

# **Hacking**

## **Curso completo**



# **Hacking**

## **Curso completo**

*Fernando Castillo*





Hacking. Curso completo  
© Fernando Castillo  
© De la edición: Ra-Ma 2022

Edición original publicada por Six Ediciones. Ciudad Autónoma de Buenos Aires, Argentina.  
Título original: Hacking ético  
Colección: USERS ebooks  
ISBN: 978-987-8414-03-4  
Derechos Reservados © Six Ediciones. Ciudad Autónoma de Buenos Aires, Argentina.

MARCAS COMERCIALES. Las designaciones utilizadas por las empresas para distinguir sus productos (hardware, software, sistemas operativos, etc.) suelen ser marcas registradas. RA-MA ha intentado a lo largo de este libro distinguir las marcas comerciales de los términos descriptivos, siguiendo el estilo que utiliza el fabricante, sin intención de infringir la marca y solo en beneficio del propietario de la misma. Los datos de los ejemplos y pantallas son ficticios a no ser que se especifique lo contrario.

RA-MA es marca comercial registrada.

Se ha puesto el máximo empeño en ofrecer al lector una información completa y precisa. Sin embargo, RA-MA Editorial no asume ninguna responsabilidad derivada de su uso ni tampoco de cualquier violación de patentes ni otros derechos de terceras partes que pudieran ocurrir. Esta publicación tiene por objeto proporcionar unos conocimientos precisos y acreditados sobre el tema tratado. Su venta no supone para el editor ninguna forma de asistencia legal, administrativa o de ningún otro tipo. En caso de precisarse asesoría legal u otra forma de ayuda experta, deben buscarse los servicios de un profesional competente.

Reservados todos los derechos de publicación en cualquier idioma.

Según lo dispuesto en el Código Penal vigente, ninguna parte de este libro puede ser reproducida, grabada en sistema de almacenamiento o transmitida en forma alguna ni por cualquier procedimiento, ya sea electrónico, mecánico, reprográfico, magnético o cualquier otro sin autorización previa y por escrito de RA-MA; su contenido está protegido por la ley vigente, que establece penas de prisión y/o multas a quienes, intencionadamente, reprodujeren o plagiaran, en todo o en parte, una obra literaria, artística o científica.

Editado por:  
RA-MA Editorial  
Calle Jarama, 3A, Polígono Industrial Igarsa  
28860 PARACUELLOS DE JARAMA, Madrid  
Teléfono: 91 658 42 80  
Fax: 91 662 81 39  
Correo electrónico: [editorial@ra-ma.com](mailto:editorial@ra-ma.com)  
Internet: [www.ra-ma.es](http://www.ra-ma.es) y [www.ra-ma.com](http://www.ra-ma.com)  
ISBN: 978-84-1944-427-1  
Depósito legal: M-22778-2022  
Maquetación: Antonio García Tomé  
Diseño de portada: Antonio García Tomé  
Filmación e impresión: Safekat  
Impreso en España en septiembre de 2022

# ÍNDICE

<b>PRÓLOGO .....</b>	<b>11</b>
<b>SOBRE ESTA OBRA.....</b>	<b>13</b>
<b>PARTE 1 .....</b>	<b>15</b>
<b>CAPÍTULO 1. ¿QUÉ SE NECESITA?.....</b>	<b>17</b>
1.1 LABORATORIO DE PRUEBAS .....	17
1.1.1 VirtualBox .....	18
1.1.2 Metasploitable .....	24
1.1.3 Kali Linux .....	33
1.1.4 Conectar ambas máquinas virtuales .....	36
1.2 ACTIVIDADES.....	39
1.2.1 Test de autoevaluación .....	39
1.2.2 Ejercicios prácticos .....	39
<b>CAPÍTULO 2. VULNERABILIDADES Y PRUEBAS .....</b>	<b>41</b>
2.1 SISTEMAS VULNERABLES .....	41
2.1.1 Metasploitable 2 .....	42
2.1.2 Metasploitable 3 .....	49
2.2 PRUEBA DE PENETRACIÓN .....	50
2.2.1 Black Box.....	50
2.2.2 White Box .....	50
2.2.3 Grey Box .....	51
2.3 ACTIVIDADES.....	51
2.3.1 Test de autoevaluación .....	51
2.3.2 Ejercicios prácticos .....	51
<b>CAPÍTULO 3. ESCANEADO CON NMAP .....</b>	<b>53</b>
3.1 OPCIONES DISPONIBLES .....	53
3.1.1 Puertos para analizar .....	58

3.1.2	Duración del escaneo .....	59
3.2	ESCANEO CON NMAP .....	59
3.3	OPCIONES ADICIONALES .....	68
3.4	INTERFAZ GRÁFICA .....	72
3.5	ACTIVIDADES.....	84
3.5.1	Test de autoevaluación .....	84
3.5.2	Ejercicios prácticos .....	84
<b>GLOSARIO PARTE 1</b> .....		<b>85</b>
<b>PARTE 2</b> .....		<b>87</b>
<b>CAPÍTULO 4. SHELL</b> .....		<b>89</b>
4.1	QUÉ ES UNA SHELL.....	90
4.1.1	Tipos de shell .....	90
4.1.2	Bash shell .....	90
4.2	SHELL SCRIPT.....	90
4.2.1	Comandos.....	94
4.2.2	Comandos básicos más usados dentro de los scripts.....	94
4.2.3	Redireccionamiento de entrada/salida en shell script.....	95
4.2.4	Uso de las comillas en el shell script.....	95
4.3	USO DE VARIABLES .....	98
4.3.1	Reasignación de variables .....	99
4.3.2	Reglas de las variables .....	100
4.4	SCRIPTS MÁS ELABORADOS .....	100
4.4.1	Tomar decisiones .....	102
4.4.2	Condicionales en bash.....	103
4.5	OPERADORES EN BASH SCRIPT.....	104
4.5.1	Operadores de comparación .....	105
4.5.2	Operadores lógicos .....	106
4.6	BUCLES EN BASH SCRIPT.....	107
4.6.1	Asignar alias a los scripts.....	108
4.6.2	Uso de funciones en bash shell .....	108
4.7	ONELINERS .....	109
4.8	USO DE CRONTAB .....	110
4.9	EJERCICIOS DE AUTOMATIZACIÓN .....	111
4.10	ACTIVIDADES .....	115
4.10.1	Test de autoevaluación .....	115
4.10.2	Ejercicios prácticos .....	116
<b>CAPÍTULO 5. CAPTURA DE INFORMACIÓN</b> .....		<b>117</b>
5.1	PROCESO DE CAPTURA DE LA INFORMACIÓN .....	118
5.2	RECONOCIMIENTO PASIVO.....	118
5.2.1	OSINT o inteligencia de fuentes abiertas.....	119

5.2.2	Maltego .....	119
5.2.3	The Harvester .....	122
5.2.4	Footprinting .....	123
5.2.5	Google dorks aplicados a yahoo.com.....	124
5.2.6	Shodan, Censys y filtros de GitHub .....	128
5.2.7	Wappalyzer para huellas dactilares .....	131
5.2.8	Búsqueda de ASN.....	131
5.2.9	Wayback Machine .....	133
5.2.10	Adquisiciones con Crunchbase .....	133
5.3	RECONOCIMIENTO ACTIVO.....	134
5.3.1	Búsqueda de subdominios.....	135
5.3.2	Uso de Amass.....	138
5.3.3	Screenshots de webs con Aquatone.....	139
5.3.4	Búsqueda de archivos sensibles con Dirsearch y FFUF .....	141
5.3.5	Descubrimiento de subdominios activos con httpx.....	142
5.3.6	FFUF .....	144
5.3.7	Análisis de archivos .js con Link Finder .....	144
5.3.8	Listado de palabras más usadas en hacking ético.....	145
5.4	ACTIVIDADES .....	146
5.4.1	Test de autoevaluación .....	146
5.4.2	Ejercicios prácticos .....	146
<b>CAPÍTULO 6. OBJETIVOS.....</b>		<b>147</b>
6.1	CLASIFICACIÓN .....	147
6.1.1	Definición del scope u objetivos del test.....	149
6.1.2	Ambiente de producción vs. ambiente de pruebas .....	151
6.1.3	Metodologías.....	152
6.1.4	Informes del pentesting .....	154
6.2	ACTIVIDADES.....	155
6.2.1	Test de autoevaluación .....	155
6.2.2	Ejercicios prácticos .....	155
<b>GLOSARIO PARTE 2.....</b>		<b>157</b>
<b>PARTE 3 .....</b>		<b>159</b>
<b>CAPÍTULO 7. RECONOCIMIENTO.....</b>		<b>161</b>
7.1	CONCEPTOS PRELIMINARES .....	162
7.2	CASO PRÁCTICO DE BÚSQUEDA DE VULNERABILIDADES .....	162
7.2.1	1. Enumerar subdominios.....	163
7.2.2	2. Filtrar subdominios .....	169
7.2.3	3. Buscar las URL en Wayback Machine y Google Dorks .....	170
7.2.4	4. OSINT aplicada para la búsqueda de datos de empleados.....	175
7.2.5	5. Capturas de pantalla de subdominios vivos .....	176
7.2.6	6. Tecnología subyacente en los subdominios.....	178
7.2.7	7. Búsqueda de archivos con extensión .js.....	178

7.2.8	8. Búsqueda de endpoints en archivos .js .....	179
7.2.9	9. Búsqueda de parámetros.....	181
7.2.10	10. Encontrar directorios .....	182
7.3	COMANDOS ÚTILES.....	183
7.3.1	HTTPX.....	183
7.3.2	FUFF .....	184
7.4	ACTIVIDADES .....	184
7.4.1	Test de autoevaluación .....	185
7.4.2	Ejercicios prácticos .....	185
<b>CAPÍTULO 8. ANÁLISIS DE VULNERABILIDADES .....</b>		<b>187</b>
8.1	¿QUÉ ES UN ANÁLISIS DE VULNERABILIDADES?.....	187
8.1.1	¿Cuáles son las vulnerabilidades más comunes en los sitios web?.....	189
8.1.2	Búsqueda de ejemplo .....	190
8.1.3	Reporte .....	195
8.2	ACTIVIDADES .....	206
8.2.1	Test de autoevaluación .....	206
8.2.2	Ejercicios prácticos .....	206
<b>CAPÍTULO 9. EXPLOTACIÓN Y POSEXPLORACIÓN .....</b>		<b>207</b>
9.1	EXPLOITS.....	208
9.1.1	Ejemplo de exploit .....	208
9.1.2	¿Cómo funcionan los exploits? .....	209
9.1.3	Términos relacionados con la etapa de explotación .....	210
9.2	POSEXPLOTACIÓN.....	211
9.2.1	Eliminar los logs.....	212
9.2.2	Ofuscar los archivos modificados .....	212
9.2.3	Sobrescribir la memoria RAM del equipo .....	213
9.2.4	Borrar el historial de comandos .....	213
9.3	PERIODICIDAD DEL TEST DE INTRUSIÓN .....	213
9.3.1	¿Cuándo es el momento de contratar un pentesting? .....	214
9.3.2	Reporte de pentest con explicación técnica.....	214
9.3.3	La importancia del reporte para los profesionales.....	214
9.3.4	¿Que ítems debe contener un reporte de pentest? .....	214
9.4	CERTIFICACIONES .....	216
9.5	ACTIVIDADES .....	219
9.5.1	Test de autoevaluación .....	219
9.5.2	Ejercicios prácticos .....	219
<b>CAPÍTULO 10. REFERENCIA DE COMANDOS NMAP.....</b>		<b>221</b>
10.1	NMAP.....	221
10.1.1	Especificación del objetivo.....	221
10.1.2	Descubrimiento de host.....	222
10.1.3	Técnicas de escaneo .....	222
10.1.4	Especificación de puerto y secuencia de escaneo.....	223

10.1.5	Detección de servicios/versiones.....	223
10.1.6	Detección del sistema operativo.....	223
10.1.7	Escanear hosts y subredes objetivo .....	224
10.1.8	Escaneo de puertos .....	224
10.1.9	Opciones para escaneo de puertos.....	224
10.1.10	Habilitar comentarios en Nmap.....	225
10.2	ACTIVIDADES .....	225
10.2.1	Test de autoevaluación .....	225
10.2.2	Ejercicios prácticos .....	226
<b>GLOSARIO PARTE 3.....</b>		<b>227</b>
<b>PARTE 4 .....</b>		<b>229</b>
<b>CAPÍTULO 11. ATAQUES MITM.....</b>		<b>231</b>
11.1	¿QUÉ ES UN ATAQUE MITM? .....	231
11.1.1	¿Cómo se perpetra un ataque MITM?.....	232
11.1.2	Tipos de ataques MITM .....	233
11.2	CÓMO PROTEGERTE DE ATAQUES MAN IN THE MIDDLE .....	234
11.2.1	Navegar por sitios seguros .....	235
11.2.2	Usar contraseñas fuertes .....	235
11.2.3	Usar WPA2-AES .....	235
11.2.4	Segmentar las redes.....	235
11.2.5	Tener una política de actualización de software.....	235
11.2.6	Usar la autenticación de dos pasos.....	236
11.2.7	Evitar conectarse a redes Wi-Fi abiertas públicas.....	236
11.2.8	No abrir enlaces de fuentes de correos desconocidas.....	236
11.2.9	Asegurar los equipos con aplicativos antivirus y antimalware .....	236
11.2.10	Usar dispositivos de protección de red.....	236
11.3	ATAQUE MITM DE ENVENENAMIENTO DE ARP CON EL USO DE ETTERCAP .....	237
11.3.1	Elaboración del ataque MITM con ettercap .....	238
11.4	ACTIVIDADES .....	243
11.4.1	Test de autoevaluación .....	243
11.4.2	Ejercicios prácticos .....	244
<b>CAPÍTULO 12. METASPLOIT.....</b>		<b>245</b>
12.1	METASPLOIT FRAMEWORK.....	245
12.1.1	Módulos de Metasploit.....	246
12.2	COMANDO MSFCONSOLE .....	252
12.3	COMANDO SET.....	253
12.4	BÚSQUEDA EN MSFCONSOLE .....	257
12.5	TRABAJAR CON MÓDULOS.....	259
12.6	SESIONES.....	264
12.7	ACTIVIDADES .....	264

12.7.1	Test de autoevaluación .....	264
12.7.2	Ejercicios prácticos .....	264
<b>CAPÍTULO 13. NESSUS .....</b>		<b>265</b>
13.1	¿QUÉ ES NESSUS? .....	265
13.1.1	Compatibilidad .....	265
13.1.2	Versiones .....	266
13.2	PLUGINS DE NESSUS .....	267
13.3	TEMPLATES DE NESSUS .....	275
13.4	AGENTES NESSUS .....	277
13.4.1	¿Cómo iniciar un escaneo utilizando Nessus Agents? .....	277
13.4.2	Ejemplo práctico de escaneo con Nessus .....	277
13.5	ACTIVIDADES .....	282
13.5.1	Test de autoevaluación .....	282
13.5.2	Ejercicios prácticos .....	282
<b>CAPÍTULO 14. ATAQUES A CONTRASEÑAS .....</b>		<b>283</b>
14.1	ALMACENAMIENTO DE CONTRASEÑAS .....	284
14.1.1	Opción 1. Almacenamiento de contraseñas en texto plano .....	284
14.1.2	Opción 2. Almacenamiento de contraseñas cifradas .....	284
14.1.3	Opción 3. Cifrado de contraseñas a través de funciones hash .....	284
14.2	ATAQUES DE FUERZA BRUTA .....	285
14.2.1	THC Hydra .....	285
14.2.2	John The Ripper .....	287
14.2.3	Aircrack-ng .....	291
14.3	ATAQUE PASSWORD SPRAYING .....	293
14.4	ATAQUE DE CREDENTIAL STUFFING .....	294
14.4.1	Mitigación para Credential stuffing .....	294
14.4.2	Fuerza bruta inversa .....	294
14.4.3	Ataques de diccionario a contraseñas .....	294
14.4.4	Ataques online .....	295
14.4.5	Ataque offline .....	295
14.5	CONSEJOS PARA PROTEGER TUS CONTRASEÑAS .....	296
14.6	GESTORES DE CONTRASEÑAS .....	297
14.6.1	Qué seguridad ofrecen los gestores de contraseñas .....	298
14.6.2	Cómo se usan los gestores de contraseñas .....	298
14.7	CIFRADOS .....	299
14.7.1	Tipos de funciones SHA .....	300
14.8	ACTIVIDADES .....	301
14.8.1	Test de autoevaluación .....	301
14.8.2	Ejercicios prácticos .....	301
<b>GLOSARIO PARTE 4 .....</b>		<b>303</b>
<b>MATERIAL ADICIONAL .....</b>		<b>305</b>



---

## PRÓLOGO

El hacking siempre ha despertado interés en todos los aficionados a la tecnología. Debes saber que no todos los hackers son delincuentes ni toda actividad relacionada con ellos es ilegal, pues existe una rama importante denominada **hacking ético**, que se preocupa de analizar los sistemas informáticos corporativos y los programas con el fin de aclarar el estado de la seguridad. En forma concreta, se trata de asumir el rol de un ciberdelincuente para simular ataques a cierto sistema y, de esta forma, evaluar el estado real de su seguridad.

Las acciones realizadas por un hacker ético tratan de adelantarse a los cibercriminales, solucionando cualquier debilidad que pueda provocar un posible ataque. Además, buscan concientizar a las compañías sobre la importancia de mantener la seguridad informática y, también, de mejorar los procesos de seguridad mediante planes de respuesta y acción ante los incidentes.



---

## SOBRE ESTA OBRA

En esta obra se revisan las acciones que puedes realizar para analizar y explotar un sistema objetivo. De esta forma, estarás en los zapatos de un hacker ético mientras realizas intrusiones en un sistema objetivo y logras obtener información o efectuar análisis de seguridad.

Se irán presentando diferentes formas de explotar y analizar un sistema objetivo, así como también aprenderás a montar un entorno de pruebas para poder ensayar tus habilidades sin utilizar sistemas externos.

### Partes de esta obra

- **Parte 1:** Aquí se presenta el concepto de hacking ético, aprenderás a configurar un entorno de pruebas, conocerás los sistemas vulnerables y el uso de Nmap.
- **Parte 2:** En este volumen revisarás a fondo el Shell Scripting, conocerás la forma en que puedes capturar información y cómo seleccionar objetivos para las tareas de análisis y extracción de información.
- **Parte 3:** Aquí se presentan los conceptos relacionados con el mapeo de vulnerabilidades de un sistema objetivo y se analiza el proceso de explotación y posexplotación.
- **Parte 4:** En esta parte aprenderás a realizar el ataque Man in the middle y conocerás a fondo Metasploit y Nessus.



***USERS***

Parte 1

# Hacking

**Entorno de  
pruebas**

**Sistemas  
vulnerables**

**Uso de Nmap**



# 1

---

## ¿QUÉ SE NECESITA?

Antes de comenzar a realizar las primeras tareas de hacking ético, debes armar el laboratorio de trabajo; esto incluye diversas herramientas que conocerás en este capítulo.

### 1.1 LABORATORIO DE PRUEBAS

---

Las tareas de **protección de sistemas y redes** requieren tener una amplia comprensión de las estrategias de ataque existentes y, también, un conocimiento acabado de cada una de las tácticas, herramientas y motivaciones de quienes realizan este tipo de ataques.

Estos conocimientos son los que definen a un **hacker ético** pues en general se trata de personas dedicadas a identificar y reparar posibles vulnerabilidades, lo que previene en forma eficiente la explotación de estas por hackers malintencionados. Entonces, un hacker ético se especializa en **pruebas de penetración** de sistemas informáticos y en el uso de **software de seguridad** con el fin de analizar, evaluar, detectar **agujeros**, fortalecer y mejorar la seguridad de un sistema o una red.

En definitiva, es un tipo especial de pirata informático conocido también como **hacker de sombrero blanco** o white hat, para separarlo de los piratas informáticos criminales o **hackers de sombrero negro**.

En esta obra aprenderás los fundamentos para realizar diversas tareas de hacking ético desde GNU/Linux. Para lograrlo, la primera tarea es configurar tu laboratorio de pruebas, es decir, un espacio donde poder ejecutar los análisis y las pruebas de seguridad necesarios, sin que debas utilizar entornos o sistemas en producción para buscar vulnerabilidades o probar herramientas de ataque, pues

podrías ser acusado de pirata informático al intentar acceder sin permiso a ciertos sistemas.

Para protegerte y teniendo en cuenta que las actividades que se deben realizar en algunos momentos pueden rayar en la línea de la ilegalidad, es una excelente idea que estas pruebas y análisis se realicen en entornos controlados, donde no sea necesario causar problemas o molestias accediendo a máquinas ajenas. A diferencia de lo que se puede pensar, configurar un laboratorio de pruebas no requiere contar con una red de ordenadores listos para ser blanco de tus ataques ni de las búsquedas de vulnerabilidades, más bien debes recrear un **sistema vulnerable** al que puedas acceder para analizar y probar tus habilidades.

La creación de un sistema vulnerable no es una tarea tan compleja, pues existen máquinas virtuales programadas con ciertas vulnerabilidades y son estas las opciones adecuadas para probar lo que aprendas a lo largo de esta colección.

Para armar tu laboratorio de pruebas necesitarás tres integrantes básicos: **VirtualBox**, **Kali Linux** y **Metasploitable**. A continuación verás cómo puedes obtenerlos e instalarlos.

### 1.1.1 VirtualBox

La **virtualización** de plataformas o sistemas puede conseguirse utilizando como base cualquier sistema operativo, por lo tanto, una distribución Linux no es la excepción. Por ejemplo, dentro de Ubuntu es posible instalar una versión de Windows o, en este caso, los sistemas que se requieren para configurar tu laboratorio de pruebas. Si estuvieras ante la necesidad de instalar y utilizar solo una aplicación creada para otro sistema, puedes usar Wine, pero al tratarse de la virtualización de un sistema operativo completo debes optar por VirtualBox u otra alternativa similar (Figura 1.1).



Figura 1.1.

Aunque lograr la virtualización gracias a VirtualBox es una tarea bastante sencilla, debes tener en cuenta un detalle importante, el sistema operativo que quieres virtualizar debe consumir menos recursos que el hardware que tiene la máquina huésped.

Esto es muy importante pues de no cumplirse podrías encontrarte con que el equipo huésped no responde durante un tiempo o que se suspende la actividad del software por seguridad.

Una de las opciones comerciales más completas y potentes para virtualizar sistemas operativos pertenece a **VMware**, pero también puedes acceder a una excelente alternativa gratuita: VirtualBox, que puede ser instalado en cualquier distribución Linux y también en Windows o MacOSX.

VirtualBox es de código abierto y multiplataforma; esto permite, entre otras cosas, que puedas crear máquinas virtuales en un sistema Windows para después trasladarlas a una computadora con GNU/Linux y lograr su funcionamiento.

Entre las características más destacadas de esta aplicación de virtualización, se encuentra la posibilidad de crear un disco duro virtual segmentado, es decir, que puede aumentar su capacidad o hacer uso de esta capacidad de almacenamiento en función del uso que le des. Por otro lado, te permite crear máquinas virtuales que pueden ser transportadas como si fueran documentos o imágenes.

Instalar VirtualBox en Linux, al igual que sucede con otras aplicaciones, puede lograrse de varias formas, por ejemplo, a través de los **repositorios oficiales** o mediante la descarga del **paquete de instalación** en sistemas Windows.

Si utilizas los repositorios oficiales, obtendrás una versión estable y funcional de VirtualBox, pero no la última versión del programa, aunque sin duda se trata de la forma más difundida y segura de realizar la instalación. VirtualBox se encuentra disponible en los repositorios de las principales distribuciones GNU/Linux, por lo tanto, solo necesitarás buscarlo en la interfaz de Centro de Aplicaciones que corresponda a tu distribución o abrir una consola de comandos para ejecutar lo siguiente para instalar VirtualBox en una distribución Ubuntu, Debian o cualquiera derivada.

```
sudo apt-get install virtualbox
```

Para Arch Linux o alguna **distribución** derivada, debes ejecutar lo siguiente:

```
sudo pacman -S virtualbox
```

Para Fedora, Red Hat o una distribución derivada, debes ejecutar el siguiente código:

```
sudo dnf install virtualbox
```

Por otra parte, si utilizas SUSE Linux, OpenSUSE o cualquier distribución derivada, tendrás que ejecutar el siguiente código:

```
sudo zypper install virtualbox
```

Una opción para instalar VirtualBox es acceder al sitio web oficial [www.virtualbox.org](http://www.virtualbox.org) para descargar el paquete de instalación preparado para tu sistema operativo. Esto te permitirá obtener la última versión de VirtualBox, pero debes considerar que puede no estar probada con ciertas distribuciones específicas, por lo que podría presentar algunas fallas (**Figura 1.2.**).

Si estás en Linux, una vez instalado puedes controlarlo en forma directa desde la Terminal de comandos.

El comando principal para controlar VirtualBox es **VBoxManage**, pero debes acompañarlo de los siguientes subcomandos o **parámetros**.

Para mostrar una lista de las máquinas virtuales, escribe lo siguiente:

```
VBoxManage list vms
```



**VirtualBox**  
Download VirtualBox

Here you will find links to VirtualBox binaries and its source code.

**VirtualBox binaries**

By downloading, you agree to the terms and conditions of the respective license.

If you're looking for the latest VirtualBox 6.0 packages, see [VirtualBox 6.0 builds](#). Please also use discontinued in 6.1. Version 6.0 will remain supported until July 2020.

If you're looking for the latest VirtualBox 5.2 packages, see [VirtualBox 5.2 builds](#). Please also use 6.0. Version 5.2 will remain supported until July 2020.

**VirtualBox 6.1.6 platform packages**

- [Windows hosts](#)
- [OS X hosts](#)
- [Linux distributions](#)
- [Solaris hosts](#)

The binaries are released under the terms of the GPL version 2.

See the [changelog](#) for what has changed.

You might want to compare the checksums to verify the integrity of downloaded packages. *The St...*

**Navigation:** About, Screenshots, Downloads, Documentation (End-user docs, Technical docs), Contribute, Community.

**Figura 1.2.** El instalador de VirtualBox está disponible para sistemas Windows, OS X, Linux y Solaris.

En la misma Terminal de comandos verás un listado como el siguiente:

```
Oracle VM VirtualBox Command Line Management Interface
Oracle Corporation
All rights reserved.
"win10" {3f157880-c642-4be2-b641-85d7aedb5090}
"Linux-Mint" {c25e1257-dfed-4789-a22b-8489c4d4df05}
"ubuntu" {fee70808-ab0e-473a-8991-d9b711773672}
"slackware" {f65d5b26-6491-4523-8c06-970cbe6844d5}
"peppermint" {32b1845f-dd72-4c8a-bfe7-8cc3e83d0109}
```

Para obtener información detallada de cada una de las máquinas disponibles, usa el siguiente comando:

```
VBoxManage list vms -l
```

Para iniciar una **máquina virtual** de VirtualBox, utiliza el comando **startvm** seguido del nombre de la máquina virtual:

```
VBoxManage startvm "slackware"
VBoxManage startvm f65d5b26-6491-4523-8c06-970cbe6844d5
```

Para pausar una máquina virtual escribe:

```
VBoxManage controlvm "slackware" pause
```

Para reanudar una máquina virtual pausada escribe:

```
VBoxManage controlvm "slackware" resume
```

Para reiniciar una máquina virtual (apagarla y encenderla nuevamente):

```
VBoxManage controlvm "slackware" reset
```

Para apagar una máquina virtual:

```
VBoxManage controlvm "slackware" poweroff
```

Para detener la máquina virtual, pero guardando su estado actual:

```
VBoxManage controlvm "slackware" savestate
```

Si necesitas crear una máquina virtual con las opciones predeterminadas, desde la Terminal de comandos ejecuta lo siguiente:

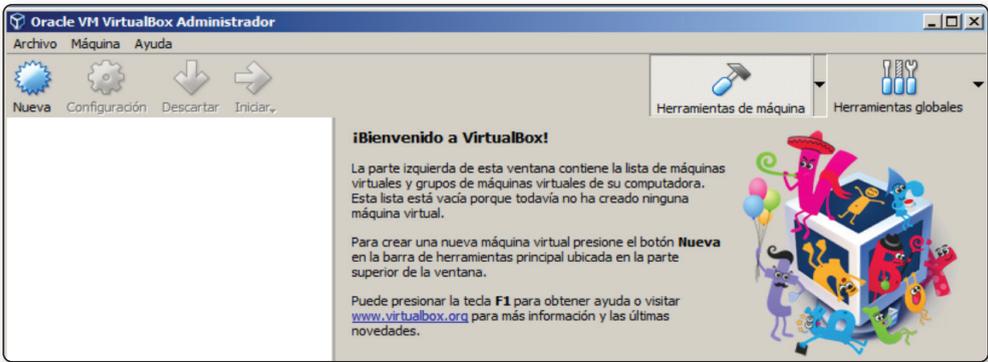
```
VBoxManage createvm -name "LinuxMint" -register
```

### 1.1.1.1 CREAR UNA MÁQUINA VIRTUAL GENERAL

Para crear una máquina virtual tanto en Windows como en Linux utilizando el apartado gráfico, deberás iniciar VirtualBox y seguir las instrucciones mencionadas a continuación.

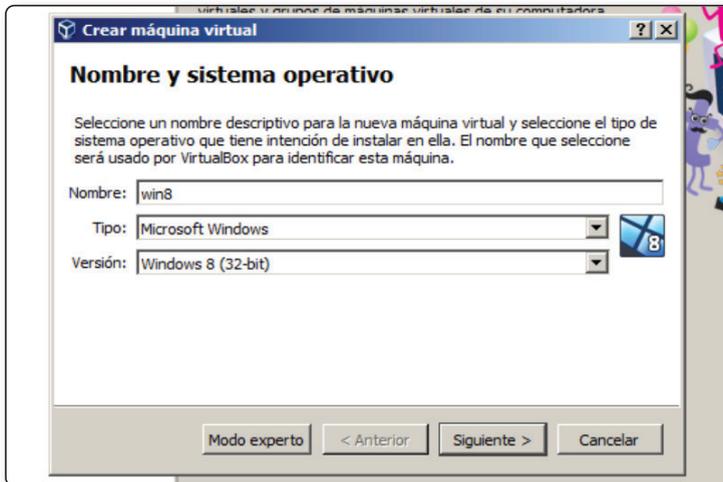
#### PASO 1

Una vez iniciado VirtualBox haces clic sobre el botón **Nueva**, que se encuentra en la barra superior de opciones.



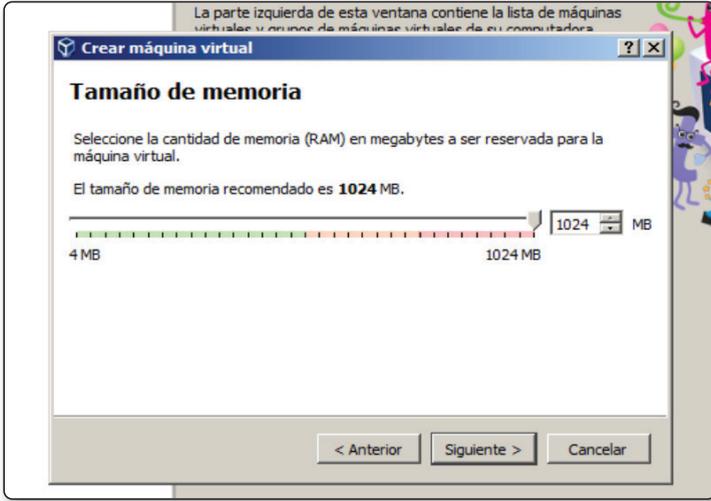
#### PASO 2

En la ventana que aparece, escribe el nombre con el que identificarás la máquina virtual, luego elige el tipo y la versión de SO que virtualizarás. Haz clic en **Siguiente**.



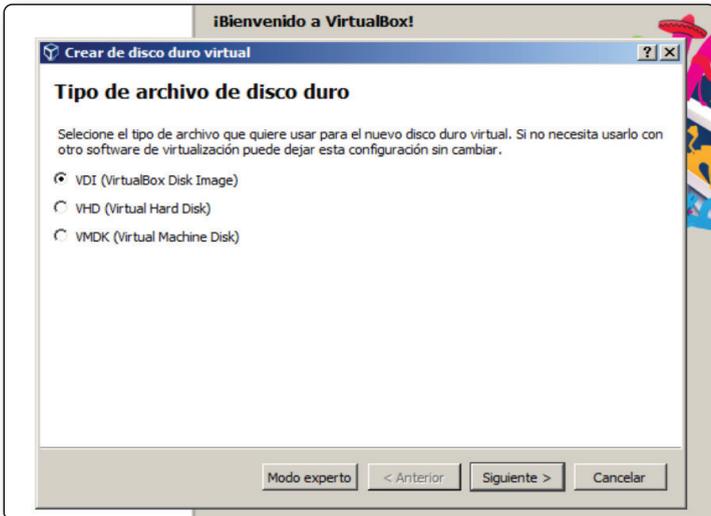
**PASO 3**

Elige el tamaño de memoria RAM que dedicarás al sistema virtualizado, teniendo cuidado de no asignar más de la mitad de RAM real del sistema huésped. Presiona **Siguiente**.



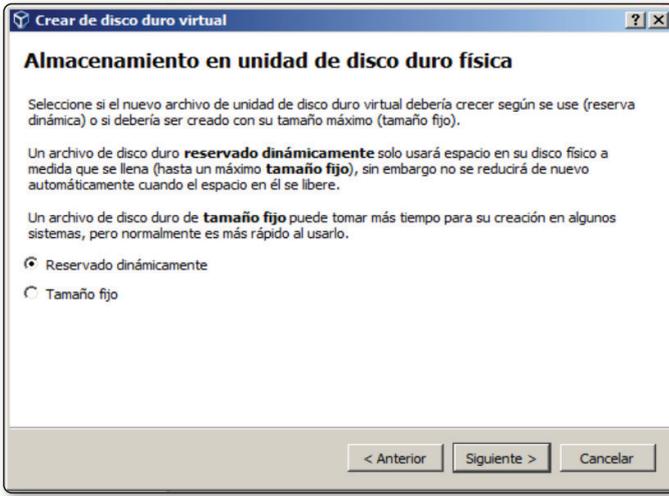
**PASO 4**

Marca la opción **Crear un disco duro virtual ahora** y haz clic sobre **Crear**. En la pantalla que se presenta, elige **VDI** (se trata de la opción predeterminada) y presiona **Siguiente**.



### PASO 5

Elige la opción **Reservado dinámicamente** y presiona **Siguiente**, luego escribe un nombre para identificarlo y desliza el control para definir el tamaño máximo; presiona **Crear**. Luego de esto, nuestra máquina virtual estará lista.



### 1.1.2 Metasploitable

Existen diversas herramientas que pueden ayudar a validar y llevar a cabo pruebas en relación con la seguridad de un sistema operativo, pero sin duda la mejor opción es realizar las pruebas de penetración y vulnerabilidad en el propio sistema operativo para comprobar los problemas de seguridad en detalle y en contexto.

En este sentido puedes hacer uso de Metasploitable, un sistema operativo diseñado teniendo en cuenta que puede ser vulnerado para que logres ensayar las pruebas de penetración con el fin de mejorar la seguridad y prevenir los ataques (Figura 1.3).



Figura 1.3.

Metasploitable te provee diversas **vulnerabilidades de seguridad** para llevar a cabo todas las pruebas necesarias que te permitan perfeccionar las técnicas de seguridad. Este sistema, en su versión Linux, no cuenta con entorno gráfico y debes utilizarlo en redes privadas debido a su tolerancia a ataques.

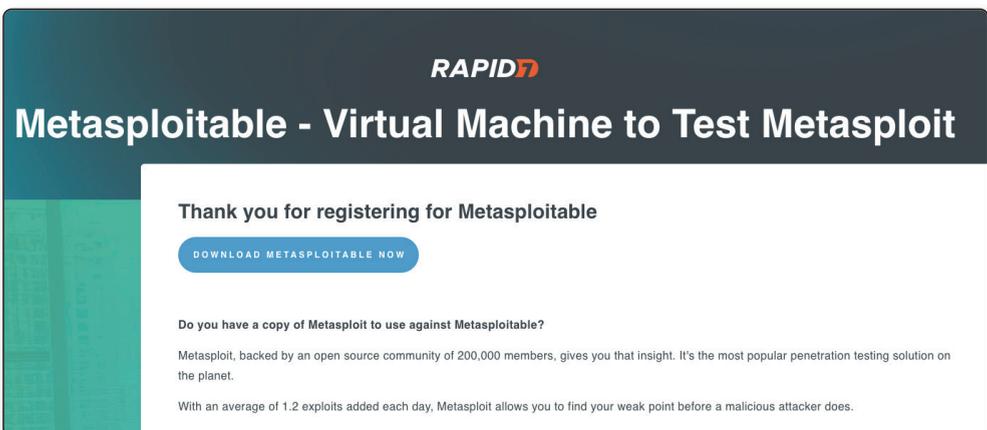
Es un sistema de **código abierto** y te permitirá realizar pruebas de vulnerabilidades en archivos incrustados, atributos de archivo, permisos, entre otros.

Para utilizar Metasploitable en tu laboratorio de pruebas, necesitas contar con dos elementos:

- VirtualBox, que ya instalaste en la sección anterior.
- Disco virtual de Metasploitable, que conseguirás a continuación.

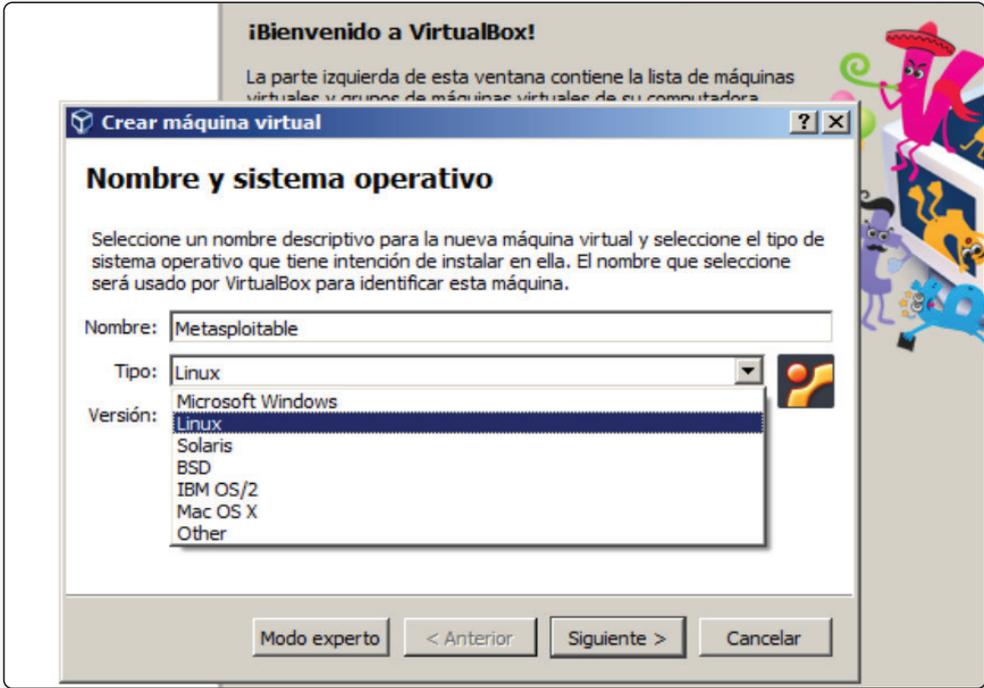
Metasploitable, en sus primeras versiones, funcionaba como una distribución Linux especialmente configurada para ser vulnerable; en su versión 3, toma como base Microsoft Windows Server 2008 R2, pero por restricciones de licencia no te pueden proporcionar una máquina ya lista, sino que debes generarla tú mismo. Para comenzar el trabajo en esta colección, se utilizará la máquina de Linux ya configurada.

Para obtener el disco de Metasploitable adecuado, debes acceder a la dirección <https://information.rapid7.com/download-metasploitable-2017.html>, completar los datos requeridos por el formulario y presionar **Submit**. Es importante tener en cuenta que, para realizar la descarga de Metasploitable, necesitas contar con un correo electrónico corporativo, de lo contrario no podrás pasar del formulario.



**Figura 1.4.** En la pantalla que se presenta haz clic sobre **DOWNLOAD METASPLOITABLE NOW**, la descarga supera los 800 MB.

Una vez que Metasploitable complete su descarga, inicia VirtualBox y haz clic sobre **Nueva**. Como nombre de la máquina escribe **Metasploitable** y elige **Linux, Ubuntu (64 bits)**.



**Figura 1.5.** Aunque no es imprescindible que el nombre de la nueva máquina virtual sea Metasploitable, es una buena idea para diferenciarla de otras máquinas ya creadas.

En la siguiente ventana deja la memoria RAM asignada en forma predeterminada, es decir, **1024 MB**. Para continuar, crea un disco duro virtual con la opción **VDI** y **Reservado dinámicamente**.

En la siguiente ventana, define una ruta donde almacenar la máquina virtual y asigna la capacidad mínima del disco duro que permite la aplicación: **10 GB**. Pulsa sobre **Crear** para completar el proceso (**Figura 1.6.**).



Figura 1.6. Luego de completar las indicaciones anteriores, la máquina virtual denominada Metasploitable ya estará creada.

### 1.1.2.1 CONFIGURAR UNA MÁQUINA VIRTUAL METASPLOITABLE

Ahora que tu máquina virtual de Metasploitable ya está creada, debes configurarla. Para ello, selecciónala y haz clic sobre **Configuración**, elige **Almacenamiento** y pulsa sobre el disco duro virtual llamado **Metasploitable.vdi**. Pulsa sobre el icono ubicado al lado del campo **Disco duro** y elige la opción **Seleccione archivo de disco duro virtual**; en la ventana desplegada, ubica el disco duro virtual de Metasploitable que descargaste en la sección anterior (Figura 1.7.).

Antes de elegir la imagen de disco, será necesario descomprimir el archivo descargado, luego de esta operación su peso superará los 1.90 GB. Haz clic sobre **Aceptar** para completar la configuración, luego presiona **Iniciar** para arrancar la máquina virtual y completar el proceso de instalación (Figuras 1.8 y 1.9.).

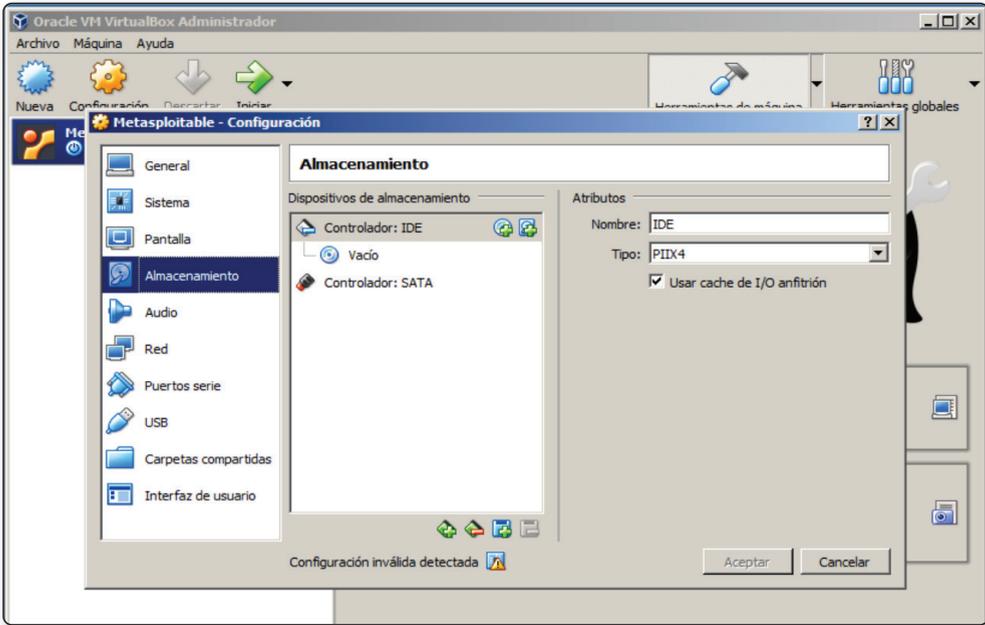


Figura 1.7. En la sección Atributos/Disco duro puedes elegir la imagen de disco virtual que utilizarás para arrancar tu máquina virtual.

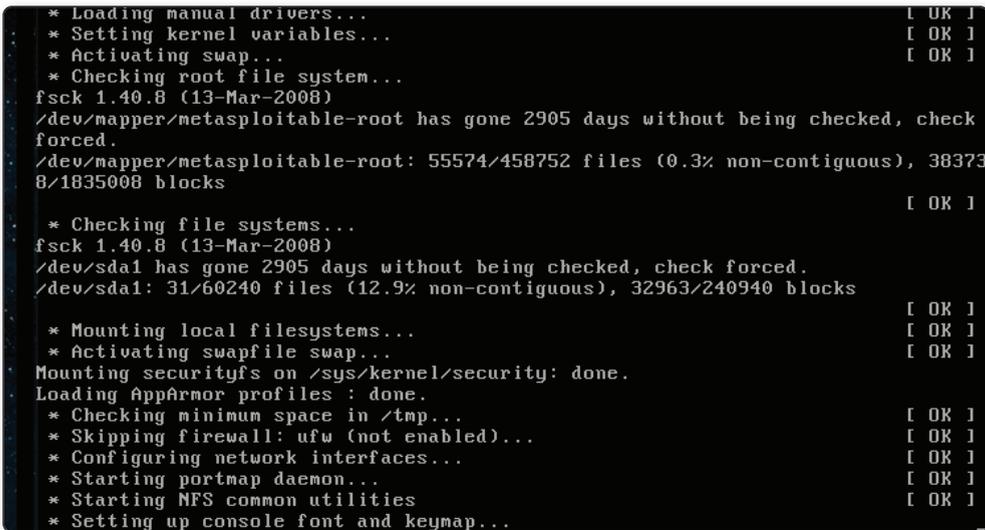


Figura 1.8. Mientras las tareas necesarias se realizan, verás la indicación de los procesos en pantalla, tal como si estuvieses realizando la instalación en una máquina física.

```
* Starting deferred execution scheduler atd [ OK ]
* Starting periodic command scheduler crond [ OK ]
* Starting Tomcat servlet engine tomcat5.5 [ OK ]
* Starting web server apache2 [ OK ]
* Running local boot scripts (/etc/rc.local)
nohup: appending output to `nohup.out'
nohup: appending output to `nohup.out' [ OK ]

Warning: Never expose this VM to an untrusted network!
Contact: msfdev[at]metasploit.com
Login with msfadmin/msfadmin to get started

metasploitable login: _
```

**Figura 1.9.** Una vez que la instalación se complete, te encontrarás con esta pantalla, que solicita los datos de ingreso al sistema.

El proceso de instalación puede tardar unos cinco minutos, una vez completado utiliza las siguientes credenciales de acceso:

Usuario: **msfadmin**  
Contraseña: **msfadmin**

### 1.1.2.2 METASPLOITABLE 2

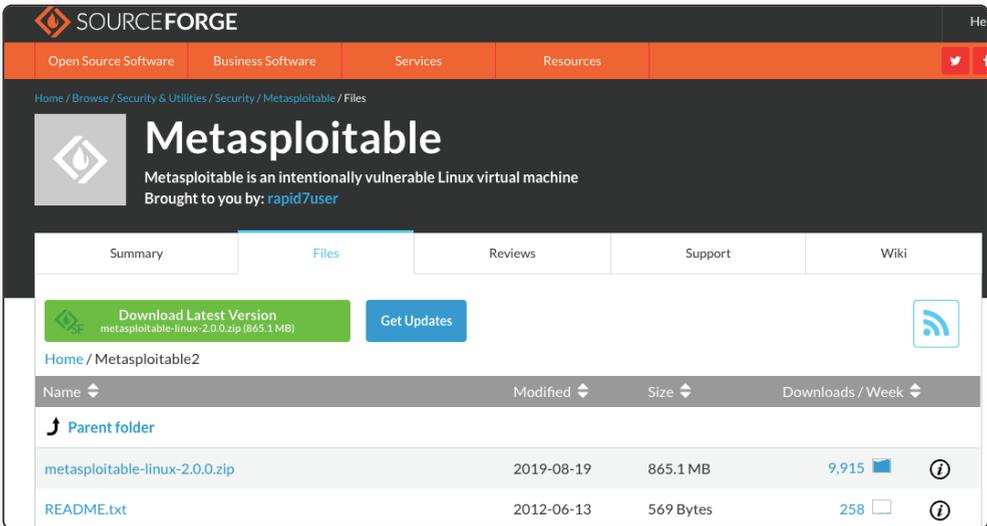
**Metasploitable 2** es una evolución de la máquina virtual original, que incorpora problemas de seguridad más actuales para ser explotadas.

Para utilizar esta opción, debes seguir un procedimiento similar al que se explicó para Metasploitable:

1. Descarga Metasploitable 2.
2. Descomprime el archivo descargado para obtener la imagen de instalación.
3. Crea una nueva máquina virtual en VirtualBox.
4. Instala Metasploitable2.

En primer lugar dirígete a la dirección:

<http://sourceforge.net/projects/metasploitable>  
para descargar Metasploitable 2.



**Figura 1.10.** En la sección Files de esta web puedes ver el detalle de la descarga, observa que el archivo que contiene la imagen comprimida posee un tamaño de 865.1 MB.

Una vez que la imagen ha sido descargada la descomprimes. Luego de esto obtendrás cuatro archivos, uno de ellos con la extensión **.vmdk**.

Ahora inicia VirtualBox y crea una nueva máquina virtual; sigue las indicaciones del asistente para configurarla, indicando el archivo con extensión **.vmdk** como imagen de disco de arranque. Para elegir los detalles puedes copiar las instrucciones indicadas en la sección anterior, sobre la instalación de Metasploitable.

Una vez que la instalación finalice, utiliza los siguientes datos para iniciar una sesión de trabajo:

Usuario: **msfadmin**  
 Contraseña: **msfadmin**

### 1.1.2.3 METASPLOITABLE 3

Hasta ahora se ha analizado la instalación de Metasploitable y Metasploitable 2 como máquinas vulnerables. Se trata de opciones que quienes están relacionados con el Ethical Hacking y el test de penetración han utilizado por años, ya sea para pruebas de explotación de red, desarrollo de exploits, evaluación de software, identificación de vulnerabilidades, entre otras opciones. Metasploitable y Metasploitable 2 se

presentan como un disco duro que se instala en forma sencilla sobre Vmware o VirtualBox en este caso, y hace algún tiempo surgió una versión 3.

**Metasploitable 3** posee una lógica diferente y, por lo tanto, su instalación se realiza de una forma también distinta.

Esta nueva máquina presenta una serie de vulnerabilidades más actuales, que resultan interesantes para desarrollar diversas habilidades relacionadas con la seguridad. A diferencia de las versiones anteriores de Metasploitable que se ofrecían como máquinas virtuales ya preparadas, esta nueva opción depende de Vagrant y Packer para compilar la imagen en el sistema. De esta forma es más dinámica y permite que los usuarios participen en su generación, está disponible con los sistemas **Windows Server 2008** y **Ubuntu 14.04** como base.

Para utilizar estas máquinas virtuales, es necesario contar con un sistema operativo compatible con las dependencias que debes instalar, además con un procesador que soporte las funciones de virtualización (VT-x o AMD-V), 4.5 GB de memoria RAM y 65 GB de espacio en el disco duro. Se trata de altas exigencias, por lo que es una buena idea comenzar con las versiones anteriores de Metasploitable, al menos en los primeros pasos en las tareas de explotar vulnerabilidades.

En cuanto a las dependencias, necesitas contar con las herramientas **Packer**, **Vagrant**, **Vagrant Reload Plugin** y, por supuesto, con un sistema de virtualización como VMWare o VirtualBox (el que se utiliza en esta colección). Es posible que te des a la tarea de compilar tú mismo la imagen o también descargar las versiones ya compiladas; esta última opción es la más sencilla.

Para compilar la máquina virtual primero instala los prerequisites.

Para instalar Packer:

```
wget https://releases.hashicorp.com/packer/1.1.3/packer_1.1.3_linux_amd64.zip?_ga=2.259436163.8806393.1516559508-2105737727.1516559508 -O packer_1.1.3_linux_amd64.zip
unzip packer_1.1.3_linux_amd64.zip
sudo mkdir /usr/local/packer
sudo mv packer /usr/local/packer/
nano ~/.profile
```

Al final del archivo agrega lo siguiente:

```
export PATH=$PATH:/usr/local/packer
```

Luego ejecuta:

```
source ~/.profile
```

Y finalmente:

```
packer -version
```

Deberías ver un mensaje que muestre la versión de Packer instalada.

Para instalar Vagrant, escribe los siguientes comandos:

```
wget https://releases.hashicorp.com/vagrant/2.0.1/vagrant_2.0.1_x86_64.deb?_ga=2.260144013.1615441003.1516560525-1954706866.1516560525 -O vagrant_2.0.1_x86_64.deb
sudo dpkg -i vagrant_2.0.1_x86_64.deb
vagrant plugin install vagrant-reload
```

Ahora, para realizar la instalación de Metasploitable3:

```
git clone https://github.com/rapid7/metasploitable3.git
cd metasploitable3/
```

Luego de esto comienza con la construcción de la máquina virtual, en el caso de la máquina de Windows 2008:

```
packer build windows_2008_r2.json
```

Ahora debes esperar a que la descarga se complete. Una vez que este proceso haya terminado, ejecuta el siguiente comando:

```
vagrant box add windows_2008_r2_virtualbox.box -name=metasploitable3
```

Después, la importas hacia VirtualBox:

```
vagrant up
```

Si no quieres realizar el proceso de **compilación** o te encuentras con algún problema complejo, puedes descargar las versiones ya compiladas, para ello procede de la siguiente forma.

```

Bringing machine 'default' up with 'virtualbox' provider...
==> default: Importing base box 'metasploitable3'...
==> default: Matching MAC address for NAT networking...
==> default: Setting the name of the VM: metasploitable3_default_1516568941153_91940
==> default: Clearing any previously set network interfaces...
==> default: Preparing network interfaces based on configuration...
    default: Adapter 1: nat
    default: Adapter 2: hostonly
==> default: Forwarding ports...
    default: 3389 (guest) => 3389 (host) (adapter 1)
    default: 22 (guest) => 2222 (host) (adapter 1)
    default: 5985 (guest) => 55985 (host) (adapter 1)
    default: 5986 (guest) => 55986 (host) (adapter 1)
==> default: Running 'pre-boot' VM customizations...
==> default: Booting VM...
==> default: Waiting for machine to boot. This may take a few minutes...
    default: WinRM address: 127.0.0.1:55985
    default: WinRM username: vagrant
    default: WinRM execution time limit: PT2H
    default: WinRM transport: negotiate
==> default: Machine booted and ready!
==> default: Checking for guest additions in VM...
==> default: Setting hostname...
==> default: Configuring and enabling network interfaces...
==> default: Mounting shared folders...
    default: /vagrant => /home/jr0llv4/Downloads/metasploitable3
==> default: Running provisioner: shell...
    default: Running: scripts/configs/enable_firewall.bat as c:\tmp\vagrant-shell.bat
    default: Ok.
==> default: Running provisioner: shell...
    default: Running: scripts/configs/configure_firewall.bat as c:\tmp\vagrant-shell.bat
    default: Ok.
    default: Ok.

```

**Figura 1.11.** Con esto ya terminas la construcción de la máquina virtual, que ya estará disponible en la lista de máquinas de VirtualBox.

Si deseas contar con Metasploitable 3 basado en Windows Server 2008, debes ejecutar lo siguiente en Vagrant:

```

Vagrant.configure("2") do |config|
  config.vm.box = "rapid7/metasploitable3-win2k8"
  config.vm.box_version = "0.1.0-weekly"
end

```

Si deseas montar Metasploitable3 basado en Ubuntu 14.04, será necesario ejecutar el siguiente box en Vagrant:

```

Vagrant.configure("2") do |config|
  config.vm.box = "rapid7/metasploitable3-ub1404"
  config.vm.box_version = "0.1.12-weekly"
end

```

### 1.1.3 Kali Linux

Kali Linux es una distribución basada en Debian; se trata de un sistema operativo desarrollado para realizar pruebas avanzadas de penetración y auditorías de seguridad que permiten conocer en tiempo real el nivel de seguridad de un sitio, una arquitectura de red, un sistema o una implementación de aplicaciones (**Figura 1.12**).

Esto se logra gracias a la integración de más de 600 herramientas diseñadas para este fin. Para agregar Kali Linux a tu laboratorio de pruebas, accede a la dirección <https://www.kali.org/downloads/> y elige la imagen de disco adecuada, ya sea para instalar o para ejecutar en forma de sistema Live, es decir, sin instalarla en el disco.



Figura 1.12.

Como una opción adicional, en este sitio web también puedes acceder a la descarga de una imagen de Kali Linux especialmente preparada para VMWare o para VirtualBox. En este caso se ha elegido la opción **Kali Linux VirtualBox 64-Bit** (Figura 1.13).

Inicia VirtualBox y haz clic sobre **Importar servicio virtualizado**; elige el archivo de Kali Linux descargado y presiona **Abrir**. Verifica los datos contenidos en la pantalla **Preferencias de servicio** y haz clic sobre **Importar** (Figura 1.14).

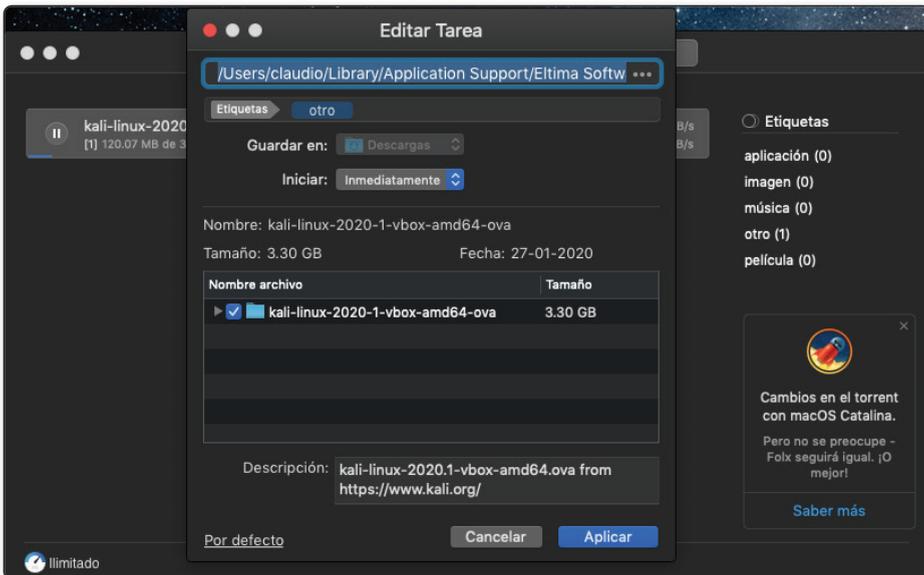
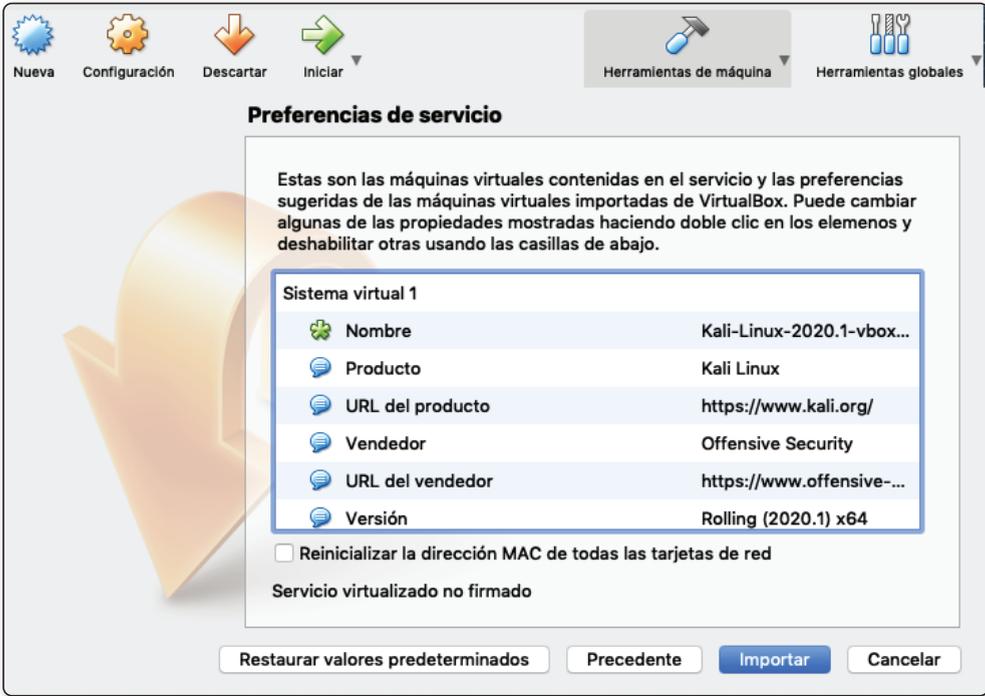


Figura 1.13. Por tratarse de un archivo que supera los 3 GB, puedes utilizar el archivo torrent para descargarlo.



**Figura 1.14.** En Preferencias de servicio se listan todos los detalles relacionados con la máquina virtual que estás importando, en este caso Kali Linux.

Luego debes esperar mientras la importación se realiza. Esto tardará varios minutos y, una vez que este proceso termine, verás la máquina Kali Linux en la lista de máquinas disponibles. Para iniciarla solo debes hacer clic sobre ella y elegir **Iniciar** en el menú superior de opciones.

Luego de iniciarla puede presentarse un error en el puerto USB 2.0, debes deshabilitarlo y volver a iniciar la máquina virtual; puedes hacerlo en la sección **Configuración/Puertos/USB**. Para solucionarlo por completo, instala el paquete de extras de VirtualBox una vez que la máquina se haya iniciado.

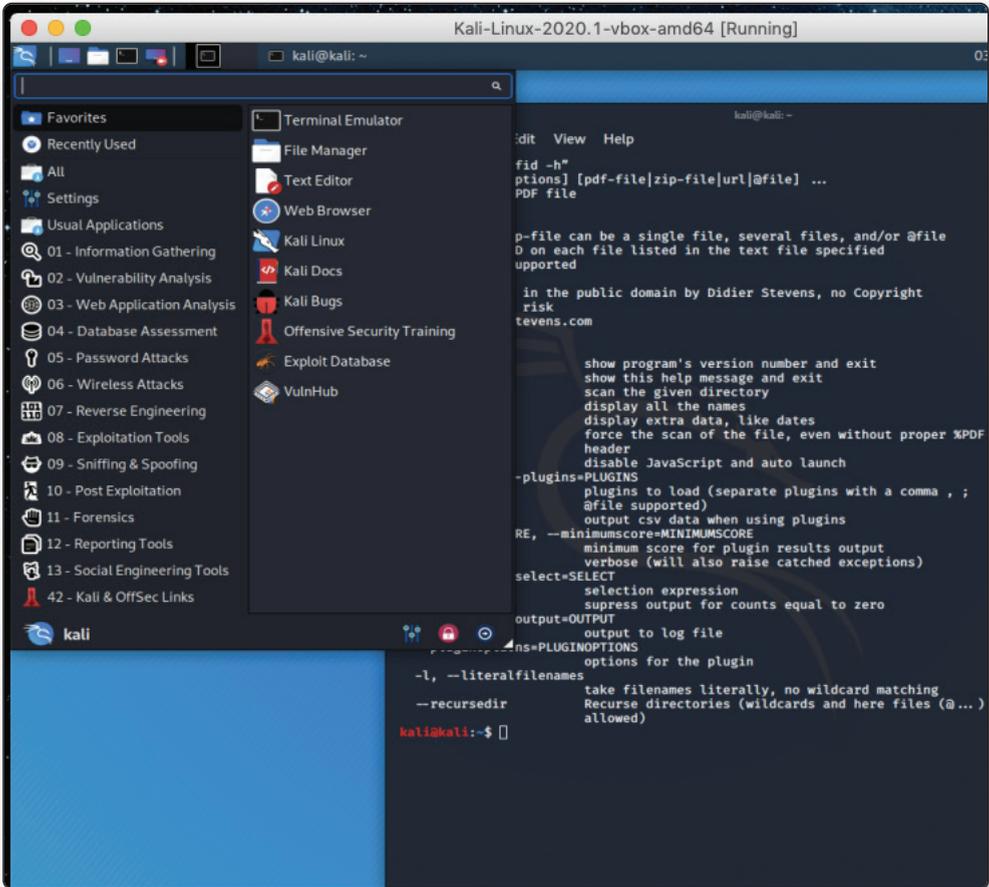


Figura 1.15. Una vez que el sistema haya arrancado, puedes utilizar kali/kali como datos de usuario/contraseña. Luego de eso te encontrarás en el escritorio de Kali Linux.

### 1.1.4 Conectar ambas máquinas virtuales

En este punto ya tienes VirtualBox instalado y dos máquinas virtuales funcionando, una con Metasploitable y otra con Kali Linux. Lo que necesitas es que ambas puedan verse, es decir, que funcionen en la misma red, de esta forma accederás a Metasploitable desde Kali Linux para realizar las tareas de análisis de seguridad.

Para lograr que ambas máquinas estén en la misma red, ajusta sus configuraciones.

En primer lugar, elige Metasploitable desde el listado de máquinas virtuales de VirtualBox y haz clic sobre **Configuración**. Ve a la sección **Red** y en **Conectado a** elige **Adaptador puente**. Repite este procedimiento en la máquina virtual de Kali Linux.

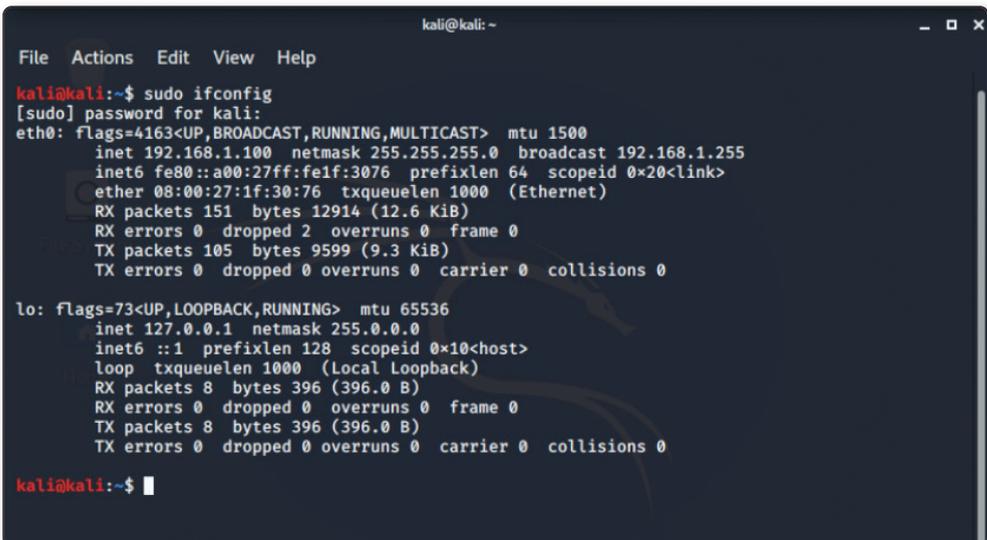
Adaptador puente es una forma útil para conectar las máquinas virtuales. Se trata de un tipo de conexión que se encarga de simular una conexión física a la red de la máquina virtual. Así, la máquina virtual quedará conectada a través de un adaptador de red creado en la máquina host en forma directa al router o al servidor de tu entorno de red.

Con este procedimiento, cada máquina virtual obtendrá una dirección IP desde la puerta de enlace a internet, por lo que dentro de tus máquinas virtuales tendrás las mismas posibilidades que si estuvieras frente a un equipo físico.

Ahora debes iniciar ambas máquinas virtuales y averiguar la **dirección IP** de cada una; para ello ejecuta el comando **ifconfig** en la terminal de Kali Linux y también de Metasploitable (**Figura 1.16** y **1.17**).

Aunque después de verificar las direcciones IP no es necesario ejecutar un **ping** para realizar un rastreo de paquetes, igual puedes hacerlo como una forma adicional de verificar la conexión entre ambas máquinas. Abre la terminal de comandos en Kali Linux y escribe lo siguiente:

```
ping 192.168.1.101
```



```
kali@kali: ~  
File Actions Edit View Help  
kali@kali:~$ sudo ifconfig  
[sudo] password for kali:  
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500  
inet 192.168.1.100 netmask 255.255.255.0 broadcast 192.168.1.255  
inet6 fe80::a00:27ff:fe1f:3076 prefixlen 64 scopeid 0<link>  
ether 08:00:27:1f:30:76 txqueuelen 1000 (Ethernet)  
RX packets 151 bytes 12914 (12.6 KiB)  
RX errors 0 dropped 2 overruns 0 frame 0  
TX packets 105 bytes 9599 (9.3 KiB)  
TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0  
  
lo: flags=73<UP,LOOPBACK,RUNNING> mtu 65536  
inet 127.0.0.1 netmask 255.0.0.0  
inet6 ::1 prefixlen 128 scopeid 0<host>  
loop txqueuelen 1000 (Local Loopback)  
RX packets 8 bytes 396 (396.0 B)  
RX errors 0 dropped 0 overruns 0 frame 0  
TX packets 8 bytes 396 (396.0 B)  
TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0  
  
kali@kali:~$
```

**Figura 1.16.** Luego de ejecutar el comando `sudo ifconfig`, verificarás que, en este caso, se ha asignado la dirección IP 192.168.1.100 a la máquina con Kali Linux.

```

msfadmin@metasploitable:~$ ifconfig
eth0      Link encap:Ethernet  HWaddr 08:00:27:5f:dd:37
          inet addr:192.168.1.101  Bcast:192.168.1.255  Mask:255.255.255.0
          inet6 addr: fe80::a00:27ff:fe5f:dd37/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:3305 errors:0 dropped:0 overruns:0 frame:0
          TX packets:1912 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:300165 (293.1 KB)  TX bytes:182893 (178.6 KB)
          Base address:0xd010 Memory:f0000000-f0020000

lo        Link encap:Local Loopback
          inet addr:127.0.0.1  Mask:255.0.0.0
          inet6 addr: ::1/128 Scope:Host
          UP LOOPBACK RUNNING  MTU:16436  Metric:1
          RX packets:214 errors:0 dropped:0 overruns:0 frame:0
          TX packets:214 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:0
          RX bytes:79221 (77.3 KB)  TX bytes:79221 (77.3 KB)

msfadmin@metasploitable:~$

```

Figura 1.17. El comando `ifconfig` en Metasploitable te informa que la dirección IP asignada a la máquina virtual es 192.168.1.101.

Luego de algunas líneas, presiona las teclas **CTRL+C** para cancelar el **ping** y ver un resumen.

```

kali@kali: ~
File Actions Edit View Help

RX packets 151 bytes 12914 (12.6 KiB)
RX errors 0 dropped 2 overruns 0 frame 0
TX packets 105 bytes 9599 (9.3 KiB)
TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

lo: flags=73<UP,LOOPBACK,RUNNING> mtu 65536
   inet 127.0.0.1 netmask 255.0.0.0
   inet6 ::1 prefixlen 128 scopeid 0<host>
   loop txqueuelen 1000 (Local Loopback)
   RX packets 8 bytes 396 (396.0 B)
   RX errors 0 dropped 0 overruns 0 frame 0
   TX packets 8 bytes 396 (396.0 B)
   TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

kali@kali:~$ ping 192.168.1.101
PING 192.168.1.101 (192.168.1.101) 56(84) bytes of data:
 64 bytes from 192.168.1.101: icmp_seq=1 ttl=64 time=5.04 ms
 64 bytes from 192.168.1.101: icmp_seq=2 ttl=64 time=0.526 ms
 64 bytes from 192.168.1.101: icmp_seq=3 ttl=64 time=0.487 ms
 64 bytes from 192.168.1.101: icmp_seq=4 ttl=64 time=0.528 ms
 64 bytes from 192.168.1.101: icmp_seq=5 ttl=64 time=0.637 ms
 64 bytes from 192.168.1.101: icmp_seq=6 ttl=64 time=0.786 ms
 64 bytes from 192.168.1.101: icmp_seq=7 ttl=64 time=0.732 ms
 64 bytes from 192.168.1.101: icmp_seq=8 ttl=64 time=0.641 ms
 64 bytes from 192.168.1.101: icmp_seq=9 ttl=64 time=0.479 ms
 64 bytes from 192.168.1.101: icmp_seq=10 ttl=64 time=0.532 ms
^C
--- 192.168.1.101 ping statistics ---
 10 packets transmitted, 10 received, 0% packet loss, time 9182ms
 rtt min/avg/max/mdev = 0.479/1.038/5.036/1.336 ms
kali@kali:~$

```

Figura 1.18. En esta imagen se observa que se transmitieron 10 paquetes y ninguno de ellos se perdió.

## 1.2 ACTIVIDADES

---

A continuación verás las preguntas y los ejercicios que deberías saber responder y resolver, para considerar aprendido el capítulo.

### 1.2.1 Test de autoevaluación

1. *¿Qué son las pruebas de penetración?*
2. *¿Qué es Metasploitable?*
3. *¿Para qué sirve Kali Linux?*

### 1.2.2 Ejercicios prácticos

1. *Instala VirtualBox en tu sistema operativo.*
2. *Configura tu laboratorio de pruebas.*
3. *Crea una máquina virtual.*
4. *Configura una máquina con Metasploitable en VirtualBox.*

