "report.html"

```
# nikto -Display 1234EP -o report.html -Format htm -Tuning 123bde -host  
www.csirt.gob.cl  
- Nikto v2.1.6
```

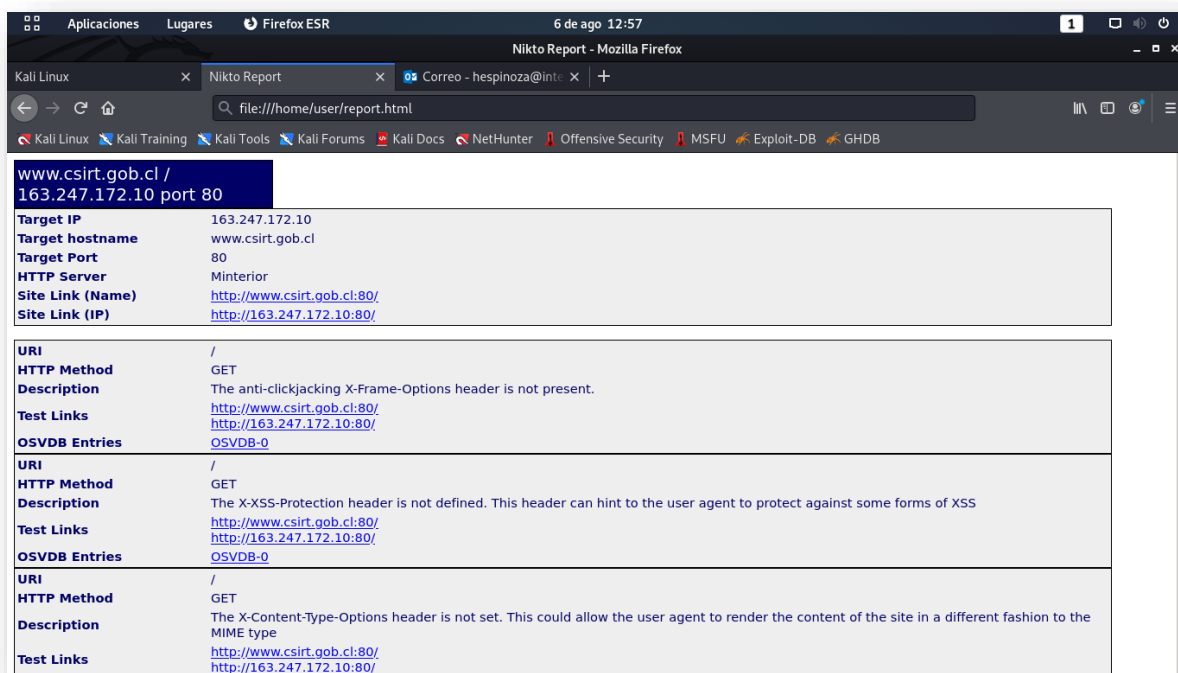
```
+ Target IP:          163.247.172.10  
+ Target Hostname:    www.csirt.gob.cl  
+ Target Port:        80  
+ Start Time:         2021-08-06 12:35:19 (GMT-4)
```

```
+ Server: No banner retrieved  
+ The anti-clickjacking X-Frame-Options header is not present.  
+ The X-XSS-Protection header is not defined. This header can hint to the user  
agent to protect against some forms of XSS  
+ The X-Content-Type-Options header is not set. This could allow the user agent  
to render the content of the site in a different fashion to the MIME type  
+ No CGI Directories found (use '-C all' to force check all possible dirs)  
+ Server banner has changed from '' to 'Minterior' which may suggest a WAF, load  
balancer or proxy is in place  
+ Multiple index files found: /index.pl, /index.htm, /index.php5, /index.html,  
/index.cfm, /index.php7, /index.cgi, /index.php4, /index.aspx, /index.xml,  
/index.php3, /index.asp, /default.htm, /index.shtml, /index.jhtml,  
/default.aspx, /index.do, /index.jsp, /default.asp, /index.php  
E:Fri Aug 6 12:36:58 2021 + ERROR: returned an error: error reading HTTP  
response  
E:Fri Aug 6 12:37:18 2021 + ERROR: returned an error: error reading HTTP  
response  
E:Fri Aug 6 12:37:38 2021 + ERROR: returned an error: error reading HTTP  
response  
E:Fri Aug 6 12:37:59 2021 + ERROR: returned an error: error reading HTTP  
response  
E:Fri Aug 6 12:38:19 2021 + ERROR: returned an error: error reading HTTP  
response  
E:Fri Aug 6 12:38:39 2021 + ERROR: returned an error: error reading HTTP  
response  
E:Fri Aug 6 12:38:59 2021 + ERROR: returned an error: error reading HTTP  
response  
E:Fri Aug 6 12:39:19 2021 + ERROR: returned an error: error reading HTTP  
response  
E:Fri Aug 6 12:39:46 2021 + ERROR: returned an error: error reading HTTP  
response  
E:Fri Aug 6 12:40:06 2021 + ERROR: returned an error: error reading HTTP  
response  
E:Fri Aug 6 12:40:26 2021 + ERROR: returned an error: error reading HTTP  
response  
E:Fri Aug 6 12:40:46 2021 + ERROR: returned an error: error reading HTTP  
response  
E:Fri Aug 6 12:41:06 2021 + ERROR: returned an error: error reading HTTP  
response  
E:Fri Aug 6 12:41:26 2021 + ERROR: returned an error: error reading HTTP  
response
```

```
E:Fri Aug 6 12:41:46 2021 + ERROR: returned an error: error reading HTTP
response
E:Fri Aug 6 12:42:06 2021 + ERROR: returned an error: error reading HTTP
response
E:Fri Aug 6 12:42:26 2021 + ERROR: returned an error: error reading HTTP
response
E:Fri Aug 6 12:42:46 2021 + ERROR: returned an error: error reading HTTP
response
+ ERROR: Error limit (20) reached for host, giving up. Last error: error reading
HTTP response
E:Fri Aug 6 12:43:06 2021 + ERROR: returned an error: error reading HTTP
response
+ Scan terminated: 19 error(s) and 4 item(s) reported on remote host
+ End Time: 2021-08-06 12:43:06 (GMT-4) (467 seconds)
-----
+ 1 host(s) tested
```

Luego de que ha finalizado la ejecución del comando, podemos revisar el reporte final con Firefox, por ejemplo, y analizar que debilidades encontró en nuestro sistema para corregirlas a la brevedad posible.



Tenga presente que es importante que estas pruebas deben ser coordinadas con el equipo de operaciones y en ambientes que estén bajo supervisión.



Antes de proceder a aplicar estos comandos revise sus políticas de seguridad de la información interna, sus códigos de ética, los NDA que haya suscrito y las cláusulas de confidencialidad de su contrato de trabajo.

Defina horarios especiales o ambientes de “test o QA” equivalentes a los de “producción”, para mitigar los posibles efectos perjudiciales en los dispositivos de seguridad, el sitio o el sistema web.

Estudie las múltiples opciones de los comandos ilustrados en esta ficha, entienda el significado de sus diferentes parámetros con el objetivo de obtener resultados específicos, para diferentes escenarios de carga o redirigir la salida a un archivo, para su inclusión en informes posteriores.

Tenga presente que para el procesamiento y análisis de los datos es relevante que vaya perfeccionando su manejo de LINUX y comandos PowerShell²⁷ (si es un usuario de Windows).

En próximas ediciones se irán reforzando estos aspectos para facilitar el manejo de los datos y resultados obtenidos, logrando así una mejor comunicación con sus equipos TIC y con el CSIRT de Gobierno.

En caso de cualquier inquietud no dude en consultarnos a soc-csirt@interior.gob.cl.

Si encuentra algún error en el documento también es importante que nos lo comunique para introducir las correcciones pertinentes en las versiones futuras de esta ficha.

²⁷ <https://devblogs.microsoft.com/scripting/table-of-basic-powershell-commands/>



Anexo I: Comandos Básicos de Linux

Comandos básicos

Los comandos son esencialmente los mismos que cualquier sistema UNIX. En las tablas que se presentan a continuación se tiene la lista de comandos más frecuentes.

1. comando “pwd2

Use el comando `pwd` para averiguar la ruta del directorio de trabajo actual (carpeta) en la que se encuentra. El comando devolverá una ruta absoluta (completa), que es básicamente una ruta de todos los directorios que comienza con una barra inclinada (/). Un ejemplo de ruta absoluta es `/home / username`.

2. comando “cd”

Para navegar por los archivos y directorios de Linux, use el comando `cd`. Requiere la ruta completa o el nombre del directorio, según el directorio de trabajo actual en el que se encuentre.

Digamos que estás en `/home / username / Documents` y quieres ir a `Photos`, un subdirectorio de `Documents`. Para hacerlo, simplemente escriba el siguiente comando: `cd Photos`.

Otro escenario es si desea cambiar a un directorio completamente nuevo, por ejemplo, `/home / username / Movies`. En este caso, debe escribir `cd` seguido de la ruta absoluta del directorio: `cd /home / username / Movies`.

Hay algunos atajos que le ayudarán a navegar rápidamente:

- `cd ..` (con dos puntos) para mover un directorio hacia arriba
- `cd` para ir directamente a la carpeta de inicio
- `cd-` (con un guion) para ir a su directorio anterior

En una nota al margen, el shell de Linux distingue entre mayúsculas y minúsculas. Por lo tanto, debe escribir el directorio del nombre exactamente como está.

3. comando “ls”

El comando `ls` se usa para ver el contenido de un directorio. De forma predeterminada, este comando mostrará el contenido de su directorio de trabajo actual.



Si desea ver el contenido de otros directorios, escriba ls y luego la ruta del directorio. Por ejemplo, ingrese ls / home / username / Documents para ver el contenido de Documents.

Hay variaciones que puede usar con el comando ls:

- ls -R también listará todos los archivos en los subdirectorios
- ls -a mostrará los archivos ocultos
- ls -al enumerará los archivos y directorios con información detallada como los permisos, el tamaño, el propietario, etc.

4. comando de “cat”

cat (abreviatura de concatenar) es uno de los comandos más utilizados en Linux. Se utiliza para enumerar el contenido de un archivo en la salida estándar (stdout). Para ejecutar este comando, escriba cat seguido del nombre del archivo y su extensión. Por ejemplo: cat file.txt.

Aquí hay otras formas de usar el comando cat :

- “cat > filename” crea un nuevo archivo
- “cat filename1 filename2> filename3” une dos archivos (1 y 2) y almacena la salida de ellos en un nuevo archivo (3)
- convertir un archivo a mayúsculas o minúsculas, “cat filename | tr az AZ> salida.txt”.

5. comando “cp”

Utilice el comando cp para copiar archivos del directorio actual a un directorio diferente. Por ejemplo, el comando cp scenery.jpg / home / username / Pictures crearía una copia de paisaje.jpg (de su directorio actual) en el directorio de Pictures.

6. comando “mv”

El uso principal del comando mv es mover archivos, aunque también se puede usar para cambiar el nombre de los archivos.

Los argumentos en mv son similares al comando cp. Debe escribir mv, el nombre del archivo y el directorio de destino. Por ejemplo: mv file.txt / home / username / Documents.



Para cambiar el nombre de los archivos, el comando de Linux es “mv oldname.ext newname.ext”.

7. comando mkdir

Utilice el comando mkdir para crear un nuevo directorio; si escribe mkdir Music, se creará un directorio llamado Music.

También hay comandos adicionales de mkdir:

- Para generar un nuevo directorio dentro de otro directorio, use este comando básico de Linux mkdir Music / Newfile
- use la opción p (padres) para crear un directorio entre dos directorios existentes. Por ejemplo, mkdir -p Music / 2020 / Newfile creará el nuevo archivo “2020”.

8. comando “rmdir”

Si necesita eliminar un directorio, use el comando rmdir. Sin embargo, rmdir solo le permite eliminar directorios vacíos.

9. comando “rm”

El comando rm se usa para eliminar directorios y su contenido. Si solo desea eliminar el directorio, como alternativa a rmdir, use rm -r.

Nota: Tenga mucho cuidado con este comando y verifique dos veces en qué directorio se encuentra. Esto eliminará todo y no se puede deshacer.

10. comando “touch”

El comando touch le permite crear un nuevo archivo en blanco a través de la línea de comandos de Linux. Como ejemplo, ingrese touch /home/username/Documents/Web.html para crear un archivo HTML titulado Web en el directorio Documentos.



11. comando “locate”

Puede usar este comando para ubicar o localizar un archivo, al igual que el comando de búsqueda en Windows. Además, el uso del argumento -i junto con este comando hará que no distinga entre mayúsculas y minúsculas, por lo que puede buscar un archivo incluso si no recuerda su nombre exacto.

Para buscar un archivo que contenga dos o más palabras, use un asterisco (*). Por ejemplo, el comando “locate -i escuela*nota” buscará cualquier archivo que contenga la palabra "escuela" y "nota", ya sea en mayúsculas o minúsculas.

12. comando “find”

Similar al comando “locate”, el uso de “find” también busca archivos y directorios. La diferencia es que el comando “find” se usa para ubicar archivos dentro de un directorio determinado.

Como ejemplo, el comando `find / home / -name notes.txt` buscará un archivo llamado notes.txt dentro del directorio de inicio y sus subdirectorios.

Otras variaciones al usar el hallazgo son:

- Para buscar archivos en el directorio actual, “find. -nombre notes.txt”
- Para buscar directorios desde la raíz, llamados home, use “find / -type d -name home”

13. comando “grep”

Otro comando básico de Linux que sin duda es útil para el uso diario es grep. Te permite buscar en todo el texto de un archivo determinado.

Para ilustrar, `grep blue notepad.txt` buscará la palabra azul en el archivo del bloc de notas. Las líneas que contienen la palabra buscada se mostrarán completamente.

14. comando “sudo”

Abreviatura de " SuperUser Do ", este comando le permite realizar tareas que requieren permisos administrativos o de root. Sin embargo, no es recomendable utilizar este comando para el uso diario porque podría ser fácil que ocurra un error si hiciste algo mal.



15. comando “df”

Utilice el comando df para obtener un informe sobre el uso de espacio en disco del sistema, que se muestra en porcentaje y KB. Si desea ver el informe en megabytes, escriba df -m.

16. comando “du”

Si desea comprobar cuánto espacio ocupa un archivo o un directorio, el comando du (Uso del disco) es la respuesta. Sin embargo, el resumen de uso del disco mostrará los números de bloque de disco en lugar del formato de tamaño habitual. Si desea verlo en bytes, kilobytes y megabytes, agregue el argumento -h a la línea de comando.

17. comando “head”

El comando head se usa para ver las primeras líneas de cualquier archivo de texto. De forma predeterminada, mostrará las primeras diez líneas, pero puede cambiar este número a su gusto. Por ejemplo, si solo desea mostrar las primeras cinco líneas, escriba head -n 5 filename.ext.

18. comando “tail”

Este tiene una función similar al comando head, pero en lugar de mostrar las primeras líneas, el comando tail mostrará las últimas diez líneas de un archivo de texto. Por ejemplo, tail -n filename.ext.

19. comando “diff”

Abreviatura de diferencia, el comando diff compara el contenido de dos archivos línea por línea. Después de analizar los archivos, generará las líneas que no coinciden. Los programadores suelen utilizar este comando cuando necesitan realizar modificaciones en el programa en lugar de reescribir todo el código fuente.

La forma más simple de este comando es diff file1.ext file2.ext



20. comando “tar”

El comando tar es el comando más utilizado para archivar varios archivos en un tarball, un formato de archivo común de Linux que es similar al formato zip, con la compresión opcional.

Este comando es bastante complejo con una larga lista de funciones, como agregar nuevos archivos a un archivo existente, enumerar el contenido de un archivo, extraer el contenido de un archivo y muchas más. Consulte algunos ejemplos prácticos para saber más sobre otras funciones.

21. comando “chmod”

chmod es otro comando de Linux, que se utiliza para cambiar los permisos de lectura, escritura y ejecución de archivos y directorios. Como este comando es bastante complicado, puede leer el tutorial completo para ejecutarlo correctamente.

22. comando “chown”

En Linux, todos los archivos pertenecen a un usuario específico. El comando chown le permite cambiar o transferir la propiedad de un archivo al nombre de usuario especificado. Por ejemplo, chown linuxuser2 file.ext hará que linuxuser2 sea el propietario del file.ext .

23. comando “jobs”

El comando jobs mostrará todos los trabajos actuales junto con sus estados. Un trabajo es básicamente un proceso que inicia el shell.

24. comando “kill”

Si tiene un programa que no responde, puede terminarlo manualmente usando el comando kill. Enviará una cierta señal a la aplicación que no funciona correctamente y le indicará a la aplicación que se cierre.

Hay un total de sesenta y cuatro señales que puede usar, pero las personas generalmente solo usan dos señales:



- SIGTERM (15): solicita que un programa deje de ejecutarse y le da algo de tiempo para guardar todo su progreso. Si no especifica la señal al ingresar el comando kill, se usará esta señal.
- SIGKILL (9): obliga a los programas a detenerse inmediatamente. El progreso no guardado se perderá.

Además de conocer las señales, también necesita conocer el número de identificación del proceso (PID) del programa que desea matar. Si no conoce el PID, simplemente ejecute el comando “ps ux”.

Después de saber qué señal desea usar y el PID del programa, ingrese la siguiente sintaxis:

kill [opción de señal] PID.

25. comando “ping”

Utilice el comando ping para verificar el estado de su conectividad a un servidor. Por ejemplo, simplemente ingresando ping google.com, el comando verificará si puede conectarse a Google y también medirá el tiempo de respuesta.

26. comando “wget”

La línea de comandos de Linux es muy útil; incluso puede descargar archivos de Internet con la ayuda del comando wget. Para hacerlo, simplemente escriba wget seguido del enlace de descarga.

27. comando “uname”

El comando uname , abreviatura de Unix Name, imprimirá información detallada sobre su sistema Linux, como el nombre de la máquina, el sistema operativo, el kernel, etc.

28. comando “top”

Como terminal equivalente al Administrador de tareas en Windows, el comando “top” mostrará una lista de procesos en ejecución y cuánta CPU usa cada proceso. Es muy útil monitorear el uso de recursos del sistema, especialmente sabiendo qué proceso debe terminarse porque consume demasiados recursos. Busque referencias sobre “htop”.



29. comando “history”

Cuando haya estado usando Linux durante un cierto período de tiempo, notará rápidamente que puede ejecutar cientos de comandos todos los días. Como tal, ejecutar el comando “history” es particularmente útil si desea revisar los comandos que ha ingresado antes.

30. comando “man”

¿Confundido acerca de la función de ciertos comandos de Linux? No se preocupe, puede aprender fácilmente cómo usarlos directamente desde el shell de Linux usando el comando man. Por ejemplo, ingresar man tail mostrará la instrucción manual del comando tail.

31. comando “echo”

Este comando se usa para mover algunos datos a un archivo. Por ejemplo, si desea agregar el texto "Hola, mi nombre es Juan" en un archivo llamado nombre.txt, debe escribir “echo Hola, mi nombre es Juan >> nombre.txt”.

32. comando “zip,unzip”

Use el comando zip para comprimir sus archivos en un archivo zip y use el comando unzip para extraer los archivos comprimidos de un archivo zip.

33. comando “hostname”

Si desea saber el nombre de su host / red, simplemente escriba hostname. Si agrega un -i al final, se mostrará la dirección IP de su red.

34. comando “useradd, userdel”

Dado que Linux es un sistema multiusuario, esto significa que más de una persona puede interactuar con el mismo sistema al mismo tiempo. useradd se usa para crear un nuevo usuario, mientras que passwd agrega una contraseña a la cuenta de ese usuario. Para agregar una nueva persona llamada John escriba, useradd John y luego para agregar su tipo de contraseña, passwd 123456789.



Eliminar un usuario es muy similar a agregar un nuevo usuario. Para eliminar el tipo de cuenta de usuario, userdel UserName

Notas:

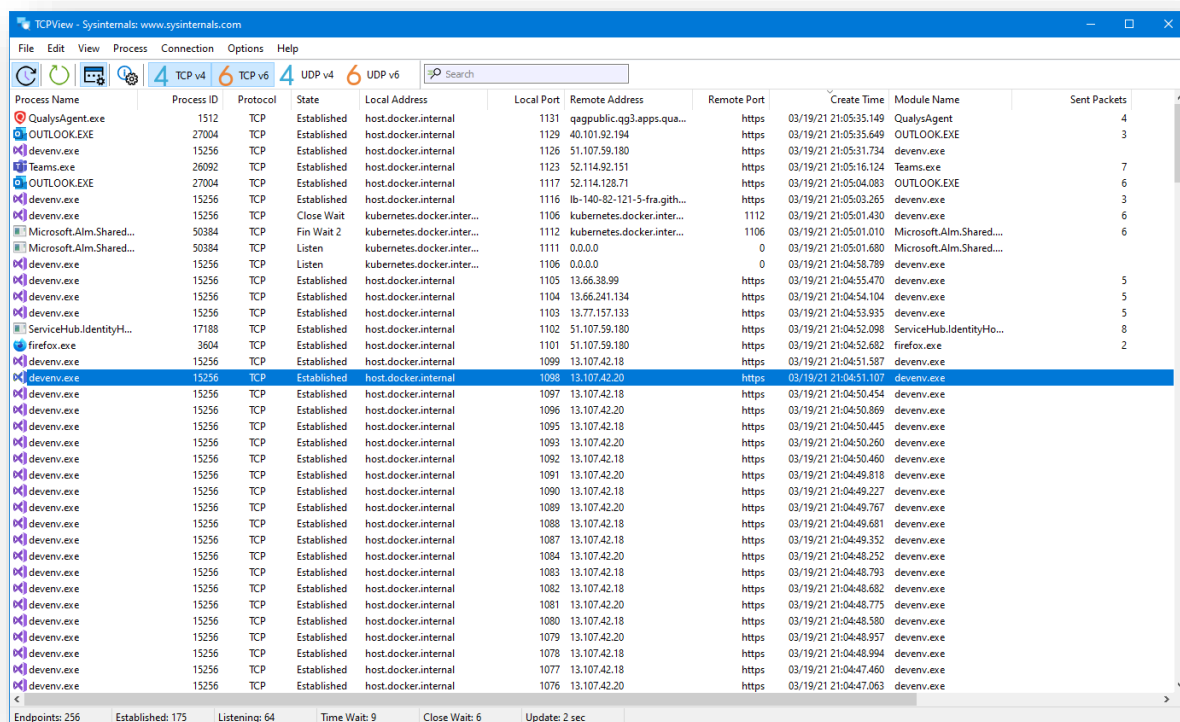
- Utilice el comando “clear” para limpiar la terminal si se llena de demasiados comandos anteriores.
- Pruebe el botón TAB para completar automáticamente lo que está escribiendo. Por ejemplo, si necesita escribir Documentos, comience a escribir un comando (vayamos con cd Docu, luego presione la tecla TAB) y el terminal completará el resto, mostrándole Documentos de cd.
- Ctrl + C y Ctrl + Z se utilizan para detener cualquier comando que esté funcionando actualmente. Ctrl + C detendrá y terminará el comando, mientras que Ctrl + Z simplemente pausará el comando.
- Si accidentalmente congela su terminal utilizando Ctrl + S, basta con descongelar usando Ctrl + Q.
- Ctrl + A lo mueve al principio de la línea, mientras que Ctrl + E lo mueve al final.
- Puede ejecutar varios comandos en un solo comando utilizando el “;” para separarlos. Por ejemplo Command1; Command2; Command3. O use && si solo desea que el siguiente comando se ejecute cuando el primero sea exitoso.



Anexo II: Comandos o aplicativos básicos para Windows: TCPView

En esta segunda versión de comandos o aplicativos para Windows mencionaremos el aplicativo “TCPView de la suite SYSINTERNALS”.

TCPView es un programa de Windows que le mostrará listados detallados de todos los puntos finales TCP y UDP en su sistema, incluidas las direcciones locales y remotas y el estado de las conexiones TCP. En Windows Server 2008, Vista y XP, TCPView también informa el nombre del proceso propietario del endpoint. TCPView proporciona un subconjunto más informativo y convenientemente presentado del programa Netstat que se envía con Windows. La descarga de TCPView incluye Tcpcvcon, una versión de línea de comandos con la misma funcionalidad.



Process Name	Process ID	Protocol	State	Local Address	Local Port	Remote Address	Remote Port	Create Time	Module Name	Sent Packets
QualysAgent.exe	1512	TCP	Established	host.docker.internal	1131	qagpublic.cg3.apps.qua...	https	03/19/21 21:05:35.149	QualysAgent	4
OUTLOOK.EXE	27004	TCP	Established	host.docker.internal	1129	40.101.92.194	https	03/19/21 21:05:35.649	OUTLOOK.EXE	3
devenv.exe	15256	TCP	Established	host.docker.internal	1126	51.107.59.180	https	03/19/21 21:05:31.734	devenv.exe	4
Teams.exe	26092	TCP	Established	host.docker.internal	1123	52.114.92.151	https	03/19/21 21:05:16.124	Teams.exe	7
OUTLOOK.EXE	27004	TCP	Established	host.docker.internal	1117	52.114.128.71	https	03/19/21 21:05:04.083	OUTLOOK.EXE	6
devenv.exe	15256	TCP	Established	host.docker.internal	1116	lb-140-82-121-5-fra.git...	https	03/19/21 21:05:03.265	devenv.exe	3
devenv.exe	15256	TCP	Close Wait	kubernetes.docker.inter...	1106	kubernetes.docker.inter...	1112	03/19/21 21:05:01.430	devenv.exe	6
Microsoft.Alm.Shared...	50384	TCP	Fin Wait 2	kubernetes.docker.inter...	1102	kubernetes.docker.inter...	1106	03/19/21 21:05:01.010	Microsoft.Alm.Shared...	6
Microsoft.Alm.Shared...	50384	TCP	Listen	kubernetes.docker.inter...	1111	0.0.0.0	0	03/19/21 21:05:01.680	Microsoft.Alm.Shared...	0
devenv.exe	15256	TCP	Listen	kubernetes.docker.inter...	1106	0.0.0.0	0	03/19/21 21:04:58.789	devenv.exe	0
devenv.exe	15256	TCP	Established	host.docker.internal	1105	13.66.38.99	https	03/19/21 21:04:55.470	devenv.exe	5
devenv.exe	15256	TCP	Established	host.docker.internal	1104	13.66.241.134	https	03/19/21 21:04:54.104	devenv.exe	5
devenv.exe	15256	TCP	Established	host.docker.internal	1103	13.77.157.133	https	03/19/21 21:04:53.935	devenv.exe	5
ServiceHub.IdentityH...	17188	TCP	Established	host.docker.internal	1102	51.107.59.180	https	03/19/21 21:04:52.098	ServiceHub.IdentityHo...	8
firefox.exe	3604	TCP	Established	host.docker.internal	1101	51.107.59.180	https	03/19/21 21:04:52.682	firefox.exe	2
devenv.exe	15256	TCP	Established	host.docker.internal	1099	13.107.42.18	https	03/19/21 21:04:51.587	devenv.exe	0
devenv.exe	15256	TCP	Established	host.docker.internal	1098	13.107.42.20	https	03/19/21 21:04:51.107	devenv.exe	0
devenv.exe	15256	TCP	Established	host.docker.internal	1097	13.107.42.18	https	03/19/21 21:04:50.454	devenv.exe	0
devenv.exe	15256	TCP	Established	host.docker.internal	1096	13.107.42.20	https	03/19/21 21:04:50.869	devenv.exe	0
devenv.exe	15256	TCP	Established	host.docker.internal	1095	13.107.42.18	https	03/19/21 21:04:50.445	devenv.exe	0
devenv.exe	15256	TCP	Established	host.docker.internal	1093	13.107.42.20	https	03/19/21 21:04:50.260	devenv.exe	0
devenv.exe	15256	TCP	Established	host.docker.internal	1092	13.107.42.18	https	03/19/21 21:04:50.460	devenv.exe	0
devenv.exe	15256	TCP	Established	host.docker.internal	1091	13.107.42.20	https	03/19/21 21:04:49.818	devenv.exe	0
devenv.exe	15256	TCP	Established	host.docker.internal	1090	13.107.42.18	https	03/19/21 21:04:49.227	devenv.exe	0
devenv.exe	15256	TCP	Established	host.docker.internal	1089	13.107.42.20	https	03/19/21 21:04:49.767	devenv.exe	0
devenv.exe	15256	TCP	Established	host.docker.internal	1088	13.107.42.18	https	03/19/21 21:04:49.681	devenv.exe	0
devenv.exe	15256	TCP	Established	host.docker.internal	1087	13.107.42.18	https	03/19/21 21:04:49.352	devenv.exe	0
devenv.exe	15256	TCP	Established	host.docker.internal	1084	13.107.42.20	https	03/19/21 21:04:48.252	devenv.exe	0
devenv.exe	15256	TCP	Established	host.docker.internal	1083	13.107.42.18	https	03/19/21 21:04:48.793	devenv.exe	0
devenv.exe	15256	TCP	Established	host.docker.internal	1082	13.107.42.18	https	03/19/21 21:04:48.682	devenv.exe	0
devenv.exe	15256	TCP	Established	host.docker.internal	1081	13.107.42.20	https	03/19/21 21:04:48.775	devenv.exe	0
devenv.exe	15256	TCP	Established	host.docker.internal	1080	13.107.42.18	https	03/19/21 21:04:48.580	devenv.exe	0
devenv.exe	15256	TCP	Established	host.docker.internal	1079	13.107.42.20	https	03/19/21 21:04:48.957	devenv.exe	0
devenv.exe	15256	TCP	Established	host.docker.internal	1078	13.107.42.18	https	03/19/21 21:04:48.994	devenv.exe	0
devenv.exe	15256	TCP	Established	host.docker.internal	1077	13.107.42.18	https	03/19/21 21:04:47.460	devenv.exe	0
devenv.exe	15256	TCP	Established	host.docker.internal	1076	13.107.42.20	https	03/19/21 21:04:47.063	devenv.exe	0

Este programa puede descargarlo desde:

<https://download.sysinternals.com/files/TCPView.zip>

Cuando inicie TCPView, enumerará todos los puntos finales TCP y UDP activos, resolviendo todas las direcciones IP en sus versiones de nombre de dominio. Puede utilizar un botón de la barra de herramientas o un elemento de menú para alternar la visualización de los nombres resueltos.



TCPView muestra el nombre del proceso que posee cada punto final, incluido el nombre del servicio (si corresponde).

De forma predeterminada, TCPView se actualiza cada segundo, pero puede utilizar el elemento de menú Opciones | Frecuencia de actualización para cambiar la frecuencia. Los puntos finales que cambian de estado de una actualización a la siguiente se resaltan en amarillo; los que se eliminan se muestran en rojo y los nuevos puntos finales se muestran en verde.

Puede cerrar las conexiones TCP / IP establecidas (aquellas etiquetadas con un estado de ESTABLECIDO) seleccionando Archivo | Cerrar conexiones, o haciendo clic con el botón derecho en una conexión y eligiendo Cerrar conexiones en el menú contextual resultante.

Puede guardar la ventana de salida de TCPView en un archivo usando el elemento del menú Guardar.

Nota adicional para “tcpvcon”:

El uso de Tcgvcon es similar al de la utilidad netstat incorporada de Windows:

Uso:

cmd

 Copiar

tcpvcon [-a] [-c] [-n] [process name or PID]

Parámetro	Descripción
-a	Mostrar todos los puntos finales (el valor predeterminado es mostrar las conexiones TCP establecidas).
-C	Imprime la salida como CSV.
-norte	No resuelva las direcciones.

Con estos tips básicos buscamos incentivarlo a explorar estas herramientas y sus múltiples usos para ciberseguridad.