



Red Hat Enterprise Linux 8

Configuración y gestión de la virtualización

Configuración de su anfitrión, creación y administración de máquinas virtuales, y comprensión de las características de virtualización en Red Hat Enterprise Linux 8

Red Hat Enterprise Linux 8 Configuración y gestión de la virtualización

Configuración de su anfitrión, creación y administración de máquinas virtuales, y comprensión de las características de virtualización en Red Hat Enterprise Linux 8

Enter your first name here. Enter your surname here.

Enter your organisation's name here. Enter your organisational division here.

Enter your email address here.

Legal Notice

Copyright © 2021 | You need to change the HOLDER entity in the en-US/Configuring_and_managing_virtualization.ent file |.

The text of and illustrations in this document are licensed by Red Hat under a Creative Commons Attribution–Share Alike 3.0 Unported license ("CC-BY-SA"). An explanation of CC-BY-SA is available at

<http://creativecommons.org/licenses/by-sa/3.0/>

. In accordance with CC-BY-SA, if you distribute this document or an adaptation of it, you must provide the URL for the original version.

Red Hat, as the licensor of this document, waives the right to enforce, and agrees not to assert, Section 4d of CC-BY-SA to the fullest extent permitted by applicable law.

Red Hat, Red Hat Enterprise Linux, the Shadowman logo, the Red Hat logo, JBoss, OpenShift, Fedora, the Infinity logo, and RHCE are trademarks of Red Hat, Inc., registered in the United States and other countries.

Linux[®] is the registered trademark of Linus Torvalds in the United States and other countries.

Java[®] is a registered trademark of Oracle and/or its affiliates.

XFS[®] is a trademark of Silicon Graphics International Corp. or its subsidiaries in the United States and/or other countries.

MySQL[®] is a registered trademark of MySQL AB in the United States, the European Union and other countries.

Node.js[®] is an official trademark of Joyent. Red Hat is not formally related to or endorsed by the official Joyent Node.js open source or commercial project.

The OpenStack[®] Word Mark and OpenStack logo are either registered trademarks/service marks or trademarks/service marks of the OpenStack Foundation, in the United States and other countries and are used with the OpenStack Foundation's permission. We are not affiliated with, endorsed or sponsored by the OpenStack Foundation, or the OpenStack community.

All other trademarks are the property of their respective owners.

Resumen

Este documento describe cómo gestionar la virtualización en Red Hat Enterprise Linux 8 (RHEL 8). Además de la información general sobre la virtualización, describe cómo gestionar la virtualización utilizando las utilidades de línea de comandos, así como el uso de la consola web.

Table of Contents

HACER QUE EL CÓDIGO ABIERTO SEA MÁS INCLUSIVO	8
PROPORCIONAR COMENTARIOS SOBRE LA DOCUMENTACIÓN DE RED HAT	9
CAPÍTULO 1. VIRTUALIZACIÓN EN RHEL 8 - UNA VISIÓN GENERAL	10
1.1. ¿QUÉ ES LA VIRTUALIZACIÓN?	10
1.2. VENTAJAS DE LA VIRTUALIZACIÓN	10
1.3. COMPONENTES DE LA MÁQUINA VIRTUAL Y SU INTERACCIÓN	11
1.4. HERRAMIENTAS E INTERFACES PARA LA GESTIÓN DE LA VIRTUALIZACIÓN	12
1.5. SOLUCIONES DE VIRTUALIZACIÓN DE RED HAT	14
CAPÍTULO 2. INTRODUCCIÓN A LA VIRTUALIZACIÓN	16
2.1. HABILITACIÓN DE LA VIRTUALIZACIÓN	16
2.2. CREACIÓN DE MÁQUINAS VIRTUALES	17
2.2.1. Creación de máquinas virtuales mediante la interfaz de línea de comandos	18
2.2.2. Creación de máquinas virtuales e instalación de sistemas operativos invitados mediante la consola web	21
2.2.2.1. Creación de máquinas virtuales mediante la consola web	21
2.2.2.2. Creación de máquinas virtuales mediante la importación de imágenes de disco a través de la consola web	23
2.2.2.3. Instalación de sistemas operativos invitados mediante la consola web	24
2.3. INICIAR MÁQUINAS VIRTUALES	25
2.3.1. Iniciar una máquina virtual mediante la interfaz de línea de comandos	25
2.3.2. Iniciar máquinas virtuales mediante la consola web	26
2.4. CONEXIÓN A MÁQUINAS VIRTUALES	27
2.4.1. Interacción con las máquinas virtuales mediante la consola web	27
2.4.1.1. Visualización de la consola gráfica de la máquina virtual en la consola web	27
2.4.1.2. Visualización de la consola gráfica en un visor remoto mediante la consola web	29
2.4.1.3. Visualización de la consola de serie de la máquina virtual en la consola web	31
2.4.2. Abrir la consola gráfica de una máquina virtual con Virt Viewer	32
2.4.3. Conexión a una máquina virtual mediante SSH	34
2.4.4. Abrir la consola de serie de una máquina virtual	35
2.4.5. Configuración de un acceso fácil a los hosts de virtualización remotos	37
2.5. APAGADO DE MÁQUINAS VIRTUALES	39
2.5.1. Apagar una máquina virtual mediante la interfaz de línea de comandos	39
2.5.2. Apagado y reinicio de máquinas virtuales mediante la consola web	40
2.5.2.1. Apagado de máquinas virtuales en la consola web	40
2.5.2.2. Reinicio de máquinas virtuales mediante la consola web	40
2.5.2.3. Envío de interrupciones no enmascarables a las máquinas virtuales mediante la consola web	41
2.6. ELIMINACIÓN DE MÁQUINAS VIRTUALES	42
2.6.1. Eliminación de máquinas virtuales mediante la interfaz de línea de comandos	42
2.6.2. Eliminación de máquinas virtuales mediante la consola web	42
2.7. INFORMACIÓN RELACIONADA	43
CAPÍTULO 3. INTRODUCCIÓN A LA VIRTUALIZACIÓN EN IBM POWER	45
3.1. HABILITACIÓN DE LA VIRTUALIZACIÓN EN IBM POWER	45
3.2. EN QUÉ SE DIFERENCIA LA VIRTUALIZACIÓN EN IBM POWER DE LA DE AMD64 E INTEL 64	46
CAPÍTULO 4. INTRODUCCIÓN A LA VIRTUALIZACIÓN EN IBM Z	49
4.1. HABILITACIÓN DE LA VIRTUALIZACIÓN EN IBM Z	49
4.2. EN QUÉ SE DIFERENCIA LA VIRTUALIZACIÓN EN IBM Z DE LA DE AMD64 E INTEL 64	50
4.3. INFORMACIÓN RELACIONADA	52

CAPÍTULO 5. GESTIÓN DE MÁQUINAS VIRTUALES EN LA CONSOLA WEB	53
5.1. VISIÓN GENERAL DE LA GESTIÓN DE MÁQUINAS VIRTUALES MEDIANTE LA CONSOLA WEB	53
5.2. CONFIGURACIÓN DE LA CONSOLA WEB PARA GESTIONAR LAS MÁQUINAS VIRTUALES	53
5.3. FUNCIONES DE GESTIÓN DE MÁQUINAS VIRTUALES DISPONIBLES EN LA CONSOLA WEB	54
5.4. DIFERENCIAS ENTRE LAS FUNCIONES DE VIRTUALIZACIÓN EN VIRTUAL MACHINE MANAGER Y LA CONSOLA WEB	55
CAPÍTULO 6. VER INFORMACIÓN SOBRE LAS MÁQUINAS VIRTUALES	58
6.1. VISUALIZACIÓN DE LA INFORMACIÓN DE LA MÁQUINA VIRTUAL MEDIANTE LA INTERFAZ DE LÍNEA DE COMANDOS	58
6.2. VISUALIZACIÓN DE LA INFORMACIÓN DE LA MÁQUINA VIRTUAL MEDIANTE LA CONSOLA WEB	60
6.2.1. Ver un resumen de la virtualización en la consola web	60
6.2.2. Visualización de la información del pool de almacenamiento mediante la consola web	61
6.2.3. Visualización de la información básica de la máquina virtual en la consola web	63
6.2.4. Ver el uso de recursos de la máquina virtual en la consola web	65
6.2.5. Visualización de la información del disco de la máquina virtual en la consola web	66
6.2.6. Ver y editar la información de la interfaz de red virtual en la consola web	67
6.3. EJEMPLO DE CONFIGURACIÓN XML DE UNA MÁQUINA VIRTUAL	69
CAPÍTULO 7. GUARDAR Y RESTAURAR MÁQUINAS VIRTUALES	74
7.1. CÓMO FUNCIONA EL GUARDADO Y LA RESTAURACIÓN DE MÁQUINAS VIRTUALES	74
7.2. GUARDAR UNA MÁQUINA VIRTUAL MEDIANTE LA INTERFAZ DE LÍNEA DE COMANDOS	74
7.3. INICIAR UNA MÁQUINA VIRTUAL MEDIANTE LA INTERFAZ DE LÍNEA DE COMANDOS	75
7.4. INICIAR MÁQUINAS VIRTUALES MEDIANTE LA CONSOLA WEB	76
CAPÍTULO 8. CLONACIÓN DE MÁQUINAS VIRTUALES	78
8.1. CÓMO FUNCIONA LA CLONACIÓN DE MÁQUINAS VIRTUALES	78
8.2. CREACIÓN DE UNA PLANTILLA DE MÁQUINA VIRTUAL	78
8.3. CLONACIÓN DE UNA MÁQUINA VIRTUAL MEDIANTE LA INTERFAZ DE LÍNEA DE COMANDOS	80
CAPÍTULO 9. MIGRACIÓN DE MÁQUINAS VIRTUALES	82
9.1. CÓMO FUNCIONA LA MIGRACIÓN DE MÁQUINAS VIRTUALES	82
9.2. VENTAJAS DE LA MIGRACIÓN DE MÁQUINAS VIRTUALES	83
9.3. LIMITACIONES PARA LA MIGRACIÓN DE MÁQUINAS VIRTUALES	83
9.4. COMPARTIR IMÁGENES DE DISCO DE MÁQUINAS VIRTUALES CON OTROS HOSTS	84
9.5. MIGRACIÓN DE UNA MÁQUINA VIRTUAL MEDIANTE LA INTERFAZ DE LÍNEA DE COMANDOS	85
9.6. HOSTS COMPATIBLES CON LA MIGRACIÓN DE MÁQUINAS VIRTUALES	87
9.7. RECURSOS ADICIONALES	88
CAPÍTULO 10. GESTIÓN DE DISPOSITIVOS VIRTUALES	89
10.1. CÓMO FUNCIONAN LOS DISPOSITIVOS VIRTUALES	89
10.2. ADJUNTAR DISPOSITIVOS A LAS MÁQUINAS VIRTUALES	90
10.3. MODIFICACIÓN DE LOS DISPOSITIVOS CONECTADOS A LAS MÁQUINAS VIRTUALES	91
10.4. ELIMINACIÓN DE DISPOSITIVOS DE LAS MÁQUINAS VIRTUALES	93
10.5. TIPOS DE DISPOSITIVOS VIRTUALES	94
10.6. GESTIÓN DE DISPOSITIVOS USB VIRTUALES	96
10.6.1. Adjuntar dispositivos USB a las máquinas virtuales	96
10.6.2. Eliminación de los dispositivos USB de las máquinas virtuales	97
10.6.3. Recursos adicionales	98
10.7. GESTIÓN DE UNIDADES ÓPTICAS VIRTUALES	98
10.7.1. Adjuntar unidades ópticas a las máquinas virtuales	98
10.7.2. Sustitución de imágenes ISO en unidades ópticas virtuales	99
10.7.3. Eliminación de imágenes ISO de unidades ópticas virtuales	100
10.7.4. Eliminación de las unidades ópticas de las máquinas virtuales	100

10.7.5. Recursos adicionales	101
10.8. GESTIÓN DE DISPOSITIVOS SR-IOV	101
10.8.1. ¿Qué es el SR-IOV?	101
10.8.2. Adjuntar dispositivos de red SR-IOV a máquinas virtuales	103
10.8.3. Dispositivos compatibles con la asignación de SR-IOV	107
10.9. ADJUNTAR DISPOSITIVOS DASD A MÁQUINAS VIRTUALES EN IBM Z	107
CAPÍTULO 11. GESTIÓN DEL ALMACENAMIENTO DE LAS MÁQUINAS VIRTUALES	110
11.1. COMPRENDER EL ALMACENAMIENTO DE LAS MÁQUINAS VIRTUALES	110
11.1.1. Almacenamiento de máquinas virtuales	110
11.1.2. Piscinas de almacenamiento	110
11.1.3. Volúmenes de almacenamiento	112
11.2. GESTIÓN DEL ALMACENAMIENTO DE LAS MÁQUINAS VIRTUALES MEDIANTE LA CLI	113
11.2.1. Visualización de la información de almacenamiento de las máquinas virtuales mediante la CLI	113
11.2.1.1. Visualización de la información del pool de almacenamiento mediante la CLI	113
11.2.1.2. Visualización de la información del volumen de almacenamiento mediante la CLI	113
11.2.2. Creación y asignación de almacenamiento para máquinas virtuales mediante la CLI	114
11.2.2.1. Creación y asignación de almacenamiento basado en directorios para máquinas virtuales mediante la CLI	115
11.2.2.1.1. Creación de grupos de almacenamiento basados en directorios mediante la CLI	115
11.2.2.1.2. Parámetros del pool de almacenamiento basado en directorios	117
11.2.2.2. Creación y asignación de almacenamiento en disco para máquinas virtuales mediante la CLI	118
11.2.2.2.1. Creación de grupos de almacenamiento basados en disco mediante la CLI	118
11.2.2.2.2. Parámetros del pool de almacenamiento en disco	120
11.2.2.3. Creación y asignación de almacenamiento basado en sistemas de archivos para máquinas virtuales mediante la CLI	121
11.2.2.3.1. Creación de grupos de almacenamiento basados en sistemas de archivos mediante la CLI	121
11.2.2.3.2. Parámetros del pool de almacenamiento basado en el sistema de archivos	123
11.2.2.4. Creación y asignación de almacenamiento GlusterFS para máquinas virtuales utilizando la CLI	124
11.2.2.4.1. Creación de pools de almacenamiento basados en GlusterFS mediante la CLI	124
11.2.2.4.2. Parámetros del pool de almacenamiento basado en GlusterFS	126
11.2.2.5. Creación y asignación de almacenamiento basado en iSCSI para máquinas virtuales mediante la CLI	127
Recomendaciones	127
Requisitos previos	128
11.2.2.5.1. Creación de pools de almacenamiento basados en iSCSI mediante la CLI	128
11.2.2.5.2. parámetros del pool de almacenamiento basado en iSCSI	129
11.2.2.5.3. Asegurar los grupos de almacenamiento iSCSI con secretos de libvirt	131
11.2.2.6. Creación y asignación de almacenamiento basado en LVM para máquinas virtuales utilizando la CLI	132
11.2.2.6.1. Creación de pools de almacenamiento basados en LVM mediante la CLI	132
11.2.2.6.2. Parámetros del pool de almacenamiento basado en LVM	134
11.2.2.7. Creación y asignación de almacenamiento en red para máquinas virtuales mediante la CLI	136
11.2.2.7.1. Creación de pools de almacenamiento basados en NFS mediante la CLI	136
11.2.2.7.2. Parámetros del pool de almacenamiento basado en NFS	137
11.2.2.8. Creación y asignación de almacenamiento basado en SCSI con dispositivos vHBA para máquinas virtuales utilizando la CLI	139
11.2.2.8.1. Creación de vHBAs	140
11.2.2.8.2. Creación de pools de almacenamiento basados en SCSI con dispositivos vHBA mediante la CLI	142
11.2.2.8.3. Parámetros para pools de almacenamiento basados en SCSI con dispositivos vHBA	144
11.2.2.9. Creación y asignación de volúmenes de almacenamiento mediante la CLI	146
11.2.3. Eliminación del almacenamiento de las máquinas virtuales mediante la CLI	147
11.2.3.1. Eliminación de grupos de almacenamiento mediante la CLI	147

11.2.3.2. Eliminación de volúmenes de almacenamiento mediante la CLI	148
11.3. GESTIÓN DEL ALMACENAMIENTO DE LAS MÁQUINAS VIRTUALES MEDIANTE LA CONSOLA WEB	149
11.3.1. Visualización de la información del pool de almacenamiento mediante la consola web	150
11.3.2. Creación de grupos de almacenamiento mediante la consola web	151
11.3.3. Eliminación de grupos de almacenamiento mediante la consola web	153
11.3.4. Desactivación de grupos de almacenamiento mediante la consola web	155
11.3.5. Creación de volúmenes de almacenamiento mediante la consola web	156
11.3.6. Eliminación de volúmenes de almacenamiento mediante la consola web	158
11.3.7. Gestión de los discos de las máquinas virtuales mediante la consola web	160
11.3.7.1. Visualización de la información del disco de la máquina virtual en la consola web	160
11.3.7.2. Añadir nuevos discos a las máquinas virtuales mediante la consola web	161
11.3.7.3. Adjuntar discos existentes a máquinas virtuales mediante la consola web	163
11.3.7.4. Separar los discos de las máquinas virtuales	166
CAPÍTULO 12. GESTIÓN DE DISPOSITIVOS NVIDIA VGPU	167
12.1. CONFIGURACIÓN DE DISPOSITIVOS NVIDIA VGPU	167
12.2. ELIMINACIÓN DE DISPOSITIVOS NVIDIA VGPU	169
12.3. OBTENCIÓN DE INFORMACIÓN DE NVIDIA VGPU SOBRE SU SISTEMA	170
12.4. SERVICIOS DE STREAMING DE ESCRITORIO REMOTO PARA NVIDIA VGPU	172
12.5. INFORMACIÓN RELACIONADA	172
CAPÍTULO 13. CONFIGURACIÓN DE LAS CONEXIONES DE RED DE LAS MÁQUINAS VIRTUALES	173
13.1. ENTENDER LA RED VIRTUAL	173
13.1.1. Cómo funcionan las redes virtuales	173
13.1.2. Configuración por defecto de la red virtual	174
13.2. USO DE LA CONSOLA WEB PARA GESTIONAR LAS INTERFACES DE RED DE LAS MÁQUINAS VIRTUALES	175
13.2.1. Ver y editar la información de la interfaz de red virtual en la consola web	175
13.2.2. Conexión de interfaces de red virtuales en la consola web	177
13.2.3. Desconexión de interfaces de red virtuales en la consola web	178
13.3. CONFIGURACIONES DE RED DE MÁQUINAS VIRTUALES RECOMENDADAS MEDIANTE LA INTERFAZ DE LÍNEA DE COMANDOS	178
13.3.1. Configuración de máquinas virtuales visibles externamente mediante la interfaz de línea de comandos	179
13.3.2. Aislar las máquinas virtuales entre sí mediante la interfaz de línea de comandos	181
13.4. CONFIGURACIONES DE RED DE MÁQUINAS VIRTUALES RECOMENDADAS MEDIANTE LA CONSOLA WEB	182
13.4.1. Configuración de máquinas virtuales visibles externamente mediante la consola web	182
13.4.2. Aislar las máquinas virtuales entre sí mediante la consola web	184
13.5. TIPOS DE CONEXIONES DE RED DE MÁQUINAS VIRTUALES	186
13.5.1. Red virtual con traducción de direcciones de red	186
13.5.2. Red virtual en modo rotado	187
13.5.3. Red virtual en modo puente	188
13.5.4. Red virtual en modo aislado	190
13.5.5. Red virtual en modo abierto	190
13.5.6. Fijación directa del dispositivo de red virtual	190
13.5.7. Comparación de los tipos de conexión de las máquinas virtuales	191
13.6. RECURSOS ADICIONALES	191
CAPÍTULO 14. COMPARTIR ARCHIVOS ENTRE EL HOST Y SUS MÁQUINAS VIRTUALES	193
14.1. COMPARTIR ARCHIVOS ENTRE EL HOST Y LAS MÁQUINAS VIRTUALES LINUX	193
14.2. COMPARTIR ARCHIVOS ENTRE EL HOST Y LAS MÁQUINAS VIRTUALES DE WINDOWS	195
CAPÍTULO 15. ASEGURAR LAS MÁQUINAS VIRTUALES	200

15.1. CÓMO FUNCIONA LA SEGURIDAD EN LAS MÁQUINAS VIRTUALES	200
15.2. MEJORES PRÁCTICAS PARA ASEGURAR LAS MÁQUINAS VIRTUALES	201
15.3. CREACIÓN DE UNA MÁQUINA VIRTUAL SECUREBOOT	202
15.4. FUNCIONES AUTOMÁTICAS PARA LA SEGURIDAD DE LAS MÁQUINAS VIRTUALES	203
15.5. BOOLEANOS DE VIRTUALIZACIÓN	204
15.6. CONFIGURACIÓN DE IBM SECURE EXECUTION EN IBM Z	205
15.7. ADJUNTAR COPROCESADORES CRIPTOGRÁFICOS A MÁQUINAS VIRTUALES EN IBM Z	208
15.8. HABILITACIÓN DE LA SEGURIDAD DE HARDWARE ESTÁNDAR EN LAS MÁQUINAS VIRTUALES DE WINDOWS	211
CAPÍTULO 16. OPTIMIZACIÓN DEL RENDIMIENTO DE LAS MÁQUINAS VIRTUALES	213
16.1. QUÉ INFLUYE EN EL RENDIMIENTO DE LAS MÁQUINAS VIRTUALES	213
El impacto de la virtualización en el rendimiento del sistema	213
Reducción de la pérdida de rendimiento de las máquinas virtuales	213
16.2. OPTIMIZACIÓN DEL RENDIMIENTO DE LAS MÁQUINAS VIRTUALES MEDIANTE EL AJUSTE	214
16.3. CONFIGURACIÓN DE LA MEMORIA DE LA MÁQUINA VIRTUAL	215
16.3.1. Añadir y eliminar la memoria de la máquina virtual mediante la consola web	215
16.3.2. Añadir y eliminar la memoria de la máquina virtual mediante la interfaz de línea de comandos	217
16.3.3. Recursos adicionales	218
16.4. OPTIMIZACIÓN DEL RENDIMIENTO DE E/S DE LAS MÁQUINAS VIRTUALES	218
16.4.1. Ajuste de la E/S en bloque en las máquinas virtuales	219
16.4.2. Estrangulamiento de E/S de disco en máquinas virtuales	219
16.4.3. Activación de la cola múltiple virtio-scsi	221
16.5. OPTIMIZACIÓN DEL RENDIMIENTO DE LA CPU DE LA MÁQUINA VIRTUAL	221
16.5.1. Añadir y eliminar CPUs virtuales mediante la interfaz de línea de comandos	221
16.5.2. Gestión de las CPUs virtuales mediante la consola web	223
16.5.3. Configuración de NUMA en una máquina virtual	224
16.5.4. Ejemplo de escenario de ajuste de rendimiento de vCPU	226
16.5.5. Desactivación de la fusión de páginas en el núcleo	231
16.6. OPTIMIZACIÓN DEL RENDIMIENTO DE LA RED DE MÁQUINAS VIRTUALES	232
16.7. HERRAMIENTAS DE SUPERVISIÓN DEL RENDIMIENTO DE LAS MÁQUINAS VIRTUALES	233
16.8. INFORMACIÓN RELACIONADA	235
CAPÍTULO 17. INSTALACIÓN Y GESTIÓN DE MÁQUINAS VIRTUALES WINDOWS	237
17.1. INSTALACIÓN DE MÁQUINAS VIRTUALES WINDOWS	237
17.2. OPTIMIZACIÓN DE LAS MÁQUINAS VIRTUALES DE WINDOWS	238
17.2.1. Instalación de controladores paravirtuales KVM para máquinas virtuales Windows	238
17.2.1.1. Cómo funcionan los controladores de Windows virtio	238
17.2.1.2. Preparación de los medios de instalación del controlador virtio en una máquina anfitriona	239
17.2.1.3. Instalación de los controladores virtio en un invitado de Windows	240
17.2.2. Habilitación de las iluminaciones de Hyper-V	243
17.2.2.1. Habilitación de las iluminaciones de Hyper-V en una máquina virtual de Windows	243
17.2.2.2. Luces configurables de Hyper-V	244
17.2.3. Configuración de los parámetros del controlador NetKVM	247
17.2.4. Parámetros del controlador NetKVM	248
17.2.5. Optimización de los procesos de fondo en las máquinas virtuales de Windows	250
17.3. INFORMACIÓN RELACIONADA	251
CAPÍTULO 18. CREACIÓN DE MÁQUINAS VIRTUALES ANIDADAS	252
18.1. CREACIÓN DE UNA MÁQUINA VIRTUAL ANIDADADA EN INTEL	252
18.2. CREACIÓN DE UNA MÁQUINA VIRTUAL ANIDADADA EN AMD	253
18.3. CREACIÓN DE UNA MÁQUINA VIRTUAL ANIDADADA EN IBM Z	254
18.4. CREACIÓN DE UNA MÁQUINA VIRTUAL ANIDADADA EN IBM POWER9	255
18.5. RESTRICCIONES Y LIMITACIONES DE LA VIRTUALIZACIÓN ANIDADADA	257

CAPÍTULO 19. DIAGNÓSTICO DE LOS PROBLEMAS DE LAS MÁQUINAS VIRTUALES	259
19.1. GENERACIÓN DE REGISTROS DE DEPURACIÓN DE MÁQUINAS VIRTUALES	259
19.1.1. Entender los registros de depuración de la máquina virtual	259
19.1.2. Activación de la configuración persistente para los registros de depuración de la máquina virtual	259
19.1.3. Activación de los registros de depuración de la máquina virtual durante el tiempo de ejecución	260
19.1.4. Adjuntar registros de depuración de máquinas virtuales a las solicitudes de asistencia	261
19.2. DUMPING DE UN NÚCLEO DE MÁQUINA VIRTUAL	262
19.2.1. Cómo funciona el volcado del núcleo de la máquina virtual	262
19.2.2. Creación de un archivo de volcado del núcleo de la máquina virtual	262
19.3. RASTREO DE PROCESOS DE MÁQUINAS VIRTUALES	263
19.4. RECURSOS ADICIONALES PARA INFORMAR DE LOS PROBLEMAS DE LAS MÁQUINAS VIRTUALES Y PROPORCIONAR REGISTROS	264
CAPÍTULO 20. COMPATIBILIDAD Y LIMITACIONES DE LAS FUNCIONES EN LA VIRTUALIZACIÓN DE RHEL 8	265
20.1. CÓMO FUNCIONA EL SOPORTE DE VIRTUALIZACIÓN DE RHEL 8	265
20.2. CARACTERÍSTICAS RECOMENDADAS EN LA VIRTUALIZACIÓN DE RHEL 8	265
20.3. CARACTERÍSTICAS NO SOPORTADAS EN LA VIRTUALIZACIÓN DE RHEL 8	266
20.4. LÍMITES DE ASIGNACIÓN DE RECURSOS EN LA VIRTUALIZACIÓN DE RHEL 8	270
20.5. UNA VISIÓN GENERAL DE LA COMPATIBILIDAD CON LAS FUNCIONES DE VIRTUALIZACIÓN	270

HACER QUE EL CÓDIGO ABIERTO SEA MÁS INCLUSIVO

Red Hat se compromete a sustituir el lenguaje problemático en nuestro código, documentación y propiedades web. Estamos empezando con estos cuatro términos: maestro, esclavo, lista negra y lista blanca. Debido a la enormidad de este esfuerzo, estos cambios se implementarán gradualmente a lo largo de varias versiones próximas. Para más detalles, consulte [el mensaje de nuestro CTO Chris Wright](#) .

PROPORCIONAR COMENTARIOS SOBRE LA DOCUMENTACIÓN DE RED HAT

Agradecemos su opinión sobre nuestra documentación. Por favor, díganos cómo podemos mejorarla. Para ello:

- Para comentarios sencillos sobre pasajes concretos:
 1. Asegúrese de que está viendo la documentación en el formato *Multi-page HTML*. Además, asegúrese de ver el botón **Feedback** en la esquina superior derecha del documento.
 2. Utilice el cursor del ratón para resaltar la parte del texto que desea comentar.
 3. Haga clic en la ventana emergente **Add Feedback** que aparece debajo del texto resaltado.
 4. Siga las instrucciones mostradas.
- Para enviar comentarios más complejos, cree un ticket de Bugzilla:
 1. Vaya al sitio web [de Bugzilla](#).
 2. Como componente, utilice **Documentation**.
 3. Rellene el campo **Description** con su sugerencia de mejora. Incluya un enlace a la(s) parte(s) pertinente(s) de la documentación.
 4. Haga clic en **Submit Bug**.

CAPÍTULO 1. VIRTUALIZACIÓN EN RHEL 8 - UNA VISIÓN GENERAL

Si no está familiarizado con el concepto de virtualización o su implementación en Linux, las siguientes secciones proporcionan una visión general de la virtualización en RHEL 8: sus fundamentos, ventajas, componentes y otras posibles soluciones de virtualización proporcionadas por Red Hat.

1.1. ¿QUÉ ES LA VIRTUALIZACIÓN?

RHEL 8 proporciona la funcionalidad *virtualization*, que permite a una máquina que ejecuta RHEL 8 *host* múltiples máquinas virtuales (VMs), también conocidas como *guests*. Las VMs utilizan el hardware físico y los recursos informáticos del host para ejecutar un sistema operativo virtualizado independiente (*guest OS*) como un proceso de espacio de usuario en el sistema operativo del host.

En otras palabras, la virtualización permite tener sistemas operativos dentro de los sistemas operativos.

Las máquinas virtuales le permiten probar con seguridad configuraciones y funciones de software, ejecutar software heredado u optimizar la eficiencia de la carga de trabajo de su hardware. Para obtener más información sobre las ventajas, consulte [Sección 1.2, "Ventajas de la virtualización"](#).

Para más información sobre lo que es la virtualización, consulte [el Portal del Cliente de Red Hat](#).

Para probar la virtualización en RHEL 8, consulte [Capítulo 2, Introducción a la virtualización](#).



NOTA

Además de la virtualización de RHEL 8, Red Hat ofrece una serie de soluciones de virtualización especializadas, cada una con un enfoque y características diferentes para el usuario. Para más información, consulte [Sección 1.5, "Soluciones de virtualización de Red Hat"](#).

1.2. VENTAJAS DE LA VIRTUALIZACIÓN

El uso de máquinas virtuales (VM) tiene las siguientes ventajas en comparación con el uso de máquinas físicas:

- **Flexible and fine-grained allocation of resources**

Una VM se ejecuta en una máquina anfitriona, que suele ser física, y también se puede asignar hardware físico para que lo utilice el SO invitado. Sin embargo, la asignación de recursos físicos a la VM se realiza a nivel de software, por lo que es muy flexible. Una VM utiliza una fracción configurable de la memoria, las CPU o el espacio de almacenamiento del host, y esa configuración puede especificar solicitudes de recursos muy precisas.

Por ejemplo, lo que el SO huésped ve como su disco puede ser representado como un archivo en el sistema de archivos del host, y el tamaño de ese disco está menos restringido que los tamaños disponibles para los discos físicos.

- **Software-controlled configurations**

Toda la configuración de una VM se guarda como datos en el host, y está bajo el control del software. Por lo tanto, una VM puede ser fácilmente creada, eliminada, clonada, migrada, operada remotamente o conectada a un almacenamiento remoto.

- **Separation from the host**

Un sistema operativo invitado se ejecuta en un núcleo virtualizado, separado del sistema

operativo anfitrión. Esto significa que se puede instalar cualquier sistema operativo en una máquina virtual, e incluso si el sistema operativo invitado se vuelve inestable o se ve comprometido, el host no se ve afectado de ninguna manera.

- **Space and cost efficiency**

Una sola máquina física puede albergar un gran número de máquinas virtuales. Por lo tanto, evita la necesidad de contar con varias máquinas físicas para realizar las mismas tareas y, por lo tanto, reduce los requisitos de espacio, energía y mantenimiento asociados al hardware físico.

- **Software compatibility**

Dado que una máquina virtual puede utilizar un sistema operativo diferente al de su anfitrión, la virtualización permite ejecutar aplicaciones que no fueron publicadas originalmente para su sistema operativo anfitrión. Por ejemplo, utilizando un sistema operativo invitado RHEL 6, puede ejecutar aplicaciones publicadas para RHEL 6 en un sistema anfitrión RHEL 8.



NOTA

No todos los sistemas operativos están soportados como SO invitado en un host RHEL 8. Para más detalles, consulte [Sección 20.2, "Características recomendadas en la virtualización de RHEL 8"](#).

1.3. COMPONENTES DE LA MÁQUINA VIRTUAL Y SU INTERACCIÓN

La virtualización en RHEL 8 consta de los siguientes componentes principales de software:

Hipervisor

La base de la creación de máquinas virtuales (VMs) en RHEL 8 es el *hypervisor*, una capa de software que controla el hardware y permite la ejecución de múltiples sistemas operativos en una máquina anfitriona.

El hipervisor incluye el módulo **Kernel-based Virtual Machine (KVM)** y los controladores del kernel de virtualización, como **virtio** y **vfiio**. Estos componentes garantizan que el kernel Linux de la máquina anfitriona proporcione recursos para la virtualización al software del espacio de usuario.

A nivel de espacio de usuario, el emulador **QEMU** simula una plataforma de hardware virtualizada completa en la que puede ejecutarse el sistema operativo invitado, y gestiona cómo se asignan los recursos en el host y se presentan al invitado.

Además, el paquete de software **libvirt** sirve como capa de gestión y comunicación, facilitando la interacción con QEMU, aplicando reglas de seguridad y proporcionando una serie de herramientas adicionales para configurar y ejecutar las máquinas virtuales.

Configuración XML

Un archivo de configuración XML basado en el host (también conocido como archivo *domain XML*) determina todos los ajustes y dispositivos de una VM específica. La configuración incluye:

- Metadatos como el nombre de la VM, la zona horaria y otra información sobre la VM.
- Una descripción de los dispositivos en la VM, incluyendo CPUs virtuales (vCPUS), dispositivos de almacenamiento, dispositivos de entrada/salida, tarjetas de interfaz de red y otro hardware, real y virtual.
- Ajustes de la VM como la cantidad máxima de memoria que puede utilizar, ajustes de reinicio y otros ajustes sobre el comportamiento de la VM.

Para más información sobre el contenido de una configuración XML, consulte el [ejemplo de configuración XML de VM](#).

Interacción de los componentes

Cuando se inicia una VM, el hipervisor utiliza la configuración XML para crear una instancia de la VM como un proceso de espacio de usuario en el host. El hipervisor también hace que el proceso de la VM sea accesible a las interfaces basadas en el host, como las utilidades **virsh**, **virt-install**, y **guestfish**, o la GUI de la consola web.

Cuando se utilizan estas herramientas de virtualización, libvirt traduce su entrada en instrucciones para QEMU. QEMU comunica las instrucciones a KVM, que se asegura de que el kernel asigne adecuadamente los recursos necesarios para llevar a cabo las instrucciones. Como resultado, QEMU puede ejecutar los cambios correspondientes en el espacio de usuario, como crear o modificar una VM, o realizar una acción en el sistema operativo invitado de la VM.

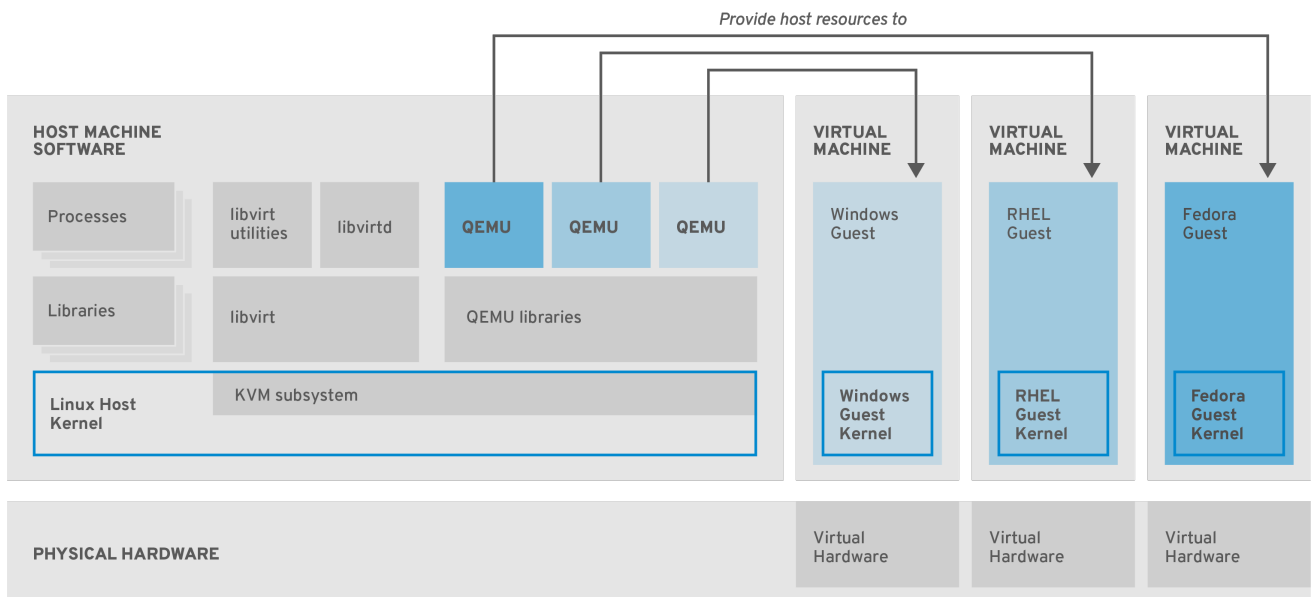


NOTA

Aunque QEMU es un componente esencial de la arquitectura, no está pensado para ser utilizado directamente en los sistemas RHEL 8, debido a cuestiones de seguridad. Por lo tanto, el uso de los comandos de **qemu-*** no está soportado por Red Hat, y se recomienda encarecidamente interactuar con QEMU utilizando libvirt.

Para más información sobre las interfaces basadas en el host, consulte [Sección 1.4, "Herramientas e interfaces para la gestión de la virtualización"](#).

Figura 1.1. Arquitectura de virtualización de RHEL 8



RHEL_7_0319

1.4. HERRAMIENTAS E INTERFACES PARA LA GESTIÓN DE LA VIRTUALIZACIÓN

Puede gestionar la virtualización en RHEL 8 mediante la interfaz de línea de comandos (CLI) o varias interfaces gráficas de usuario (GUI).

Interfaz de línea de comandos

La CLI es el método más potente para gestionar la virtualización en RHEL 8. Los principales comandos de la CLI para la gestión de máquinas virtuales (VM) son:

- **virsh** - Una versátil utilidad de línea de comandos de virtualización y shell con una gran variedad de propósitos, dependiendo de los argumentos proporcionados. Por ejemplo:
 - Iniciar y apagar una VM - **virsh start** y **virsh shutdown**
 - Listado de máquinas virtuales disponibles - **virsh list**
 - Creación de una VM a partir de un archivo de configuración - **virsh create**
 - Entrar en un shell de virtualización - **virsh**

Para más información, consulte la página de manual **virsh(1)**.

- **virt-install** - Una utilidad CLI para crear nuevas VMs. Para más información, consulte la página de manual **virt-install(1)**.
- **virt-xml** - Una utilidad para editar la configuración de una VM.
- **guestfish** - Una utilidad para examinar y modificar imágenes de disco de VM. Para más información, consulte la página de manual **guestfish(1)**.

Interfaces gráficas

Puede utilizar las siguientes interfaces gráficas para gestionar la virtualización en RHEL 8:

- El **RHEL 8 web console**, también conocido como *Cockpit*, proporciona una interfaz gráfica de usuario accesible de forma remota y fácil de usar para gestionar las máquinas virtuales y los hosts de virtualización. Para obtener instrucciones sobre la gestión básica de la virtualización con la consola web, consulte [Capítulo 5, Gestión de máquinas virtuales en la consola web](#).
- La aplicación Virtual Machine Manager (**virt-manager**) proporciona una interfaz gráfica de usuario especializada para gestionar las máquinas virtuales y los hosts de virtualización.



IMPORTANTE

Aunque sigue siendo compatible con RHEL 8, **virt-manager** ha quedado obsoleto. La consola web está destinada a convertirse en su reemplazo en una versión posterior. Por lo tanto, se recomienda que se familiarice con la consola web para gestionar la virtualización en una GUI.

Sin embargo, en RHEL 8, es posible que algunas funciones sólo sean accesibles desde **virt-manager** o desde la línea de comandos. Para más detalles, consulte [Sección 5.4, "Diferencias entre las funciones de virtualización en Virtual Machine Manager y la consola web"](#).

- La aplicación **Gnome Boxes** es una interfaz gráfica ligera para ver y acceder a máquinas virtuales y sistemas remotos. Gnome Boxes está diseñado principalmente para su uso en sistemas de escritorio.



IMPORTANTE

Gnome Boxes se proporciona como parte del entorno de escritorio GNOME y es compatible con RHEL 8, pero Red Hat recomienda que se utilice la consola web para gestionar la virtualización en una GUI.

Recursos adicionales

- Para obtener instrucciones sobre la gestión básica de la virtualización mediante la CLI y la GUI, consulte [Capítulo 2, Introducción a la virtualización](#).

1.5. SOLUCIONES DE VIRTUALIZACIÓN DE RED HAT

Los siguientes productos de Red Hat están contruidos sobre las características de virtualización de RHEL 8 y amplían las capacidades de virtualización de KVM disponibles en RHEL 8. Además, muchas [limitaciones de la virtualización de RHEL 8](#) no se aplican a estos productos:

Red Hat Virtualization (RHV)

RHV está diseñado para ofrecer escalabilidad y rendimiento de clase empresarial, y permite la gestión de toda la infraestructura virtual, incluidos los hosts, las máquinas virtuales, las redes, el almacenamiento y los usuarios desde una interfaz gráfica centralizada.

Red Hat Virtualization puede ser utilizado por las empresas que ejecutan grandes implementaciones o aplicaciones de misión crítica. Entre los ejemplos de grandes implantaciones adecuadas para Red Hat Virtualization se encuentran las bases de datos, las plataformas comerciales y los sistemas de mensajería que deben funcionar de forma continua sin ningún tiempo de inactividad.

Para más información sobre Red Hat Virtualization, consulte [el Portal del Cliente de Red Hat](#) o el [conjunto de documentación de Red Hat Virtualization](#).

Para descargar una versión de evaluación de 60 días totalmente compatible con Red Hat Virtualization, consulte <https://access.redhat.com/products/red-hat-virtualization/evaluation>

Red Hat OpenStack Platform (RHOSP)

Red Hat OpenStack Platform ofrece una base integrada para crear, implementar y escalar una nube [OpenStack](#) pública o privada segura y fiable.

Para más información sobre Red Hat OpenStack Platform, consulte [el Portal del Cliente de Red Hat](#) o el [conjunto de documentación de Red Hat OpenStack Platform](#).



NOTA

Para más detalles sobre las características de virtualización no soportadas en RHEL pero sí en RHV o RHOSP, consulte [Sección 20.3, "Características no soportadas en la virtualización de RHEL 8"](#).

Además, algunos productos de Red Hat proporcionan *operating-system-level virtualization*, también conocido como **containerization**:

- Los contenedores son instancias aisladas del sistema operativo anfitrión y operan sobre un núcleo de sistema operativo existente. Para más información sobre los contenedores, consulte el [Portal del Cliente de Red Hat](#).

- Los contenedores no tienen la versatilidad de la virtualización KVM, pero son más ligeros y flexibles de manejar. Para una comparación más detallada, consulte la [Introducción a los contenedores de Linux](#).

CAPÍTULO 2. INTRODUCCIÓN A LA VIRTUALIZACIÓN

Para empezar a utilizar [la virtualización en RHEL 8](#), siga los siguientes pasos. El método por defecto para esto es utilizar la interfaz de línea de comandos (CLI), pero para la comodidad del usuario, algunos de los pasos se pueden completar en la [GUI de la consola web](#).

1. Habilite el módulo de virtualización e instale los paquetes de virtualización – véase [Sección 2.1, “Habilitación de la virtualización”](#).
2. Crear una máquina virtual (VM):
 - Para la CLI, véase [Sección 2.2.1, “Creación de máquinas virtuales mediante la interfaz de línea de comandos”](#).
 - Para la interfaz gráfica de usuario, véase [Sección 2.2.2, “Creación de máquinas virtuales e instalación de sistemas operativos invitados mediante la consola web”](#).
3. Inicie la máquina virtual:
 - Para la CLI, véase [Sección 2.3.1, “Iniciar una máquina virtual mediante la interfaz de línea de comandos”](#).
 - Para la interfaz gráfica de usuario, véase [Sección 2.3.2, “Iniciar máquinas virtuales mediante la consola web”](#).
4. Conéctate a la VM:
 - Para la CLI, véase [Sección 2.4.3, “Conexión a una máquina virtual mediante SSH”](#) o [Sección 2.4.2, “Abrir la consola gráfica de una máquina virtual con Virt Viewer”](#).
 - Para la interfaz gráfica de usuario, véase [Sección 2.4.1, “Interacción con las máquinas virtuales mediante la consola web”](#).



NOTA

Actualmente, la consola web sólo proporciona un subconjunto de funciones de gestión de máquinas virtuales, por lo que se recomienda utilizar la línea de comandos para el uso avanzado de la virtualización en RHEL 8.

2.1. HABILITACIÓN DE LA VIRTUALIZACIÓN

Para utilizar la virtualización en RHEL 8, debe activar el módulo de virtualización, instalar los paquetes de virtualización y asegurarse de que su sistema está configurado para alojar máquinas virtuales (VM).

Requisitos previos

- Red Hat Enterprise Linux 8 está [instalado y registrado](#) en su máquina anfitriona.
- Su sistema cumple los siguientes requisitos de hardware para funcionar como host de virtualización:
 - La arquitectura de su máquina anfitriona [soporta la virtualización KVM](#).
 - Los siguientes recursos mínimos del sistema están disponibles:

- 6 GB de espacio libre en disco para el host, más otros 6 GB para cada máquina virtual prevista.
- 2 GB de RAM para el host, más otros 2 GB para cada máquina virtual prevista.

Procedimiento

1. Instale los paquetes del módulo de virtualización de RHEL 8:

```
# yum module install virt
```

2. Instale los paquetes **virt-install** y **virt-viewer**:

```
# yum install virt-install virt-viewer
```

3. Inicie el servicio **libvirtd**.

```
# systemctl start libvirtd
```

4. Compruebe que su sistema está preparado para ser un host de virtualización:

```
# virt-host-validate
[...]
QEMU: Checking for device assignment IOMMU support      : PASS
QEMU: Checking if IOMMU is enabled by kernel            : WARN (IOMMU appears to be
disabled in kernel. Add intel_iommu=on to kernel cmdline arguments)
LXC: Checking for Linux >= 2.6.26                      : PASS
[...]
LXC: Checking for cgroup 'blkio' controller mount-point : PASS
LXC: Checking if device /sys/fs/fuse/connections exists : FAIL (Load the 'fuse' module to
enable /proc/ overrides)
```

5. Si todas las comprobaciones de **virt-host-validate** devuelven un valor de **PASS**, su sistema está preparado para [crear máquinas virtuales](#).

Si alguna de las comprobaciones devuelve un valor de **FAIL**, siga las instrucciones mostradas para solucionar el problema.

Si alguna de las comprobaciones devuelve un valor de **WARN**, considere seguir las instrucciones mostradas para mejorar las capacidades de virtualización.

Información adicional

- Tenga en cuenta que si la virtualización no está soportada por su CPU anfitriona, **virt-host-validate** genera la siguiente salida:

```
QEMU: Comprobando la virtualización del hardware: FAIL (Sólo hay CPUs emuladas, el
rendimiento será significativamente limitado)
```

Sin embargo, al intentar crear máquinas virtuales en un sistema anfitrión de este tipo, se producirá un fallo, en lugar de tener problemas de rendimiento.

2.2. CREACIÓN DE MÁQUINAS VIRTUALES

Para crear una máquina virtual (VM) en RHEL 8, utilice la [interfaz de línea de comandos](#) o la [consola web de RHEL 8](#).

Requisitos previos

- La virtualización está [instalada y habilitada](#) en su sistema.
- Tiene una cantidad suficiente de recursos del sistema para asignar a sus VMs, como espacio en disco, RAM o CPUs. Los valores recomendados pueden variar significativamente en función de las tareas previstas y la carga de trabajo de las máquinas virtuales.



AVISO

La instalación desde un dispositivo de CD-ROM o DVD-ROM del host no es posible en RHEL 8. Si selecciona un CD-ROM o DVD-ROM como fuente de instalación cuando utiliza cualquier método de instalación de VM disponible en RHEL 8, la instalación fallará. Para más información, consulte la [Base de conocimientos de Red Hat](#).

2.2.1. Creación de máquinas virtuales mediante la interfaz de línea de comandos

Para crear una máquina virtual (VM) en su host RHEL 8 utilizando la utilidad **virt-install**, siga las siguientes instrucciones.

Requisitos previos

- La virtualización está [activada](#) en su sistema anfitrión.
- Una fuente de instalación del sistema operativo (SO) está disponible localmente o en una red. Puede ser uno de los siguientes:
 - Una imagen ISO de un medio de instalación
 - Una imagen de disco de una instalación VM existente
- Opcional: Se puede proporcionar un archivo Kickstart para una configuración más rápida y sencilla de la instalación.

Procedimiento

Para crear una VM e iniciar la instalación de su sistema operativo, utilice el comando **virt-install**, junto con los siguientes argumentos obligatorios:

- El nombre de la nueva máquina
- La cantidad de memoria asignada
- El número de CPUs virtuales asignadas (vCPUs)
- El tipo y el tamaño del almacenamiento asignado
- El tipo y la ubicación de la fuente de instalación del sistema operativo

Según el método de instalación elegido, las opciones y valores necesarios pueden variar. Consulte los ejemplos siguientes:

- Lo siguiente crea una VM llamada **demo-guest1** que instala el SO Windows 10 desde una imagen ISO almacenada localmente en el archivo `/home/username/Downloads/Win10install.iso`. A esta VM también se le asignan 2048 MiB de RAM y 2 vCPUs, y se configura automáticamente un disco virtual qcow2 de 80 GiB para la VM.

```
# virt-install --name demo-guest1 --memory 2048 --vcpus 2 --disk size=80 --os-variant win10 --cdrom /home/username/Downloads/Win10install.iso
```

- Lo siguiente crea una VM llamada **demo-guest2** que utiliza la imagen `/home/username/Downloads/rhel8.iso` para ejecutar un SO RHEL 8 desde un CD en vivo. No se asigna espacio en disco a esta VM, por lo que los cambios realizados durante la sesión no se conservarán. Además, la VM se asigna con 4096 MiB de RAM y 4 vCPUs.

```
# virt-install --name demo-guest2 --memory 4096 --vcpus 4 --disk none --livecd --os-variant rhel8.0 --cdrom /home/username/Downloads/rhel8.iso
```

- Lo siguiente crea una VM RHEL 8 llamada **demo-guest3** que se conecta a una imagen de disco existente, `/home/username/backup/disk.qcow2`. Esto es similar a mover físicamente un disco duro entre máquinas, por lo que el sistema operativo y los datos disponibles para `demo-guest3` están determinados por cómo se manejó la imagen anteriormente. Además, esta VM está asignada con 2048 MiB de RAM y 2 vCPUs.

```
# virt-install --name demo-guest3 --memory 2048 --vcpus 2 --os-variant rhel8.0 --import --disk /home/username/backup/disk.qcow2
```

Tenga en cuenta que la opción **--os-variant** es muy recomendable cuando se importa una imagen de disco. Si no se proporciona, el rendimiento de la VM creada se verá afectado negativamente.

- Lo siguiente crea una VM llamada **demo-guest4** que se instala desde la <http://example.com/OS-install> URL. Para que la instalación se inicie con éxito, la URL debe contener un árbol de instalación del SO que funcione. Además, el SO se configura automáticamente utilizando el archivo kickstart de `/home/username/ks.cfg`. Esta VM también está asignada con 2048 MiB de RAM, 2 vCPUs, y un disco virtual qcow2 de 160 GiB.

```
# virt-install --name demo-guest4 --memory 2048 --vcpus 2 --disk size=160 --os-variant rhel8.0 --location http://example.com/OS-install --initrd-inject /home/username/ks.cfg --extra-args="ks=file:/ks.cfg console=tty0 console=ttyS0,115200n8"
```

- Lo siguiente crea una VM llamada **demo-guest5** que se instala desde un archivo de imagen **RHEL8.iso** en modo de sólo texto, sin gráficos. Conecta la consola del huésped a la consola serial. La VM tiene 16384 MiB de memoria, 16 vCPUs, y 280 GiB de disco. Este tipo de instalación es útil cuando se conecta a un host a través de un enlace de red lento.

```
# virt-install --name demo-guest5 --memory 16384 --vcpus 16 --disk size=280 --os-variant rhel8.0 --location RHEL8.iso --graphics none --extra-args='console=ttyS0'
```

- Lo siguiente crea una VM llamada **demo-guest6**, que tiene la misma configuración que `demo-guest5`, pero reside en el host remoto 10.0.0.1.

```
# virt-install --connect qemu ssh://root@10.0.0.1/system --name demo-guest6 --
memory 16384 --vcpus 16 --disk size=280 --os-variant rhel8.0 --location RHEL8.iso --
graphics none --extra-args='console=ttyS0'
```

Si la VM se crea con éxito, se abre una ventana de [virt-viewer](#) con una consola gráfica de la VM y se inicia la instalación del SO huésped.

Solución de problemas

- Si **virt-install** falla con un error **cannot find default network**:

- a. Asegúrese de que el paquete *libvirt-daemon-config-network* está instalado:

```
# yum info libvirt-daemon-config-network
Installed Packages
Name      : libvirt-daemon-config-network
[...]
```

- b. Compruebe que la red por defecto **libvirt** está activa y configurada para iniciarse automáticamente:

```
# virsh net-list --all
Name    State  Autostart  Persistent
-----
default active    yes        yes
```

- c. Si no lo está, active la red por defecto y configúrela para que se inicie automáticamente:

```
# virsh net-autostart default
Network default marked as autostarted

# virsh net-start default
Network default started
```

- i. Si la activación de la red por defecto falla con el siguiente error, el paquete *libvirt-daemon-config-network* no se ha instalado correctamente.

```
error: failed to get network 'default'
error: Network not found: no network with matching name 'default'
```

Para solucionarlo, vuelva a instalar *libvirt-daemon-config-network*.

```
# yum reinstall libvirt-daemon-config-network
```

- ii. Si la activación de la red por defecto falla con un error similar al siguiente, se ha producido un conflicto entre la subred de la red por defecto y una interfaz existente en el host.

```
error: Failed to start network default
error: internal error: Network is already in use by interface ens2
```

Para solucionarlo, utilice el comando **virsh net-edit default** y cambie los valores de 192.168.122.* en la configuración a una subred que no esté ya en uso en el host.

Recursos adicionales

- Se pueden especificar otras opciones para **virt-install** para configurar aún más la VM y su instalación del SO. Para más detalles, consulte la página man **virt-install**.
- Si ya tiene una VM funcional, puede clonarla para crear rápidamente una nueva VM con la misma configuración y datos. Para más detalles, consulte [Capítulo 8, Clonación de máquinas virtuales](#).

2.2.2. Creación de máquinas virtuales e instalación de sistemas operativos invitados mediante la consola web

Para gestionar máquinas virtuales (VMs) en una GUI en un host RHEL 8, utilice la consola web. Las siguientes secciones proporcionan información sobre cómo utilizar la consola web de RHEL 8 para crear máquinas virtuales e instalar sistemas operativos invitados en ellas.

2.2.2.1. Creación de máquinas virtuales mediante la consola web

Para crear una máquina virtual (VM) en el equipo anfitrión al que está conectada la consola web, siga las siguientes instrucciones.

Requisitos previos

- Para utilizar la consola web para gestionar las máquinas virtuales, [instale el complemento VM de la consola web](#).
- Tiene una cantidad suficiente de recursos del sistema para asignar a sus VMs, como espacio en disco, RAM o CPUs. Los valores recomendados pueden variar significativamente en función de las tareas previstas y la carga de trabajo de las máquinas virtuales.

Procedimiento

1. En la interfaz de **Máquinas Virtuales** de la consola web, haga clic en **Create VM**. Aparece el cuadro de diálogo Crear nueva máquina virtual.

Create New Virtual Machine ✕

Name

Installation Type ▼
Download an OS

Operating System ▼
Fedora 31

Storage ▼
Create New Volume

Size 20 GiB ▼
 10 42

Memory 2 GiB ▼
 1 3

Immediately Start VM

2. Introduzca la configuración básica de la VM que desea crear.

- **Name** - El nombre de la máquina virtual.
- **Installation Type** - La instalación puede utilizar un medio de instalación local, una URL, un arranque de red PXE o descargar un sistema operativo de un conjunto limitado de sistemas operativos.
- **Operating System** - El sistema operativo de la VM. Tenga en cuenta que Red Hat sólo ofrece soporte para un [conjunto limitado de sistemas operativos para huéspedes](#) .
- **Storage** - El tipo de almacenamiento con el que configurar la VM.
- **Size** - La cantidad de espacio de almacenamiento con la que configurar la VM.
- **Memory** - La cantidad de memoria con la que configurar la VM.
- **Run unattended installation** - Si se ejecuta o no la instalación sin supervisión.
- **Immediately Start VM** - Si la VM se iniciará o no inmediatamente después de ser creada.

3. Haga clic en **Crear**.

Se crea la VM. Si se selecciona la casilla **Immediately Start VM**, la VM se iniciará inmediatamente y comenzará a instalar el sistema operativo invitado.

Recursos adicionales

- Para obtener información sobre la instalación de un sistema operativo en una VM, consulte [Sección 2.2.2.3, "Instalación de sistemas operativos invitados mediante la consola web"](#) .

2.2.2.2. Creación de máquinas virtuales mediante la importación de imágenes de disco a través de la consola web

Para crear una máquina virtual (VM) importando una imagen de disco de una instalación VM existente, siga las siguientes instrucciones.

Requisitos previos

- Para utilizar la consola web para gestionar las máquinas virtuales, [instale el complemento VM de la consola web](#).
- Tiene una cantidad suficiente de recursos del sistema para asignar a sus VMs, como espacio en disco, RAM o CPUs. Los valores recomendados pueden variar significativamente en función de las tareas previstas y la carga de trabajo de las máquinas virtuales.
- Asegúrese de tener una imagen de disco de una instalación de VM existente

Procedimiento

1. En la interfaz de **Máquinas Virtuales** de la consola web, haga clic en **Import VM**. Aparece el cuadro de diálogo Importar una máquina virtual.

Import A Virtual Machine ✕

Name

Connection System Session

Installation Source ▼

Operating System ▼

Memory 0 15 ▼

Immediately Start VM

2. Introduzca la configuración básica de la VM que desea crear.
 - **Name** - El nombre de la máquina virtual.
 - **Connection** - El tipo de conexión libvirt, sistema o sesión.
 - **Installation Source** - La imagen de disco existente de una VM en el sistema anfitrión.

- **Operating System** - El sistema operativo de la VM. Tenga en cuenta que Red Hat sólo ofrece soporte para un [conjunto limitado de sistemas operativos para huéspedes](#).
- **Memory** - La cantidad de memoria con la que configurar la VM.
- **Immediately Start VM** - Si la VM se iniciará o no inmediatamente después de ser creada.

3. Haga clic en **Importar**.

2.2.2.3. Instalación de sistemas operativos invitados mediante la consola web

La primera vez que se carga una máquina virtual (VM), se debe instalar un sistema operativo en la VM.



NOTA

Si la casilla *Immediately Start VM* del diálogo Crear nueva máquina virtual está marcada, la rutina de instalación del sistema operativo se inicia automáticamente cuando se crea la VM.

Requisitos previos

- Para utilizar la consola web para gestionar las máquinas virtuales, [instale el complemento VM de la consola web](#).
- Debe haber una máquina virtual en la que instalar un sistema operativo.

Procedimiento

1. En la interfaz de **máquinas virtuales**, haga clic en la fila de la máquina virtual en la que desea instalar un sistema operativo invitado.

La fila se expande para revelar el panel de Visión General con información básica sobre la VM seleccionada y controles para instalar y eliminar la VM.

Name	Connection	State	
> grid2	Session	shut off	Run ⋮
▼ Oasis_v1	Session	shut off	Install ⋮
<p>Overview Usage Disks Network Interfaces Consoles</p> <p>General</p> <p>Memory 2 GiB</p> <p>vCPUs 2</p> <p>CPU Type host</p> <p>Boot Order disk</p> <p>Autostart <input type="checkbox"/> Run when host boots</p> <p>Hypervisor Details</p> <p>Emulated Machine pc-q35-4.1</p> <p>Firmware UEFI</p>			
> Tron_3.0	System	shut off	Run ⋮

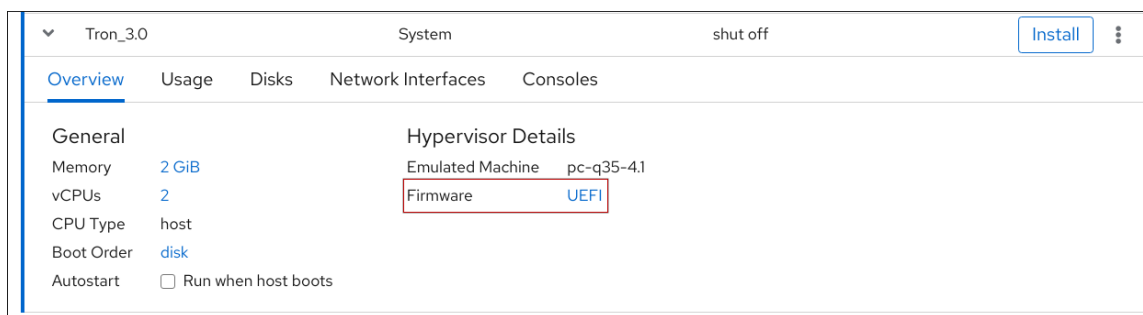
2. **Optional:** Cambiar el firmware.



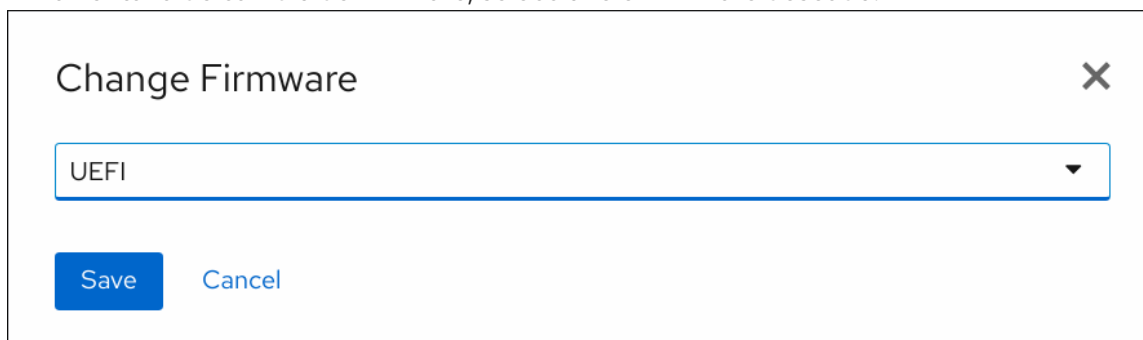
NOTA

Puede cambiar el firmware sólo si no ha seleccionado la casilla *Immediately Start VM* en el cuadro de diálogo **Create New Virtual Machine** y el sistema operativo no ha sido ya instalado en la VM.

- a. Haga clic en el firmware resaltado.



b. En la ventana de cambio de firmware, seleccione el firmware deseado.



c. Haga clic en **Guardar**.

3. Haga clic en **Instalar**.

La rutina de instalación del sistema operativo se ejecuta en la consola de la VM.

Solución de problemas

- Si la rutina de instalación falla, la VM debe ser borrada y recreada.

2.3. INICIAR MÁQUINAS VIRTUALES

Para iniciar una máquina virtual (VM) en RHEL 8, puede utilizar [la interfaz de línea de comandos](#) o [la GUI de la consola web](#).

Requisitos previos

- Antes de poder iniciar una VM, ésta debe ser creada e, idealmente, también instalada con un SO. Para obtener instrucciones para hacerlo, consulte [Sección 2.2, "Creación de máquinas virtuales"](#).

2.3.1. Iniciar una máquina virtual mediante la interfaz de línea de comandos

Puede utilizar la interfaz de línea de comandos para iniciar una máquina virtual (VM) apagada o restaurar una VM guardada. Siga el procedimiento siguiente.

Requisitos previos

- Una VM inactiva que ya está definida.
- El nombre de la máquina virtual.
- Para las máquinas virtuales remotas:
 - La dirección IP del host donde se encuentra la VM.

- Privilegios de acceso a la raíz del host.

Procedimiento

- Para una VM local, utilice la utilidad **virsh start**.
Por ejemplo, el siguiente comando inicia la VM *demo-guest1*.

```
# virsh start demo-guest1
Domain demo-guest1 started
```

- Para una VM ubicada en un host remoto, utilice la utilidad **virsh start** junto con la conexión SSH de QEMU al host.
Por ejemplo, el siguiente comando inicia la VM *demo-guest1* en el host 192.168.123.123.

```
# virsh -c qemu+ssh://root@192.168.123.123/system start demo-guest1

root@192.168.123.123's password:
Last login: Mon Feb 18 07:28:55 2019

Domain demo-guest1 started
```

Recursos adicionales

- Para más argumentos **virsh start**, utilice **virsh start --help**.
- Para simplificar la gestión de las máquinas virtuales en hosts remotos, consulte la [modificación de la configuración de libvirt y SSH](#).
- Puede utilizar la utilidad **virsh autostart** para configurar una VM para que se inicie automáticamente cuando el host arranque. Para obtener más información sobre el arranque automático, consulte la página de ayuda **virsh autostart**.

2.3.2. Iniciar máquinas virtuales mediante la consola web

Si una máquina virtual (VM) está en el estado **shut off**, puede iniciarla utilizando la consola web de RHEL 8.

Requisitos previos

- Una VM inactiva que ya está definida.
- El nombre de la máquina virtual.
- Para utilizar la consola web para gestionar las máquinas virtuales, [instale el complemento VM de la consola web](#).

Procedimiento

1. En la interfaz de **máquinas virtuales**, haga clic en la fila de la máquina virtual que desea iniciar. La fila se expande para revelar el panel de Visión General con información básica sobre la VM seleccionada y controles para apagar y eliminar la VM.
2. Haga clic en **Ejecutar**.
La VM se inicia, y puedes [conectarte a su consola o salida gráfica](#).

3. **Optional:** Para configurar la VM para que se inicie automáticamente cuando se inicie el host, haga clic en la casilla **Autostart**.

Recursos adicionales

- Para obtener información sobre cómo apagar una máquina virtual, consulte [Sección 2.5.2.1, "Apagado de máquinas virtuales en la consola web"](#).
- Para obtener información sobre cómo reiniciar una máquina virtual, consulte [Sección 2.5.2.2, "Reinicio de máquinas virtuales mediante la consola web"](#).
- Para obtener información sobre el envío de una interrupción no enmascarable a una VM, consulte [Sección 2.5.2.3, "Envío de interrupciones no enmascarables a las máquinas virtuales mediante la consola web"](#).

2.4. CONEXIÓN A MÁQUINAS VIRTUALES

Para interactuar con una máquina virtual (VM) en RHEL 8, es necesario conectarse a ella realizando una de las siguientes acciones:

- Cuando utilice la interfaz de la consola web, utilice el panel de Máquinas Virtuales en la interfaz de la consola web. Para obtener más información, consulte [Sección 2.4.1, "Interacción con las máquinas virtuales mediante la consola web"](#).
- Si necesita interactuar con una pantalla gráfica de la VM sin utilizar la consola web, utilice la aplicación Virt Viewer. Para más detalles, consulte [Sección 2.4.2, "Abrir la consola gráfica de una máquina virtual con Virt Viewer"](#).
- Cuando no es posible o no es necesaria una visualización gráfica, utilice [una conexión de terminal SSH](#).
- Cuando la máquina virtual no es accesible desde su sistema utilizando una red, utilice [la consola virsh](#).

Si las máquinas virtuales a las que se conecta están en un host remoto en lugar de uno local, puede configurar opcionalmente su sistema para [un acceso más conveniente a los hosts remotos](#).

Requisitos previos

- Las máquinas virtuales con las que se quiere interactuar están [instaladas](#) e [iniciadas](#).

2.4.1. Interacción con las máquinas virtuales mediante la consola web

Para interactuar con una máquina virtual (VM) en la consola web de RHEL 8, es necesario conectarse a la consola de la VM. Esto incluye tanto las consolas gráficas como las seriales.

- Para interactuar con la interfaz gráfica de la VM en la consola web, utilice [la consola gráfica](#).
- Para interactuar con la interfaz gráfica de la VM en un visor remoto, utilice [la consola gráfica de los visores remotos](#).
- Para interactuar con la CLI de la VM en la consola web, utilice [la consola serie](#).

2.4.1.1. Visualización de la consola gráfica de la máquina virtual en la consola web

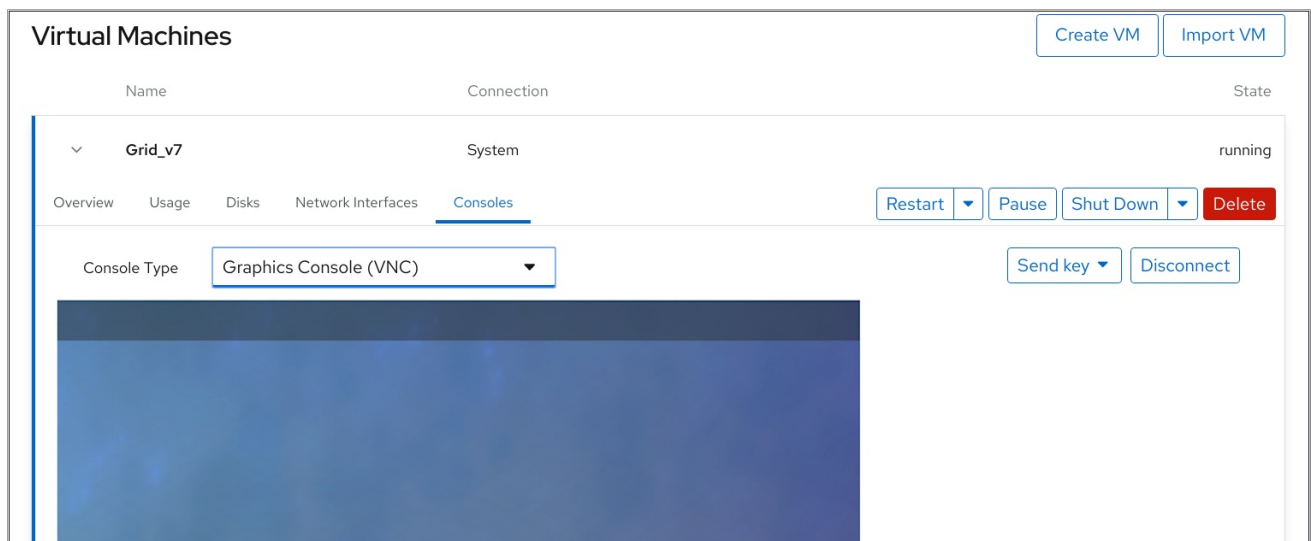
Mediante la interfaz de la consola de la máquina virtual (VM), puede ver la salida gráfica de una VM seleccionada en la consola web de RHEL 8.

Requisitos previos

- Para utilizar la consola web para gestionar las máquinas virtuales, [instale el complemento VM de la consola web](#).
- Asegúrese de que tanto el host como la VM admiten una interfaz gráfica.

Procedimiento

1. En la interfaz de **máquinas virtuales**, haga clic en la fila de la máquina virtual cuya consola gráfica desee ver.
La fila se expande para revelar el panel de Visión General con información básica sobre la VM seleccionada y controles para apagar y eliminar la VM.
2. Haga clic en **Consolas**.
La consola gráfica aparece en la interfaz web.



Puede interactuar con la consola de la VM utilizando el ratón y el teclado de la misma manera que interactúa con una máquina real. La pantalla de la consola de la VM refleja las actividades que se realizan en la VM.



NOTA

El host en el que se ejecuta la consola web puede interceptar determinadas combinaciones de teclas, como **Ctrl+Alt+Del**, impidiendo que se envíen a la VM.

Para enviar estas combinaciones de teclas, haga clic en el menú **Enviar tecla** y seleccione la secuencia de teclas que desea enviar.

Por ejemplo, para enviar la combinación **Ctrl+Alt+Del** a la VM, haga clic en el menú de la **tecla Enviar** y seleccione la entrada del menú **Ctrl Alt Supr**.

Recursos adicionales

- Para obtener instrucciones sobre la visualización de la consola gráfica en un visor remoto, consulte [Sección 2.4.1.2, "Visualización de la consola gráfica en un visor remoto mediante la consola web"](#).
- Para obtener instrucciones sobre la visualización de la consola serie en la consola web, consulte [Sección 2.4.1.3, "Visualización de la consola de serie de la máquina virtual en la consola web"](#).

2.4.1.2. Visualización de la consola gráfica en un visor remoto mediante la consola web

Puede ver la consola gráfica de una máquina virtual (VM) seleccionada en un visor remoto, como virt-viewer. Para obtener instrucciones, consulte más abajo.



NOTA

Puedes lanzar Virt Viewer desde la consola web. Otros visores remotos VNC y SPICE se pueden lanzar manualmente.

Requisitos previos

- Para utilizar la consola web para gestionar las máquinas virtuales, [instale el complemento VM de la consola web](#).
- Asegúrese de que tanto el host como la VM admiten una interfaz gráfica.
- Antes de poder ver la consola gráfica en Virt Viewer, Virt Viewer debe estar instalado en la máquina a la que está conectada la consola web.
Para ver la información sobre la instalación de Virt Viewer, seleccione la **Consola de Gráficos en el Tipo de Consola del Visor de Escritorio** y haga clic en **More Information** en la ventana de Consolas.

[▼ More Information](#)

Clicking "Launch Remote Viewer" will download a .v file and launch *Remote Viewer*.

Remote Viewer is available for most operating systems. To install it, search for it in GNOME Software or run the following:

- **RHEL, CentOS:** `sudo yum install virt-viewer`
- **Fedora:** `sudo dnf install virt-viewer`
- **Ubuntu, Debian:** `sudo apt-get install virt-viewer`
- **Windows:** Download the MSI from virt-manager.org

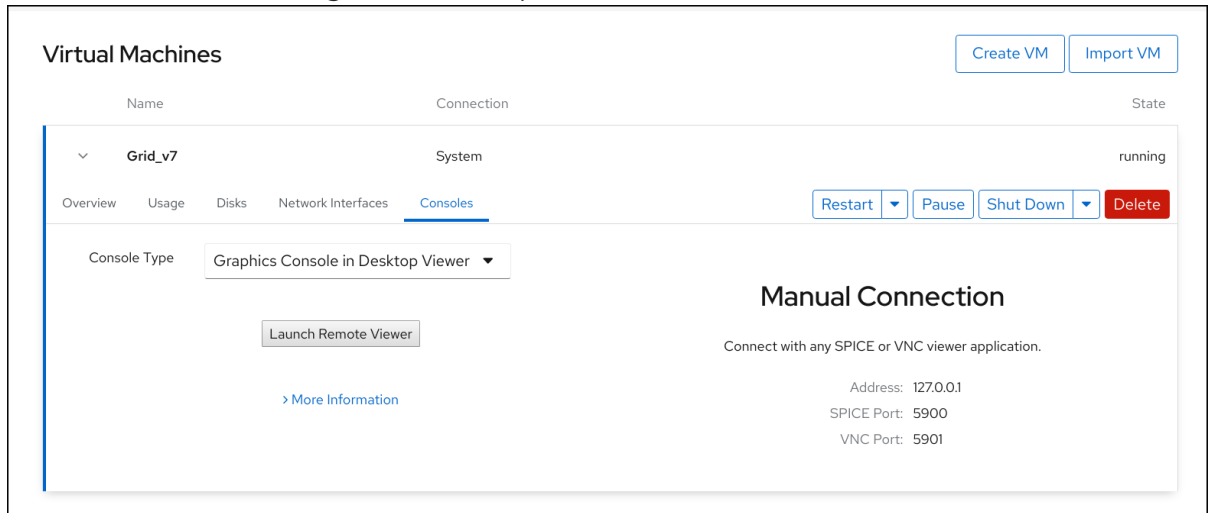


NOTA

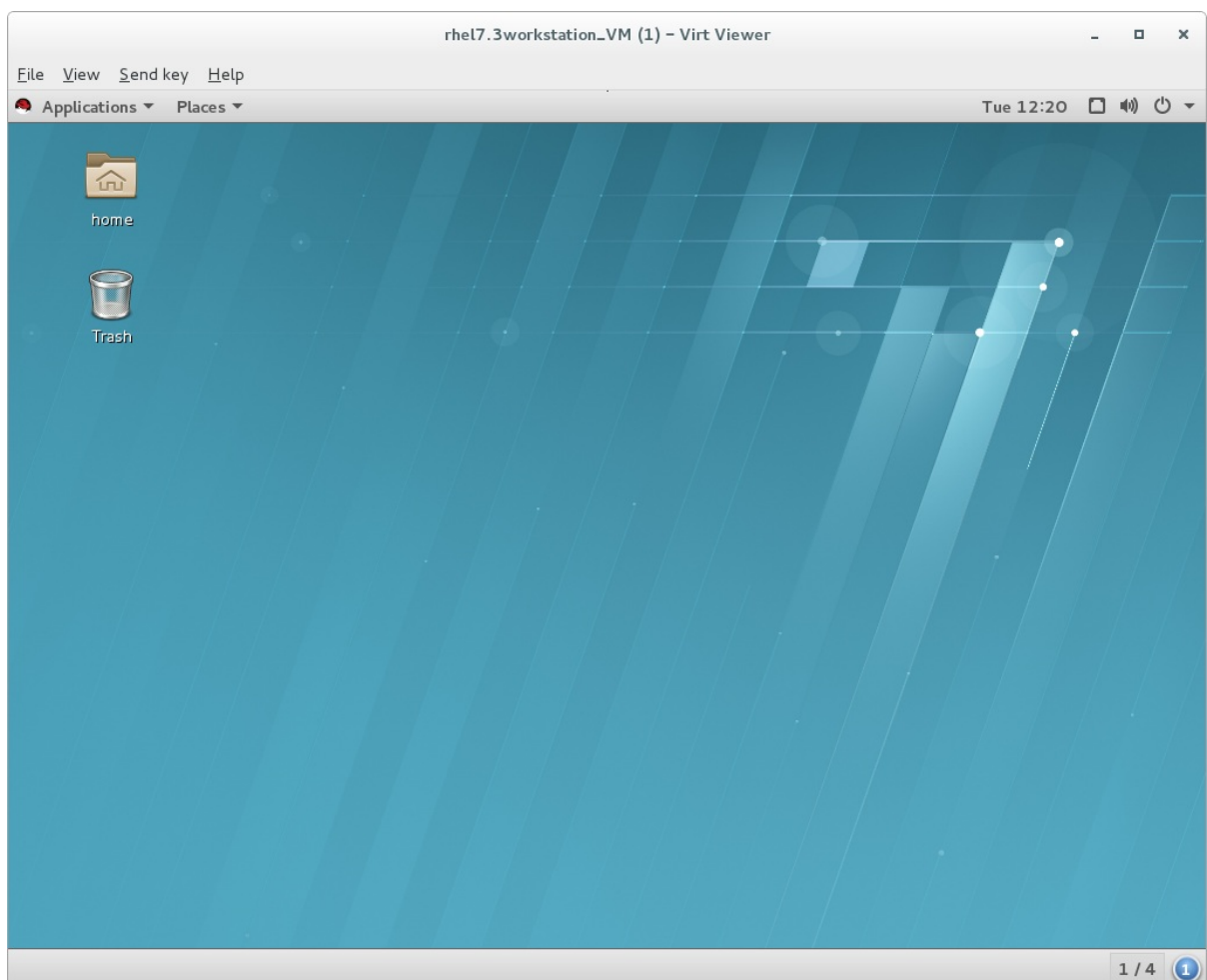
Algunas extensiones y complementos del navegador no permiten que la consola web abra el Virt Viewer.

Procedimiento

1. En la interfaz de **máquinas virtuales**, haga clic en la fila de la máquina virtual cuya consola gráfica desee ver.
La fila se expande para revelar el panel de Visión General con información básica sobre la VM seleccionada y controles para apagar y eliminar la VM.
2. Haga clic en **Consolas**.
La consola gráfica aparece en la interfaz web.
3. Seleccione la **consola de gráficos en el tipo de consola del visor de escritorio**.



4. Haga clic en **Lanzar el Visor Remoto**.
La consola gráfica aparece en Virt Viewer.



Puede interactuar con la consola de la VM utilizando el ratón y el teclado de la misma manera que interactúa con una máquina real. La pantalla de la consola de la VM refleja las actividades que se realizan en la VM.



NOTA

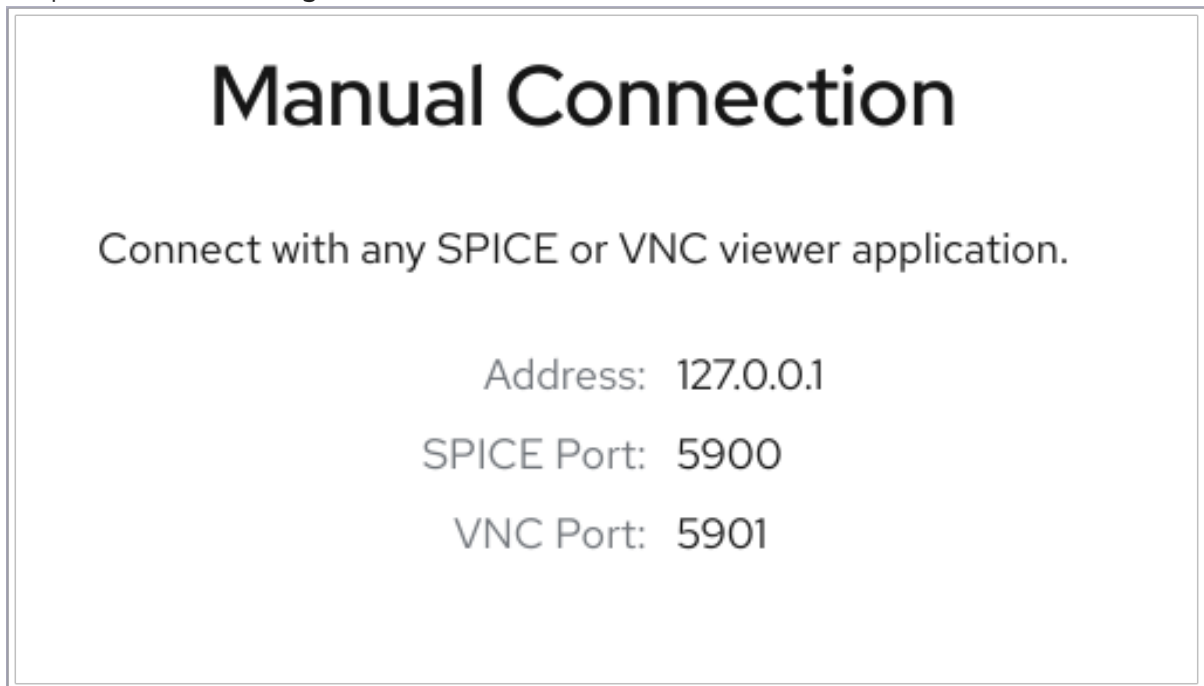
El servidor en el que se ejecuta la consola web puede interceptar determinadas combinaciones de teclas, como **Ctrl+Alt+Del**, impidiendo que se envíen a la VM.

Para enviar estas combinaciones de teclas, haga clic en el menú **Enviar tecla** y seleccione la secuencia de teclas que desea enviar.

Por ejemplo, para enviar la combinación **Ctrl+Alt+Del** a la VM, haga clic en el menú de la **tecla Enviar** y seleccione la entrada del menú **Ctrl Alt Supr**.

Solución de problemas

- Si el lanzamiento de una consola gráfica del visor remoto en la consola web no funciona o no es óptimo, puede utilizar la información de **Manual Connection**, que aparece en la parte derecha del panel de la **consola gráfica**.



Introduzca la información en una aplicación de visualización SPICE o VNC, como Virt Viewer.

Recursos adicionales

- Para obtener instrucciones sobre la visualización de la consola gráfica en la consola web, consulte [Sección 2.4.1.1, “Visualización de la consola gráfica de la máquina virtual en la consola web”](#).
- Para obtener instrucciones sobre la visualización de la consola serie en la consola web, consulte [Sección 2.4.1.3, “Visualización de la consola de serie de la máquina virtual en la consola web”](#).

2.4.1.3. Visualización de la consola de serie de la máquina virtual en la consola web

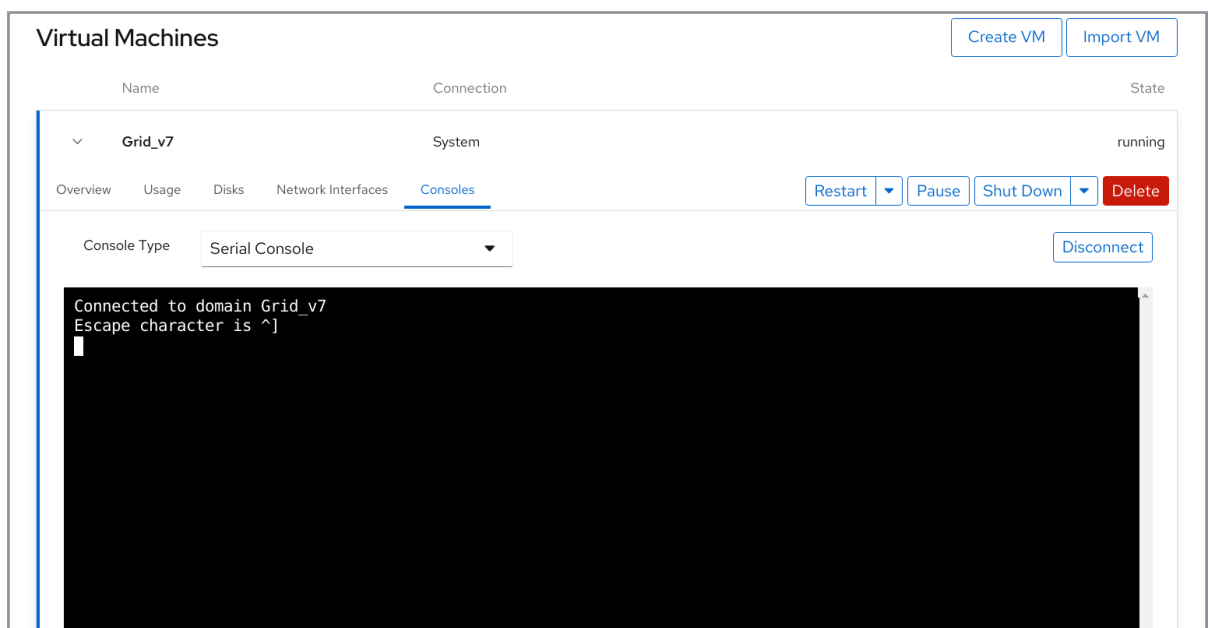
Puede ver la consola serie de una máquina virtual (VM) seleccionada en la consola web de RHEL 8. Esto es útil cuando la máquina anfitriona o la VM no están configuradas con una interfaz gráfica.

Requisitos previos

- Para utilizar la consola web para gestionar las máquinas virtuales, [instale el complemento VM de la consola web](#).

Procedimiento

1. En el panel de **máquinas virtuales**, haga clic en la fila de la máquina virtual cuya consola de serie desea ver.
La fila se expande para revelar el panel de Visión General con información básica sobre la VM seleccionada y controles para apagar y eliminar la VM.
2. Haga clic en **Consolas**.
La consola gráfica aparece en la interfaz web.
3. Seleccione el Tipo de Consola **Serial**.
La consola serie aparece en la interfaz web.



Puedes desconectar y volver a conectar la consola serie de la VM.

- Para desconectar la consola serie de la VM, haga clic en **Desconectar**.
- Para volver a conectar la consola serie a la VM, haga clic en **Reconectar**.

Recursos adicionales

- Para obtener instrucciones sobre la visualización de la consola gráfica en la consola web, consulte [Sección 2.4.1.1, "Visualización de la consola gráfica de la máquina virtual en la consola web"](#).
- Para obtener instrucciones sobre la visualización de la consola gráfica en un visor remoto, consulte [Sección 2.4.1.2, "Visualización de la consola gráfica en un visor remoto mediante la consola web"](#).

2.4.2. Abrir la consola gráfica de una máquina virtual con Virt Viewer

Para conectarse a una consola gráfica de una máquina virtual (VM) KVM y abrirla en la aplicación de escritorio **Virt Viewer**, siga el siguiente procedimiento.

Requisitos previos

- Tu sistema, así como la máquina virtual a la que te conectas, deben soportar pantallas gráficas.
- Si la VM de destino se encuentra en un host remoto, se necesita conexión y privilegios de acceso de root al host.
- **Optional:** Si la VM de destino se encuentra en un host remoto, configure su libvirt y SSH para [un acceso más conveniente a los hosts remotos](#).

Procedimiento

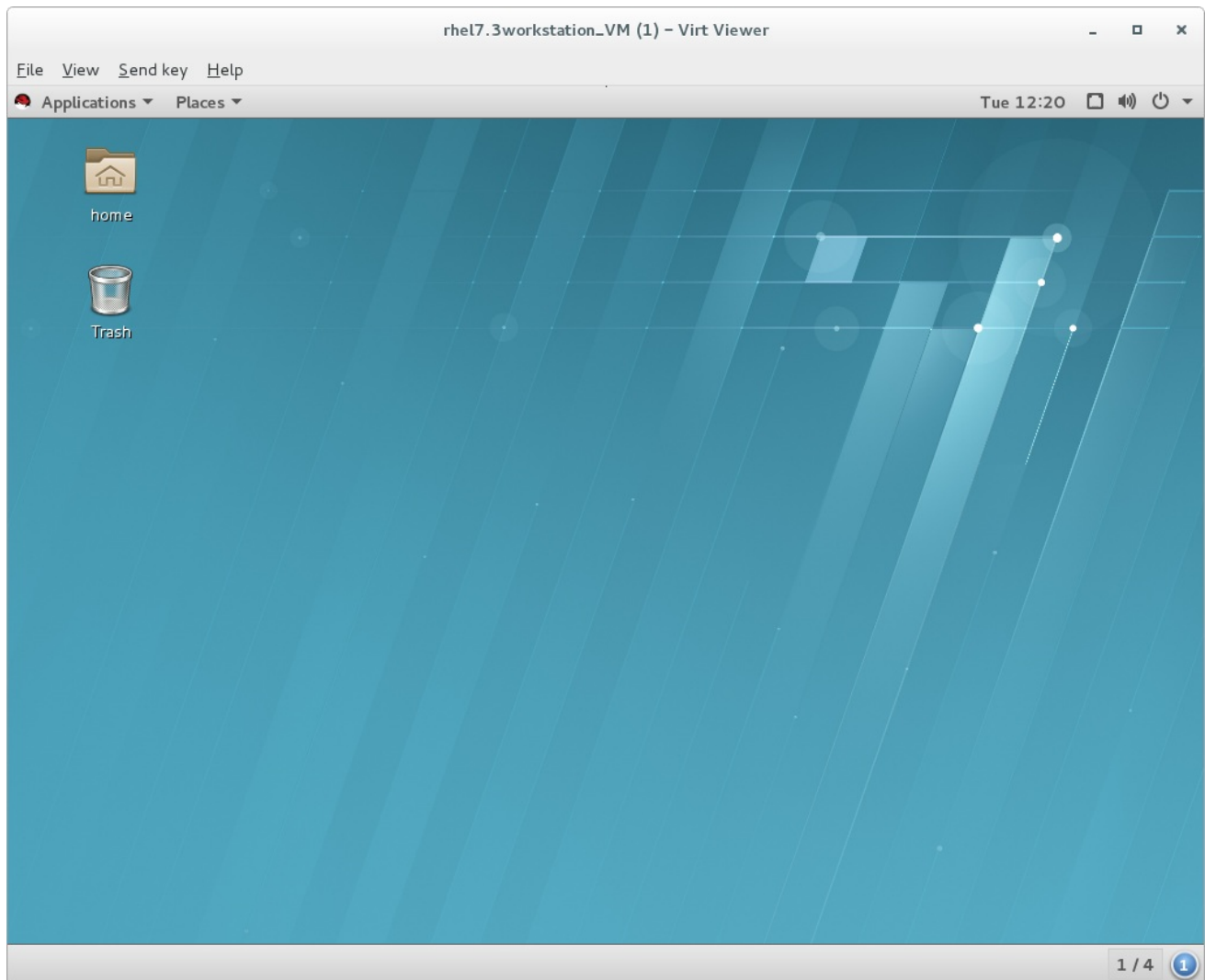
- Para conectarse a una VM local, utilice el siguiente comando y sustituya *guest-name* por el nombre de la VM a la que desea conectarse:

```
# virt-viewer guest-name
```

- Para conectarse a una VM remota, utilice el comando **virt-viewer** con el protocolo SSH. Por ejemplo, el siguiente comando se conecta como root a una VM llamada *guest-name*, ubicada en el sistema remoto 10.0.0.1. La conexión también requiere la autenticación de root para 10.0.0.1.

```
# virt-viewer --direct --connect qemu+ssh://root@10.0.0.1/system guest-name  
root@10.0.0.1's password:
```

Si la conexión funciona correctamente, la visualización de la VM se muestra en la ventana **Virt Viewer**.



Puede interactuar con la consola de la VM utilizando el ratón y el teclado de la misma manera que interactúa con una máquina real. La pantalla de la consola de la VM refleja las actividades que se realizan en la VM.

Recursos adicionales

- Para más información sobre el uso de Virt Viewer, consulte la página man **virt-viewer**.
- La conexión a las máquinas virtuales en un host remoto puede simplificarse [modificando la configuración de libvirt y SSH](#).
- Para la gestión de las máquinas virtuales en una GUI interactiva en RHEL 8, puede utilizar la interfaz de la consola web. Para obtener más información, consulte [Sección 2.4.1, "Interacción con las máquinas virtuales mediante la consola web"](#).

2.4.3. Conexión a una máquina virtual mediante SSH

Para interactuar con el terminal de una máquina virtual (VM) utilizando el protocolo de conexión SSH, siga el siguiente procedimiento:

Requisitos previos

- Tiene conexión de red y privilegios de acceso a la raíz de la máquina virtual de destino.

- Si la VM de destino se encuentra en un host remoto, también tiene privilegios de conexión y acceso de root a ese host.
- El componente **libvirt-nss** está instalado y habilitado en el host de la máquina virtual. Si no lo está, haga lo siguiente:

- a. Instale el paquete **libvirt-nss**:

```
# yum install libvirt-nss
```

- b. Edite el archivo `/etc/nsswitch.conf` y añada **libvirt_guest** a la línea **hosts**:

```
[...]
passwd:    compat
shadow:    compat
group:     compat
hosts:     files libvirt_guest dns
[...]
```

Procedimiento

1. **Optional:** Cuando se conecte a una máquina virtual remota, primero debe conectarse mediante SSH a su host físico. El siguiente ejemplo demuestra la conexión a una máquina anfitriona 10.0.0.1 utilizando sus credenciales de root:

```
# ssh root@10.0.0.1
root@10.0.0.1's password:
Last login: Mon Sep 24 12:05:36 2018
root~#
```

2. Utilice el nombre de la VM y las credenciales de acceso del usuario para conectarse a ella. Por ejemplo, lo siguiente se conecta a la VM "testguest1" Utilizando sus credenciales de root:

```
# ssh root@testguest1
root@testguest1's password:
Last login: Wed Sep 12 12:05:36 2018
root~]#
```

Solución de problemas

- Si no conoce el nombre de la VM, puede listar todas las VMs disponibles en el host utilizando el comando **virsh list --all**:

```
# virsh list --all
Id Name                               State
-----
 2 testguest1                          running
- testguest2                          shut off
```

2.4.4. Abrir la consola de serie de una máquina virtual

Utilizando el comando **virsh console**, es posible conectarse a la consola serie de una máquina virtual (VM).

Esto es útil cuando la VM:

- No proporciona protocolos VNC o SPICE, y por lo tanto no ofrece visualización de vídeo para las herramientas GUI.
- No tiene conexión de red, por lo que no se puede interactuar con él [mediante SSH](#).

Requisitos previos

- La VM debe tener la consola serie configurada en su línea de comandos del kernel. Para verificar esto, la salida del comando **cat /proc/cmdline** en la VM debe incluir `console=ttyS0`. Por ejemplo:

```
# cat /proc/cmdline
BOOT_IMAGE=vmlinuz-3.10.0-948.el7.x86_64 root=/dev/mapper/rhel-root ro console=tty0
console=ttyS0,9600n8 rd.lvm.lv=rhel/root rd.lvm.lv=rhel/swap rhgb
```

Si la consola serial no está configurada correctamente en una VM, al usar **virsh console** para conectarse a la VM, se conecta a una consola huésped que no responde. Sin embargo, puedes salir de la consola que no responde utilizando el acceso directo **Ctrl]**.

- Para configurar la consola serial en la VM, haga lo siguiente:
 - a. En la máquina virtual, edite el archivo `/etc/default/grub` y añada `console=ttyS0` a la línea que comienza con `GRUB_CMDLINE_LINUX`.
 - b. Borre las opciones del kernel que puedan impedir que sus cambios surtan efecto.

```
# grub2-editenv - unset kernelopts
```

- c. Recarga la configuración de Grub:

```
# grub2-mkconfig -o /boot/grub2/grub.cfg
Generating grub configuration file ...
Found linux image: /boot/vmlinuz-3.10.0-948.el7.x86_64
Found initrd image: /boot/initramfs-3.10.0-948.el7.x86_64.img
[...]
done
```

- d. Reinicie la máquina virtual.

Procedimiento

1. En su sistema anfitrión, utilice el comando **virsh console**. El siguiente ejemplo se conecta a la VM `guest1`, si el controlador libvirt soporta el manejo seguro de la consola:

```
# virsh console guest1 --safe
Connected to domain guest1
Escape character is ^]

Subscription-name
Kernel 3.10.0-948.el7.x86_64 on an x86_64

localhost login:
```


2. Puedes interactuar con la consola virsh de la misma manera que con una interfaz de línea de comandos estándar.

Recursos adicionales

- Para más información sobre la consola serie de la VM, consulte la página man de virsh.

2.4.5. Configuración de un acceso fácil a los hosts de virtualización remotos

Cuando se gestionan máquinas virtuales en un sistema anfitrión remoto utilizando las utilidades de libvirt, se recomienda utilizar la sintaxis **-c qemu ssh://root@hostname/system**. Por ejemplo, para utilizar el comando **virsh list** como root en el host 10.0.0.1:

```
# virsh -c qemu+ssh://root@10.0.0.1/system list
```

```
root@10.0.0.1's password:
Last login: Mon Feb 18 07:28:55 2019
```

Id	Name	State
1	remote-guest	running

Sin embargo, para mayor comodidad, puede eliminar la necesidad de especificar los detalles de la conexión en su totalidad modificando la configuración de SSH y libvirt. Por ejemplo, podrás hacer:

```
# virsh -c remote-host list
```

```
root@10.0.0.1's password:
Last login: Mon Feb 18 07:28:55 2019
```

Id	Name	State
1	remote-guest	running

Para habilitar esta mejora, siga las siguientes instrucciones.

Procedimiento

1. Edite o cree el archivo `~/.ssh/config` y añada lo siguiente, donde *host-alias* es un nombre abreviado asociado a un host remoto específico, y *hosturl* es la dirección URL del host.

```
Host host-alias
  User      root
  Hostname  hosturl
```

Por ejemplo, lo siguiente configura el alias *tyrannosaurus* para root@10.0.0.1:

```
Host tyrannosaurus
  User      root
  Hostname  10.0.0.1
```

2. Edita o crea el archivo `/etc/libvirt/libvirt.conf`, y añada lo siguiente, donde *qemu-host-alias* es un alias de host que las utilidades QEMU y libvirt asociarán con el host previsto:

```
uri_aliases = [
    "qemu-host-alias=qemu+ssh://host-alias/system",
]
```

Por ejemplo, lo siguiente utiliza el alias *tyrannosaurus* configurado en el paso anterior para configurar el alias *t-rex*, que significa **qemu ssh://10.0.0.1/system**:

```
uri_aliases = [
    "t-rex=qemu+ssh://tyrannosaurus/system",
]
```

- Como resultado, puede gestionar las VMs remotas utilizando utilidades basadas en libvirt en el sistema local con un parámetro añadido **-c qemu-host-alias** parámetro. Esto realiza automáticamente los comandos a través de SSH en el host remoto.

Por ejemplo, a continuación se enumeran las máquinas virtuales del host remoto 10.0.0.1, cuya conexión se configuró como *t-rex* en los pasos anteriores:

```
$ virsh -c t-rex list
```

```
root@10.0.0.1's password:
Last login: Mon Feb 18 07:28:55 2019
```

```
Id Name          State
-----
1  velociraptor  running
```

- Optional:** Si desea utilizar las utilidades de libvirt exclusivamente en un único host remoto, también puede establecer una conexión específica como objetivo por defecto para las utilidades basadas en libvirt. Para ello, edite el archivo **/etc/libvirt/libvirt.conf** y establezca el valor del parámetro **uri_default** en *qemu-host-alias*. Por ejemplo, lo siguiente utiliza el alias de host *t-rex* configurado en los pasos anteriores como objetivo por defecto de libvirt.

```
# These can be used in cases when no URI is supplied by the application
# (@uri_default also prevents probing of the hypervisor driver).
#
uri_default = "t-rex"
```

Como resultado, todos los comandos basados en libvirt se ejecutarán automáticamente en el host remoto especificado.

```
$ virsh list
```

```
root@10.0.0.1's password:
Last login: Mon Feb 18 07:28:55 2019
```

```
Id Name          State
-----
1  velociraptor  running
```

Sin embargo, esto no es recomendable si también quiere gestionar máquinas virtuales en su host local o en diferentes hosts remotos.

Recursos adicionales

- Al conectarse a un host remoto, puede evitar tener que proporcionar la contraseña de root al sistema remoto. Para ello, utilice uno o varios de los siguientes métodos:
 - [Configurar el acceso SSH basado en claves al host](#) remoto.
 - Utilice la multiplexación de la conexión SSH para conectarse al sistema remoto.
 - Configure un ticket de autenticación kerberos en el sistema remoto. Para obtener instrucciones, consulte [Autenticación Kerberos en Gestión de identidades](#).
- Entre las utilidades que pueden utilizar la opción **-c** (o **--connect**) y la configuración de acceso al host remoto descrita anteriormente se encuentran:
 - [virt-install](#)
 - [virt-viewer](#)
 - `virsh`
 - `virt-manager`

2.5. APAGADO DE MÁQUINAS VIRTUALES

Para apagar una máquina virtual en ejecución en Red Hat Enterprise Linux 8, utilice [la interfaz de línea de comandos](#) o [la GUI de la consola web](#).

2.5.1. Apagar una máquina virtual mediante la interfaz de línea de comandos

Para apagar una máquina virtual (VM) con capacidad de respuesta, realice una de las siguientes acciones:

- Utilice un comando de apagado apropiado para el SO invitado mientras [esté conectado al mismo](#).
- Utilice el comando **virsh shutdown** en el host:
 - Si la VM está en un host local:

```
# virsh shutdown demo-guest1
Domain demo-guest1 is being shutdown
```

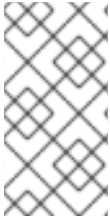
- Si la VM está en un host remoto, en este ejemplo 10.0.0.1:

```
# virsh -c qemu+ssh://root@10.0.0.1/system shutdown demo-guest1

root@10.0.0.1's password:
Last login: Mon Feb 18 07:28:55 2019
Domain demo-guest1 is being shutdown
```

Para forzar el cierre de un huésped, por ejemplo, si no responde, utilice el comando **virsh destroy** en el host:

```
# virsh destroy demo-guest1
Domain demo-guest1 destroyed
```



NOTA

El comando **virsh destroy** no borra o elimina la configuración de la VM o las imágenes de disco. Sólo destruye la instancia de la VM en ejecución. Sin embargo, en casos raros, este comando puede causar la corrupción del sistema de archivos de la VM, por lo que el uso de **virsh destroy** sólo se recomienda si todos los demás métodos de apagado han fallado.

2.5.2. Apagado y reinicio de máquinas virtuales mediante la consola web

Utilizando la consola web de RHEL 8, puede [apagar](#) o [reiniciar](#) las máquinas virtuales en funcionamiento. También puede enviar una interrupción no enmascarable a una máquina virtual que no responda.

2.5.2.1. Apagado de máquinas virtuales en la consola web

Si una máquina virtual (VM) está en el estado **running**, puede apagarla utilizando la consola web de RHEL 8.

Requisitos previos

- Para utilizar la consola web para gestionar las máquinas virtuales, [instale el complemento VM de la consola web](#).

Procedimiento

1. En la interfaz de **máquinas virtuales**, haga clic en la fila de la máquina virtual que desea apagar. La fila se expande para revelar el panel de Visión General con información básica sobre la VM seleccionada y controles para apagar y eliminar la VM.
2. Haga clic en **Apagar**.
La máquina virtual se apaga.

Solución de problemas

- Si la VM no se apaga, haga clic en el botón Menú **⋮** junto al botón **Apagar** y seleccione **Forzar apagado**.
- Para apagar una máquina virtual que no responde, también puede enviar una interrupción no enmascarable. Para más información, consulte [Sección 2.5.2.3, "Envío de interrupciones no enmascarables a las máquinas virtuales mediante la consola web"](#).

Recursos adicionales

- Para obtener información sobre cómo iniciar una máquina virtual, consulte [Sección 2.3.2, "Iniciar máquinas virtuales mediante la consola web"](#).
- Para obtener información sobre cómo reiniciar una máquina virtual, consulte [Sección 2.5.2.2, "Reinicio de máquinas virtuales mediante la consola web"](#).

2.5.2.2. Reinicio de máquinas virtuales mediante la consola web

Si una máquina virtual (VM) está en el estado **running**, puede reiniciarla utilizando la consola web de RHEL 8.


Requisitos previos

- Para utilizar la consola web para gestionar las máquinas virtuales, [instale el complemento VM de la consola web](#).

Procedimiento

1. En la interfaz de **máquinas virtuales**, haga clic en la fila de la máquina virtual que desea reiniciar. La fila se expande para revelar el panel de Visión General con información básica sobre la VM seleccionada y controles para apagar y eliminar la VM.
2. Haga clic en **Reiniciar**. La máquina virtual se apaga y se reinicia.

Solución de problemas

- Si la VM no se reinicia, haga clic en el botón Menú  junto al botón **Reiniciar** y seleccione **Forzar reinicio**.
- Para reiniciar una máquina virtual que no responde, también puede enviar una interrupción no enmascarable. Para más información, consulte [Sección 2.5.2.3, "Envío de interrupciones no enmascarables a las máquinas virtuales mediante la consola web"](#).

Recursos adicionales

- Para obtener información sobre cómo iniciar una máquina virtual, consulte [Sección 2.3.2, "Iniciar máquinas virtuales mediante la consola web"](#).
- Para obtener información sobre cómo apagar una máquina virtual, consulte [Sección 2.5.2.1, "Apagado de máquinas virtuales en la consola web"](#).


2.5.2.3. Envío de interrupciones no enmascarables a las máquinas virtuales mediante la consola web

El envío de una interrupción no enmascarable (NMI) puede hacer que una máquina virtual (VM) que no responde responda o se apague. Por ejemplo, puede enviar la **Ctrl+Alt+Del** NMI a una VM que no responde a la entrada estándar.

Requisitos previos

- Para utilizar la consola web para gestionar las máquinas virtuales, [instale el complemento VM de la consola web](#).

Procedimiento

1. En la interfaz de **máquinas virtuales**, haga clic en la fila de la máquina virtual a la que desea enviar un NMI. La fila se expande para revelar el panel de Visión General con información básica sobre la VM seleccionada y controles para apagar y eliminar la VM.
2. Haz clic en el botón Menú  junto al botón **Apagar** y selecciona **Enviar interrupción no enmascarable**. Se envía un NMI a la VM.

Recursos adicionales

- Para obtener información sobre cómo iniciar una máquina virtual, consulte [Sección 2.3.2, “Iniciar máquinas virtuales mediante la consola web”](#).
- Para obtener información sobre cómo reiniciar una máquina virtual, consulte [Sección 2.5.2.2, “Reinicio de máquinas virtuales mediante la consola web”](#).
- Para obtener información sobre cómo apagar una máquina virtual, consulte [Sección 2.5.2.1, “Apagado de máquinas virtuales en la consola web”](#).

2.6. ELIMINACIÓN DE MÁQUINAS VIRTUALES

Para eliminar máquinas virtuales en Red Hat Enterprise Linux 8, utilice la [interfaz de línea de comandos](#) o la [GUI de la consola web](#).

2.6.1. Eliminación de máquinas virtuales mediante la interfaz de línea de comandos

Para eliminar una máquina virtual (VM), puede eliminar su configuración XML y los archivos de almacenamiento asociados del host utilizando la línea de comandos. Siga el procedimiento siguiente:

Requisitos previos

- Haz una copia de seguridad de los datos importantes de la máquina virtual.
- Apague la máquina virtual.
- Asegúrese de que ninguna otra máquina virtual utilice el mismo almacenamiento asociado.

Procedimiento

- Utilice la utilidad **virsh undefine**.
Por ejemplo, el siguiente comando elimina la VM *guest1*, sus volúmenes de almacenamiento asociados y la RAM no volátil, si la hay.

```
# virsh undefine guest1 --remove-all-storage --nvram
Domain guest1 has been undefined
Volume 'vda'(/home/images/guest1.qcow2) removed.
```

Recursos adicionales

- Para otros argumentos de **virsh undefine**, utilice **virsh undefine --help** o consulte la página de manual **virsh**.

2.6.2. Eliminación de máquinas virtuales mediante la consola web


Para eliminar una máquina virtual (VM) y sus archivos de almacenamiento asociados del host al que está conectada la consola web de RHEL 8, siga el siguiente procedimiento:

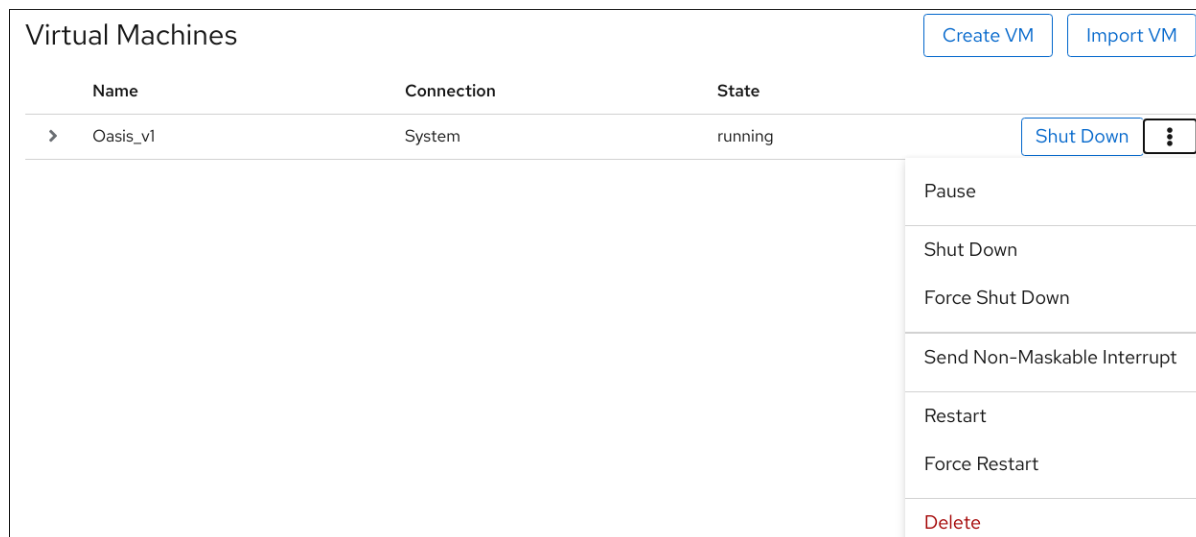
Requisitos previos

- Para utilizar la consola web para gestionar las máquinas virtuales, instale el [complemento VM de la consola web](#).
- Haz una copia de seguridad de los datos importantes de la máquina virtual.

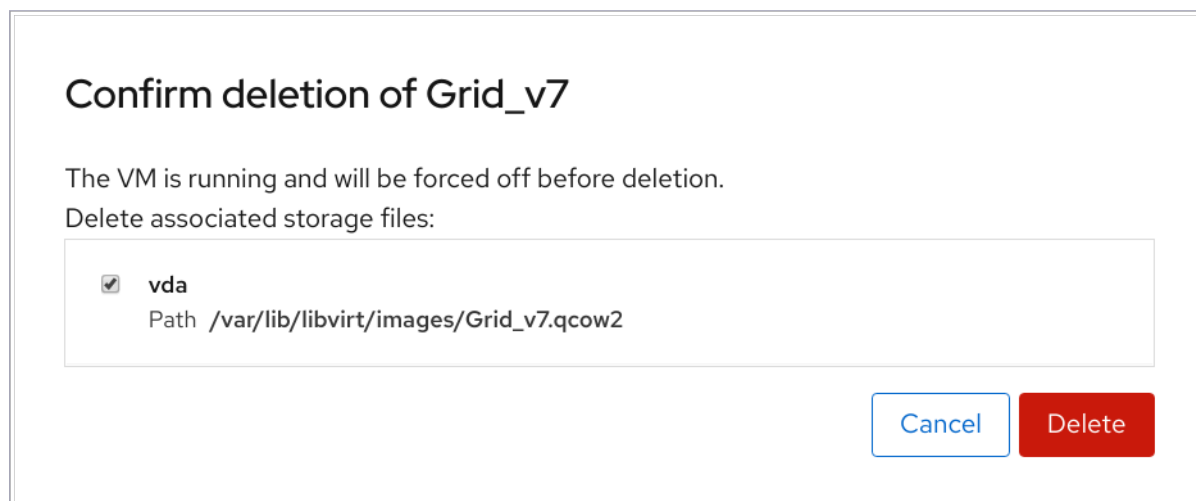
- Apague la máquina virtual.
- Asegúrese de que ninguna otra máquina virtual utilice el mismo almacenamiento asociado.

Procedimiento

1. En la interfaz de **Máquinas Virtuales**, haga clic en el botón Menú  de la VM que desea eliminar.
Aparece un menú desplegable con controles para varias operaciones de VM.



2. Haga clic en **Eliminar**.
Aparece un diálogo de confirmación.



3. **Optional:** Para eliminar todos o algunos de los archivos de almacenamiento asociados a la VM, seleccione las casillas de verificación junto a los archivos de almacenamiento que desee eliminar.
4. Haga clic en **Eliminar**.
La VM y los archivos de almacenamiento seleccionados se eliminan.

2.7. INFORMACIÓN RELACIONADA

- La información anterior se aplica a las arquitecturas AMD64 e Intel 64. Si desea utilizar la virtualización de RHEL8 en otras arquitecturas soportadas, se necesitan diferentes procedimientos de configuración y ciertas características pueden estar restringidas o funcionar

de manera diferente. Para más detalles, consulte la sección correspondiente más abajo:

- [Capítulo 3, *Introducción a la virtualización en IBM POWER*](#)
- [Capítulo 4, *Introducción a la virtualización en IBM Z*](#)

CAPÍTULO 3. INTRODUCCIÓN A LA VIRTUALIZACIÓN EN IBM POWER

Puede utilizar la virtualización KVM cuando utilice RHEL 8 en hardware IBM POWER8 o POWER9. Sin embargo, [habilitar el hipervisor K VM](#) en su sistema requiere pasos adicionales en comparación con la virtualización en arquitecturas AMD64 e Intel64. Algunas características de virtualización de RHEL 8 también tienen [una funcionalidad diferente o restringida](#) en IBM POWER.

Aparte de la información de las siguientes secciones, el uso de la virtualización en IBM POWER funciona igual que en AMD64 e Intel 64. Por lo tanto, puede consultar otra documentación de virtualización de RHEL 8 para obtener más información sobre el uso de la virtualización en IBM POWER.

3.1. HABILITACIÓN DE LA VIRTUALIZACIÓN EN IBM POWER

Para configurar un hipervisor KVM y crear máquinas virtuales (VM) en un sistema IBM POWER8 o IBM POWER9 que ejecute RHEL 8, siga las siguientes instrucciones.

Requisitos previos

- RHEL 8 está instalado y registrado en su máquina anfitriona.
- Los siguientes recursos mínimos del sistema están disponibles:
 - 6 GB de espacio libre en disco para el host, más otros 6 GB para cada máquina virtual prevista.
 - 2 GB de RAM para el host, más otros 2 GB para cada máquina virtual prevista.
- Su tipo de máquina con CPU debe soportar la virtualización IBM POWER. Para comprobarlo, consulte la información de la plataforma en su archivo `/proc/cpuinfo`.

```
# grep ^platform /proc/cpuinfo/  
platform      : PowerNV
```

Si la salida de este comando incluye la entrada **PowerNV**, está ejecutando un tipo de máquina PowerNV y puede utilizar la virtualización en IBM POWER.

Procedimiento

1. Cargar el módulo del kernel KVM-HV

```
# modprobe kvm_hv
```

2. Compruebe que el módulo del kernel KVM está cargado

```
# lsmod | grep kvm
```

Si KVM se ha cargado con éxito, la salida de este comando incluye **kvm_hv**.

3. Instale los paquetes en el módulo de virtualización:

```
# yum module install virt
```

4. Instale el paquete **virt-install**:

```
# yum install virt-install
```

5. Inicie el servicio **libvirtd**.

```
# systemctl start libvirtd
```

6. Compruebe que su sistema está preparado para ser un host de virtualización:

```
# virt-host-validate
[...]
QEMU: Checking if device /dev/vhost-net exists           : PASS
QEMU: Checking if device /dev/net/tun exists            : PASS
QEMU: Checking for cgroup 'memory' controller support   : PASS
QEMU: Checking for cgroup 'memory' controller mount-point : PASS
[...]
QEMU: Checking for cgroup 'blkio' controller support    : PASS
QEMU: Checking for cgroup 'blkio' controller mount-point : PASS
QEMU: Checking if IOMMU is enabled by kernel            : PASS
```

7. Si todas las comprobaciones de **virt-host-validate** devuelven un valor de **PASS**, su sistema está preparado para [crear máquinas virtuales](#).

Si alguna de las comprobaciones devuelve un valor de **FAIL**, siga las instrucciones mostradas para solucionar el problema.

Si alguna de las comprobaciones devuelve un valor de **WARN**, considere seguir las instrucciones mostradas para mejorar las capacidades de virtualización.

Información adicional

- Tenga en cuenta que si la virtualización no está soportada por su CPU anfitriona, **virt-host-validate** genera la siguiente salida:

```
QEMU: Comprobando la virtualización del hardware: FAIL (Sólo hay CPUs emuladas, el rendimiento será significativamente limitado)
```

Sin embargo, al intentar crear máquinas virtuales en un sistema anfitrión de este tipo, se producirá un fallo, en lugar de tener problemas de rendimiento.

3.2. EN QUÉ SE DIFERENCIA LA VIRTUALIZACIÓN EN IBM POWER DE LA DE AMD64 E INTEL 64

La virtualización KVM en RHEL 8 en sistemas IBM POWER es diferente de KVM en sistemas AMD64 e Intel 64 en varios aspectos, especialmente:

Requisitos de memoria

Las máquinas virtuales en IBM POWER consumen más memoria. Por lo tanto, la asignación de memoria mínima recomendada para una máquina virtual (VM) en un host IBM POWER es de 2 GB de RAM.

Protocolos de visualización

El protocolo SPICE no es compatible con los sistemas IBM POWER. Para mostrar la salida gráfica de una VM, utilice el protocolo **VNC**. Además, sólo se admiten los siguientes dispositivos de tarjetas gráficas virtuales:

- **vga** - sólo se admite en el modo **-vga std** y no en el modo **-vga cirrus**.
- **virtio-vga**
- **virtio-gpu**

SMBIOS

La configuración de SMBIOS no está disponible.

Errores de asignación de memoria

Las máquinas virtuales POWER8, incluidas las del modo de compatibilidad, pueden fallar con un error similar al siguiente:

```
qemu-kvm: No se ha podido asignar el HPT de KVM de orden 33 (intente con un maxmem menor): No se puede asignar memoria
```

Esto es mucho más probable que ocurra en las máquinas virtuales que utilizan RHEL 7.3 y anteriores como sistema operativo invitado.

Para solucionar el problema, aumente la reserva de memoria CMA disponible para la tabla de páginas hash (HPT) del huésped añadiendo **kvm_cma_resv_ratio=memory** a la línea de comandos del kernel del huésped, donde *memory* es el porcentaje de la memoria del huésped que debe reservarse para el pool de CMA (por defecto es 5).

Páginas enormes

Las páginas enormes transparentes (THP) no proporcionan ninguna ventaja de rendimiento notable en las máquinas virtuales IBM POWER8. Sin embargo, las máquinas virtuales IBM POWER9 pueden beneficiarse de las THP como se esperaba.

Además, el tamaño de las páginas enormes estáticas en los sistemas IBM POWER8 es de 16 MiB y 16 GiB, frente a 2 MiB y 1 GiB en AMD64, Intel 64 e IBM POWER9. Como consecuencia, para migrar una VM configurada con páginas enormes estáticas desde un host IBM POWER8 a un host IBM POWER9, primero debe configurar páginas enormes de 1 GiB en la VM.

kvm-reloj

El servicio **kvm-clock** no tiene que ser configurado para la gestión del tiempo en las máquinas virtuales en IBM POWER9.

pvpanic

Los sistemas IBM POWER9 no soportan el dispositivo **pvpanic**. Sin embargo, una funcionalidad equivalente está disponible y activada por defecto en esta arquitectura. Para activarla en una VM, utilice el elemento de configuración XML **<on_crash>** con el valor **preserve**.

Además, asegúrese de eliminar el elemento **<panic>** de la sección **<devices>**, ya que su presencia puede hacer que la máquina virtual no arranque en los sistemas IBM POWER.

Host de un solo hilo

En los sistemas IBM POWER8, la máquina anfitriona debe ejecutarse en **single-threaded mode** para soportar las máquinas virtuales. Esto se configura automáticamente si se instalan los paquetes *qemu-kvm*. Sin embargo, las máquinas virtuales que se ejecutan en hosts de un solo hilo pueden seguir utilizando varios hilos.

Dispositivos periféricos

Varios dispositivos periféricos admitidos en los sistemas AMD64 e Intel 64 no son compatibles con los sistemas IBM POWER, o bien se admite un dispositivo diferente como sustitución.

- Los dispositivos utilizados para la jerarquía PCI-E, incluidos **ioh3420** y **xio3130-downstream**, no son compatibles. Esta funcionalidad se sustituye por múltiples puentes raíz PCI independientes proporcionados por el dispositivo **spapr-pci-host-bridge**.
- Los controladores PCI UHCI y EHCI no son compatibles. Utilice en su lugar las controladoras OHCI y XHCI.
- Los dispositivos IDE, incluyendo el CD-ROM virtual IDE (**ide-cd**) y el disco virtual IDE (**ide-hd**), no son compatibles. Utilice en su lugar los dispositivos **virtio-scsi** y **virtio-blk**.
- Los NIC PCI emulados (**rtl8139**) no son compatibles. Utilice el dispositivo **virtio-net** en su lugar.
- Los dispositivos de sonido, incluyendo **intel-hda**, **hda-output**, y **AC97**, no son compatibles.
- Los dispositivos de redirección USB, incluyendo **usb-redirect** y **usb-tablet**, no son compatibles.

v2v y p2v

Las utilidades **virt-v2v** y **virt-p2v** sólo son compatibles con la arquitectura AMD64 e Intel 64, y no se proporcionan en IBM POWER.

Fuentes adicionales

- Para una comparación de las características de virtualización soportadas y no soportadas en las arquitecturas de sistema soportadas por Red Hat, consulte [Sección 20.5, “Una visión general de la compatibilidad con las funciones de virtualización”](#).

CAPÍTULO 4. INTRODUCCIÓN A LA VIRTUALIZACIÓN EN IBM Z

Puede utilizar la virtualización KVM cuando utilice RHEL 8 en hardware IBM Z. Sin embargo, [habilitar el hipervisor K VM](#) en su sistema requiere pasos adicionales en comparación con la virtualización en arquitecturas AMD64 e Intel 64. Algunas características de virtualización de RHEL 8 también tienen [una funcionalidad diferente o restringida](#) en IBM Z.

Aparte de la información de las siguientes secciones, el uso de la virtualización en IBM Z funciona igual que en AMD64 e Intel 64. Por lo tanto, puede ver otra documentación de virtualización de RHEL 8 para obtener más información al utilizar la virtualización en IBM Z.

4.1. HABILITACIÓN DE LA VIRTUALIZACIÓN EN IBM Z

Para configurar un hipervisor KVM y crear máquinas virtuales (VM) en un sistema IBM Z que ejecute RHEL 8, siga las siguientes instrucciones.

Requisitos previos

- RHEL 8 está instalado y registrado en su máquina anfitriona.
- Los siguientes recursos mínimos del sistema están disponibles:
 - 6 GB de espacio libre en disco para el host, más otros 6 GB para cada máquina virtual prevista.
 - 2 GB de RAM para el host, más otros 2 GB para cada máquina virtual prevista.
- Su sistema host IBM Z utiliza una CPU z13 o posterior.
- RHEL 8 se instala en una partición lógica (LPAR). Además, la LPAR admite las funciones de virtualización de *start-interpretive execution* (SIE). Para comprobarlo, busque **sie** en su archivo **/proc/cpuinfo**.

```
# grep sie /proc/cpuinfo/
features      : esan3 zarch stfle msa ldisp eimm dfp edat etf3eh highgrps te sie
```

Procedimiento

1. Cargue el módulo del kernel KVM:

```
# modprobe kvm
```

2. Compruebe que el módulo del kernel KVM está cargado:

```
# lsmod | grep kvm
```

Si KVM se ha cargado con éxito, la salida de este comando incluye **kvm**:

3. Instale los paquetes en el módulo de virtualización:

```
# yum module install virt
```

4. Instale el paquete **virt-install**:

```
# yum install virt-install
```

5. Inicie el servicio **libvirtd**.

```
# systemctl start libvirtd
```

6. Compruebe que su sistema está preparado para ser un host de virtualización:

```
# virt-host-validate
[...]
QEMU: Checking if device /dev/kvm is accessible      : PASS
QEMU: Checking if device /dev/vhost-net exists      : PASS
QEMU: Checking if device /dev/net/tun exists        : PASS
QEMU: Checking for cgroup 'memory' controller support : PASS
QEMU: Checking for cgroup 'memory' controller mount-point : PASS
[...]
```

7. Si todas las comprobaciones de **virt-host-validate** devuelven un valor de **PASS**, su sistema está preparado para [crear máquinas virtuales](#).

Si alguna de las comprobaciones devuelve un valor de **FAIL**, siga las instrucciones mostradas para solucionar el problema.

Si alguna de las comprobaciones devuelve un valor de **WARN**, considere seguir las instrucciones mostradas para mejorar las capacidades de virtualización.

Información adicional

- Tenga en cuenta que si la virtualización no está soportada por su CPU anfitriona, **virt-host-validate** genera la siguiente salida:

```
QEMU: Comprobando la virtualización del hardware: FAIL (Sólo hay CPUs emuladas, el rendimiento será significativamente limitado)
```

Sin embargo, al intentar crear máquinas virtuales en un sistema anfitrión de este tipo, se producirá un fallo, en lugar de tener problemas de rendimiento.

4.2. EN QUÉ SE DIFERENCIA LA VIRTUALIZACIÓN EN IBM Z DE LA DE AMD64 E INTEL 64

La virtualización KVM en RHEL 8 en sistemas IBM Z difiere de KVM en sistemas AMD64 e Intel 64 en lo siguiente:

No hay salida gráfica

La visualización de la salida gráfica de la VM no es posible cuando se conecta a la VM utilizando el protocolo VNC. Esto se debe a que la utilidad **gnome-desktop** no es compatible con IBM Z.

Además, el protocolo de visualización SPICE no funciona en IBM Z.

Dispositivos PCI y USB

Los dispositivos virtuales PCI y USB no son compatibles con IBM Z. Esto también significa que **virtio-**-pci*** no están soportados, y que en su lugar se deben utilizar dispositivos **virtio-**-ccw*** deben ser utilizados en su lugar. Por ejemplo, utilice **virtio-net-ccw** en lugar de **virtio-net-pci**.

Tenga en cuenta que se admite la conexión directa de dispositivos PCI, también conocida como PCI passthrough.

Sistema operativo invitado soportado

Red Hat sólo admite máquinas virtuales alojadas en IBM Z si utilizan RHEL 7 o RHEL 8 como sistema operativo invitado.

Orden de arranque del dispositivo

IBM Z no soporta el elemento de configuración `<boot dev='device'>` Elemento de configuración XML. Para definir el orden de arranque del dispositivo, utilice el elemento `<boot order='number'>` en la sección `<devices>` del XML. Por ejemplo:

```
<disk type='file' device='disk'>
  <driver name='qemu' type='qcow2'/>
  <source file='/path/to/qcow2'/>
  <target dev='vda' bus='virtio'/>
  <address type='ccw' cssid='0xfe' ssid='0x0' devno='0x0000'/>
  <boot order='2'>
</disk>
```



NOTA

El uso de `<boot order='number'>` para la gestión del orden de arranque también es preferible en hosts AMD64 e Intel 64.

Memoria de conexión en caliente

Añadir memoria a una VM en ejecución no es posible en IBM Z. Tenga en cuenta que eliminar memoria de una VM en ejecución (*memory hot unplug*) tampoco es posible en IBM Z, así como en AMD64 e Intel 64.

Topología NUMA

La topología de acceso no uniforme a la memoria (NUMA) para las CPU no es compatible con **libvirt** en IBM Z. Por lo tanto, el ajuste del rendimiento de las vCPU utilizando NUMA no es posible en estos sistemas.

vfio-ap

Las máquinas virtuales en un host IBM Z pueden utilizar el passthrough de dispositivos criptográficos *vfio-ap*, que no está soportado en otras arquitecturas.

SMBIOS

La configuración de SMBIOS no está disponible en IBM Z.

Dispositivos de vigilancia

Si utiliza dispositivos de vigilancia en su VM en un host IBM Z, utilice el modelo **diag288**. Por ejemplo:

```
<devices>
  <watchdog model='diag288' action='poweroff'/>
</devices>
```

kvm-reloj

El servicio **kvm-clock** es específico de los sistemas AMD64 e Intel 64, y no es necesario configurarlo para la gestión del tiempo de las máquinas virtuales en IBM Z.

v2v y p2v

Las utilidades **virt-v2v** y **virt-p2v** sólo son compatibles con la arquitectura AMD64 e Intel 64, y no se proporcionan en IBM Z.

Virtualización anidada

La creación de máquinas virtuales anidadas requiere una configuración diferente en IBM Z que en AMD64 e Intel 64. Para más detalles, consulte [Capítulo 18, Creación de máquinas virtuales anidadas](#).

Fuentes adicionales

- Para una comparación de las características de virtualización soportadas y no soportadas en las arquitecturas de sistema soportadas por Red Hat, consulte [Sección 20.5, "Una visión general de la compatibilidad con las funciones de virtualización"](#).

4.3. INFORMACIÓN RELACIONADA

- Cuando se configura una VM en un sistema IBM Z, se recomienda proteger el SO invitado de la vulnerabilidad `\ "Spectre\ "`. Para ello, utilice el comando **virsh edit** para modificar la configuración XML de la VM y configurar su CPU de una de las siguientes maneras:
 - Utilice el modelo de CPU del host, por ejemplo, como se indica a continuación:

```
<cpu mode='host-model' check='partial'>
  <model fallback='allow'/>
</cpu>
```

Esto hace que las funciones **ppa15** y **bpb** estén disponibles para el huésped si el host las soporta.

- Si utiliza un modelo de host específico, añada las funciones **ppa15** y **pbp**. El siguiente ejemplo utiliza el modelo de CPU zEC12:

```
<cpu mode='custom' match='exact' check='partial'>
  <model fallback='allow'>zEC12</model>
  <feature policy='force' name='ppa15'/>
  <feature policy='force' name='bpb'/>
</cpu>
```

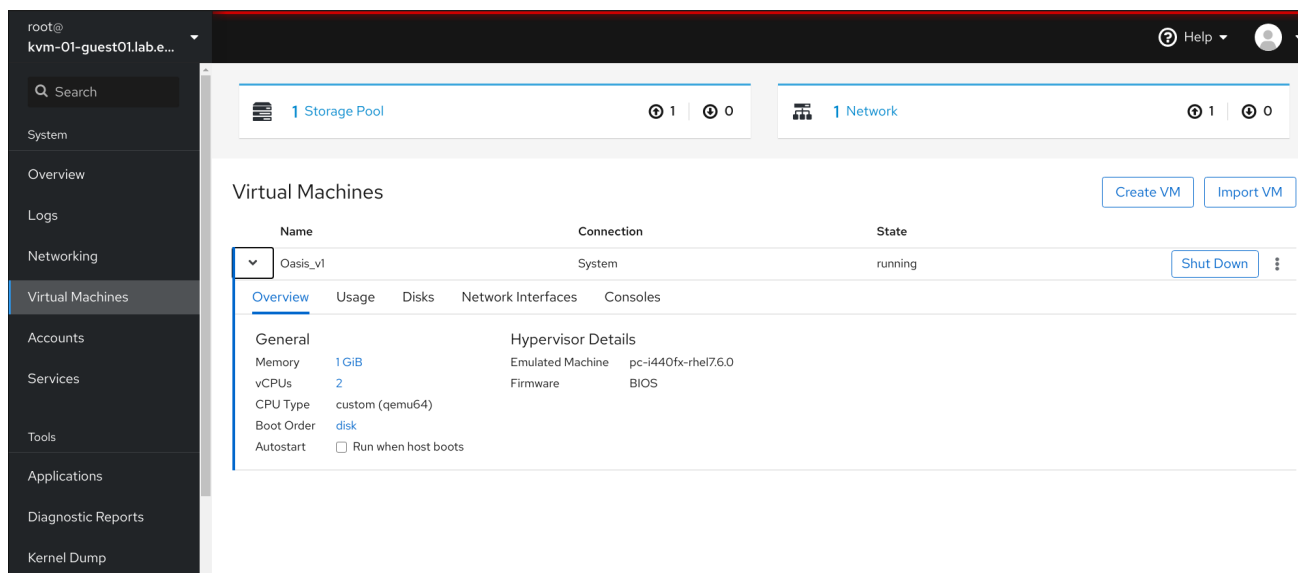
Tenga en cuenta que cuando utilice la función **ppa15** con los modelos de CPU **z114** y **z196** en una máquina anfitriona que utilice una CPU z12, también deberá utilizar el último nivel de microcódigo (paquete 95 o posterior).

- Tenga en cuenta que la ejecución de KVM en el sistema operativo z/VM no es compatible.
- Para obtener información sobre cómo adjuntar dispositivos DASD a las máquinas virtuales en los hosts IBM Z, consulte [Sección 10.9, "Adjuntar dispositivos DASD a máquinas virtuales en IBM Z"](#).
- Para obtener instrucciones sobre el uso del cifrado de hardware de IBM Z en las máquinas virtuales, consulte [Sección 15.7, "Adjuntar coprocesadores criptográficos a máquinas virtuales en IBM Z"](#).
- Para obtener instrucciones sobre la configuración de IBM Z Secure Execution para sus máquinas virtuales, consulte [Sección 15.6, "Configuración de IBM Secure Execution en IBM Z"](#).
- Para obtener información sobre la configuración de la virtualización anidada en hosts IBM Z, consulte [Sección 18.3, "Creación de una máquina virtual anidada en IBM Z"](#).

CAPÍTULO 5. GESTIÓN DE MÁQUINAS VIRTUALES EN LA CONSOLA WEB

Gestione sus máquinas virtuales en una consola web de RHEL 8 y conozca las capacidades de gestión de la virtualización.

Para gestionar las máquinas virtuales en una interfaz gráfica en un host RHEL 8, puede utilizar el panel **Virtual Machines** en la [consola web de RHEL 8](#).



5.1. VISIÓN GENERAL DE LA GESTIÓN DE MÁQUINAS VIRTUALES MEDIANTE LA CONSOLA WEB

La consola web de RHEL 8 es una interfaz basada en la web para la administración del sistema. Como una de sus características, la consola web proporciona una vista gráfica de las máquinas virtuales (VMs) en el sistema anfitrión, y hace posible crear, acceder y configurar estas VMs.

Tenga en cuenta que para utilizar la consola web para gestionar sus máquinas virtuales en RHEL 8, debe instalar primero [un complemento de la consola web](#) para la virtualización.

Próximos pasos

- Para obtener instrucciones sobre cómo habilitar la gestión de máquinas virtuales en su consola web, consulte [Configuración de la consola web para gestionar máquinas virtuales](#).
- Para obtener una lista completa de las acciones de gestión de máquinas virtuales que ofrece la consola web, consulte [Funciones de gestión de máquinas virtuales disponibles en la consola web](#).
- Para ver una lista de funciones que actualmente no están disponibles en la consola web pero que pueden utilizarse en la aplicación **virt-manager**, consulte [Diferencias entre las funciones de virtualización en el Administrador de máquinas virtuales y la consola web](#).

5.2. CONFIGURACIÓN DE LA CONSOLA WEB PARA GESTIONAR LAS MÁQUINAS VIRTUALES

Antes de utilizar la consola web de RHEL 8 para gestionar máquinas virtuales (VMs), debe instalar el plug-in de máquina virtual de la consola web en el host.

Requisitos previos

- Asegúrese de que la consola web está instalada y habilitada en su máquina.

```
# systemctl status cockpit.socket
cockpit.socket - Cockpit Web Service Socket
Loaded: loaded (/usr/lib/systemd/system/cockpit.socket
[...]
```

Si este comando devuelve **Unit cockpit.socket could not be found**, siga el documento [Instalación de la consola web](#) para habilitar la consola web.

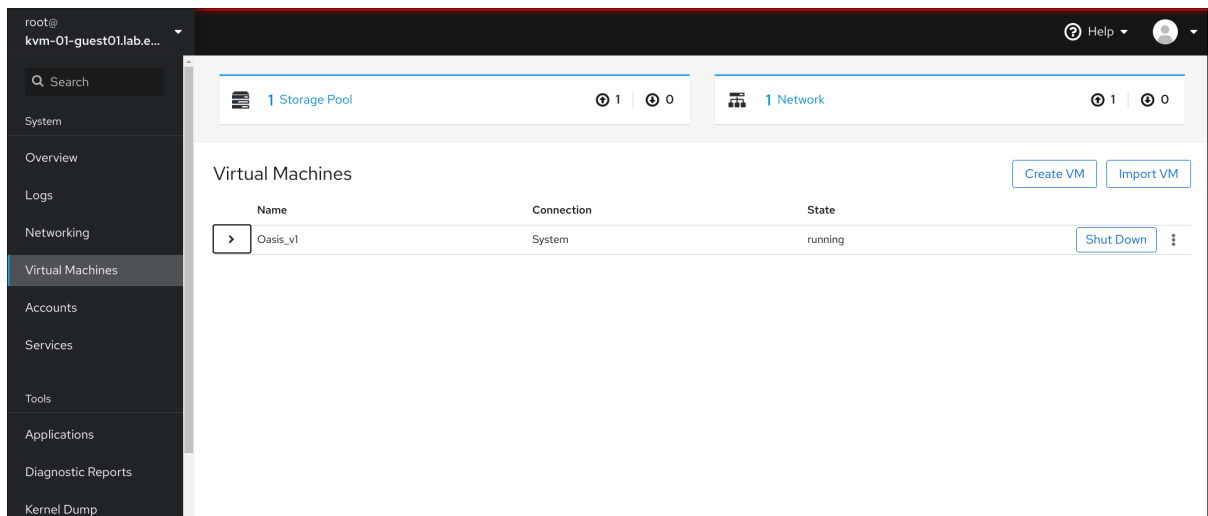
Procedimiento

- Instale el complemento **cockpit-machines**.

```
# yum install cockpit-machines
```

Verificación

- Si la instalación se realiza correctamente, las **máquinas virtuales** aparecen en el menú lateral de la consola web.



Recursos adicionales

- Para obtener instrucciones sobre la conexión a la consola web, así como otra información sobre el uso de la consola web, consulte el [Managing systems using the RHEL 8 web console](#) documento.

5.3. FUNCIONES DE GESTIÓN DE MÁQUINAS VIRTUALES DISPONIBLES EN LA CONSOLA WEB

Mediante la consola web de RHEL 8, puede realizar las siguientes acciones para gestionar las máquinas virtuales (VM) de su sistema.

Tabla 5.1. Tareas de gestión de máquinas virtuales que se pueden realizar en la consola web de RHEL 8

Tarea	Para más detalles, consulte
Crear una VM e instalarla con un sistema operativo invitado	Sección 2.2.2, "Creación de máquinas virtuales e instalación de sistemas operativos invitados mediante la consola web"
Borrar una VM	Sección 2.6.2, "Eliminación de máquinas virtuales mediante la consola web"
Iniciar, apagar y reiniciar la máquina virtual	Sección 2.3.2, "Iniciar máquinas virtuales mediante la consola web" y Sección 2.5.2, "Apagado y reinicio de máquinas virtuales mediante la consola web"
Conectarse e interactuar con una VM utilizando una variedad de consolas	Sección 2.4.1, "Interacción con las máquinas virtuales mediante la consola web"
Ver una variedad de información sobre la MV	Sección 6.2, "Visualización de la información de la máquina virtual mediante la consola web"
Ajustar la memoria del host asignada a una VM	Sección 16.3.1, "Añadir y eliminar la memoria de la máquina virtual mediante la consola web"
Gestionar las conexiones de red para la VM	Sección 13.2, "Uso de la consola web para gestionar las interfaces de red de las máquinas virtuales"
Gestionar el almacenamiento de la VM disponible en el host y adjuntar discos virtuales a la VM	Sección 11.3, "Gestión del almacenamiento de las máquinas virtuales mediante la consola web"
Configurar los ajustes de la CPU virtual de la VM	Sección 16.5.2, "Gestión de las CPUs virtuales mediante la consola web"

5.4. DIFERENCIAS ENTRE LAS FUNCIONES DE VIRTUALIZACIÓN EN VIRTUAL MACHINE MANAGER Y LA CONSOLA WEB

La aplicación Virtual Machine Manager (**virt-manager**) está soportada en RHEL 8, pero ha quedado obsoleta. La consola web está destinada a convertirse en su reemplazo en una versión mayor posterior. Por lo tanto, se recomienda que se familiarice con la consola web para gestionar la virtualización en una GUI.

Sin embargo, en RHEL 8, algunas tareas de gestión de máquinas virtuales sólo pueden realizarse en **virt-manager** o en la línea de comandos. La siguiente tabla destaca las funciones que están disponibles en **virt-manager** pero que no están disponibles en la consola web de RHEL 8.0.

Si una función está disponible en una versión menor posterior de RHEL 8, la versión mínima de RHEL 8 aparece en la columna *Support in web console introduced*.

Tabla 5.2. Tareas de gestión de máquinas virtuales que no se pueden realizar mediante la consola web en RHEL 8.0

Tarea	Soporte en la consola web introducido	Método alternativo usando CLI
Configurar una máquina virtual para que se inicie cuando el host arranque	RHEL 8.1	virsh autostart
Suspender una máquina virtual	RHEL 8.1	virsh suspend
Reanudación de una máquina virtual suspendida	RHEL 8.1	virsh resume
Creación de pools de almacenamiento de directorios del sistema de archivos	RHEL 8.1	virsh pool-define-as
Creación de pools de almacenamiento NFS	RHEL 8.1	virsh pool-define-as
Creación de grupos de almacenamiento de dispositivos de disco físico	RHEL 8.1	virsh pool-define-as
Creación de grupos de almacenamiento de volúmenes LVM	RHEL 8.1	virsh pool-define-as
Creación de pools de almacenamiento basados en particiones	<i>CURRENTLY UNAVAILABLE</i>	virsh pool-define-as
Creación de pools de almacenamiento basados en GlusterFS	<i>CURRENTLY UNAVAILABLE</i>	virsh pool-define-as
Creación de pools de almacenamiento basados en vHBA con dispositivos SCSI	<i>CURRENTLY UNAVAILABLE</i>	virsh pool-define-as
Creación de pools de almacenamiento basados en Multipath	<i>CURRENTLY UNAVAILABLE</i>	virsh pool-define-as
Creación de pools de almacenamiento basados en RBD	<i>CURRENTLY UNAVAILABLE</i>	virsh pool-define-as
Creación de un nuevo volumen de almacenamiento	RHEL 8.1	virsh vol-create

Tarea	Soporte en la consola web introducido	Método alternativo usando CLI
Añadir una nueva red virtual	RHEL 8.1	virsh net-create o virsh net-define
Borrar una red virtual	RHEL 8.1	virsh net-undefine
Creación de un puente desde la interfaz de una máquina anfitriona a una máquina virtual	<i>CURRENTLY UNAVAILABLE</i>	virsh iface-bridge
Creación de una instantánea	<i>CURRENTLY UNAVAILABLE</i>	virsh snapshot-create-as
Revertir a una instantánea	<i>CURRENTLY UNAVAILABLE</i>	virsh snapshot-revert
Borrar una instantánea	<i>CURRENTLY UNAVAILABLE</i>	virsh snapshot-delete
Clonar una máquina virtual	<i>CURRENTLY UNAVAILABLE</i>	virt-clone
Migración de una máquina virtual a otra máquina anfitriona	<i>CURRENTLY UNAVAILABLE</i>	virsh migrate

Recursos adicionales

- Para obtener información sobre el Administrador de Máquinas Virtuales, consulte [la documentación de RHEL 7](#).

CAPÍTULO 6. VER INFORMACIÓN SOBRE LAS MÁQUINAS VIRTUALES

Cuando necesite ajustar o solucionar cualquier aspecto de su despliegue de virtualización en RHEL 8, el primer paso que debe realizar normalmente es ver la información sobre el estado actual y la configuración de sus máquinas virtuales. Para ello, puede utilizar [la interfaz de línea de comandos](#) o [la consola web](#). También puede ver la información en la [configuración XML](#) de la máquina virtual.

6.1. VISUALIZACIÓN DE LA INFORMACIÓN DE LA MÁQUINA VIRTUAL MEDIANTE LA INTERFAZ DE LÍNEA DE COMANDOS

Para obtener información sobre las máquinas virtuales (VM) en su host y sus configuraciones, utilice uno o más de los siguientes comandos.

Procedimiento

- Para obtener una lista de VMs en su host:

```
# virsh list --all
Id Name          State
-----
1  testguest1     running
-  testguest2     shut off
-  testguest3     shut off
-  testguest4     shut off
```

- Para obtener información básica sobre una máquina virtual específica:

```
# virsh dominfo testguest1
Id:          1
Name:        testguest1
UUID:        a973666f-2f6e-415a-8949-75a7a98569e1
OS Type:     hvm
State:       running
CPU(s):      2
CPU time:    188.3s
Max memory:  4194304 KiB
Used memory: 4194304 KiB
Persistent:  yes
Autostart:   disable
Managed save: no
Security model: selinux
Security DOI: 0
Security label: system_u:system_r:svirt_t:s0:c486,c538 (enforcing)
```

- Para obtener la configuración XML completa de una VM específica:

```
# virsh dumpxml testguest2

<domain type='kvm' id='1'>
  <name>testguest2</name>
```

```
<uuid>a973434f-2f6e-4ěša-8949-76a7a98569e1</uuid>
<metadata>
[...]
```

- Para obtener información sobre los discos y otros dispositivos de bloque de una VM:

```
# virsh domblklist testguest3
Target Source
-----
vda    /var/lib/libvirt/images/testguest3.qcow2
sda    -
sdb    /home/username/Downloads/virt-p2v-1.36.10-1.el7.iso
```

Para obtener instrucciones sobre la gestión del almacenamiento de una máquina virtual, consulte [Capítulo 11, Gestión del almacenamiento de las máquinas virtuales](#) .

- Para obtener información sobre los sistemas de archivos de una VM y sus puntos de montaje:

```
# virsh domfsinfo testguest3
Mountpoint Name Type Target
-----
/          dm-0 xfs
/boot     vda1 xfs
```

- Para obtener más detalles sobre las vCPUs de una VM específica:

```
# virsh vcpuinfo testguest4
VCPU:      0
CPU:       3
State:     running
CPU time:  103.1s
CPU Affinity: yyyy

VCPU:      1
CPU:       0
State:     running
CPU time:  88.6s
CPU Affinity: yyyy
```

Para configurar y optimizar las vCPUs de su VM, consulte [Sección 16.5, "Optimización del rendimiento de la CPU de la máquina virtual"](#) .

- Para listar todas las interfaces de red virtuales de su host:

```
# virsh net-list --all
Name      State Autostart Persistent
-----
default   active yes      yes
labnet    active yes      yes
```

Para obtener información sobre una interfaz específica:

```
# virsh net-info default
Name:      default
```

```

UUID:      c699f9f6-9202-4ca8-91d0-6b8cb9024116
Active:    yes
Persistent: yes
Autostart: yes
Bridge:    virbr0

```

Para obtener más detalles sobre las interfaces de red, las redes VM y las instrucciones para configurarlas, consulte [Capítulo 13, Configuración de las conexiones de red de las máquinas virtuales](#).

- Para obtener instrucciones sobre cómo ver la información sobre los grupos de almacenamiento y los volúmenes de almacenamiento en el host, consulte [Sección 11.2.1, "Visualización de la información de almacenamiento de las máquinas virtuales mediante la CLI"](#).

6.2. VISUALIZACIÓN DE LA INFORMACIÓN DE LA MÁQUINA VIRTUAL MEDIANTE LA CONSOLA WEB

Utilizando la consola web de RHEL 8, puede ver información sobre el almacenamiento virtual y las máquinas virtuales a las que está conectada la consola web.

6.2.1. Ver un resumen de la virtualización en la consola web

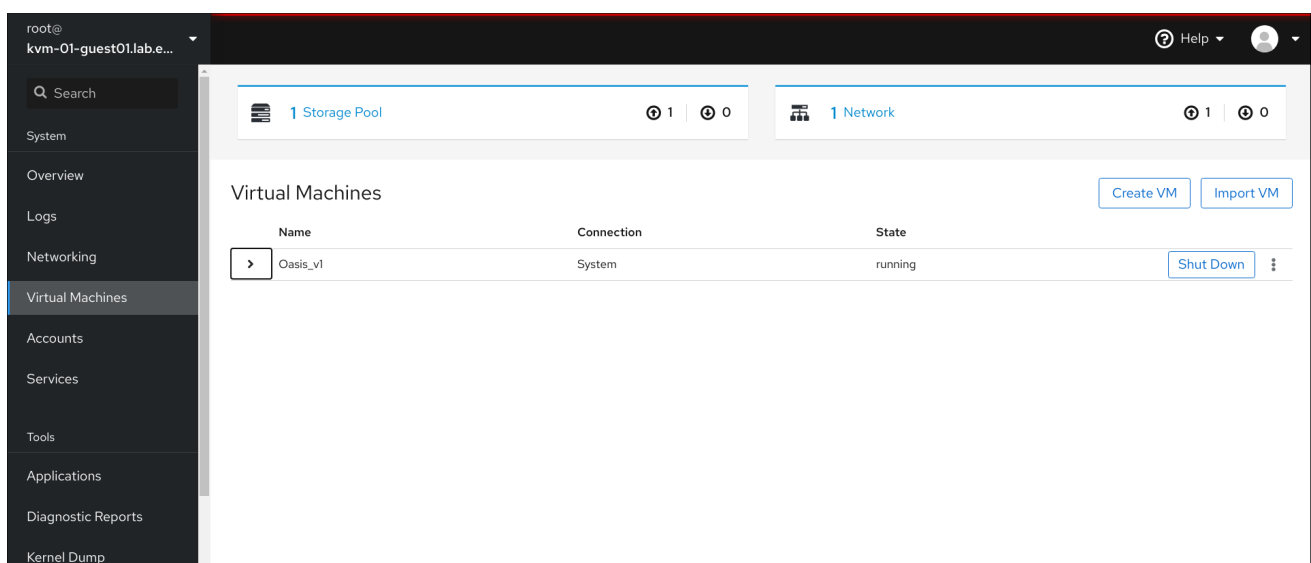
El siguiente procedimiento describe cómo ver un resumen de las máquinas virtuales (VM) y el almacenamiento virtual disponible al que está conectada la sesión de la consola web.

Requisitos previos

- Para utilizar la consola web para gestionar las máquinas virtuales, [instale el complemento VM de la consola web](#).

Procedimiento

- Haga clic en **Máquinas Virtuales** en el menú lateral de la consola web. Aparece un cuadro de diálogo con información sobre el almacenamiento disponible y las máquinas virtuales a las que está conectada la consola web.



La información incluye lo siguiente:

- **Storage Pools** - El número de grupos de almacenamiento a los que puede acceder la consola web y su estado.
- **Networks** - El número de redes a las que puede acceder la consola web y su estado.
- **Name** - El nombre de la máquina virtual.
- **Connection** - El tipo de conexión libvirt, sistema o sesión.
- **State** - El estado de la máquina virtual.

Recursos adicionales

- Si desea obtener instrucciones para ver información detallada sobre los grupos de almacenamiento a los que puede acceder la sesión de la consola web, consulte [Sección 6.2.2, "Visualización de la información del pool de almacenamiento mediante la consola web"](#).
- Para obtener instrucciones sobre cómo ver la información básica de una VM seleccionada a la que está conectada la sesión de la consola web, consulte [Sección 6.2.3, "Visualización de la información básica de la máquina virtual en la consola web"](#).
- Para obtener instrucciones sobre cómo ver el uso de recursos de una VM seleccionada a la que está conectada la sesión de la consola web, consulte [Sección 6.2.4, "Ver el uso de recursos de la máquina virtual en la consola web"](#).
- Para obtener instrucciones sobre cómo ver la información del disco de una VM seleccionada a la que está conectada la sesión de la consola web, consulte [Sección 6.2.5, "Visualización de la información del disco de la máquina virtual en la consola web"](#).
- Para obtener instrucciones sobre cómo ver la información de la interfaz de red virtual sobre una VM seleccionada a la que está conectada la sesión de la consola web, consulte [Sección 6.2.6, "Ver y editar la información de la interfaz de red virtual en la consola web"](#).

6.2.2. Visualización de la información del pool de almacenamiento mediante la consola web

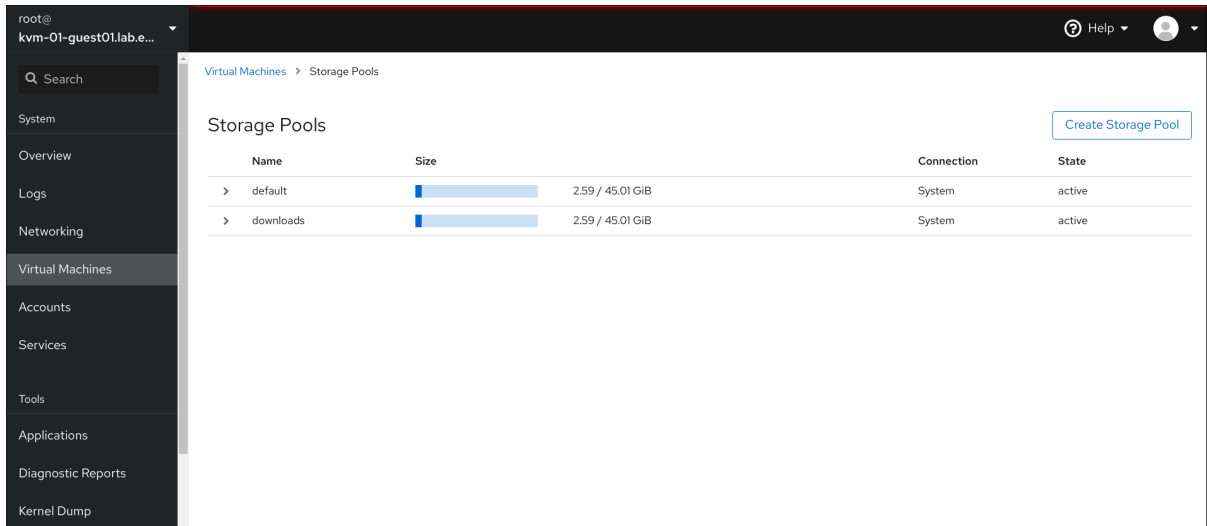
El siguiente procedimiento describe cómo ver información detallada sobre los pools de almacenamiento de las máquinas virtuales (VM) a los que puede acceder la sesión de la consola web.

Requisitos previos

- Para utilizar la consola web para gestionar las máquinas virtuales, [instale el complemento VM de la consola web](#).

Procedimiento

1. Haga clic en **Storage Pools** en la parte superior de la interfaz de **Máquinas Virtuales**. Aparece la ventana Storage Pools, que muestra una lista de pools de almacenamiento configurados.



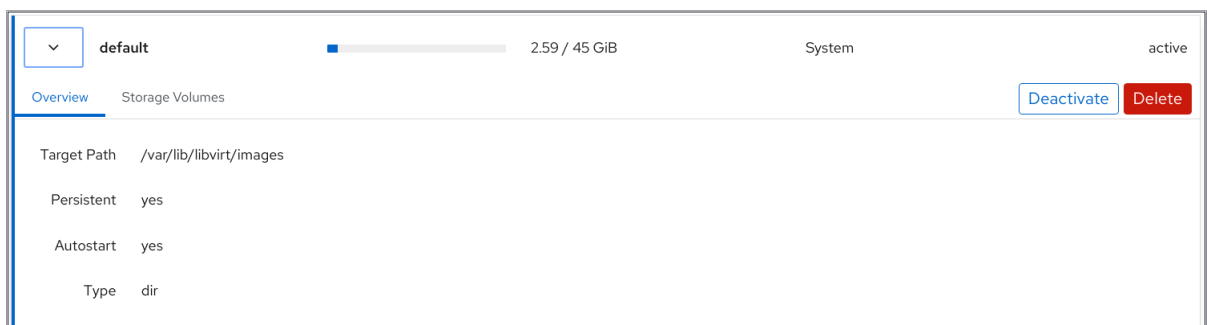
La información incluye lo siguiente:

- **Name** - El nombre del pool de almacenamiento.
- **Size** - El tamaño del pool de almacenamiento.
- **Connection** - La conexión utilizada para acceder al pool de almacenamiento.
- **State** - El estado del pool de almacenamiento.

2. Haga clic en la fila del almacén cuya información desea ver.

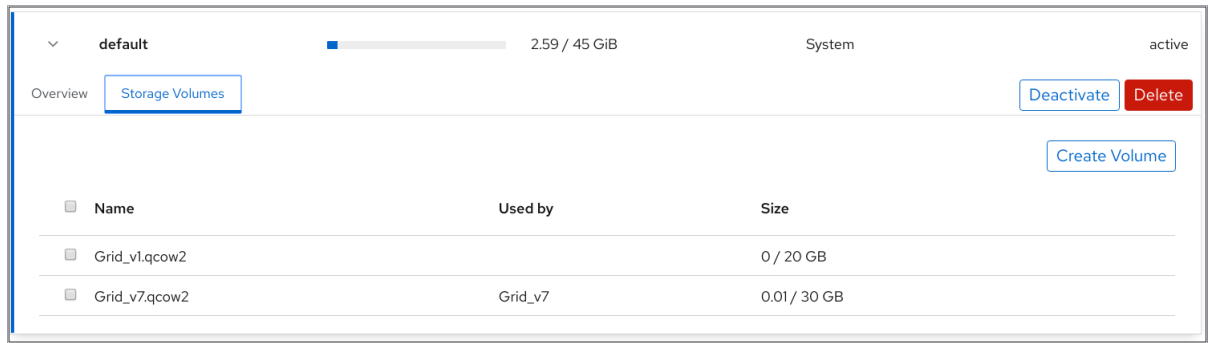
La fila se expande para revelar el panel de Visión General con la siguiente información sobre el pool de almacenamiento seleccionado:

- **Path** - La ruta de acceso al pool de almacenamiento.
- **Persistent** - Si el pool de almacenamiento es persistente o no.
- **Autostart** - Si el pool de almacenamiento se inicia automáticamente o no.
- **Type** - El tipo de pool de almacenamiento.



3. Para ver una lista de los volúmenes de almacenamiento creados a partir del pool de almacenamiento, haga clic en **Volúmenes de almacenamiento**.

Aparece el panel de volúmenes de almacenamiento, que muestra una lista de volúmenes de almacenamiento configurados con sus tamaños y la cantidad de espacio utilizado.



Recursos adicionales

- Para obtener instrucciones sobre cómo ver la información de todas las máquinas virtuales a las que está conectada la sesión de la consola web, consulte [Sección 6.2.1, “Ver un resumen de la virtualización en la consola web”](#).
- Para obtener instrucciones sobre cómo ver la información básica de una VM seleccionada a la que está conectada la sesión de la consola web, consulte [Sección 6.2.3, “Visualización de la información básica de la máquina virtual en la consola web”](#).
- Para obtener instrucciones sobre cómo ver el uso de recursos de una VM seleccionada a la que está conectada la sesión de la consola web, consulte [Sección 6.2.4, “Ver el uso de recursos de la máquina virtual en la consola web”](#).
- Para obtener instrucciones sobre cómo ver la información del disco de una VM seleccionada a la que está conectada la sesión de la consola web, consulte [Sección 6.2.5, “Visualización de la información del disco de la máquina virtual en la consola web”](#).
- Para obtener instrucciones sobre cómo ver la información de la interfaz de red virtual sobre una VM seleccionada a la que está conectada la sesión de la consola web, consulte [Sección 6.2.6, “Ver y editar la información de la interfaz de red virtual en la consola web”](#).

6.2.3. Visualización de la información básica de la máquina virtual en la consola web

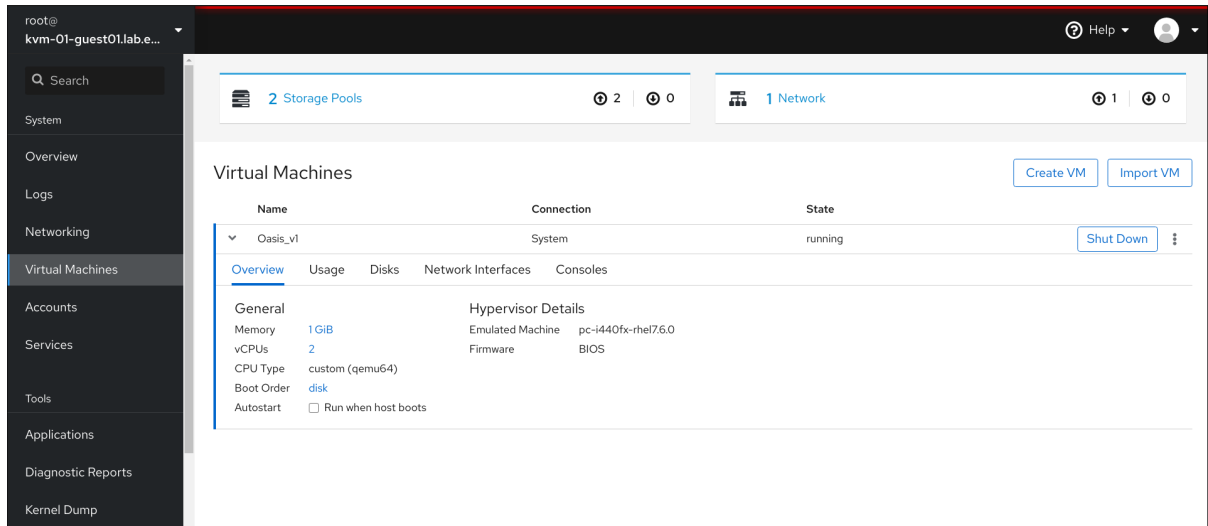
A continuación se describe cómo ver la información básica de una máquina virtual (VM) seleccionada a la que está conectada la sesión de la consola web.

Requisitos previos

- Para utilizar la consola web para gestionar las máquinas virtuales, [instale el complemento VM de la consola web](#).

Procedimiento

1. Haga clic en **Máquinas Virtuales** en el menú lateral de la consola web.
2. Haga clic en la fila de la máquina virtual cuya información desea ver.
La fila se expande para revelar el panel de Visión General con información básica sobre la VM seleccionada y controles para apagar y eliminar la VM.
3. Si se selecciona otra pestaña, haga clic en **Resumen**.



La información incluye los siguientes datos generales de VM:

- **Memory** - La cantidad de memoria asignada a la VM.
- **vCPUs** - El número de CPUs virtuales configuradas para la VM.
- **CPU Type** - La arquitectura de las CPUs virtuales configuradas para la VM.
- **Boot Order** - El orden de arranque configurado para la VM.
- **Autostart** - Si el autoarranque está activado o no para la máquina virtual.

La información también incluye los siguientes detalles del hipervisor:

- **Emulated Machine** - El tipo de máquina emulada por la VM.
- **Firmware** - El firmware de la máquina virtual.

Recursos adicionales

- Para obtener instrucciones sobre cómo ver la información de todas las máquinas virtuales a las que está conectada la sesión de la consola web, consulte [Sección 6.2.1, "Ver un resumen de la virtualización en la consola web"](#).
- Para obtener instrucciones sobre cómo ver la información de los grupos de almacenamiento a los que está conectada la sesión de la consola web, consulte [Sección 6.2.2, "Visualización de la información del pool de almacenamiento mediante la consola web"](#).
- Para obtener instrucciones sobre cómo ver el uso de recursos de una VM seleccionada a la que está conectada la sesión de la consola web, consulte [Sección 6.2.4, "Ver el uso de recursos de la máquina virtual en la consola web"](#).
- Para obtener instrucciones sobre cómo ver la información del disco de una VM seleccionada a la que está conectada la sesión de la consola web, consulte [Sección 6.2.5, "Visualización de la información del disco de la máquina virtual en la consola web"](#).
- Para obtener instrucciones sobre cómo ver la información de la interfaz de red virtual sobre una VM seleccionada a la que está conectada la sesión de la consola web, consulte [Sección 6.2.6, "Ver y editar la información de la interfaz de red virtual en la consola web"](#).

- Para ver información más detallada sobre las CPUs virtuales y configurar las CPUs virtuales configuradas para una VM, consulte [Sección 16.5.2, “Gestión de las CPUs virtuales mediante la consola web”](#).

6.2.4. Ver el uso de recursos de la máquina virtual en la consola web

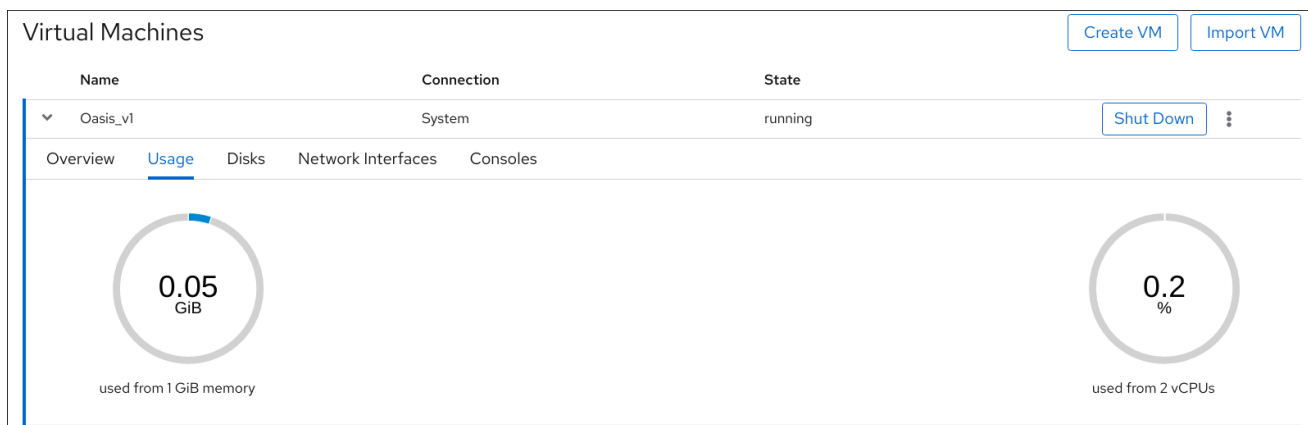
El siguiente procedimiento describe cómo ver la información de uso de la memoria y la CPU virtual de una máquina virtual (VM) seleccionada a la que está conectada la sesión de la consola web.

Requisitos previos

- Para utilizar la consola web para gestionar las máquinas virtuales, [instale el complemento VM de la consola web](#).

Procedimiento

1. En la interfaz de **máquinas virtuales**, haga clic en la fila de la máquina virtual cuya información desea ver.
La fila se expande para revelar el panel de Visión General con información básica sobre la VM seleccionada y controles para apagar y eliminar la VM.
2. Haga clic en **Uso**.
Aparece el panel de Uso con información sobre el uso de la memoria y la CPU virtual de la VM.



Recursos adicionales

- Para obtener instrucciones sobre cómo ver la información de todas las máquinas virtuales a las que está conectada la sesión de la consola web, consulte [Sección 6.2.1, “Ver un resumen de la virtualización en la consola web”](#).
- Para obtener instrucciones sobre cómo ver la información de los grupos de almacenamiento a los que está conectada la sesión de la consola web, consulte [Sección 6.2.2, “Visualización de la información del pool de almacenamiento mediante la consola web”](#).
- Para obtener instrucciones sobre cómo ver la información básica de una VM seleccionada a la que está conectada la sesión de la consola web, consulte [Sección 6.2.3, “Visualización de la información básica de la máquina virtual en la consola web”](#).
- Para obtener instrucciones sobre cómo ver la información del disco de una VM seleccionada a la que está conectada la sesión de la consola web, consulte [Sección 6.2.5, “Visualización de la información del disco de la máquina virtual en la consola web”](#).

- Para obtener instrucciones sobre cómo ver la información de la interfaz de red virtual sobre una VM seleccionada a la que está conectada la sesión de la consola web, consulte [Sección 6.2.6, “Ver y editar la información de la interfaz de red virtual en la consola web”](#).

6.2.5. Visualización de la información del disco de la máquina virtual en la consola web

El siguiente procedimiento describe cómo ver la información del disco de una máquina virtual (VM) a la que está conectada la sesión de la consola web.

Requisitos previos

Para utilizar la consola web para gestionar las máquinas virtuales, [instale el complemento VM de la consola web](#).

Procedimiento

1. Haga clic en la fila de la máquina virtual cuya información desea ver.
La fila se expande para revelar el panel de Visión General con información básica sobre la VM seleccionada y controles para apagar y eliminar la VM.
2. Haga clic en **Discos**.
Aparece el panel de Discos con información sobre los discos asignados a la VM.

The screenshot shows the 'Virtual Machines' management interface. At the top, there are buttons for 'Create VM' and 'Import VM'. Below that is a table of VMs with columns for Name, Connection, and State. The VM 'Oasis_v1' is selected, and its details are expanded. The 'Disks' tab is active, showing a table of disks with columns for Device, Used, Capacity, Bus, Access, and Source. A single disk is listed with 0.00 GiB used and 9 GiB capacity. There are 'Add Disk', 'Remove', and 'Edit' buttons for the disk.

Name	Connection	State
Oasis_v1	System	running

Device	Used	Capacity	Bus	Access	Source
disk	0.00 GiB	9 GiB	virtio	Writeable	File /var/lib/libvirt/images/Oasis_v1.qcow2

La información incluye lo siguiente:

- **Device** - El tipo de dispositivo del disco.
- **Used** - La cantidad de disco que se utiliza.
- **Capacity** - El tamaño del disco.
- **Bus** - El tipo de bus del disco.
- **Access** - Si el disco es de escritura o de sólo lectura.
- **Source** - El dispositivo o archivo de disco.

Recursos adicionales

- Para obtener instrucciones sobre cómo ver la información de todas las máquinas virtuales a las que está conectada la sesión de la consola web, consulte [Sección 6.2.1, “Ver un resumen de la virtualización en la consola web”](#).

- Para obtener instrucciones sobre cómo ver la información de los grupos de almacenamiento a los que está conectada la sesión de la consola web, consulte [Sección 6.2.2, “Visualización de la información del pool de almacenamiento mediante la consola web”](#).
- Para obtener instrucciones sobre cómo ver la información básica de una VM seleccionada a la que está conectada la sesión de la consola web, consulte [Sección 6.2.3, “Visualización de la información básica de la máquina virtual en la consola web”](#).
- Para obtener instrucciones sobre cómo ver el uso de recursos de una VM seleccionada a la que está conectada la sesión de la consola web, consulte [Sección 6.2.4, “Ver el uso de recursos de la máquina virtual en la consola web”](#).
- Para obtener instrucciones sobre cómo ver la información de la interfaz de red virtual sobre una VM seleccionada a la que está conectada la sesión de la consola web, consulte [Sección 6.2.6, “Ver y editar la información de la interfaz de red virtual en la consola web”](#).

6.2.6. Ver y editar la información de la interfaz de red virtual en la consola web

Utilizando la consola web de RHEL 8, puede ver y modificar las interfaces de red virtuales en una máquina virtual (VM) seleccionada:

Requisitos previos

- Para utilizar la consola web para gestionar las máquinas virtuales, [instale el complemento VM de la consola web](#).

Procedimiento

1. En la interfaz de **máquinas virtuales**, haga clic en la fila de la máquina virtual cuya información desea ver.

La fila se expande para revelar el panel de Visión General con información básica sobre la VM seleccionada y controles para apagar y eliminar la VM.

2. Haga clic en **Interfaces de red**.

Aparece el panel Interfaces de red con información sobre la interfaz de red virtual configurada para la VM.

The screenshot shows the 'Virtual Machines' management interface. At the top right, there are buttons for 'Create VM' and 'Import VM'. Below this is a table of virtual machines. The first row is expanded to show details for 'Oasis_v1', which is in a 'running' state. Underneath, there are tabs for 'Overview', 'Usage', 'Disks', 'Network Interfaces', and 'Consoles'. The 'Network Interfaces' tab is active, showing a table with columns: Type, Model type, MAC Address, IP Address, Source, and State. One interface is listed with Type 'network', Model type 'virtio', MAC Address '52:54:00:c0:3e:f5', IP Address 'Unknown', Source 'default', and State 'up'. To the right of this table is an 'Add Network Interface' button. Below the table are 'Unplug', 'Edit', and 'Delete' buttons. A 'Shut Down' button is also visible in the top right of the VM details panel.

La información incluye lo siguiente:

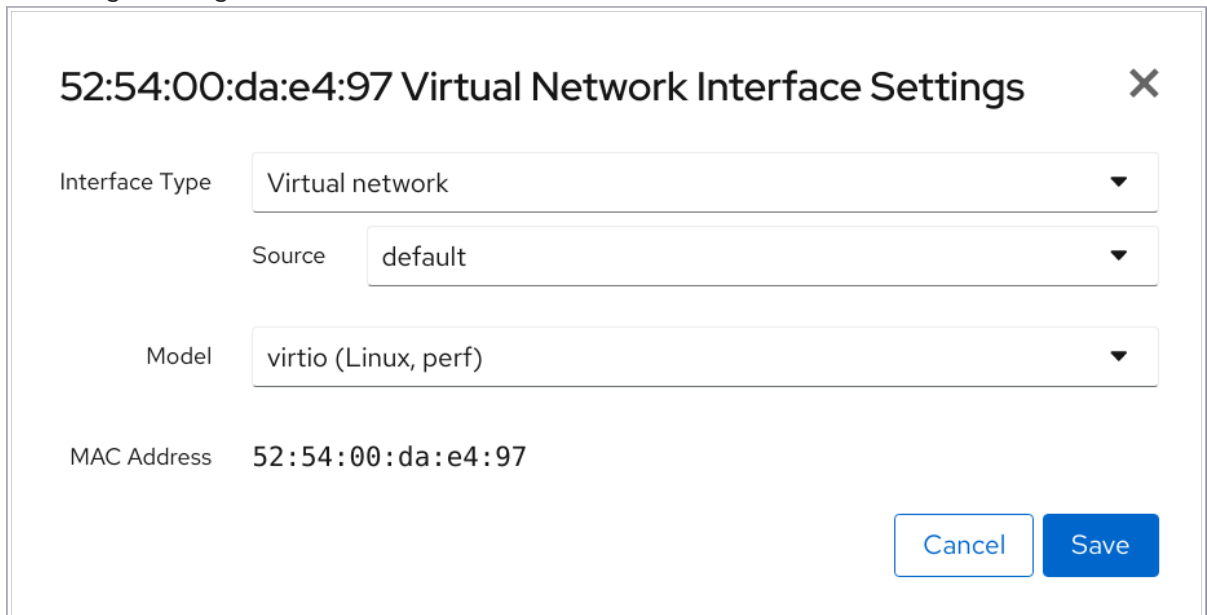
- **Type** - El tipo de interfaz de red para la VM. Los tipos incluyen red virtual, puente a LAN y conexión directa.



NOTA

La conexión Ethernet genérica no es compatible con RHEL 8.2.

- **Model type** - El modelo de la interfaz de red virtual.
 - **MAC Address** - La dirección MAC de la interfaz de red virtual.
 - **IP Address** - La dirección IP de la interfaz de red virtual.
 - **Source** - El origen de la interfaz de red. Depende del tipo de red.
 - **State** - El estado de la interfaz de red virtual.
3. Para editar la configuración de la interfaz de red virtual, haga clic en **Editar**. Se abre el cuadro de diálogo Configuración de la interfaz de red virtual.



52:54:00:da:e4:97 Virtual Network Interface Settings ✕

Interface Type: Virtual network ▼

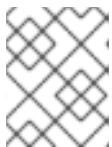
Source: default ▼

Model: virtio (Linux, perf) ▼

MAC Address: 52:54:00:da:e4:97

Cancel Save

4. Cambiar el tipo de interfaz, la fuente o el modelo.
5. Haga clic en **Guardar**. La interfaz de red se modifica.



NOTA

Los cambios en la configuración de la interfaz de red virtual sólo tienen efecto después de reiniciar la VM.

Recursos adicionales

- Para obtener instrucciones sobre cómo ver la información de todas las máquinas virtuales a las que está conectada la sesión de la consola web, consulte [Sección 6.2.1, "Ver un resumen de la virtualización en la consola web"](#).
- Para obtener instrucciones sobre cómo ver la información de los grupos de almacenamiento a los que está conectada la sesión de la consola web, consulte [Sección 6.2.2, "Visualización de la información del pool de almacenamiento mediante la consola web"](#).
- Para obtener instrucciones sobre cómo ver la información básica de una VM seleccionada a la que está conectada la sesión de la consola web, consulte [Sección 6.2.3, "Visualización de la información básica de la máquina virtual en la consola web"](#).
- Para obtener instrucciones sobre cómo ver el uso de recursos de una VM seleccionada a la que está conectada la sesión de la consola web, consulte [Sección 6.2.4, "Ver el uso de recursos de la máquina virtual en la consola web"](#).

- Para obtener instrucciones sobre cómo ver la información del disco de una VM seleccionada a la que está conectada la sesión de la consola web, consulte [Sección 6.2.5, “Visualización de la información del disco de la máquina virtual en la consola web”](#).

6.3. EJEMPLO DE CONFIGURACIÓN XML DE UNA MÁQUINA VIRTUAL

La configuración XML de una VM, también denominada *domain XML*, determina los ajustes y componentes de la VM. La siguiente tabla muestra secciones de un ejemplo de configuración XML de una máquina virtual (VM) y explica su contenido.

Para obtener la configuración XML de una VM, puede utilizar el comando **virsh dumpxml** seguido del nombre de la VM.

```
# virsh dumpxml testquest1
```

Tabla 6.1. Ejemplo de configuración XML

Sección XML del dominio	Descripción
<pre><domain type='kvm'> <name>Testquest1</name> <uuid>ec6fbaa1-3eb4-49da-bf61-bb02fbec4967</uuid> <memory unit='KiB'>1048576</memory> <currentMemory unit='KiB'>1048576</currentMemory></pre>	<p>Esta es una máquina virtual KVM llamada <i>Testquest1</i>, con 1024 MiB de RAM asignados.</p>
<pre><vcpu placement='static'>1</vcpu></pre>	<p>La VM se asigna con una sola CPU virtual (vCPU).</p> <p>Para obtener información sobre la configuración de vCPUs, consulte Sección 16.5, “Optimización del rendimiento de la CPU de la máquina virtual”.</p>
<pre><os> <type arch='x86_64' machine='pc-q35-4.1'>hvm</type> <boot dev='hd'> </os></pre>	<p>La arquitectura de la máquina está configurada para la arquitectura AMD64 e Intel 64, y utiliza el tipo de máquina Intel Q35 para determinar la compatibilidad de las características. El sistema operativo está configurado para arrancar desde el disco duro.</p> <p>Para obtener información sobre la creación de una máquina virtual con un sistema operativo instalado, consulte Sección 2.2.2, “Creación de máquinas virtuales e instalación de sistemas operativos invitados mediante la consola web”.</p>

Sección XML del dominio	Descripción
<pre data-bbox="164 241 478 436"><features> <acpi/> <apic/> <vmport state='off'/> </features></pre>	<p data-bbox="1015 219 1422 320">Las funciones del hipervisor acpi y apic están desactivadas y el puerto IO de VMWare está desactivado.</p>
<pre data-bbox="164 593 726 627"><cpu mode='host-model' check='partial'></pre>	<p data-bbox="1015 555 1433 981">Las definiciones de la CPU del host desde el XML de capacidades (que se puede obtener con virsh capabilities) se copian automáticamente en la configuración XML de la VM. Por lo tanto, cuando la VM se inicia, libvirt elige un modelo de CPU que es similar a la CPU del host, y luego añade características adicionales para aproximarse al modelo del host lo más posible.</p>
<pre data-bbox="164 1070 710 1243"><clock offset='utc'> <timer name='rtc' tickpolicy='catchup'/> <timer name='pit' tickpolicy='delay'/> <timer name='hpet' present='no'/> </clock></pre>	<p data-bbox="1015 1034 1406 1243">El reloj del hardware virtual de la VM utiliza la zona horaria UTC. Además, se configuran tres temporizadores diferentes para la sincronización con el hipervisor QEMU.</p>
<pre data-bbox="164 1352 694 1456"><on_poweroff>destroy</on_poweroff> <on_reboot>restart</on_reboot> <on_crash>destroy</on_crash></pre>	<p data-bbox="1015 1317 1412 1525">Cuando la VM se apaga, o su SO termina inesperadamente, libvirt termina la VM y libera todos sus recursos asignados. Cuando la VM se reinicia, libvirt la reinicia con la misma configuración.</p>
<pre data-bbox="164 1615 646 1758"><pm> <suspend-to-mem enabled='no'/> <suspend-to-disk enabled='no'/> </pm></pre>	<p data-bbox="1015 1579 1428 1680">Los estados de suspensión ACPI S3 y S4 están desactivados para esta máquina virtual.</p>

Sección XML del dominio	Descripción
<pre> <devices> <emulator>/usr/bin/qemu-kvm</emulator> <disk type='file' device='disk'> <driver name='qemu' type='qcow2'/> <source file='/var/lib/libvirt/images/Testguest.qcow2'/> <target dev='hda' bus='ide'/> <address type='drive' controller='0' bus='0' target='0' unit='0'/> </disk> <disk type='file' device='cdrom'> <driver name='qemu' type='raw'/> <target dev='hdb' bus='ide'/> <readonly/> <address type='drive' controller='0' bus='0' target='0' unit='1'/> </disk> </pre>	<p>La VM utiliza el archivo binario /usr/bin/qemu-kvm para la emulación. Además, tiene dos discos conectados. El primer disco es un disco duro virtualizado basado en el /var/lib/libvirt/images/Testguest.qcow2 almacenado en el host, y su nombre de dispositivo lógico está configurado como hda.</p>
<pre> <controller type='usb' index='0' model='qemu-xhci' ports='15'> <address type='pci' domain='0x0000' bus='0x02' slot='0x00' function='0x0'/> </controller> <controller type='sata' index='0'> <address type='pci' domain='0x0000' bus='0x00' slot='0x1f' function='0x2'/> </controller> <controller type='pci' index='0' model='pcie-root'/> <controller type='pci' index='1' model='pcie-root-port'> <model name='pcie-root-port'/> <target chassis='1' port='0x10'/> <address type='pci' domain='0x0000' bus='0x00' slot='0x02' function='0x0' multifunction='on'/> </controller> <controller type='pci' index='2' model='pcie-root-port'> <model name='pcie-root-port'/> <target chassis='2' port='0x11'/> <address type='pci' domain='0x0000' bus='0x00' slot='0x02' function='0x1'/> </controller> <controller type='pci' index='3' model='pcie-root-port'> <model name='pcie-root-port'/> <target chassis='3' port='0x12'/> <address type='pci' domain='0x0000' bus='0x00' slot='0x02' function='0x2'/> </controller> <controller type='pci' index='4' model='pcie-root-port'> <model name='pcie-root-port'/> <target chassis='4' port='0x13'/> </pre>	<p>La VM utiliza un único controlador para conectar dispositivos USB, y un controlador raíz para dispositivos PCI-Express (PCIe). Además, se dispone de un controlador virtio-serial, que permite a la VM interactuar con el host de diversas formas, como la consola serie.</p> <p>Para más información sobre los dispositivos virtuales, consulte Sección 10.5, "Tipos de dispositivos virtuales".</p>

Sección XML del dominio	Descripción
<pre> <address type='pci' domain='0x0000' bus='0x00' slot='0x02' function='0x3'/> </controller> <controller type='pci' index='5' model='pcie-root-port'> <model name='pcie-root-port'/> <target chassis='5' port='0x14'/> <address type='pci' domain='0x0000' bus='0x00' slot='0x02' function='0x4'/> </controller> <controller type='pci' index='6' model='pcie-root-port'> <model name='pcie-root-port'/> <target chassis='6' port='0x15'/> <address type='pci' domain='0x0000' bus='0x00' slot='0x02' function='0x5'/> </controller> <controller type='pci' index='7' model='pcie-root-port'> <model name='pcie-root-port'/> <target chassis='7' port='0x16'/> <address type='pci' domain='0x0000' bus='0x00' slot='0x02' function='0x6'/> </controller> <controller type='virtio-serial' index='0'> <address type='pci' domain='0x0000' bus='0x03' slot='0x00' function='0x0'/> </controller> </pre>	
<pre> <interface type='network'> <mac address='52:54:00:65:29:21'/> <source network='default'/> <model type='rtl8139'/> <address type='pci' domain='0x0000' bus='0x00' slot='0x03' function='0x0'/> </interface> </pre>	<p>Se configura una interfaz de red en la VM que utiliza la red virtual default y el modelo de dispositivo de red rtl8139.</p> <p>Para obtener información sobre la configuración de la interfaz de red, consulte Sección 16.6, “Optimización del rendimiento de la red de máquinas virtuales”.</p>
<pre> <serial type='pty'> <target type='isa-serial' port='0'> <model name='isa-serial'/> </target> </serial> <console type='pty'> <target type='serial' port='0'/> </console> <channel type='unix'> <target type='virtio' name='org.qemu.guest_agent.0'/> <address type='virtio-serial' controller='0' bus='0' port='1'/> </channel> <channel type='spicevmc'> <target type='virtio' name='com.redhat.spice.0'/> <address type='virtio-serial' controller='0' bus='0' port='2'/> </channel> </pre>	<p>Una consola serial pty se configura en la VM, lo que permite una comunicación rudimentaria de la VM con el host. La consola utiliza el canal UNIX en el puerto 1, y el paravirtualizado SPICE en el puerto 2. Esto se configura automáticamente y no se recomienda cambiar esta configuración.</p> <p>Para obtener más información sobre la interacción con las máquinas virtuales, consulte Sección 2.4.1, “Interacción con las máquinas virtuales mediante la consola web”.</p>

Sección XML del dominio	Descripción
<pre data-bbox="220 259 703 434"><input type='tablet' bus='usb'> <address type='usb' bus='0' port='1'/> </input> <input type='mouse' bus='ps2'/> <input type='keyboard' bus='ps2'/></pre>	<p data-bbox="1015 224 1422 501">La máquina virtual utiliza un puerto virtual usb, que está configurado para recibir la entrada de la tableta, y un puerto virtual ps2 configurado para recibir la entrada del ratón y del teclado. Esto se configura automáticamente y no se recomienda cambiar estos ajustes.</p>
<pre data-bbox="220 595 919 873"><graphics type='spice' autoport='yes' listen='127.0.0.1'> <listen type='address' address='127.0.0.1'/> <image compression='off'/> </graphics> <graphics type='vnc' port='-1' autoport='yes' listen='127.0.0.1'> <listen type='address' address='127.0.0.1'/> </graphics></pre>	<p data-bbox="1015 560 1374 725">La máquina virtual utiliza los protocolos vnc y SPICE para renderizar su salida gráfica, y la compresión de imágenes está desactivada.</p>
<pre data-bbox="220 985 836 1330"><sound model='ich6'> <address type='pci' domain='0x0000' bus='0x00' slot='0x04' function='0x0'/> </sound> <video> <model type='qxl' ram='65536' vram='65536' vgamem='16384' heads='1' primary='yes'/> <address type='pci' domain='0x0000' bus='0x00' slot='0x02' function='0x0'/> </video></pre>	<p data-bbox="1015 949 1430 1227">Un dispositivo de sonido ICH6 HDA está configurado para la VM, y el dispositivo de framebuffer paravirtualizado QEMU QXL está configurado como acelerador de vídeo. Esto se configura automáticamente y no se recomienda cambiar estos ajustes.</p>
<pre data-bbox="220 1456 836 1877"><redirdev bus='usb' type='spicevmc'> <address type='usb' bus='0' port='1'/> </redirdev> <redirdev bus='usb' type='spicevmc'> <address type='usb' bus='0' port='2'/> </redirdev> <memballoon model='virtio'> <address type='pci' domain='0x0000' bus='0x00' slot='0x07' function='0x0'/> </memballoon> </devices> </domain></pre>	<p data-bbox="1015 1420 1422 1621">La VM tiene dos redireccionadores para conectar dispositivos USB de forma remota, y la ampliación de la memoria está activada. Esto se configura automáticamente y no se recomienda cambiar estos ajustes.</p>

CAPÍTULO 7. GUARDAR Y RESTAURAR MÁQUINAS VIRTUALES

Para liberar recursos del sistema, puede apagar una máquina virtual (VM) que se esté ejecutando en ese sistema. Sin embargo, cuando vuelva a necesitar la VM, deberá arrancar el sistema operativo (SO) invitado y reiniciar las aplicaciones, lo que puede llevar un tiempo considerable. Para reducir este tiempo de inactividad y permitir que la carga de trabajo de la VM empiece a funcionar antes, puede utilizar la función de guardar y restaurar para evitar por completo la secuencia de apagado y arranque del SO.

Esta sección proporciona información sobre cómo guardar las VMs, así como sobre cómo restaurarlas al mismo estado sin un arranque completo de la VM.

7.1. CÓMO FUNCIONA EL GUARDADO Y LA RESTAURACIÓN DE MÁQUINAS VIRTUALES

Al guardar una máquina virtual (VM), se guarda su memoria y el estado de los dispositivos en el disco del host, y se detiene inmediatamente el proceso de la VM. Puede guardar una VM que esté en estado de ejecución o de pausa, y al restaurarla, la VM volverá a ese estado.

Este proceso libera recursos de RAM y CPU en el sistema anfitrión a cambio de espacio en disco, lo que puede mejorar el rendimiento del sistema anfitrión. Cuando se restaura la VM, al no tener que arrancar el SO invitado, se evita también el largo periodo de arranque.

Para guardar una VM, puede utilizar la interfaz de línea de comandos (CLI). Para obtener instrucciones, consulte [Guardar máquinas virtuales mediante la interfaz de línea de comandos](#).

Para restaurar una VM puede utilizar la [CLI](#) o la [GUI de la consola web](#).

7.2. GUARDAR UNA MÁQUINA VIRTUAL MEDIANTE LA INTERFAZ DE LÍNEA DE COMANDOS

Para guardar una máquina virtual (VM) utilizando la línea de comandos, siga el procedimiento siguiente.

Requisitos previos

- Asegúrese de tener suficiente espacio en el disco para guardar la VM y su configuración. Tenga en cuenta que el espacio ocupado por la VM depende de la cantidad de RAM asignada a esa VM.
- Asegúrese de que la máquina virtual es persistente.
- **Optional:** Haga una copia de seguridad de los datos importantes de la máquina virtual si es necesario.

Procedimiento

- Utilice la utilidad **virsh managedsave**.
Por ejemplo, el siguiente comando detiene la VM *demo-guest1* y guarda su configuración.

```
# virsh managedsave demo-guest1
Domain demo-guest1 saved by libvirt
```

El archivo VM guardado se encuentra por defecto en el directorio `/var/lib/libvirt/qemu/save` como **demo-guest1.save**.

La próxima vez que se **inicie** la máquina virtual, se restaurará automáticamente el estado guardado en el archivo anterior.

Verificación

- Puedes asegurarte de que la VM está en un estado guardado o apagado utilizando la utilidad **virsh list**.

Para listar las máquinas virtuales que tienen habilitado el almacenamiento gestionado, utilice el siguiente comando. Las máquinas virtuales que aparecen en *saved* tienen habilitado el almacenamiento gestionado.

```
# virsh list --managed-save --all
Id Name State
-----
- demo-guest1 saved
- demo-guest2 shut off
```

Para listar las máquinas virtuales que tienen una imagen guardada gestionada:

```
# virsh list --with-managed-save --all
Id Name State
-----
- demo-guest1 shut off
```

Tenga en cuenta que para listar las máquinas virtuales guardadas que están en estado de apagado, debe utilizar las opciones **--all** o **--inactive** con el comando.

Solución de problemas

- Si el archivo guardado de la VM se corrompe o es ilegible, la restauración de la VM iniciará un arranque estándar de la VM en su lugar.

Recursos adicionales

- Para obtener más argumentos de **virsh managedsave**, utilice **virsh managedsave --help** o consulte la página de manual **virsh**.
- Para obtener instrucciones sobre la restauración de una VM guardada mediante la interfaz de línea de comandos, consulte [Sección 7.3, "Iniciar una máquina virtual mediante la interfaz de línea de comandos"](#).
- Para obtener instrucciones sobre cómo restaurar una VM guardada utilizando la consola web, consulte [Sección 7.4, "Iniciar máquinas virtuales mediante la consola web"](#).

7.3. INICIAR UNA MÁQUINA VIRTUAL MEDIANTE LA INTERFAZ DE LÍNEA DE COMANDOS

Puede utilizar la interfaz de línea de comandos para iniciar una máquina virtual (VM) apagada o restaurar una VM guardada. Siga el procedimiento siguiente.

Requisitos previos

- Una VM inactiva que ya está definida.

- El nombre de la máquina virtual.
- Para las máquinas virtuales remotas:
 - La dirección IP del host donde se encuentra la VM.
 - Privilegios de acceso a la raíz del host.

Procedimiento

- Para una VM local, utilice la utilidad **virsh start**.
Por ejemplo, el siguiente comando inicia la VM *demo-guest1*.

```
# virsh start demo-guest1
Domain demo-guest1 started
```

- Para una VM ubicada en un host remoto, utilice la utilidad **virsh start** junto con la conexión SSH de QEMU al host.
Por ejemplo, el siguiente comando inicia la VM *demo-guest1* en el host 192.168.123.123.

```
# virsh -c qemu+ssh://root@192.168.123.123/system start demo-guest1
root@192.168.123.123's password:
Last login: Mon Feb 18 07:28:55 2019

Domain demo-guest1 started
```

Recursos adicionales

- Para más argumentos **virsh start**, utilice **virsh start --help**.
- Para simplificar la gestión de las máquinas virtuales en hosts remotos, consulte la [modificación de la configuración de libvirt y SSH](#).
- Puede utilizar la utilidad **virsh autostart** para configurar una VM para que se inicie automáticamente cuando el host arranque. Para obtener más información sobre el arranque automático, consulte la página de ayuda **virsh autostart**.

7.4. INICIAR MÁQUINAS VIRTUALES MEDIANTE LA CONSOLA WEB

Si una máquina virtual (VM) está en el estado **shut off**, puede iniciarla utilizando la consola web de RHEL 8.

Requisitos previos

- Una VM inactiva que ya está definida.
- El nombre de la máquina virtual.
- Para utilizar la consola web para gestionar las máquinas virtuales, [instale el complemento VM de la consola web](#).

Procedimiento

1. En la interfaz de **máquinas virtuales**, haga clic en la fila de la máquina virtual que desea iniciar. La fila se expande para revelar el panel de Visión General con información básica sobre la VM seleccionada y controles para apagar y eliminar la VM.
2. Haga clic en **Ejecutar**.
La VM se inicia, y puedes [conectarte a su consola o salida gráfica](#) .
3. **Optional:** Para configurar la VM para que se inicie automáticamente cuando se inicie el host, haga clic en la casilla **Autostart**.

Recursos adicionales

- Para obtener información sobre cómo apagar una máquina virtual, consulte [Sección 2.5.2.1, "Apagado de máquinas virtuales en la consola web"](#).
- Para obtener información sobre cómo reiniciar una máquina virtual, consulte [Sección 2.5.2.2, "Reinicio de máquinas virtuales mediante la consola web"](#).
- Para obtener información sobre el envío de una interrupción no enmascarable a una VM, consulte [Sección 2.5.2.3, "Envío de interrupciones no enmascarables a las máquinas virtuales mediante la consola web"](#).

CAPÍTULO 8. CLONACIÓN DE MÁQUINAS VIRTUALES

Para crear rápidamente una nueva máquina virtual (VM) con un conjunto específico de propiedades, puede *clone* una VM existente.

La clonación crea una nueva VM que utiliza su propia imagen de disco para el almacenamiento, pero la mayor parte de la configuración del clon y los datos almacenados son idénticos a la VM de origen. Esto permite preparar una serie de VMs optimizadas para una determinada tarea sin necesidad de optimizar cada VM individualmente.

8.1. CÓMO FUNCIONA LA CLONACIÓN DE MÁQUINAS VIRTUALES

La clonación de una máquina virtual (VM) copia la configuración XML de la VM de origen y sus imágenes de disco, y realiza ajustes en las configuraciones para asegurar la unicidad de la nueva VM. Esto incluye cambiar el nombre de la VM y asegurarse de que utiliza los clones de las imágenes de disco. No obstante, los datos almacenados en los discos virtuales del clon son idénticos a los de la VM de origen.

Este proceso es más rápido que crear una nueva VM e instalarla con un sistema operativo invitado, y puede utilizarse para generar rápidamente VMs con una configuración y contenido específicos.

Si está planeando crear múltiples clones de una VM, primero cree una VM *template* que no contenga:

- Ajustes únicos, como la configuración MAC de red persistente, que pueden impedir que los clones funcionen correctamente.
- Datos sensibles, como claves SSH y archivos de contraseñas.

Para obtener instrucciones, consulte [Sección 8.2, "Creación de una plantilla de máquina virtual"](#).

Para clonar una VM, puede utilizar la CLI de RHEL 8. Para obtener más detalles, consulte [Sección 8.3, "Clonación de una máquina virtual mediante la interfaz de línea de comandos"](#).

8.2. CREACIÓN DE UNA PLANTILLA DE MÁQUINA VIRTUAL

Para garantizar que los clones de máquinas virtuales (VM) se ejecuten correctamente, elimine la información y las configuraciones que son exclusivas de la VM de origen, como las claves SSH o la configuración MAC de red persistente, antes de clonar la VM de origen.

Esto crea una VM *template*, que se puede utilizar para crear fácilmente y con seguridad clones de VM.

Requisitos previos

- La utilidad **virt-sysprep** está instalada en su host:

```
# yum install /usr/bin/virt-sysprep
```

- La VM prevista como plantilla se apaga.
- Debe saber dónde se encuentra la imagen de disco de la máquina virtual de origen y ser el propietario del archivo de imagen de disco de la máquina virtual.
Tenga en cuenta que las imágenes de disco para las máquinas virtuales creadas en la [sesión de sistema](#) de libvirt se encuentran por defecto en el directorio `/var/lib/libvirt/images` y son propiedad del usuario root:

```
# ls -la /var/lib/libvirt/images
```

```
-rw-----. 1 root root 9665380352 Jul 23 14:50 a-really-important-vm.qcow2
-rw-----. 1 root root 8591507456 Jul 26 2017 an-actual-vm-that-i-use.qcow2
-rw-----. 1 root root 8591507456 Jul 26 2017 totally-not-a-fake-vm.qcow2
-rw-----. 1 root root 10739318784 Sep 20 17:57 another-vm-example.qcow2
```

- **Optional:** Cualquier dato importante del disco de la VM ha sido respaldado. Si desea conservar la VM de origen intacta, [clónela](#) primero y edite el clon para crear una plantilla.

Procedimiento

1. Asegúrese de que ha iniciado la sesión como propietario de la imagen de disco de la máquina virtual:

```
# whoami
root
```

2. **Optional:** Copiar la imagen de disco de la VM.

```
# cp /var/lib/libvirt/images/a-really-important-vm.qcow2 /var/lib/libvirt/images/a-really-
important-vm-original.qcow2
```

Esto se utiliza más tarde para verificar que la VM se convirtió con éxito en una plantilla.

3. Utilice el siguiente comando y sustituya `/var/lib/libvirt/images/a-really-important-vm.qcow2` por la ruta de acceso a la imagen de disco de la máquina virtual de origen.

```
# virt-sysprep -a /var/lib/libvirt/images/a-really-important-vm.qcow2
[ 0.0] Examining the guest ...
[ 7.3] Performing "abrt-data" ...
[ 7.3] Performing "backup-files" ...
[ 9.6] Performing "bash-history" ...
[ 9.6] Performing "blkid-tab" ...
[...]
```

Verificación

- Para confirmar que el proceso fue exitoso, compare la imagen de disco modificada con la original. El siguiente ejemplo muestra la creación exitosa de una plantilla:

```
# virt-diff -a /var/lib/libvirt/images/a-really-important-vm-orig.qcow2 -A
/var/lib/libvirt/images/a-really-important-vm.qcow2
-- 0644    1001 /etc/group-
-- 0000    797 /etc/gshadow-
= - 0444    33 /etc/machine-id
[...]
```

```
-- 0600    409 /home/username/.bash_history
-d 0700    6 /home/username/.ssh
-- 0600    868 /root/.bash_history
[...]
```

Recursos adicionales

- Usando el comando **virt-sysprep** como se muestra arriba se realiza la preparación estándar de la plantilla de la VM. Para obtener más información, consulte la sección **OPERATIONS** en la página de manual **virt-sysprep**.
Para personalizar las operaciones específicas que desea que realice **virt-sysprep**, utilice la opción **--operations**, y especifique las operaciones previstas como una lista separada por comas.
- Para obtener instrucciones sobre la clonación de una plantilla de VM, consulte [Sección 8.3, "Clonación de una máquina virtual mediante la interfaz de línea de comandos"](#).

8.3. CLONACIÓN DE UNA MÁQUINA VIRTUAL MEDIANTE LA INTERFAZ DE LÍNEA DE COMANDOS

Para crear rápidamente una nueva máquina virtual (VM) con un conjunto específico de propiedades, por ejemplo para fines de prueba, puede clonar una VM existente. Para hacerlo mediante la CLI, siga las siguientes instrucciones.

Requisitos previos

- La máquina virtual de origen se apaga.
- Asegúrese de que hay suficiente espacio en el disco para almacenar las imágenes de disco clonadas.
- **Optional:** Cuando cree varios clones de máquinas virtuales, elimine los datos y la configuración exclusivos de la máquina virtual de origen para garantizar que las máquinas virtuales clonadas funcionen correctamente. Para obtener instrucciones, consulte [Sección 8.2, "Creación de una plantilla de máquina virtual"](#).

Procedimiento

1. Utilice la utilidad **virt-clone** con las opciones adecuadas para su entorno y caso de uso.

Sample use cases

- El siguiente comando clona una VM local llamada *doppelganger* y crea la VM *doppelganger-clone*. También crea la imagen de disco *doppelganger-clone.qcow2* en la misma ubicación que la imagen de disco de la VM original, y con los mismos datos:

```
# virt-clone --original doppelganger --auto-clone
Allocating 'doppelganger-clone.qcow2' | 50.0 GB 00:05:37

Clone 'doppelganger-clone' created successfully.
```

- El siguiente comando clona una VM llamada *geminus1*, y crea una VM local llamada *geminus2*, que utiliza sólo dos de los múltiples discos de *geminus1*.

```
# virt-clone --original geminus1 --name geminus2 --file
/var/lib/libvirt/images/disk1.qcow2 --file /var/lib/libvirt/images/disk2.qcow2
Allocating 'disk1-clone.qcow2' | 78.0 GB 00:05:37
Allocating 'disk2-clone.qcow2' | 80.0 GB 00:05:37

Clone 'geminus2' created successfully.
```

- Para clonar su VM a un host diferente, migre la VM sin desinstalarla en el host local. Por

ejemplo, los siguientes comandos clonan la VM *geminus2* creada anteriormente al sistema remoto 10.0.0.1, incluyendo sus discos locales. Tenga en cuenta que el uso de estos comandos también requiere privilegios de root para 10.0.0.1.

```
# virsh migrate --offline --persistent geminus2 qemu+ssh://root@10.0.0.1/system
root@10.0.0.1's password:

# scp /var/lib/libvirt/images/disk1-clone.qcow2
root@10.0.0.1/user@remote_host.com://var/lib/libvirt/images/

# scp /var/lib/libvirt/images/disk2-clone.qcow2
root@10.0.0.1/user@remote_host.com://var/lib/libvirt/images/
```

Verificación

Para verificar que la VM ha sido clonada con éxito y funciona correctamente:

1. Confirme que el clon se ha añadido a la lista de máquinas virtuales de su host.

```
# virsh list --all
Id Name State
-----
- doppelganger shut off
- doppelganger-clone shut off
```

2. Inicie el clon y observe si arranca.

```
# virsh start doppelganger-clone
Domain doppelganger-clone started
```

Recursos adicionales

- Para ver otras opciones para clonar máquinas virtuales, consulte la página de manual **virt-clone**.
- Para obtener más detalles sobre cómo mover los clones de VM a un host diferente, incluyendo información sobre la solución de problemas, consulte [Capítulo 9, Migración de máquinas virtuales](#).

CAPÍTULO 9. MIGRACIÓN DE MÁQUINAS VIRTUALES

Si el host actual de una máquina virtual (VM) se vuelve inadecuado o ya no se puede utilizar, o si desea redistribuir la carga de trabajo de alojamiento, puede migrar la VM a otro host KVM.

9.1. CÓMO FUNCIONA LA MIGRACIÓN DE MÁQUINAS VIRTUALES

La parte esencial de la migración de una máquina virtual (VM) es copiar la configuración XML de una VM a una máquina anfitriona diferente. Si la VM migrada no se apaga, la migración también transfiere el estado de la memoria de la VM y cualquier dispositivo virtualizado a una máquina anfitriona de destino. Para que la VM siga funcionando en el host de destino, las imágenes de disco de la VM deben seguir estando disponibles para ella.

Por defecto, la VM migrada es transitoria en el host de destino, y permanece definida también en el host de origen.

Puede migrar una VM en funcionamiento utilizando las migraciones *live* o *non-live*. Para migrar una VM apagada, debe utilizar una migración *offline*. Para más detalles, consulte la siguiente tabla.

Tabla 9.1. Tipos de migración de máquinas virtuales

Tipo de migración	Descripción	Caso de uso	Requisitos de almacenamiento
Live migration	La VM continúa ejecutándose en la máquina anfitriona de origen mientras KVM transfiere las páginas de memoria de la VM al anfitrión de destino. Cuando la migración está casi completa, KVM suspende muy brevemente la VM y la reanuda en el host de destino.	Es útil para las máquinas virtuales que requieren un tiempo de actividad constante. Sin embargo, las máquinas virtuales que modifican las páginas de memoria más rápido de lo que KVM puede transferirlas, como las máquinas virtuales sometidas a una gran carga de E/S, no pueden ser migradas en vivo, y en su lugar debe utilizarse <i>non-live migration</i> .	Las imágenes de disco de la VM deben estar ubicadas en una red compartida , accesible tanto para el host de origen como para el host de destino.
Non-live migration	Suspende la VM, copia su configuración y su memoria al host de destino, y reanuda la VM.	Crea un tiempo de inactividad para la VM, pero generalmente es más fiable que la migración en vivo. Se recomienda para las máquinas virtuales con mucha carga de E/S.	Las imágenes de disco de la VM deben estar ubicadas en una red compartida , accesible tanto para el host de origen como para el host de destino.

Tipo de migración	Descripción	Caso de uso	Requisitos de almacenamiento
Offline migration	Mueve la configuración de la VM al host de destino	Recomendado para los VM de cierre.	Las imágenes de disco de la VM no tienen que estar disponibles en una red compartida, y pueden copiarse o moverse manualmente al host de destino.

Recursos adicionales

- Para más información sobre las ventajas de la migración de máquinas virtuales, consulte [Sección 9.2, “Ventajas de la migración de máquinas virtuales”](#).
- Para obtener instrucciones sobre la configuración del almacenamiento compartido para la migración de máquinas virtuales, consulte [Sección 9.4, “Compartir imágenes de disco de máquinas virtuales con otros hosts”](#).

9.2. VENTAJAS DE LA MIGRACIÓN DE MÁQUINAS VIRTUALES

La migración de máquinas virtuales (VM) puede ser útil para:

Equilibrio de la carga

Las máquinas virtuales se pueden trasladar a máquinas con menor uso si su host se sobrecarga, o si otro host está infrautilizado.

Independencia del hardware

Cuando necesite actualizar, añadir o eliminar dispositivos de hardware en la máquina anfitriona, puede reubicar de forma segura las máquinas virtuales en otros hosts. Esto significa que las máquinas virtuales no experimentan ningún tiempo de inactividad para las mejoras de hardware.

Ahorro de energía

Las máquinas virtuales pueden redistribuirse a otros hosts, y los sistemas de host sin carga pueden así apagarse para ahorrar energía y reducir los costes durante los periodos de bajo uso.

Migración geográfica

Las máquinas virtuales pueden trasladarse a otra ubicación física para reducir la latencia o cuando sea necesario por otros motivos.

9.3. LIMITACIONES PARA LA MIGRACIÓN DE MÁQUINAS VIRTUALES

Antes de migrar máquinas virtuales (VM) en RHEL 8, asegúrese de conocer las limitaciones de la migración.

- La migración de almacenamiento en vivo no puede realizarse en RHEL 8, pero puede migrar el almacenamiento mientras la VM está apagada. Tenga en cuenta que la migración de almacenamiento en vivo está disponible en [Red Hat Virtualization](#).
- La migración de máquinas virtuales desde o hacia una [sesión de usuario de libvirt](#) no es fiable y, por tanto, no se recomienda.

- Las máquinas virtuales que utilizan ciertas características y configuraciones no funcionarán correctamente si se migran, o la migración fallará. Estas características incluyen:
 - Paso de dispositivos
 - Asignación de dispositivos SR-IOV
 - Dispositivos mediados, como las vGPUs
 - Acceso no uniforme a la memoria (NUMA)

9.4. COMPARTIR IMÁGENES DE DISCO DE MÁQUINAS VIRTUALES CON OTROS HOSTS

Para realizar una migración en vivo de una máquina virtual (VM) entre [hosts KVM compatibles](#), se requiere un almacenamiento compartido de VM. Esta sección proporciona instrucciones para compartir una imagen de VM almacenada localmente con el host de origen y el host de destino utilizando el protocolo NFS.

Requisitos previos

- La VM destinada a la migración se apaga.
- **Optional:** Se dispone de un sistema de host para alojar el almacenamiento que no es el host de origen ni el de destino, pero tanto el host de origen como el de destino pueden alcanzarlo a través de la red. Esta es la solución óptima para el almacenamiento compartido y es recomendada por Red Hat.
- Asegúrese de que no se utiliza el bloqueo de archivos NFS, ya que no está soportado en KVM.
- El NFS está instalado y habilitado en los hosts de origen y destino. Si no lo está:

- a. Instale los paquetes NFS:

```
# yum install nfs-utils
```

- b. Asegúrese de que los puertos para NFS en **iptables** (como el 2049) están abiertos en el cortafuegos.

```
# firewall-cmd --permanent --add-service=nfs
# firewall-cmd --permanent --add-service=mountd
# firewall-cmd --permanent --add-service=rpc-bind
# firewall-cmd --permanent --add-port=2049/tcp
# firewall-cmd --permanent --add-port=2049/udp
# firewall-cmd --reload
```

- c. Inicie el servicio NFS.

```
# systemctl start nfs-server
```

Procedimiento

1. Conéctate al host que proporcionará el almacenamiento compartido. En este ejemplo, es el host **cargo-bay**:


```
# ssh root@cargo-bay
root@cargo-bay's password:
Last login: Mon Sep 24 12:05:36 2019
root~#
```

2. Cree un directorio que contenga la imagen de disco y que se comparta con los hosts de migración.

```
# mkdir /var/lib/libvirt/shared-images
```

3. Copie la imagen de disco de la VM desde el host de origen al directorio recién creado. Por ejemplo, lo siguiente copia la imagen de disco de la VM **wanderer1** al directorio **/var/lib/libvirt/shared-images/** en el host ``cargo-bay``:

```
# scp /var/lib/libvirt/images/wanderer1.qcow2 root@cargo-bay:/var/lib/libvirt/shared-
images/wanderer1.qcow2
```

4. En el host que desee utilizar para compartir el almacenamiento, añada el directorio compartido al archivo **/etc/exports**. El siguiente ejemplo comparte el directorio **/var/lib/libvirt/shared-images** con los hosts **source-example** y **dest-example**:

```
/var/lib/libvirt/shared-images source-example(rw,no_root_squash) dest-
example(rw,no_root_squash)
```

5. Tanto en el host de origen como en el de destino, monte el directorio compartido en el directorio **/var/lib/libvirt/images**:

```
# mount cargo-bay:/var/lib/libvirt/shared-images /var/lib/libvirt/images
```

Verificación

- Para verificar que el proceso ha sido exitoso, inicie la VM en el host de origen y observe si arranca correctamente.

Recursos adicionales

- Para obtener información detallada sobre la configuración de NFS, la apertura de tablas IP y la configuración del cortafuegos, consulte [Exportación de recursos compartidos NFS](#).

9.5. MIGRACIÓN DE UNA MÁQUINA VIRTUAL MEDIANTE LA INTERFAZ DE LÍNEA DE COMANDOS

Si el host actual de una máquina virtual (VM) se vuelve inadecuado o ya no se puede utilizar, o si desea redistribuir la carga de trabajo de alojamiento, puede migrar la VM a otro host KVM. Esta sección proporciona instrucciones y ejemplos para varios escenarios de tales migraciones.

Requisitos previos

- Tanto el host de origen como el de destino utilizan el hipervisor KVM.
- El host de origen y el host de destino pueden comunicarse entre sí a través de la red. Utilice la utilidad **ping** para comprobarlo.

- Para que la migración sea compatible con Red Hat, el host de origen y el host de destino deben utilizar sistemas operativos y tipos de máquinas específicos. Para asegurarse de que este es el caso, consulte la [tabla de compatibilidad de la migración de VM](#) .
- Red Hat recomienda que las imágenes de disco de las VMs que serán migradas estén ubicadas en una localización de red separada y accesible tanto para el host de origen como para el host de destino. Esto es opcional para la migración fuera de línea, pero se requiere para la migración de una VM en ejecución.
Para obtener instrucciones para configurar dicho almacenamiento compartido de VM, consulte [Sección 9.4, “Compartir imágenes de disco de máquinas virtuales con otros hosts”](#) .
- Cuando se migra una VM existente en una red de grifo de puente público, los hosts de origen y destino deben estar ubicados en la misma red. De lo contrario, la red de la VM no funcionará después de la migración.

Procedimiento

1. Asegúrese de que el servicio **libvirtd** está activado y en funcionamiento.

```
# systemctl enable libvirtd.service
# systemctl restart libvirtd.service
```

2. Utilice el comando **virsh migrate** con las opciones adecuadas a sus necesidades de migración.

- A continuación se migra la VM **wanderer1** desde su host local a la sesión del sistema del host **dest-example**. La VM seguirá funcionando durante la migración.

```
# virsh migrate --persistent --live wanderer1 qemu ssh://dest-example/system
```

- Lo siguiente le permite realizar ajustes manuales en la configuración de la VM de **wanderer2** que se ejecuta en su host local, y luego migra la VM al host de **dest-example**. La VM migrada utilizará automáticamente la configuración actualizada.

```
# virsh dumpxml --migratable wanderer2 >wanderer2.xml
# vi wanderer2.xml
# virsh migrate --live --persistent --xml wanderer2.xml wanderer2 qemu+ssh://dest-example/system
```

Este procedimiento puede ser útil, por ejemplo, cuando el host de destino necesita utilizar una ruta diferente para acceder al almacenamiento compartido de la VM o cuando se configura una función específica del host de destino.

- Lo siguiente suspende la VM **wanderer3** del host **source-example**, la migra al host **dest-example** y le indica que utilice la configuración XML ajustada, proporcionada por el archivo **wanderer3-alt.xml**. Cuando se completa la migración, **libvirt** reanuda la VM en el host de destino.

```
# virsh migrate wanderer3 qemu ssh://source-example/system qemu ssh://dest-example/system --xml wanderer3-alt.xml
```

Después de la migración, la VM permanece en el estado suspendido en el host de origen, y la copia migrada se elimina después de que se apague.

- Lo siguiente elimina la VM **wanderer4** apagada del host **source-example**, y mueve su configuración al host **dest-example**.

```
# virsh migrate --offline --persistent --undefinesource wanderer4 qemu
ssh://source-example/system qemu ssh://dest-example/system
```

Tenga en cuenta que este tipo de migración no requiere mover la imagen de disco de la VM al almacenamiento compartido. Sin embargo, para que la VM sea utilizable en el host de destino, es necesario migrar la imagen de disco de la VM. Por ejemplo:

```
# scp root@source-example:/var/lib/libvirt/images/wanderer4.qcow2 root@dest-
example:/var/lib/libvirt/images/wanderer4.qcow2
```

- Espera a que se complete la migración. El proceso puede tardar algún tiempo dependiendo del ancho de banda de la red, la carga del sistema y el tamaño de la VM. Si no se utiliza la opción **--verbose** para **virsh migrate**, la CLI no muestra ningún indicador de progreso excepto los errores.

Cuando la migración está en curso, puede utilizar la utilidad **virsh domjobinfo** para mostrar las estadísticas de migración.

Verificación

- En el host de destino, liste las VMs disponibles para verificar si la VM ha sido migrada:

```
# virsh list
Id Name                               State
-----
10 wanderer1                          running
```

Tenga en cuenta que si la migración aún está en marcha, este comando mostrará el estado de la máquina virtual como **paused**.

Solución de problemas

- En algunos casos, el host de destino no será compatible con ciertos valores de la configuración XML de la VM migrada, como el nombre de la red o el tipo de CPU. Como resultado, la VM no podrá arrancar en el host de destino. Para solucionar estos problemas, puede actualizar los valores problemáticos mediante el comando **virsh edit**.
- Si una migración en vivo está tardando mucho en completarse, esto puede deberse a que la VM está bajo una gran carga y demasiadas páginas de memoria están cambiando para que la migración en vivo sea posible. Para solucionar este problema, cambie la migración a una no viva suspendiendo la VM.

```
# virsh suspend wanderer1
```

Recursos adicionales

- Para más opciones y ejemplos de migración de máquinas virtuales, utilice **virsh migrate --help** o consulte la página de manual **virsh**.

9.6. HOSTS COMPATIBLES CON LA MIGRACIÓN DE MÁQUINAS VIRTUALES

Para que la migración de la máquina virtual (VM) funcione correctamente y sea soportada por Red Hat, los hosts de origen y destino deben ser versiones de RHEL y tipos de máquina específicos. La siguiente tabla muestra las rutas de migración de VM soportadas.

Tabla 9.2. Compatibilidad con la migración en vivo

Método de migración	Tipo de liberación	Ejemplo	Estado de la ayuda
Adelante	Lanzamiento importante	7.6 → 8.1	En los sistemas RHEL 7 compatibles: tipos de máquina i440fx y q35
Hacia atrás	Lanzamiento importante	8.1 → 7.6	En los sistemas RHEL 8 compatibles: tipos de máquina i440fx y q35
Adelante	Versión menor	8.0.1 → 8.1	En los sistemas RHEL 7 compatibles: tipos de máquina i440fx y q35 en RHEL 7.6.0 y posteriores. En los sistemas RHEL 8 compatibles: tipo de máquina q35 .
Hacia atrás	Versión menor	8.1 → 8.0.1	En los sistemas RHEL 7 compatibles. Totalmente compatible con los tipos de máquina i440fx y q35 . En los sistemas RHEL 8 compatibles: tipo de máquina q35 .

Recursos adicionales

- Para obtener información sobre las versiones actualmente soportadas de RHEL 7 y RHEL 8, consulte [Red Hat Knowledgebase](#).

9.7. RECURSOS ADICIONALES

- También puede migrar VMs desde un hipervisor no-KVM a un host RHEL 7 o RHEL 8. Esto también se conoce como **V2V conversion**, y puede encontrar información adicional e instrucciones en la [base de conocimientos de Red Hat](#).

CAPÍTULO 10. GESTIÓN DE DISPOSITIVOS VIRTUALES

Una de las formas más eficaces de gestionar la funcionalidad, las características y el rendimiento de una máquina virtual (VM) es ajustar su *virtual devices*.

Las siguientes secciones proporcionan una [visión general](#) de lo que son los dispositivos virtuales, e instrucciones sobre cómo se pueden [conectar](#), [modificar](#) o [eliminar](#) de una VM.

10.1. CÓMO FUNCIONAN LOS DISPOSITIVOS VIRTUALES

Lo básico

Al igual que las máquinas físicas, las máquinas virtuales (VM) requieren dispositivos especializados para proporcionar funciones al sistema, como potencia de procesamiento, memoria, almacenamiento, redes o gráficos. Los sistemas físicos suelen utilizar dispositivos de hardware para estos fines. Sin embargo, dado que las máquinas virtuales funcionan como implementos de software, necesitan utilizar en su lugar abstracciones de software de dichos dispositivos, denominadas *virtual devices*.

Los dispositivos virtuales conectados a una VM pueden ser configurados cuando [se crea la VM](#), y también pueden ser gestionados en una VM existente. Generalmente, los dispositivos virtuales se pueden conectar o desconectar de una VM sólo cuando la VM está apagada, pero algunos se pueden añadir o eliminar cuando la VM está en funcionamiento. Esta característica se conoce como dispositivo *hot plug* y *hot unplug*.

Al crear una nueva VM, **libvirt** crea y configura automáticamente un conjunto de dispositivos virtuales esenciales por defecto, a menos que el usuario especifique lo contrario. Estos se basan en la arquitectura del sistema anfitrión y el tipo de máquina, y normalmente incluyen:

- la CPU
- memoria
- un teclado
- un controlador de interfaz de red (NIC)
- varios controladores de dispositivos
- una tarjeta de vídeo
- una tarjeta de sonido

Para gestionar los dispositivos virtuales una vez creada la VM, utilice la interfaz de línea de comandos (CLI). Sin embargo, para gestionar los dispositivos de [almacenamiento virtual](#) y [las NIC](#), también puede utilizar la consola web de RHEL 8.

Rendimiento o flexibilidad

Para algunos tipos de dispositivos, RHEL 8 admite varias implementaciones, a menudo con un compromiso entre rendimiento y flexibilidad.

Por ejemplo, el almacenamiento físico utilizado para los discos virtuales puede ser representado por archivos en varios formatos, como **qcow2** o **raw**, y presentado a la VM utilizando una variedad de controladores:

- un controlador emulado

- **virtio-scsi**
- **virtio-blk**

Un controlador emulado es más lento que un controlador **virtio**, porque los dispositivos **virtio** están diseñados específicamente para la virtualización. Por otro lado, las controladoras emuladas permiten ejecutar sistemas operativos que no tienen controladores para los dispositivos **virtio**. Del mismo modo, **virtio-scsi** ofrece un soporte más completo para los comandos SCSI, y hace posible adjuntar un mayor número de discos a la VM. Por último, **virtio-blk** ofrece un mejor rendimiento que tanto **virtio-scsi** como los controladores emulados, pero una gama más limitada de casos de uso. Por ejemplo, adjuntar un disco físico como dispositivo LUN a una VM no es posible cuando se utiliza **virtio-blk**.

Para más información sobre los tipos de dispositivos virtuales, consulte [Sección 10.5, “Tipos de dispositivos virtuales”](#).

Recursos adicionales

- Para obtener instrucciones sobre cómo conectar, eliminar o modificar los dispositivos de almacenamiento de la VM utilizando la CLI, consulte [Capítulo 11, Gestión del almacenamiento de las máquinas virtuales](#).
- Para obtener instrucciones sobre cómo gestionar los discos de las máquinas virtuales mediante la consola web, consulte [Sección 11.3.7, “Gestión de los discos de las máquinas virtuales mediante la consola web”](#).
- Para obtener instrucciones sobre cómo gestionar las NIC de las VM mediante la consola web, consulte [Sección 13.2, “Uso de la consola web para gestionar las interfaces de red de las máquinas virtuales”](#).
- Para obtener instrucciones sobre cómo crear y gestionar las vGPUs de NVIDIA, consulte [Capítulo 12, Gestión de dispositivos NVIDIA vGPU](#).

10.2. ADJUNTAR DISPOSITIVOS A LAS MÁQUINAS VIRTUALES

A continuación se ofrece información general sobre la creación y la conexión de dispositivos virtuales a sus máquinas virtuales (VM) mediante la interfaz de línea de comandos (CLI). Algunos dispositivos también pueden adjuntarse a las VM [mediante la consola web de RHEL 8](#).

Requisitos previos

- Obtenga las opciones necesarias para el dispositivo que pretende adjuntar a una VM. Para ver las opciones disponibles para un dispositivo específico, utilice el comando **virt-xml --device=?** para ver las opciones disponibles para un dispositivo específico. Por ejemplo:

```
# virt-xml --network=?
--network options:
[...]
address.unit
boot_order
clearxml
driver_name
[...]
```

Procedimiento

1. Para adjuntar un dispositivo a una VM, utilice el comando **virt-xml --add-device**, incluyendo la definición del dispositivo y las opciones requeridas:
 - Por ejemplo, el siguiente comando crea una imagen de disco *newdisk* qcow2 de 20 GB en el directorio **/var/lib/libvirt/images/**, y la adjunta como disco virtual a la máquina virtual *testguest* que se está ejecutando en el siguiente arranque de la máquina virtual:

```
# virt-xml testguest --add-device --disk
/var/lib/libvirt/images/newdisk.qcow2,format=qcow2,size=20
Domain 'testguest' defined successfully.
Changes will take effect after the domain is fully powered off.
```

- Lo siguiente adjunta una unidad flash USB, conectada como dispositivo 004 en el bus 002 en el host, a la VM *testguest2* mientras la VM se está ejecutando:

```
# virt-xml testguest2 --add-device --update --hostdev 002.004
Device hotplug successful.
Domain 'testguest2' defined successfully.
```

La combinación bus-dispositivo para definir el USB se puede obtener mediante el comando **lsusb**.

Verificación

Para comprobar que el dispositivo se ha añadido, realice una de las siguientes acciones:

- Utilice el comando **virsh dumpxml** y compruebe si la definición XML del dispositivo se ha añadido a la sección **<devices>** en la configuración XML de la máquina virtual. Por ejemplo, la siguiente salida muestra la configuración de la VM *testguest* y confirma que se ha añadido el dispositivo de disco flash USB 002.004.

```
# virsh dumpxml testguest
[...]
<hostdev mode='subsystem' type='usb' managed='yes'>
  <source>
    <vendor id='0x4146'/>
    <product id='0x902e'/>
    <address bus='2' device='4'/>
  </source>
  <alias name='hostdev0'/>
  <address type='usb' bus='0' port='3'/>
</hostdev>
[...]
```

- Ejecute la VM y compruebe si el dispositivo está presente y funciona correctamente.

Recursos adicionales

- Para más información sobre el uso del comando **virt-xml**, utilice **man virt-xml**.

10.3. MODIFICACIÓN DE LOS DISPOSITIVOS CONECTADOS A LAS MÁQUINAS VIRTUALES

El siguiente procedimiento proporciona instrucciones generales para modificar los dispositivos virtuales utilizando la interfaz de línea de comandos (CLI). Algunos dispositivos conectados a su VM, como los discos y las NIC, también pueden modificarse [mediante la consola web de RHEL 8](#).

Requisitos previos

- Obtenga las opciones necesarias para el dispositivo que pretende adjuntar a una VM. Para ver las opciones disponibles para un dispositivo específico, utilice el comando **virt-xml --device=?** para ver las opciones disponibles para un dispositivo específico. Por ejemplo:

```
# virt-xml --network=?
--network options:
[...]
address.unit
boot_order
clearxml
driver_name
[...]
```

- **Optional:** Haga una copia de seguridad de la configuración XML de su VM utilizando **virsh dumpxml *vm-name*** y enviando la salida a un archivo. Por ejemplo, lo siguiente respalda la configuración de su VM *Motoko* como el archivo **motoko.xml**:

```
# virsh dumpxml Motoko > motoko.xml
# cat motoko.xml
<domain type='kvm' xmlns:qemu='http://libvirt.org/schemas/domain/qemu/1.0'>
  <name>Motoko</name>
  <uuid>ede29304-fe0c-4ca4-abcd-d246481acd18</uuid>
  [...]
</domain>
```

Procedimiento

1. Utilice el comando **virt-xml --edit**, incluyendo la definición del dispositivo y las opciones requeridas:
Por ejemplo, lo siguiente borra la configuración de `<cpu>` del cierre *testguest* VM y lo establece en *host-model*:

```
# virt-xml testguest --edit --cpu host-model,clearxml=yes
Domain 'testguest' defined successfully.
```

Verificación

Para verificar que el dispositivo ha sido modificado, realice cualquiera de las siguientes acciones:

- Ejecute la VM y compruebe si el dispositivo está presente y refleja las modificaciones.
- Utilice el comando **virsh dumpxml** y vea si la definición XML del dispositivo ha sido modificada en la configuración XML de la máquina virtual.
Por ejemplo, la siguiente salida muestra la configuración de la VM *testguest* y confirma que el modo de CPU ha sido configurado como *host-model*.

```
# virsh dumpxml testguest
[...]
```



```
<cpu mode='host-model' check='partial'>
  <model fallback='allow'/>
</cpu>
[...]
```

Solución de problemas

- Si la modificación de un dispositivo hace que su VM no pueda arrancar, utilice la utilidad **virsh define** para restaurar la configuración XML recargando el archivo de configuración XML del que hizo una copia de seguridad anteriormente.

```
# virsh define testguest.xml
```



NOTA

Para pequeños cambios en la configuración XML de su VM, puede utilizar el comando **virsh edit** - por ejemplo **virsh edit testguest**. Sin embargo, no utilice este método para cambios más extensos, ya que es más probable que rompa la configuración de manera que podría impedir el arranque de la VM.

Recursos adicionales

- Para más detalles sobre el uso del comando **virt-xml**, utilice **man virt-xml**.

10.4. ELIMINACIÓN DE DISPOSITIVOS DE LAS MÁQUINAS VIRTUALES

A continuación se ofrece información general para eliminar dispositivos virtuales de sus máquinas virtuales (VM) mediante la interfaz de línea de comandos (CLI). Algunos dispositivos, como los discos o las NIC, también pueden eliminarse de las VM [mediante la consola web de RHEL 8](#).

Requisitos previos

- **Optional:** Haga una copia de seguridad de la configuración XML de su VM utilizando **virsh dumpxml vm-name** y enviando la salida a un archivo. Por ejemplo, lo siguiente respalda la configuración de su VM *Motoko* como el archivo **motoko.xml**:

```
# virsh dumpxml Motoko > motoko.xml
# cat motoko.xml
<domain type='kvm' xmlns:qemu='http://libvirt.org/schemas/domain/qemu/1.0'>
  <name>Motoko</name>
  <uuid>ede29304-fe0c-4ca4-abcd-d246481acd18</uuid>
  [...]
</domain>
```

Procedimiento

1. Utilice el comando **virt-xml --remove-device**, incluyendo una definición del dispositivo. Por ejemplo:
 - Lo siguiente elimina el dispositivo de almacenamiento marcado como *vdb* de la VM *testguest* en ejecución después de que se apague:

```
# virt-xml testguest --remove-device --disk target=vdb
Domain 'testguest' defined successfully.
Changes will take effect after the domain is fully powered off.
```

- Lo siguiente elimina inmediatamente un dispositivo de unidad flash USB de la VM *testguest2* en ejecución:

```
# virt-xml testguest2 --remove-device --update --hostdev type=usb
Device hotunplug successful.
Domain 'testguest2' defined successfully.
```

Solución de problemas

- Si la eliminación de un dispositivo hace que su VM no pueda arrancar, utilice la utilidad **virsh define** para restaurar la configuración XML recargando el archivo de configuración XML del que hizo una copia de seguridad anteriormente.

```
# virsh define testguest.xml
```

Recursos adicionales

- Para más detalles sobre el uso del comando **virt-xml**, utilice **man virt-xml**.

10.5. TIPOS DE DISPOSITIVOS VIRTUALES

La virtualización en RHEL 8 puede presentar varios tipos distintos de dispositivos virtuales que puede adjuntar a las máquinas virtuales (VM):

Dispositivos emulados

Los dispositivos emulados son implementaciones de software de dispositivos físicos ampliamente utilizados. Los controladores diseñados para los dispositivos físicos también son compatibles con los dispositivos emulados. Por lo tanto, los dispositivos emulados pueden utilizarse con gran flexibilidad. Sin embargo, al tener que emular fielmente un tipo concreto de hardware, los dispositivos emulados pueden sufrir una importante pérdida de rendimiento en comparación con los correspondientes dispositivos físicos o con dispositivos virtuales más optimizados.

Se admiten los siguientes tipos de dispositivos emulados:

- CPUs virtuales (vCPUs), con una gran variedad de modelos de CPU disponibles. El impacto de la emulación en el rendimiento depende en gran medida de las diferencias entre la CPU anfitriona y la vCPU emulada.
- Componentes del sistema emulados, como los controladores de bus PCI
- Controladores de almacenamiento emulados, como SATA, SCSI o incluso IDE
- Dispositivos de sonido emulados, como ICH9, ICH6 o AC97
- Tarjetas gráficas emuladas, como las tarjetas VGA o QXL
- Dispositivos de red emulados, como el rtl8139

Dispositivos paravirtuales

La paravirtualización proporciona un método rápido y eficiente para exponer dispositivos virtuales a las máquinas virtuales. Los dispositivos paravirtualizados exponen interfaces diseñadas específicamente para su uso en VMs, y por lo tanto aumentan significativamente el rendimiento del dispositivo. RHEL 8 proporciona dispositivos paravirtualizados a las VMs utilizando la API *virtio* como una capa entre el hipervisor y la VM. El inconveniente de este enfoque es que requiere un controlador de dispositivo específico en el sistema operativo invitado.

Se recomienda utilizar dispositivos paravirtualizados en lugar de dispositivos emulados para las VM siempre que sea posible, especialmente si están ejecutando aplicaciones intensivas de E/S. Los dispositivos paravirtualizados disminuyen la latencia de E/S y aumentan el rendimiento de E/S, en algunos casos acercándolos al rendimiento bare-metal. Otros dispositivos paravirtualizados también añaden funcionalidad a las VMs que no está disponible de otra manera.

Se admiten los siguientes tipos de dispositivos paravirtualizados:

- El dispositivo de red paravirtualizado (**virtio-net**).
- Controladores de almacenamiento paravirtualizados:
 - **virtio-blk** - proporciona una emulación de dispositivo de bloque.
 - **virtio-scsi** - proporciona una emulación SCSI más completa.
- El reloj paravirtualizado.
- El dispositivo serial paravirtualizado (**virtio-serial**).
- El dispositivo globo (**virtio-balloon**), utilizado para compartir información sobre el uso de la memoria del huésped con el hipervisor.
Sin embargo, hay que tener en cuenta que el dispositivo del globo también requiere que se instale el servicio del globo.
- El generador de números aleatorios paravirtualizado (**virtio-rng**).
- La tarjeta gráfica paravirtualizada (**QXL**).

Dispositivos compartidos físicamente

Algunas plataformas de hardware permiten a las máquinas virtuales acceder directamente a varios dispositivos y componentes de hardware. Este proceso se conoce como *device assignment* o *passthrough*.

Cuando se conecta de esta manera, algunos aspectos del dispositivo físico están directamente disponibles para la VM como lo estarían para una máquina física. Esto proporciona un rendimiento superior para el dispositivo cuando se utiliza en la VM. Sin embargo, los dispositivos conectados físicamente a una VM no están disponibles para el host, y tampoco pueden ser migrados.

Sin embargo, algunos dispositivos pueden ser *shared* a través de múltiples VMs. Por ejemplo, un único dispositivo físico puede, en ciertos casos, proporcionar múltiples *mediated devices*, que pueden ser asignados a distintas máquinas virtuales.

Se admiten los siguientes tipos de dispositivos de paso:

- Asignación de dispositivos de E/S de funciones virtuales (VFIO): expone de forma segura los dispositivos a las aplicaciones o máquinas virtuales utilizando DMA reforzado por hardware y aislamiento de interrupciones.

- USB, PCI y SCSI passthrough: exponen los buses estándar comunes de la industria directamente a las máquinas virtuales para que sus características específicas estén disponibles para el software invitado.
- Virtualización de E/S de raíz única (SR-IOV): una especificación que permite el aislamiento reforzado por hardware de los recursos PCI Express. Esto hace que sea seguro y eficaz dividir un único recurso PCI físico en funciones PCI virtuales. Se utiliza habitualmente para las tarjetas de interfaz de red (NIC).
- Virtualización de N_Port ID (NPIV) - una tecnología de Fibre Channel para compartir un único adaptador de bus de host (HBA) físico con múltiples puertos virtuales.
- GPUs y vGPUs - aceleradores para tipos específicos de cargas de trabajo gráficas o de computación. Algunas GPU pueden conectarse directamente a una máquina virtual, mientras que algunos tipos también ofrecen la posibilidad de crear GPU virtuales (vGPU) que comparten el hardware físico subyacente.

10.6. GESTIÓN DE DISPOSITIVOS USB VIRTUALES

Cuando se utiliza una máquina virtual (VM), se puede acceder y controlar un dispositivo USB, como una unidad flash o una cámara web, que está conectado al sistema anfitrión. En este escenario, el sistema anfitrión pasa el control del dispositivo a la VM. Esto también se conoce como USB-passthrough.

Las siguientes secciones proporcionan información sobre el uso de la línea de comandos para:

- [Adjuntar un dispositivo USB](#) a una VM
- [Eliminar un dispositivo USB](#) de una VM

10.6.1. Adjuntar dispositivos USB a las máquinas virtuales

Para adjuntar un dispositivo USB a una máquina virtual (VM), puede incluir la información del dispositivo USB en el archivo de configuración XML de la VM.

Requisitos previos

- Asegúrese de que el dispositivo que desea pasar a la VM está conectado al host.

Procedimiento

1. Localiza los valores del bus y del dispositivo del USB que quieres conectar a la VM. Por ejemplo, el siguiente comando muestra una lista de dispositivos USB conectados al host. El dispositivo que utilizaremos en este ejemplo está conectado en el bus 001 como dispositivo 005.

```
# lsusb
[...]
Bus 001 Device 003: ID 2567:0a2b Intel Corp.
Bus 001 Device 005: ID 0407:6252 Kingston River 2.0
[...]
```

2. Utilice la utilidad **virt-xml** junto con el argumento **--add-device**. Por ejemplo, el siguiente comando adjunta una unidad flash USB a la VM **Library**.

■

```
# virt-xml Library --add-device --hostdev 001.005
```

Domain 'Library' defined successfully.



NOTA

Para adjuntar un dispositivo USB a una máquina virtual en funcionamiento, añada el argumento **--update** al comando anterior.

Pasos de verificación

- Ejecute la VM y compruebe si el dispositivo está presente y funciona como se espera.
- Utilice el comando **virsh dumpxml** para ver si la definición XML del dispositivo se ha añadido a la sección <devices> en el archivo de configuración XML de la máquina virtual.

```
# virsh dumpxml Library
```

```
[...]
```

```
<hostdev mode='subsystem' type='usb' managed='yes'>
```

```
<source>
```

```
<vendor id='0x0407'>
```

```
<product id='0x6252'>
```

```
<address bus='1' device='5'>
```

```
</source>
```

```
<alias name='hostdev0'>
```

```
<address type='usb' bus='0' port='3'>
```

```
</hostdev>
```

```
[...]
```

Recursos adicionales

- Para otros argumentos, consulte la página man de virt-xml(1).

10.6.2. Eliminación de los dispositivos USB de las máquinas virtuales

Para eliminar un dispositivo USB de una máquina virtual (VM), puede eliminar la información del dispositivo USB de la configuración XML de la VM.

Procedimiento

1. Localiza los valores del bus y del dispositivo del USB que quieres eliminar de la VM. Por ejemplo, el siguiente comando muestra una lista de dispositivos USB conectados al host. El dispositivo que utilizaremos en este ejemplo está conectado en el bus 001 como dispositivo 005.

```
# lsusb
```

```
[...]
```

```
Bus 001 Device 003: ID 2567:0a2b Intel Corp.
```

```
Bus 001 Device 005: ID 0407:6252 Kingston River 2.0
```

```
[...]
```

2. Utilice la utilidad **virt-xml** junto con el argumento **--remove-device**. Por ejemplo, el siguiente comando elimina una unidad flash USB, conectada al host como dispositivo 005 en el bus 001, de la VM **Library**.

```
# virt-xml Library --remove-device --hostdev 001.005
Domain 'Library' defined successfully.
```



NOTA

Para eliminar un dispositivo USB de una máquina virtual en funcionamiento, añada el argumento **--update** al comando anterior.

Pasos de verificación

- Ejecute la VM y compruebe si el dispositivo ha sido eliminado de la lista de dispositivos.

Recursos adicionales

- Para otros argumentos, consulte la página man de virt-xml(1).

10.6.3. Recursos adicionales

- Para obtener información sobre la gestión de otros tipos de dispositivos, consulte [Sección 10.2, “Adjuntar dispositivos a las máquinas virtuales”](#).

10.7. GESTIÓN DE UNIDADES ÓPTICAS VIRTUALES

Cuando se utiliza una máquina virtual (VM), se puede acceder a la información almacenada en una imagen ISO en el host. Para ello, conecte la imagen ISO a la VM como una unidad óptica virtual, como una unidad de CD o una unidad de DVD.

Las siguientes secciones proporcionan información sobre el uso de la línea de comandos para:

- [Adjuntar una unidad y una imagen ISO](#) a una VM
- [Sustituir una imagen ISO](#) en una unidad óptica virtual
- [Eliminar una imagen ISO](#) de una unidad óptica virtual
- [Eliminar una unidad de](#) la VM

10.7.1. Adjuntar unidades ópticas a las máquinas virtuales

Para adjuntar una imagen ISO como unidad óptica virtual, edite el archivo de configuración XML de la máquina virtual (VM) y añada la nueva unidad.

Requisitos previos

- Debe almacenar la imagen ISO en el host local.
- Debe conocer la ruta de acceso a la imagen ISO.

Procedimiento

- Utilice la utilidad **virt-xml** con el argumento **--add-device**.
Por ejemplo, el siguiente comando adjunta la imagen ISO **Doc10**, almacenada en el directorio **/MC/tank/**, a la VM **DN1**.

```
# virt-xml DN1 --add-device --disk /MC/tank/Doc10.iso,device=cdrom
Domain 'DN1' defined successfully.
```

Pasos de verificación

- Ejecute la VM y compruebe si el dispositivo está presente y funciona como se espera.

Recursos adicionales

- Para otros argumentos, consulte la página man de virt-xml(1).

10.7.2. Sustitución de imágenes ISO en unidades ópticas virtuales

Para sustituir una imagen ISO adjunta como unidad óptica virtual a una máquina virtual (VM), edite el archivo de configuración XML de la VM y especifique la sustitución.

Requisitos previos

- Debe almacenar la imagen ISO en el host local.
- Debe conocer la ruta de acceso a la imagen ISO.

Procedimiento

1. Localice el dispositivo de destino donde el CD-ROM está conectado a la VM. Puede encontrar esta información en el archivo de configuración XML de la VM. Por ejemplo, el siguiente comando muestra el archivo de configuración XML de la VM **DN1**, donde el dispositivo de destino para el CD-ROM es **sda**.

```
# virsh dumpxml DN1
...
<disk>
...
  <source file='/MC/tank/Doc10.iso'/>
  <target dev='sda' bus='sata'/>
...
</disk>
...
```

2. Utilice la utilidad **virt-xml** con el argumento **--edit**. Por ejemplo, el siguiente comando sustituye la imagen ISO **Doc10**, adjunta a la máquina virtual **DN1** en el destino **sda**, por la imagen ISO **DrDN** almacenada en el directorio **/Dvrs/current/**.

```
# virt-xml DN1 --edit target=sda --disk /Dvrs/current/DrDN.iso
Domain 'DN1' defined successfully.
```

Pasos de verificación

- Ejecute la VM y compruebe si el dispositivo se sustituye y funciona como se espera.

Recursos adicionales

- Para otros argumentos, consulte la página man de virt-xml(1).

10.7.3. Eliminación de imágenes ISO de unidades ópticas virtuales

Para eliminar una imagen ISO de una unidad óptica virtual conectada a una máquina virtual (VM), edite el archivo de configuración XML de la VM.

Procedimiento

1. Localice el dispositivo de destino donde el CD-ROM está conectado a la VM. Puede encontrar esta información en el archivo de configuración XML de la VM.
Por ejemplo, el siguiente comando muestra el archivo de configuración XML de la VM **DN1**, donde el dispositivo de destino para el CD-ROM es **sda**.

```
# virsh dumpxml DN1
...
<disk>
  ...
  <source file='/Dvrs/current/DrDN'/>
  <target dev='sda' bus='sata'/>
  ...
</disk>
...
```

2. Utilice la utilidad **virt-xml** con el argumento **--edit**.
Por ejemplo, el siguiente comando elimina la imagen ISO **DrDN** de la unidad de CD conectada a la VM **DN1**.

```
# virt-xml DN1 --edit target=sda --disk path=
Domain 'DN1' defined successfully.
```

Pasos de verificación

- Ejecuta la VM y comprueba que la imagen ya no está disponible.

Recursos adicionales

- Para otros argumentos, consulte la página man de virt-xml(1).

10.7.4. Eliminación de las unidades ópticas de las máquinas virtuales

Para eliminar una unidad óptica conectada a una máquina virtual (VM), edite el archivo de configuración XML de la VM.

Procedimiento

1. Localice el dispositivo de destino donde el CD-ROM está conectado a la VM. Puede encontrar esta información en el archivo de configuración XML de la VM.
Por ejemplo, el siguiente comando muestra el archivo de configuración XML de la VM **DN1**, donde el dispositivo de destino para el CD-ROM es **sda**.

```
# virsh dumpxml DN1
...
<disk type='file' device='cdrom'>
  <driver name='qemu' type='raw'/>
  ...
</disk>
...
```



```
<target dev='sda' bus='sata'/>
...
</disk>
...
```

- Utilice la utilidad **virt-xml** con el argumento **--remove-device**.

Por ejemplo, el siguiente comando elimina la unidad óptica conectada como destino **sda** de la VM **DN1**.

```
# virt-xml DN1 --remove-device --disk target=sda
Domain 'DN1' defined successfully.
```

Pasos de verificación

- Confirme que el dispositivo ya no aparece en el archivo de configuración XML de la máquina virtual.

Recursos adicionales

- Para otros argumentos, consulte la página man de virt-xml(1).

10.7.5. Recursos adicionales

- Para obtener información sobre la gestión de otros tipos de dispositivos, consulte [Sección 10.2, “Adjuntar dispositivos a las máquinas virtuales”](#).

10.8. GESTIÓN DE DISPOSITIVOS SR-IOV

Un dispositivo virtual emulado suele utilizar más CPU y memoria que un dispositivo de red de hardware. Esto puede limitar el rendimiento de una máquina virtual (VM). Sin embargo, si alguno de los dispositivos de su host de virtualización es compatible con la virtualización de E/S de raíz única (SR-IOV), puede utilizar esta función para mejorar el rendimiento del dispositivo, y posiblemente también el rendimiento general de sus máquinas virtuales.

10.8.1. ¿Qué es el SR-IOV?

La virtualización de E/S de raíz única (SR-IOV) es una especificación que permite que un único dispositivo PCI Express (PCIe) presente varios dispositivos PCI independientes, denominados *virtual functions* (VFs), al sistema anfitrión. Cada uno de estos dispositivos:

- Es capaz de proporcionar el mismo o similar servicio que el dispositivo PCIe original.
- Aparece en una dirección diferente en el bus PCI del host.
- Se puede asignar a una VM diferente mediante la asignación de VFIO.

Por ejemplo, un único dispositivo de red con capacidad SR-IOV puede presentar VFs a múltiples VMs. Mientras que todas las VFs utilizan la misma tarjeta física, la misma conexión de red y el mismo cable de red, cada una de las VMs controla directamente su propio dispositivo de red de hardware y no utiliza recursos adicionales del host.

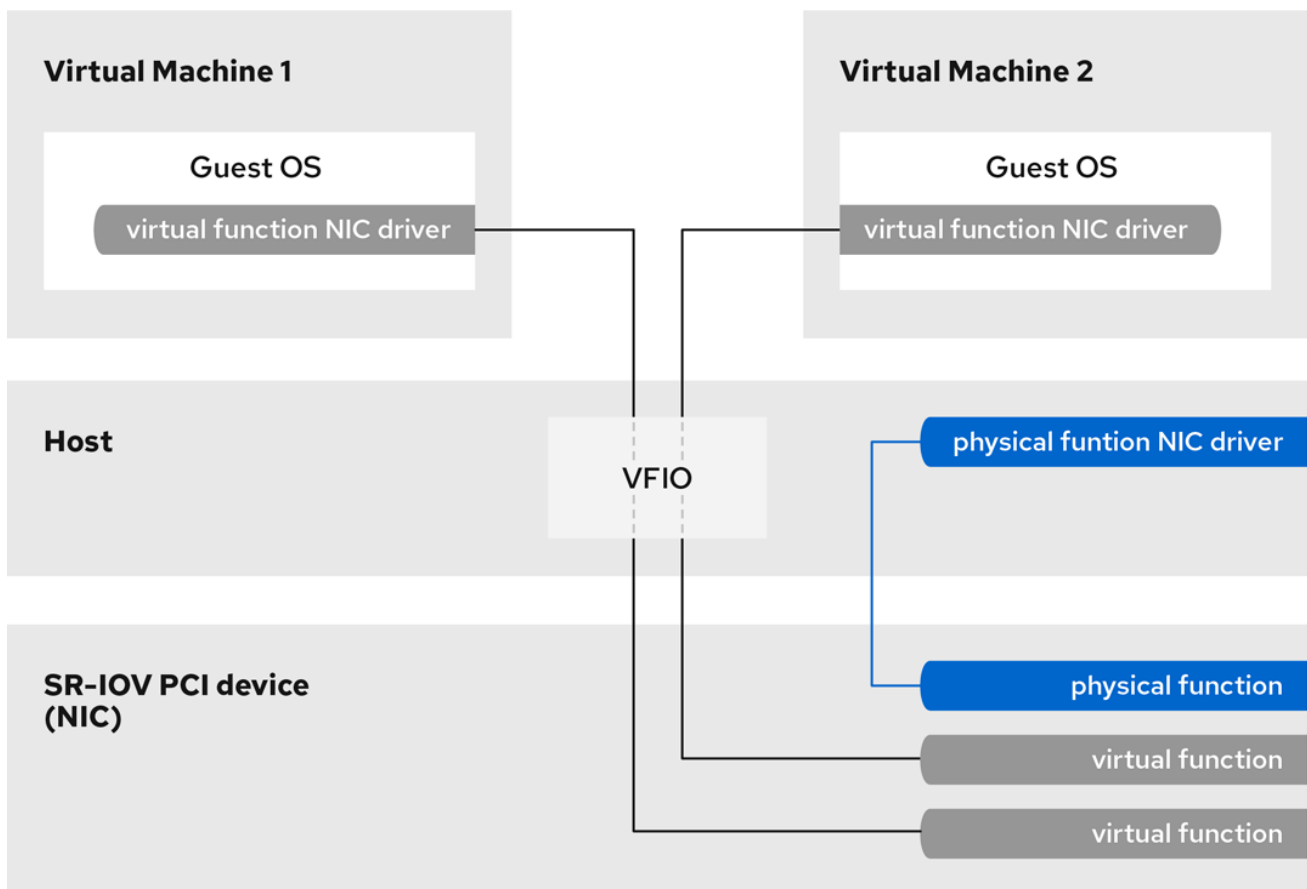
Cómo funciona el SR-IOV

La funcionalidad SR-IOV es posible gracias a la introducción de las siguientes funciones PCIe:

- **Physical functions (PFs)** - Una función PCIe que proporciona la funcionalidad de su dispositivo (por ejemplo, la red) al host, pero también puede crear y gestionar un conjunto de FPs. Cada dispositivo con capacidad SR-IOV tiene uno o más PFs.
- **Virtual functions (VFs)** - Funciones PCIe ligeras que se comportan como dispositivos independientes. Cada VF se deriva de un PF. El número máximo de VFs que puede tener un dispositivo depende del hardware del mismo. Cada VF sólo puede ser asignado a una sola VM a la vez, pero una VM puede tener múltiples VFs asignados.

Las VMs reconocen los VFs como dispositivos virtuales. Por ejemplo, un VF creado por un dispositivo de red SR-IOV aparece como una tarjeta de red para una VM a la que se asigna, de la misma manera que una tarjeta de red física aparece para el sistema anfitrión.

Figura 10.1. Arquitectura SR-IOV



Beneficios

Las principales ventajas de utilizar VFs SR-IOV en lugar de dispositivos emulados son:

- Mejora del rendimiento
- Uso reducido de los recursos de la CPU y la memoria del host

Por ejemplo, una VF conectada a una VM como vNIC rinde casi al mismo nivel que una NIC física, y mucho mejor que las NIC paravirtualizadas o emuladas. En particular, cuando se utilizan múltiples VFs simultáneamente en un solo host, los beneficios de rendimiento pueden ser significativos.

Desventajas

- Para modificar la configuración de un FP, primero debe cambiar el número de VFs expuestos por el FP a cero. Por lo tanto, también debe eliminar los dispositivos proporcionados por estos VFs de la VM a la que están asignados.
- Una VM con dispositivos asignados a VFIOs, incluyendo VFs SR-IOV, no puede ser migrada a otro host. En algunos casos, se puede sortear esta limitación emparejando el dispositivo asignado con un dispositivo emulado. Por ejemplo, puede [vincular](#) un VF de red asignado a una vNIC emulada y eliminar el VF antes de la migración.
- Además, los dispositivos asignados por VFIO requieren el pinning de la memoria de la VM, lo que aumenta el consumo de memoria de la VM y evita el uso del ballooning de memoria en la VM.

Recursos adicionales

- Para ver una lista de los tipos de dispositivos que admiten SR-IOV, consulte [Sección 10.8.3, "Dispositivos compatibles con la asignación de SR-IOV"](#).

10.8.2. Adjuntar dispositivos de red SR-IOV a máquinas virtuales

Para adjuntar un dispositivo de red SR-IOV a una máquina virtual (VM) en un host Intel o AMD, debe crear una función virtual (VF) desde una interfaz de red con capacidad SR-IOV en el host y asignar la VF como dispositivo a una VM específica. Para más detalles, consulte las siguientes instrucciones.

Requisitos previos

- La CPU y el firmware de su host soportan la Unidad de Gestión de Memoria de E/S (IOMMU).
 - Si se utiliza una CPU Intel, ésta debe ser compatible con la tecnología de virtualización Intel para E/S dirigida (VT-d).
 - Si se utiliza una CPU de AMD, debe ser compatible con la función AMD-Vi.
- El sistema anfitrión utiliza el Servicio de Control de Acceso (ACS) para proporcionar aislamiento de acceso directo a la memoria (DMA) para la topología PCIe. Verifique esto con el proveedor del sistema.
Para obtener información adicional, consulte [Consideraciones sobre el hardware para implementar SR-IOV](#).
- El dispositivo de red físico soporta SR-IOV. Para verificar si algún dispositivo de red de su sistema admite SR-IOV, utilice el comando **lspci -v** y busque **Single Root I/O Virtualization (SR-IOV)** en la salida.

```
# lspci -v
[...]
02:00.0 Ethernet controller: Intel Corporation 82576 Gigabit Network Connection (rev 01)
Subsystem: Intel Corporation Gigabit ET Dual Port Server Adapter
Flags: bus master, fast devsel, latency 0, IRQ 16, NUMA node 0
Memory at fcba0000 (32-bit, non-prefetchable) [size=128K]
[...]
Capabilities: [150] Alternative Routing-ID Interpretation (ARI)
Capabilities: [160] Single Root I/O Virtualization (SR-IOV)
Kernel driver in use: igb
Kernel modules: igb
[...]
```

- La interfaz de red del host que desea utilizar para crear VFs está en funcionamiento. Por ejemplo, para activar la interfaz `eth1` y verificar que se está ejecutando:

```
# ip link set eth1 up
# ip link show eth1
8: eth1: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc mq state UP mode
DEFAULT qlen 1000
    link/ether a0:36:9f:8f:3f:b8 brd ff:ff:ff:ff:ff:ff
    vf 0 MAC 00:00:00:00:00:00, spoof checking on, link-state auto
    vf 1 MAC 00:00:00:00:00:00, spoof checking on, link-state auto
    vf 2 MAC 00:00:00:00:00:00, spoof checking on, link-state auto
    vf 3 MAC 00:00:00:00:00:00, spoof checking on, link-state auto
```

- Para que la asignación de dispositivos SR-IOV funcione, la función IOMMU debe estar habilitada en la BIOS y el kernel del host. Para ello:

- En un host Intel, active VT-d:

- Si su host Intel utiliza múltiples entradas de arranque:

- A. Edite el archivo `/etc/default/grub` y añada los parámetros `intel_iommu=on` y `iommu=pt` al final de la línea `GRUB_CMDLINE_LINUX`:

```
GRUB_CMDLINE_LINUX="crashkernel=auto resume=/dev/mapper/rhel_dell-
per730-27-swap rd.lvm.lv=rhel_dell-per730-27/root rd.lvm.lv=rhel_dell-per730-
27/swap console=ttyS0,115200n81 intel_iommu=on iommu=pt"
```

- B. Regenerar la configuración de GRUB:

```
# grub2-mkconfig -o /boot/grub2/grub.cfg
```

- C. Reinicia el host.

- Si su host Intel utiliza una única entrada de arranque:

- A. Regenerar la configuración de GRUB con el parámetro `intel_iommu=on`:

```
# grubby --args="intel_iommu=on" --update-kernel DEFAULT
```

- B. Reinicia el host.

- En un host AMD, active AMD-Vi:

- Si su host AMD utiliza múltiples entradas de arranque:

- A. Edite el archivo `/etc/default/grub` y añada los parámetros `iommu=pt` y `amd_iommu=on` al final de la línea `GRUB_CMDLINE_LINUX`:

```
GRUB_CMDLINE_LINUX="crashkernel=auto resume=/dev/mapper/rhel_dell-
per730-27-swap rd.lvm.lv=rhel_dell-per730-27/root rd.lvm.lv=rhel_dell-per730-
27/swap console=ttyS0,115200n81 iommu=pt amd_iommu=on"
```

- B. Regenerar la configuración de GRUB:

```
# grub2-mkconfig -o /boot/grub2/grub.cfg
```

C. Reinicia el host.

- Si su host AMD utiliza una única entrada de arranque:

A. Regenerar la configuración de GRUB con el parámetro **iommu=pt**:

```
# grubby --args="iommu=pt" --update-kernel DEFAULT
```

B. Reinicia el host.

Procedimiento

1. **Optional:** Confirme el número máximo de VFs que su dispositivo de red puede utilizar. Para ello, utilice el siguiente comando y sustituya *eth1* por su dispositivo de red compatible con SR-IOV.

```
# cat /sys/class/net/eth1/device/sriov_totalvfs
7
```

2. Utilice el siguiente comando para crear una función virtual (VF):

```
# echo VF-number > /sys/class/net/network-interface/device/sriov_numvfs
```

En el comando, reemplaza:

- *VF-number* con el número de VFs que desea crear en el FP.
- *network-interface* con el nombre de la interfaz de red para la que se crearán los VF.

El siguiente ejemplo crea 2 VFs desde la interfaz de red *eth1*:

```
# echo 2 > /sys/class/net/eth1/device/sriov_numvfs
```

3. Compruebe que se han añadido los VF:

```
# lspci | grep Ethernet
01:00.0 Ethernet controller: Intel Corporation Ethernet Controller 10-Gigabit X540-AT2 (rev 01)
01:00.1 Ethernet controller: Intel Corporation Ethernet Controller 10-Gigabit X540-AT2 (rev 01)
07:00.0 Ethernet controller: Intel Corporation I350 Gigabit Network Connection (rev 01)
07:00.1 Ethernet controller: Intel Corporation I350 Gigabit Network Connection (rev 01)
```

4. Haz que los FVs creados sean persistentes creando una regla udev para la interfaz de red que utilizaste para crear los FVs. Por ejemplo, para la interfaz *eth1*, cree el archivo **/etc/udev/rules.d/eth1.rules** y añada la siguiente línea:

```
ACTION=="add", SUBSYSTEM=="net", ENV{ID_NET_DRIVER}=="ixgbe",
ATTR{device/sriov_numvfs}="2"
```

Esto garantiza que los dos VF que utilizan el controlador **ixgbe** estarán automáticamente disponibles para la interfaz **eth1** cuando se inicie el host.



AVISO

Actualmente, este comando no funciona correctamente cuando se intenta hacer persistentes los FV en los adaptadores Broadcom NetXtreme II BCM57810. Además, actualmente no es fiable adjuntar VFs basados en estos adaptadores a VMs de Windows.

- Utilice el comando **virsh nodedev-list** para verificar que *libvirt* reconoce los dispositivos VF añadidos. Por ejemplo, lo siguiente muestra que los FPs 01:00.0 y 07:00.0 del ejemplo anterior se han convertido con éxito en VFs:

```
# virsh nodedev-list | grep pci_
pci_0000_01_00_0
pci_0000_01_00_1
pci_0000_07_10_0
pci_0000_07_10_1
[...]
```

- Obtenga los valores **bus**, **slot**, y **function** de un FP y uno de sus correspondientes VF. Por ejemplo, para **pci_0000_01_00_0** y **pci_0000_01_00_1**:

```
# virsh nodedev-dumpxml pci_0000_01_00_0
<device>
  <name>pci_0000_01_00_0</name>
  <path>/sys/devices/pci0000:00/0000:00:01.0/0000:01:00.0</path>
  <parent>pci_0000_00_01_0</parent>
  <driver>
    <name>ixgbe</name>
  </driver>
  <capability type='pci'>
    <domain>0</domain>
    <bus>1</bus>
    <slot>0</slot>
    <function>0</function>
[...]
```

```
# virsh nodedev-dumpxml pci_0000_01_00_1
<device>
  <name>pci_0000_01_00_1</name>
  <path>/sys/devices/pci0000:00/0000:00:01.0/0000:01:00.1</path>
  <parent>pci_0000_00_01_0</parent>
  <driver>
    <name>vfio-pci</name>
  </driver>
  <capability type='pci'>
    <domain>0</domain>
    <bus>1</bus>
    <slot>0</slot>
    <function>1</function>
[...]
```

7. Cree un archivo XML temporal y añada una configuración en utilizando los valores **bus**, **slot**, y **function** que obtuvo en el paso anterior. Por ejemplo:

```
<interface type='hostdev' managed='yes'>
  <source>
    <address type='pci' domain='0x0000' bus='0x03' slot='0x10' function='0x2'/>
  </source>
</interface>
```

8. Añada el VF a una VM utilizando el archivo XML temporal. Por ejemplo, lo siguiente adjunta un VF guardado en el **/tmp/holdmyfunction.xml** a una VM *testguest1* en ejecución y asegura que esté disponible después de que la VM se reinicie:

```
# virsh attach-device testguest1 /tmp/holdmyfunction.xml --live --config
Device attached successfully.
```

Si esto tiene éxito, el sistema operativo invitado detecta una nueva tarjeta de interfaz de red.

10.8.3. Dispositivos compatibles con la asignación de SR-IOV

No todos los dispositivos pueden utilizarse para SR-IOV. Los siguientes dispositivos han sido probados y verificados como compatibles con SR-IOV en RHEL 8.

Dispositivos de red

- Controlador Intel 82599ES 10 Gigabit Ethernet - utiliza el controlador **ixgbe**
- Controlador Intel Ethernet Serie XL710 - utiliza el controlador **i40e**
- Tarjetas adaptadoras Ethernet Mellanox ConnectX-5: utilice el controlador **mlx5_core**
- Adaptador de red Intel Ethernet XXV710 - utiliza el controlador **i40e**
- Controlador Intel 82576 Gigabit Ethernet - utiliza el controlador **igb**
- Broadcom NetXtreme II BCM57810 - utiliza el controlador **bnx2x**

10.9. ADJUNTAR DISPOSITIVOS DASD A MÁQUINAS VIRTUALES EN IBM Z

Los dispositivos de almacenamiento de acceso directo (DASD) proporcionan una serie de características de almacenamiento específicas. Utilizando la función **vfio-ccw**, puede asignar DASDs como dispositivos mediados a sus máquinas virtuales (VMs) en hosts IBM Z. Esto, por ejemplo, hace posible que la VM acceda a un conjunto de datos de z/OS, o que comparta los DASDs asignados con una máquina z/OS.

Requisitos previos

- Su sistema anfitrión utiliza la arquitectura de hardware IBM Z y soporta el protocolo FICON.
- La máquina virtual de destino utiliza un sistema operativo invitado Linux.
- Los módulos del kernel necesarios se han cargado en el host. Para comprobarlo, utilice:

```
# lsmod | grep vfio
```

La salida debe contener los siguientes módulos:

- **vfio_ccw**
 - **vfio_mdev**
 - **vfio_iommu_type1**
- Tiene un dispositivo DASD de repuesto para uso exclusivo de la VM.

Procedimiento

1. Obtenga el identificador del dispositivo DASD. La utilidad **lsdasd** lo muestra como **Bus-ID**.

```
# lsdasd
Bus-ID  Status  Name    Device Type      BlkSz Size   Blocks
=====
=====
0.0.002c active  dasdh   94:0  ECKD      4096 21129MB 5409180
```

En los siguientes comandos de este procedimiento, sustituya **0.0.002c** por el identificador de dispositivo detectado de su dispositivo.

2. Obtener la ruta del subcanal del dispositivo DASD.

```
# lscss | grep 0.0.002c
0.0.002c 0.0.24ac 3390/0c 3990/e9 yes f0 f0 ff 01021112 00000000
```

En este ejemplo, la ruta del subcanal se detecta como **0.0.24ac**. En los siguientes comandos de este procedimiento, sustituya **0.0.24ac** por la ruta de subcanal detectada de su dispositivo.

3. Desvincula el dispositivo DASD de su subcanal en el host.

```
# echo 0.0.002c > /sys/bus/ccw/devices/0.0.002c/driver/unbind
```

4. Desvincula el subcanal del controlador del subcanal de E/S.

```
# echo 0.0.24ac > /sys/bus/css/devices/0.0.24ac/driver/unbind
```

5. Vincula el subcanal al controlador de paso de **vfio_ccw**.

```
# echo 0.0.24ac > /sys/bus/css/drivers/vfio_ccw/bind
```

6. Generar un UUID.

```
# uuidgen
30820a6f-b1a5-4503-91ca-0c10ba12345a
```

7. Crear el dispositivo mediado por DASD utilizando el UUID generado


```
# echo "30820a6f-b1a5-4503-91ca-0c10ba12345a" >
/sys/bus/css/devices/0.0.24ac/mdev_supported_types/vfio_ccw-io/create
```

- Adjunte el dispositivo mediado a la VM. Para ello, utilice la utilidad **virsh edit** para editar la configuración XML de la VM, añada la siguiente sección al XML y sustituya el valor **uuid** por el UUID que generó en el paso anterior.

```
<hostdev mode='subsystem' type='mdev' model='vfio-ccw'>
  <source>
    <address uuid="30820a6f-b1a5-4503-91ca-0c10ba12345a"/>
  </source>
</hostdev>
```

Verificación

- Obtenga el identificador que **libvirt** asignó al dispositivo DASD mediado. Para ello, visualice la configuración XML de la VM y busque un dispositivo **vfio-ccw**.

```
# virsh dumpxml vm-name

<domain>
[...]
  <hostdev mode='subsystem' type='mdev' managed='no' model='vfio-ccw'>
    <source>
      <address uuid='10620d2f-ed4d-437b-8aff-beda461541f9'/>
    </source>
    <alias name='hostdev0'/>
    <address type='ccw' cssid='0xfe' ssid='0x0' devno='0x0009'/>
  </hostdev>
[...]
</domain>
```

En este ejemplo, el identificador asignado al dispositivo es **0.0.0009**.

- Inicie sesión en el sistema operativo invitado de la VM y confirme que el dispositivo aparece en la lista. Por ejemplo:

```
# lscss | grep 0.0.0009
0.0.0009 0.0.0007 3390/0c 3990/e9   f0 f0 ff 12212231 00000000
```

- Poner el dispositivo en línea. Por ejemplo:

```
# chccwdev -e 0.0.0009
Setting device 0.0.0009 online
Done
```

CAPÍTULO 11. GESTIÓN DEL ALMACENAMIENTO DE LAS MÁQUINAS VIRTUALES

Puede gestionar el almacenamiento de las máquinas virtuales mediante la [CLI](#) o la [consola web](#).

Esta documentación proporciona información sobre cómo gestionar el almacenamiento de las máquinas virtuales mediante el comando **virsh**.

11.1. COMPRENDER EL ALMACENAMIENTO DE LAS MÁQUINAS VIRTUALES

Las siguientes secciones proporcionan información sobre el almacenamiento para las máquinas virtuales (VM), incluyendo información sobre los pools de almacenamiento, los volúmenes de almacenamiento y cómo se utilizan para proporcionar almacenamiento a las VM.

11.1.1. Almacenamiento de máquinas virtuales

A continuación se ofrece información sobre cómo se utilizan los pools de almacenamiento y los volúmenes de almacenamiento para crear almacenamiento para las máquinas virtuales (VM).

Un *storage pool* es una cantidad de almacenamiento gestionada por el host y reservada para su uso por las VMs. *Storage volumes* puede crearse a partir del espacio de los pools de almacenamiento. Cada volumen de almacenamiento puede ser asignado a una VM como un dispositivo de bloque, como un disco, en un bus de huésped.

Los grupos de almacenamiento y los volúmenes se gestionan mediante **libvirt**. Con el protocolo remoto **libvirt**, se pueden gestionar todos los aspectos del almacenamiento de las máquinas virtuales. Estas operaciones pueden realizarse en un host remoto. Como resultado, una aplicación de gestión que utiliza **libvirt**, como la consola web de RHEL, puede utilizarse para realizar todas las tareas necesarias para configurar el almacenamiento de una VM.

La API **libvirt** puede utilizarse para consultar la lista de volúmenes del pool de almacenamiento o para obtener información sobre la capacidad, la asignación y el almacenamiento disponible en el pool de almacenamiento. Un volumen de almacenamiento en el pool de almacenamiento puede ser consultado para obtener información como la asignación y la capacidad, que puede ser diferente para los volúmenes dispersos.

Para los grupos de almacenamiento que lo soportan, la API **libvirt** puede utilizarse para crear, clonar, redimensionar y eliminar volúmenes de almacenamiento. Las API también pueden utilizarse para cargar datos en volúmenes de almacenamiento, descargar datos de volúmenes de almacenamiento o borrar datos de volúmenes de almacenamiento.

Una vez que se inicia un pool de almacenamiento, se puede asignar un volumen de almacenamiento a una VM utilizando el nombre del pool de almacenamiento y el nombre del volumen de almacenamiento en lugar de la ruta del host al volumen en los archivos de configuración XML de la VM.

11.1.2. Piscinas de almacenamiento

Un pool de almacenamiento es un archivo, directorio o dispositivo de almacenamiento, gestionado por **libvirt** para proporcionar almacenamiento a las máquinas virtuales (VM). Los pools de almacenamiento se dividen en volúmenes de almacenamiento que almacenan las imágenes de las VMs o se adjuntan a las VMs como almacenamiento adicional. Varias máquinas virtuales pueden compartir el mismo pool de almacenamiento, lo que permite una mejor asignación de los recursos de almacenamiento.

Los pools de almacenamiento pueden ser persistentes o transitorios:

- Un grupo de almacenamiento persistente sobrevive a un reinicio del sistema de la máquina anfitriona.
- Un pool de almacenamiento transitorio sólo existe hasta que el host se reinicia.

El comando **virsh pool-define** se utiliza para crear un pool de almacenamiento persistente, y el comando **virsh pool-create** se utiliza para crear un pool de almacenamiento transitorio.

Tipos de almacenamiento del pool de almacenamiento

Los pools de almacenamiento pueden ser locales o en red (compartidos):

- **Local storage pools**

Los grupos de almacenamiento local están conectados directamente al servidor anfitrión. Incluyen los directorios locales, los discos conectados directamente, las particiones físicas y los grupos de volúmenes de Logical Volume Management (LVM) en dispositivos locales.

Los grupos de almacenamiento local son útiles para el desarrollo, las pruebas y las pequeñas implantaciones que no requieren migración o un gran número de máquinas virtuales.

- **Networked (shared) storage pools**

Los grupos de almacenamiento en red incluyen dispositivos de almacenamiento compartidos a través de una red utilizando protocolos estándar.

Ejemplo de uso del pool de almacenamiento

Para ilustrar las opciones disponibles para la gestión de pools de almacenamiento, a continuación se describe un ejemplo de servidor NFS que utiliza **mount -t nfs nfs.example.com:/path/to/share /path/to/data**.

Un administrador de almacenamiento podría definir un Pool de Almacenamiento NFS en el host de virtualización para describir la ruta del servidor exportado y la ruta de destino del cliente. Esto permitirá a **libvirt** realizar el montaje de forma automática cuando se inicie **libvirt** o según sea necesario mientras se ejecute **libvirt**. Los archivos con el directorio exportado del servidor NFS aparecen como volúmenes de almacenamiento dentro del pool de almacenamiento NFS.

Cuando el volumen de almacenamiento se añade a la VM, el administrador no necesita añadir la ruta de destino al volumen. Sólo necesita añadir el pool de almacenamiento y el volumen de almacenamiento por su nombre. Por lo tanto, si la ruta del cliente de destino cambia, no afecta a la VM.

Cuando se inicia el pool de almacenamiento, **libvirt** monta el disco compartido en el directorio especificado, igual que si el administrador del sistema iniciara la sesión y ejecutara **mount nfs.example.com:/path/to/share /vmdata**. Si el pool de almacenamiento está configurado para autoiniciarse, **libvirt** se asegura de que el disco compartido NFS se monte en el directorio especificado cuando se inicie **libvirt**.

Una vez iniciado el pool de almacenamiento, los archivos del disco compartido NFS se reportan como volúmenes de almacenamiento, y las rutas de los volúmenes de almacenamiento pueden ser consultadas usando la API **libvirt**. Las rutas de los volúmenes de almacenamiento pueden entonces copiarse en la sección de la definición XML de una VM que describe el almacenamiento de origen para los dispositivos de bloque de la VM. En el caso de NFS, una aplicación que utilice la API **libvirt** puede crear y eliminar volúmenes de almacenamiento en el pool de almacenamiento (archivos en el recurso compartido NFS) hasta el límite del tamaño del pool (la capacidad de almacenamiento del recurso compartido).

Detener (destruir) un pool de almacenamiento elimina la abstracción de los datos, pero mantiene los datos intactos.

No todos los tipos de pool de almacenamiento soportan la creación y eliminación de volúmenes. Detener el pool de almacenamiento (**pool-destroy**) deshace la operación de inicio, en este caso, desmontar el recurso compartido NFS. Los datos del recurso compartido no se modifican con la operación de destrucción, a pesar de lo que sugiere el nombre del comando. Para más detalles, consulte **man virsh**.

Tipos de grupos de almacenamiento admitidos y no admitidos

La siguiente es una lista de los tipos de pool de almacenamiento soportados por RHEL:

- Grupos de almacenamiento basados en directorios
- Grupos de almacenamiento en disco
- Grupos de almacenamiento basados en particiones
- Grupos de almacenamiento GlusterFS
- grupos de almacenamiento basados en iSCSI
- Grupos de almacenamiento basados en LVM
- Grupos de almacenamiento basados en NFS
- Grupos de almacenamiento basados en SCSI con dispositivos vHBA
- Grupos de almacenamiento basados en rutas múltiples
- Grupos de almacenamiento basados en RBD

La siguiente es una lista de tipos de pool de almacenamiento **libvirt** que no son compatibles con RHEL:

- Grupos de almacenamiento basados en Sheepdog
- Grupos de almacenamiento basados en Vstorage
- Grupos de almacenamiento basados en ZFS

11.1.3. Volúmenes de almacenamiento

Los pools de almacenamiento se dividen en **storage volumes**. Los volúmenes de almacenamiento son abstracciones de particiones físicas, volúmenes lógicos LVM, imágenes de disco basadas en archivos y otros tipos de almacenamiento manejados por **libvirt**. Los volúmenes de almacenamiento se presentan a las VMs como dispositivos de almacenamiento local, como discos, independientemente del hardware subyacente.

En la máquina anfitriona, se hace referencia a un volumen de almacenamiento por su nombre y un identificador para el grupo de almacenamiento del que deriva. En la línea de comandos **virsh**, esto toma la forma **--pool storage_pool volume_name**.

Por ejemplo, para mostrar información sobre un volumen llamado *firstimage* en el grupo *guest_images*.

```
# virsh vol-info --pool guest_images firstimage
Name:          firstimage
```

```
Type:      block
Capacity:  20.00 GB
Allocation: 20.00 GB
```

11.2. GESTIÓN DEL ALMACENAMIENTO DE LAS MÁQUINAS VIRTUALES MEDIANTE LA CLI

La siguiente documentación proporciona información sobre cómo gestionar el almacenamiento de la máquina virtual (VM) utilizando la utilidad de línea de comandos **virsh**.

A través de **virsh**, puede añadir, eliminar y modificar el almacenamiento de la máquina virtual, así como ver la información sobre el almacenamiento de la máquina virtual.



NOTA

En muchos casos, el almacenamiento para una VM se crea al mismo tiempo que [la VM](#). Por lo tanto, la siguiente información se refiere principalmente a la gestión avanzada del almacenamiento de la VM.

11.2.1. Visualización de la información de almacenamiento de las máquinas virtuales mediante la CLI

A continuación se proporciona información sobre cómo ver la información de los grupos de almacenamiento y los volúmenes de almacenamiento mediante la CLI.

11.2.1.1. Visualización de la información del pool de almacenamiento mediante la CLI

Utilizando la CLI, puede ver una lista de todos los grupos de almacenamiento con detalles limitados o completos sobre los grupos de almacenamiento. También puede filtrar los grupos de almacenamiento de la lista.

Procedimiento

- Utilice el comando **virsh pool-list** para ver la información del pool de almacenamiento.

```
# virsh pool-list --all --details
Name           State  Autostart Persistent  Capacity Allocation Available
default        running yes      yes      48.97 GiB 23.93 GiB 25.03 GiB
Downloads      running yes      yes      175.62 GiB 62.02 GiB 113.60 GiB
RHEL8-Storage-Pool running yes      yes      214.62 GiB 93.02 GiB 168.60 GiB
```

Recursos adicionales

- Para obtener información sobre las opciones disponibles en **virsh pool-list**, utilice el comando **virsh pool-list --help**.

11.2.1.2. Visualización de la información del volumen de almacenamiento mediante la CLI

A continuación se proporciona información sobre cómo ver la información de los grupos de almacenamiento. Puede ver una lista de todos los grupos de almacenamiento en un grupo de almacenamiento especificado y detalles sobre un grupo de almacenamiento especificado.

Procedimiento

1. Utilice el comando **virsh vol-list** para listar los volúmenes de almacenamiento en un pool de almacenamiento especificado.

```
# virsh vol-list --pool RHEL8-Storage-Pool --details
Name                Path                                Type Capacity Allocation
-----
.bash_history       /home/VirtualMachines/.bash_history file 18.70 KiB 20.00 KiB
.bash_logout        /home/VirtualMachines/.bash_logout file 18.00 B 4.00 KiB
.bash_profile       /home/VirtualMachines/.bash_profile file 193.00 B 4.00 KiB
.bashrc             /home/VirtualMachines/.bashrc      file 1.29 KiB 4.00 KiB
.git-prompt.sh     /home/VirtualMachines/.git-prompt.sh file 15.84 KiB 16.00 KiB
.gitconfig          /home/VirtualMachines/.gitconfig   file 167.00 B 4.00 KiB
RHEL8_Volume.qcow2 /home/VirtualMachines/RHEL8_Volume.qcow2 file 60.00 GiB
13.93 GiB
```

2. Utilice el comando **virsh vol-info** para listar los volúmenes de almacenamiento en un pool de almacenamiento especificado.

```
# vol-info --pool RHEL8-Storage-Pool --vol RHEL8_Volume.qcow2
Name:      RHEL8_Volume.qcow2
Type:      file
Capacity:  60.00 GiB
Allocation: 13.93 GiB
```

11.2.2. Creación y asignación de almacenamiento para máquinas virtuales mediante la CLI

A continuación se presenta un procedimiento de alto nivel para crear y asignar almacenamiento para máquinas virtuales (VM):

1. Create storage pools

Cree uno o más pools de almacenamiento a partir de los medios de almacenamiento disponibles. Para obtener una lista de los tipos de grupos de almacenamiento admitidos, consulte [Tipos de grupos de almacenamiento](#).

- Para crear grupos de almacenamiento persistentes, utilice los comandos **virsh pool-define-as** y **virsh pool-define**.
El comando **virsh pool-define-as** coloca las opciones en la línea de comandos. El comando **virsh pool-define** utiliza un archivo XML para las opciones del pool.
- Para crear grupos de almacenamiento temporal, utilice los comandos **virsh pool-create** y **virsh pool-create-as**.
El comando **virsh pool-create-as** coloca las opciones en la línea de comandos. El comando **virsh pool-create** utiliza un archivo XML para las opciones del pool.

2. Create storage volumes

Cree uno o más [volúmenes de almacenamiento](#) a partir de los pools de almacenamiento disponibles.

3. Assign storage devices to a VM

Asignar uno o más dispositivos de almacenamiento abstraídos de los volúmenes de almacenamiento a una VM.

Las siguientes secciones proporcionan información sobre la creación y asignación de almacenamiento mediante la CLI:

- [Almacenamiento basado en directorios](#)
- [Almacenamiento en disco](#)
- [Almacenamiento basado en sistemas de archivos](#)
- [Almacenamiento basado en GlusterFS](#)
- [almacenamiento basado en iSCSI](#)
- [Almacenamiento basado en LVM](#)
- [Almacenamiento basado en NFS](#)
- [Almacenamiento basado en SCSI con vHBA](#)

11.2.2.1. Creación y asignación de almacenamiento basado en directorios para máquinas virtuales mediante la CLI

A continuación se proporciona información sobre la creación de grupos de almacenamiento basados en directorios y volúmenes de almacenamiento, y la asignación de volúmenes a máquinas virtuales.

11.2.2.1.1. Creación de grupos de almacenamiento basados en directorios mediante la CLI

A continuación se ofrecen instrucciones para crear grupos de almacenamiento basados en directorios.

Requisitos previos

- Asegúrese de que su hipervisor admite grupos de almacenamiento de directorio:

```
# virsh pool-capabilities | grep "'dir' supported='yes'"
```

Si el comando muestra alguna salida, los grupos de directorios son compatibles.

Procedimiento

1. Create a storage pool

Utilice el comando **virsh pool-define-as** para definir y crear un pool de almacenamiento de tipo directorio. Por ejemplo, para crear un pool de almacenamiento llamado **guest_images_dir** que utilice el directorio **/guest_images**:

```
# virsh pool-define-as guest_images_dir dir --target "/guest_images"
Pool guest_images_dir defined
```

Si ya tiene una configuración XML del pool de almacenamiento que desea crear, también puede definir el pool basándose en el XML. Para más detalles, consulte [Sección 11.2.2.1.2, "Parámetros del pool de almacenamiento basado en directorios"](#).

2. Create the storage pool target path

Utilice el comando **virsh pool-build** para crear una ruta de destino de pool de almacenamiento para un pool de almacenamiento de sistema de archivos preformateado, inicializar el dispositivo de origen de almacenamiento y definir el formato de los datos.

```
# virsh pool-build guest_images_dir
Pool guest_images_dir built

# ls -la /guest_images
total 8
drwx-----. 2 root root 4096 May 31 19:38 .
dr-xr-xr-x. 25 root root 4096 May 31 19:38 ..
```

3. Verify that the pool was created

Utilice el comando **virsh pool-list** para verificar que el pool fue creado.

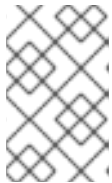
```
# virsh pool-list --all

Name           State   Autostart
-----
default        active  yes
guest_images_dir  inactive no
```

4. Start the storage pool

Utilice el comando **virsh pool-start** para montar el pool de almacenamiento.

```
# virsh pool-start guest_images_dir
Pool guest_images_dir started
```



NOTA

El comando **virsh pool-start** sólo es necesario para los pools de almacenamiento persistente. Los pools de almacenamiento transitorio se inician automáticamente cuando se crean.

5. [Optional] Turn on autostart

Por defecto, un pool de almacenamiento definido con el comando **virsh** no está configurado para iniciarse automáticamente cada vez que libvirt se inicia. Utilice el comando **virsh pool-autostart** para configurar el pool de almacenamiento para que se inicie automáticamente.

```
# virsh pool-autostart guest_images_dir
Pool guest_images_dir marked as autostarted
```

Verificación

1. Utilice el comando **virsh pool-list** para verificar el **Autostart** estado.

```
# virsh pool-list --all

Name           State   Autostart
-----
default        active  yes
guest_images_dir  inactive yes
```

2. Compruebe que el grupo de almacenamiento se ha creado correctamente, que los tamaños indicados son los esperados y que el estado es el siguiente **running**.


```
# virsh pool-info guest_images_dir
Name:      guest_images_dir
UUID:      c7466869-e82a-a66c-2187-dc9d6f0877d0
State:     running
Persistent: yes
Autostart: yes
Capacity:  458.39 GB
Allocation: 197.91 MB
Available: 458.20 GB
```

11.2.2.1.2. Parámetros del pool de almacenamiento basado en directorios

Esta sección proporciona información sobre los parámetros XML necesarios para un pool de almacenamiento basado en directorios y un ejemplo.

Puede definir un pool de almacenamiento basado en la configuración XML de un archivo específico. Por ejemplo:

```
# virsh pool-define ~/guest_images.xml
Pool defined from guest_images_dir
```

Parámetros

La siguiente tabla proporciona una lista de los parámetros necesarios para el archivo XML de un pool de almacenamiento basado en directorios.

Tabla 11.1. Parámetros del pool de almacenamiento basado en directorios

Descripción	XML
El tipo de pool de almacenamiento	<code><pool type='dir'></code>
El nombre del pool de almacenamiento	<code><name>name</name></code>
La ruta que especifica el objetivo. Esta será la ruta utilizada para el pool de almacenamiento.	<code><target> <path>target_path</path> </target></code>

Ejemplo

El siguiente es un ejemplo de un archivo XML para un pool de almacenamiento basado en el directorio `/guest_images`:

```
<pool type='dir'>
  <name>dirpool</name>
  <target>
    <path>/guest_images</path>
  </target>
</pool>
```

Recursos adicionales

Para obtener más información sobre la creación de grupos de almacenamiento basados en directorios, consulte [Sección 11.2.2.1.1, "Creación de grupos de almacenamiento basados en directorios mediante la CLI"](#).

11.2.2.2. Creación y asignación de almacenamiento en disco para máquinas virtuales mediante la CLI

A continuación se ofrece información sobre la creación de pools de almacenamiento basados en disco y volúmenes de almacenamiento y la asignación de volúmenes a máquinas virtuales.

11.2.2.2.1. Creación de grupos de almacenamiento basados en disco mediante la CLI

A continuación se ofrecen instrucciones para crear pools de almacenamiento basados en disco.

Recomendaciones

Tenga en cuenta lo siguiente antes de crear un pool de almacenamiento basado en disco:

- Dependiendo de la versión de **libvirt** que se utilice, dedicar un disco a un pool de almacenamiento puede reformatear y borrar todos los datos almacenados actualmente en el dispositivo de disco. Se recomienda encarecidamente realizar una copia de seguridad de los datos del dispositivo de almacenamiento antes de crear un pool de almacenamiento.
- Las máquinas virtuales no deben tener acceso de escritura a discos enteros o dispositivos de bloque (por ejemplo, **/dev/sdb**). Utilice particiones (por ejemplo, **/dev/sdb1**) o volúmenes LVM. Si pasas un dispositivo de bloque completo a una VM, la VM probablemente lo particionará o creará sus propios grupos LVM en él. Esto puede hacer que la máquina anfitriona detecte estas particiones o grupos LVM y cause errores.

Requisitos previos

- Asegúrese de que su hipervisor admite grupos de almacenamiento basados en disco:

```
# virsh pool-capabilities | grep "'disk' supported='yes'"
```

Si el comando muestra alguna salida, los pools basados en disco son compatibles.

Procedimiento

1. Create a storage pool

Utilice el comando **virsh pool-define-as** para definir y crear un pool de almacenamiento de tipo disco. Por ejemplo, para crear un pool de almacenamiento llamado **guest_images_disk** que utiliza la partición **/dev/sdb1**, y se monta en el directorio **/dev**:

```
# virsh pool-define-as guest_images_disk disk gpt --source-dev=/dev/sdb1 --target /dev
Pool guest_images_disk defined
```

Si ya tiene una configuración XML del pool de almacenamiento que desea crear, también puede definir el pool basándose en el XML. Para más detalles, consulte [Sección 11.2.2.2.2, "Parámetros del pool de almacenamiento en disco"](#).

2. Create the storage pool target path

Utilice el comando **virsh pool-build** para crear una ruta de destino de pool de almacenamiento para un pool de almacenamiento de sistema de archivos preformateado, inicializar el dispositivo de origen de almacenamiento y definir el formato de los datos.

```
# virsh pool-build guest_images_disk
Pool guest_images_disk built
```



NOTA

La construcción de la ruta de destino sólo es necesaria para los pools de almacenamiento basados en discos, sistemas de archivos y lógicos. Si **libvirt** detecta que el formato de datos del dispositivo de almacenamiento de origen difiere del tipo de pool de almacenamiento seleccionado, la construcción falla, a menos que se especifique la opción **overwrite** a menos que se especifique la opción

3. Verify that the pool was created

Utilice el comando **virsh pool-list** para verificar que el pool fue creado.

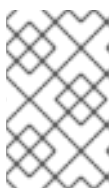
```
# virsh pool-list --all

Name           State   Autostart
-----
default        active  yes
guest_images_disk  inactive no
```

4. Start the storage pool

Utilice el comando **virsh pool-start** para montar el pool de almacenamiento.

```
# virsh pool-start guest_images_disk
Pool guest_images_disk started
```



NOTA

El comando **virsh pool-start** sólo es necesario para los pools de almacenamiento persistente. Los pools de almacenamiento transitorio se inician automáticamente cuando se crean.

5. [Optional] Turn on autostart

Por defecto, un pool de almacenamiento definido con el comando **virsh** no está configurado para iniciarse automáticamente cada vez que **libvirtd** se inicia. Utilice el comando **virsh pool-autostart** para configurar el pool de almacenamiento para que se inicie automáticamente.

```
# virsh pool-autostart guest_images_disk
Pool guest_images_disk marked as autostarted
```

Verificación

1. Utilice el comando **virsh pool-list** para verificar el **Autostart** estado.

```
# virsh pool-list --all
```

```

Name          State   Autostart
-----
default       active  yes
guest_images_disk  inactive  yes

```

2. Compruebe que el grupo de almacenamiento se ha creado correctamente, que los tamaños indicados son los esperados y que el estado es el siguiente **running**.

```

# virsh pool-info guest_images_disk
Name:          guest_images_disk
UUID:          c7466869-e82a-a66c-2187-dc9d6f0877d0
State:         running
Persistent:    yes
Autostart:     yes
Capacity:      458.39 GB
Allocation:    197.91 MB
Available:     458.20 GB

```

11.2.2.2.2. Parámetros del pool de almacenamiento en disco

Esta sección proporciona información sobre los parámetros XML necesarios para un pool de almacenamiento basado en disco y un ejemplo.

Puede definir un pool de almacenamiento basado en la configuración XML de un archivo específico. Por ejemplo:

```

# virsh pool-define ~/guest_images.xml
Pool defined from guest_images_disk

```

Parámetros

La siguiente tabla proporciona una lista de los parámetros necesarios para el archivo XML de un pool de almacenamiento basado en disco.

Tabla 11.2. Parámetros del pool de almacenamiento en disco

Descripción	XML
El tipo de pool de almacenamiento	<code><pool type='disk'></code>
El nombre del pool de almacenamiento	<code><name>name</name></code>
La ruta que especifica el dispositivo de almacenamiento. Por ejemplo, <code>/dev/sdb</code> .	<code><source> <path>source_path</path> </source></code>
La ruta que especifica el dispositivo de destino. Esta será la ruta utilizada para el pool de almacenamiento.	<code><target> <path>target_path</path> </target></code>

Ejemplo

A continuación se muestra un ejemplo de archivo XML para un pool de almacenamiento basado en disco:

```
<pool type='disk'>
  <name>phy_disk</name>
  <source>
    <device path='/dev/sdb'>
      <format type='gpt'>
    </source>
  <target>
    <path>/dev</path>
  </target>
</pool>
```

Recursos adicionales

Para más información sobre la creación de pools de almacenamiento basados en disco, consulte [Sección 11.2.2.1, "Creación de grupos de almacenamiento basados en disco mediante la CLI"](#) .

11.2.2.3. Creación y asignación de almacenamiento basado en sistemas de archivos para máquinas virtuales mediante la CLI

A continuación se ofrece información sobre la creación de pools de almacenamiento basados en sistemas de archivos y volúmenes de almacenamiento, y la asignación de volúmenes a máquinas virtuales.

11.2.2.3.1. Creación de grupos de almacenamiento basados en sistemas de archivos mediante la CLI

A continuación se ofrecen instrucciones para crear grupos de almacenamiento basados en sistemas de archivos.

Recomendaciones

No utilice este procedimiento para asignar un disco entero como pool de almacenamiento (por ejemplo, `/dev/sdb`). Las máquinas virtuales no deben tener acceso de escritura a discos enteros o dispositivos de bloque. Este método sólo debe utilizarse para asignar particiones (por ejemplo, `/dev/sdb1`) a grupos de almacenamiento.

Requisitos previos

- Asegúrese de que su hipervisor admite grupos de almacenamiento basados en sistemas de archivos:

```
# virsh pool-capabilities | grep "'fs' supported='yes'"
```

Si el comando muestra alguna salida, los pools basados en archivos son compatibles.

Procedimiento

1. Create a storage pool

Utilice el comando **virsh pool-define-as** para definir y crear un pool de almacenamiento de tipo sistema de archivos. Por ejemplo, para crear un pool de almacenamiento llamado **guest_images_fs** que utiliza la partición `/dev/sdc1`, y se monta en el directorio `/guest_images`:

```
# virsh pool-define-as guest_images_fs fs --source-dev /dev/sdc1 --target
/guest_images
Pool guest_images_fs defined
```

Si ya tiene una configuración XML del pool de almacenamiento que desea crear, también puede definir el pool basándose en el XML. Para más detalles, consulte [Sección 11.2.2.3.2, “Parámetros del pool de almacenamiento basado en el sistema de archivos”](#).

2. Define the storage pool target path

Utilice el comando **virsh pool-build** para crear una ruta de destino de pool de almacenamiento para un pool de almacenamiento de sistema de archivos preformateado, inicializar el dispositivo de origen de almacenamiento y definir el formato de los datos.

```
# virsh pool-build guest_images_fs
Pool guest_images_fs built

# ls -la /guest_images
total 8
drwx-----. 2 root root 4096 May 31 19:38 .
dr-xr-xr-x. 25 root root 4096 May 31 19:38 ..
```

3. Verify that the pool was created

Utilice el comando **virsh pool-list** para verificar que el pool fue creado.

```
# virsh pool-list --all

Name           State   Autostart
-----
default        active  yes
guest_images_fs  inactive no
```

4. Start the storage pool

Utilice el comando **virsh pool-start** para montar el pool de almacenamiento.

```
# virsh pool-start guest_images_fs
Pool guest_images_fs started
```



NOTA

El comando **virsh pool-start** sólo es necesario para los pools de almacenamiento persistente. Los pools de almacenamiento transitorio se inician automáticamente cuando se crean.

5. [Optional] Turn on autostart

Por defecto, un pool de almacenamiento definido con el comando **virsh** no está configurado para iniciarse automáticamente cada vez que libvirtd se inicia. Utilice el comando **virsh pool-autostart** para configurar el pool de almacenamiento para que se inicie automáticamente.

```
# virsh pool-autostart guest_images_fs
Pool guest_images_fs marked as autostarted
```

6. Verify the *Autostart* state

Utilice el comando **virsh pool-list** para verificar el **Autostart** estado.

```
# virsh pool-list --all

Name          State  Autostart
-----
default       active yes
guest_images_fs  inactive yes
```

7. Verify the storage pool

Compruebe que el grupo de almacenamiento se ha creado correctamente, que los tamaños indicados son los esperados y que el estado es el siguiente **running**. Compruebe que hay un directorio **lost found** en la ruta de destino en el sistema de archivos, lo que indica que el dispositivo está montado.

```
# virsh pool-info guest_images_fs
Name:      guest_images_fs
UUID:      c7466869-e82a-a66c-2187-dc9d6f0877d0
State:     running
Persistent: yes
Autostart: yes
Capacity:  458.39 GB
Allocation: 197.91 MB
Available: 458.20 GB

# mount | grep /guest_images
/dev/sdc1 on /guest_images type ext4 (rw)

# ls -la /guest_images
total 24
drwxr-xr-x. 3 root root 4096 May 31 19:47 .
dr-xr-xr-x. 25 root root 4096 May 31 19:38 ..
drwx-----. 2 root root 16384 May 31 14:18 lost+found
```

11.2.2.3.2. Parámetros del pool de almacenamiento basado en el sistema de archivos

A continuación se ofrece información sobre los parámetros necesarios para un pool de almacenamiento basado en un sistema de archivos y un ejemplo.

Puede definir un pool de almacenamiento basado en la configuración XML de un archivo específico. Por ejemplo:

```
# virsh pool-define ~/guest_images.xml
Pool defined from guest_images_fs
```

Parámetros

La siguiente tabla proporciona una lista de los parámetros necesarios para el archivo XML de un pool de almacenamiento basado en un sistema de archivos.

Tabla 11.3. Parámetros del pool de almacenamiento basado en el sistema de archivos

Descripción	XML
El tipo de pool de almacenamiento	<code><pool type='fs'></code>
El nombre del pool de almacenamiento	<code><name>name</name></code>
La ruta que especifica la partición. Por ejemplo, <code>/dev/sdc1</code>	<code><source> <device path=device_path /></code>
El tipo de sistema de archivos, por ejemplo <code>ext4</code> .	<code><format type=fs_type /> </source></code>
La ruta que especifica el objetivo. Esta será la ruta utilizada para el pool de almacenamiento.	<code><target> <path>path-to-pool</path> </target></code>

Ejemplo

El siguiente es un ejemplo de archivo XML para un pool de almacenamiento basado en la partición `/dev/sdc1`:

```
<pool type='fs'>
  <name>guest_images_fs</name>
  <source>
    <device path='/dev/sdc1'>
    <format type='auto'>
  </source>
  <target>
    <path>guest_images</path>
  </target>
</pool>
```

Recursos adicionales

Para obtener más información sobre la creación de grupos de almacenamiento basados en sistemas de archivos, consulte [Sección 11.2.2.3.1, "Creación de grupos de almacenamiento basados en sistemas de archivos mediante la CLI"](#).

11.2.2.4. Creación y asignación de almacenamiento GlusterFS para máquinas virtuales utilizando la CLI

A continuación se ofrece información sobre la creación de pools de almacenamiento y volúmenes de almacenamiento basados en GlusterFS, y la asignación de volúmenes a máquinas virtuales.

11.2.2.4.1. Creación de pools de almacenamiento basados en GlusterFS mediante la CLI

GlusterFS es un sistema de archivos de espacio de usuario que utiliza el sistema de archivos en espacio de usuario (FUSE). A continuación se ofrecen instrucciones para crear grupos de almacenamiento basados en GlusterFS.

Requisitos previos

- Antes de crear un pool de almacenamiento basado en GlusterFS en un host, prepare un Gluster.
 - a. Obtenga la dirección IP del servidor Gluster listando su estado con el siguiente comando:

```
# gluster volume status
Status of volume: gluster-vol1
Gluster process          Port Online Pid
-----
Brick 222.111.222.111:/gluster-vol1  49155 Y  18634

Task Status of Volume gluster-vol1
-----
There are no active volume tasks
```

- b. Si no está instalado, instale el paquete **glusterfs-fuse**.
- c. Si no está habilitado, habilite el booleano **virt_use_fusefs**. Compruebe que está habilitado.

```
# setsebool virt_use_fusefs on
# getsebool virt_use_fusefs
virt_use_fusefs --> on
```

- Asegúrese de que su hipervisor admite grupos de almacenamiento basados en GlusterFS:

```
# virsh pool-capabilities | grep "'gluster' supported='yes'"
```

Si el comando muestra alguna salida, los pools basados en GlusterFS son compatibles.

Procedimiento

1. Create a storage pool

Utilice el comando **virsh pool-define-as** para definir y crear un pool de almacenamiento basado en GlusterFS. Por ejemplo, para crear un pool de almacenamiento llamado **guest_images_glusterfs** que utiliza un servidor Gluster llamado **gluster-vol1** con IP **111.222.111.222**, y se monta en el directorio raíz del servidor Gluster:

```
# virsh pool-define-as --name guest_images_glusterfs --type gluster --source-host
111.222.111.222 --source-name gluster-vol1 --source-path /
Pool guest_images_glusterfs defined
```

Si ya tiene una configuración XML del pool de almacenamiento que desea crear, también puede definir el pool basándose en el XML. Para más detalles, consulte [Sección 11.2.2.4.2, "Parámetros del pool de almacenamiento basado en GlusterFS"](#).

2. Verify that the pool was created

Utilice el comando **virsh pool-list** para verificar que el pool fue creado.

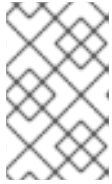
```
# virsh pool-list --all

Name          State  Autostart
-----
default       active yes
guest_images_glusterfs  inactive no
```

3. Start the storage pool

Utilice el comando **virsh pool-start** para montar el pool de almacenamiento.

```
# virsh pool-start guest_images_glusterfs
Pool guest_images_glusterfs started
```



NOTA

El comando **virsh pool-start** sólo es necesario para los pools de almacenamiento persistente. Los pools de almacenamiento transitorio se inician automáticamente cuando se crean.

4. [Optional] Turn on autostart

Por defecto, un pool de almacenamiento definido con el comando **virsh** no está configurado para iniciarse automáticamente cada vez que libvirtd se inicia. Utilice el comando **virsh pool-autostart** para configurar el pool de almacenamiento para que se inicie automáticamente.

```
# virsh pool-autostart guest_images_glusterfs
Pool guest_images_glusterfs marked as autostarted
```

Verificación

1. Utilice el comando **virsh pool-list** para verificar el **Autostart** estado.

```
# virsh pool-list --all

Name           State   Autostart
-----
default        active  yes
guest_images_glusterfs  inactive  yes
```

2. Compruebe que el grupo de almacenamiento se ha creado correctamente, que los tamaños indicados son los esperados y que el estado es el siguiente **running**.

```
# virsh pool-info guest_images_glusterfs
Name:          guest_images_glusterfs
UUID:          c7466869-e82a-a66c-2187-dc9d6f0877d0
State:         running
Persistent:    yes
Autostart:     yes
Capacity:      458.39 GB
Allocation:    197.91 MB
Available:     458.20 GB
```

11.2.2.4.2. Parámetros del pool de almacenamiento basado en GlusterFS

A continuación se ofrece información sobre los parámetros necesarios para un pool de almacenamiento basado en GlusterFS y un ejemplo.

Puede definir un pool de almacenamiento basado en la configuración XML de un archivo específico. Por ejemplo:

```
# virsh pool-define ~/guest_images.xml
Pool defined from guest_images_glusterfs
```

Parámetros

La siguiente tabla proporciona una lista de los parámetros necesarios para el archivo XML de un pool de almacenamiento basado en GlusterFS.

Tabla 11.4. Parámetros del pool de almacenamiento basado en GlusterFS

Descripción	XML
El tipo de pool de almacenamiento	<code><pool type='gluster'></code>
El nombre del pool de almacenamiento	<code><name>name</name></code>
El nombre de host o la dirección IP del servidor Gluster	<code><source></code> <code><name=gluster-name /></code>
La ruta en el servidor Gluster utilizada para el pool de almacenamiento.	<code><dir path=gluster-path /></code> <code></source></code>

Ejemplo

El siguiente es un ejemplo de un archivo XML para un pool de almacenamiento basado en el sistema de archivos Gluster en 111.222.111.222:

```
<pool type='gluster'>
  <name>Gluster_pool</name>
  <source>
    <host name='111.222.111.222' />
    <dir path='/' />
    <name>gluster-vol1 </name>
  </source>
</pool>
```

Para obtener más información sobre la creación de grupos de almacenamiento basados en sistemas de archivos, consulte [Sección 11.2.2.4.1, "Creación de pools de almacenamiento basados en GlusterFS mediante la CLI"](#).

11.2.2.5. Creación y asignación de almacenamiento basado en iSCSI para máquinas virtuales mediante la CLI

A continuación se ofrece información sobre la creación de pools de almacenamiento basados en iSCSI y volúmenes de almacenamiento, la protección de los pools de almacenamiento basados en iSCSI con **libvirt** secrets y la asignación de volúmenes a máquinas virtuales.

Recomendaciones

Internet Small Computer System Interface (iSCSI) es un protocolo de red para compartir dispositivos de almacenamiento. iSCSI conecta iniciadores (clientes de almacenamiento) con objetivos (servidores de almacenamiento) utilizando instrucciones SCSI sobre la capa IP.

El uso de dispositivos basados en iSCSI para almacenar máquinas virtuales permite opciones de

almacenamiento más flexibles, como el uso de iSCSI como dispositivo de almacenamiento en bloque. Los dispositivos iSCSI utilizan un objetivo Linux-IO (LIO). Este es un objetivo SCSI multiprotocolo para Linux. Además de iSCSI, LIO también soporta Fibre Channel y Fibre Channel over Ethernet (FCoE).

Si necesita impedir el acceso a un grupo de almacenamiento iSCSI, puede asegurarlo utilizando un [secreto de libvirt](#).

Requisitos previos

- Antes de crear un pool de almacenamiento basado en iSCSI, debe crear objetivos iSCSI. Puede crear objetivos iSCSI utilizando el paquete **targetcli**, que proporciona un conjunto de comandos para crear objetivos iSCSI respaldados por software. Para más información e instrucciones sobre la creación de objetivos iSCSI, consulte el documento [Gestión de dispositivos de almacenamiento](#).

11.2.2.5.1. Creación de pools de almacenamiento basados en iSCSI mediante la CLI

A continuación se ofrecen instrucciones para crear pools de almacenamiento basados en iSCSI.

Requisitos previos

- Asegúrese de que su hipervisor admite grupos de almacenamiento basados en iSCSI:

```
# virsh pool-capabilities | grep "'iscsi' supported='yes'"
```

Si el comando muestra alguna salida, los pools basados en iSCSI son compatibles.

Procedimiento

1. Create a storage pool

Utilice el comando **virsh pool-define-as** para definir y crear un pool de almacenamiento de tipo iSCSI. Por ejemplo, para crear un pool de almacenamiento llamado **guest_images_iscsi** que utilice el IQN de **iqn.2010-05.com.example.server1:iscsirhel7guest** en la ruta **server1.example.com**, y se monte en la ruta **/dev/disk/by-path**:

```
# virsh pool-define-as --name guest_images_iscsi --type iscsi --source-host
server1.example.com --source-dev iqn.2010-05.com.example.server1:iscsirhel7guest --
target /dev/disk/by-path
Pool guest_images_iscsi defined
```

Si ya tiene una configuración XML del pool de almacenamiento que desea crear, también puede definir el pool basándose en el XML. Para más detalles, consulte [Sección 11.2.2.5.2, "parámetros del pool de almacenamiento basado en iSCSI"](#).

2. Verify that the pool was created

Utilice el comando **virsh pool-list** para verificar que el pool fue creado.

```
# virsh pool-list --all

Name          State   Autostart
-----
default       active  yes
guest_images_iscsi  inactive no
```

3. Start the storage pool

Utilice el comando **virsh pool-start** para montar el pool de almacenamiento.

```
# virsh pool-start guest_images_iscsi
Pool guest_images_iscsi started
```



NOTA

El comando **virsh pool-start** sólo es necesario para los pools de almacenamiento persistente. Los pools de almacenamiento transitorio se inician automáticamente cuando se crean.

4. [Optional] Turn on autostart

Por defecto, un pool de almacenamiento definido con el comando **virsh** no está configurado para iniciarse automáticamente cada vez que libvirtd se inicia. Utilice el comando **virsh pool-autostart** para configurar el pool de almacenamiento para que se inicie automáticamente.

```
# virsh pool-autostart guest_images_iscsi
Pool guest_images_iscsi marked as autostarted
```

Verificación

1. Utilice el comando **virsh pool-list** para verificar el **Autostart** estado.

```
# virsh pool-list --all

Name           State   Autostart
-----
default        active  yes
guest_images_iscsi  inactive  yes
```

2. Compruebe que el grupo de almacenamiento se ha creado correctamente, que los tamaños indicados son los esperados y que el estado es el siguiente **running**. Compruebe que hay un directorio **lost found** en la ruta de destino en el sistema de archivos, lo que indica que el dispositivo está montado.

```
# virsh pool-info guest_images_iscsi
Name:          guest_images_iscsi
UUID:          c7466869-e82a-a66c-2187-dc9d6f0877d0
State:         running
Persistent:    yes
Autostart:     yes
Capacity:      458.39 GB
Allocation:    197.91 MB
Available:     458.20 GB
```

11.2.2.5.2. parámetros del pool de almacenamiento basado en iSCSI

A continuación se ofrece información sobre los parámetros necesarios para un pool de almacenamiento basado en iSCSI y un ejemplo.

Puede definir un pool de almacenamiento basado en la configuración XML de un archivo específico. Por ejemplo:

```
# virsh pool-define ~/guest_images.xml
Pool defined from guest_images_iscsi
```

Parámetros

La siguiente tabla proporciona una lista de los parámetros necesarios para el archivo XML de un pool de almacenamiento basado en iSCSI.

Tabla 11.5. parámetros del pool de almacenamiento basado en iSCSI

Descripción	XML
El tipo de pool de almacenamiento	<code><pool type='iscsi'></code>
El nombre del pool de almacenamiento	<code><name>name</name></code>
El nombre del anfitrión	<code><source></code> <code><host name=hostname /></code>
El iSCSI IQN	<code><device path= iSCSI_IQN /></code> <code></source></code>
La ruta que especifica el objetivo. Esta será la ruta utilizada para el pool de almacenamiento.	<code><target></code> <code><path>/dev/disk/by-path</path></code> <code></target></code>
Opcional] El IQN del iniciador iSCSI. Sólo es necesario cuando la ACL restringe el LUN a un iniciador concreto.	<code><initiator></code> <code><iqn name='initiator0' /></code> <code></initiator></code>



NOTA

El IQN del iniciador iSCSI puede determinarse mediante el comando **virsh find-storage-pool-sources-as** `iscsi`.

Ejemplo

El siguiente es un ejemplo de un archivo XML para un pool de almacenamiento basado en el dispositivo iSCSI especificado:

```
<pool type='iscsi'>
  <name>iSCSI_pool</name>
  <source>
    <host name='server1.example.com' />
    <device path='iqn.2010-05.com.example.server1:iscsirhel7guest' />
  </source>
  <target>
    <path>/dev/disk/by-path</path>
  </target>
</pool>
```

Recursos adicionales

Para obtener más información sobre la creación de pools de almacenamiento basados en iSCSI, consulte [Sección 11.2.2.5.1, "Creación de pools de almacenamiento basados en iSCSI mediante la CLI"](#) .

11.2.2.5.3. Asegurar los grupos de almacenamiento iSCSI con secretos de libvirt

Los parámetros de nombre de usuario y contraseña pueden configurarse con **virsh** para asegurar un pool de almacenamiento iSCSI. Puede configurar esto antes o después de definir el pool, pero el pool debe iniciarse para que la configuración de autenticación surta efecto.

A continuación se ofrecen instrucciones para asegurar los grupos de almacenamiento basados en iSCSI con **libvirt** secrets.



NOTA

Este procedimiento es necesario si a **user_ID** y **password** se definieron al crear el objetivo iSCSI.

Procedimiento

1. Cree un archivo secreto de libvirt con un nombre de usuario del protocolo de autenticación por desafío y apretón de manos (CHAP). Por ejemplo:

```
<secret ephemeral='no' private='yes'>
  <description>Passphrase for the iSCSI example.com server</description>
  <usage type='iscsi'>
    <target>iscsirhel7secret</target>
  </usage>
</secret>
```

2. Defina el secreto de libvirt con el comando **virsh secret-define**.
virsh secret-define secret.xml
3. Verifique el UUID con el comando **virsh secret-list**.

```
# virsh secret-list
UUID                               Usage
-----
2d7891af-20be-4e5e-af83-190e8a922360  iscsi iscsirhel7secret
```

4. Asigne un secreto al UUID en la salida del paso anterior utilizando el comando **virsh secret-set-value**. Esto asegura que el nombre de usuario y la contraseña de CHAP están en una lista de secretos controlada por libvirt. Por ejemplo:

```
# virsh secret-set-value --interactive 2d7891af-20be-4e5e-af83-190e8a922360
Enter new value for secret:
Secret value set
```

5. Añade una entrada de autenticación en el archivo XML del pool de almacenamiento utilizando el comando **virsh edit**, y añada un elemento **<auth>** elemento, especificando **authentication type, username y secret usage**.

Por ejemplo:

```
<pool type='iscsi'>
  <name>iscsirhel7pool</name>
```

```

<source>
  <host name='192.168.122.1'/>
  <device path='iqn.2010-05.com.example.server1:iscsirhel7guest'/>
  <auth type='chap' username='redhat'>
    <secret usage='iscsirhel7secret'/>
  </auth>
</source>
<target>
  <path>/dev/disk/by-path</path>
</target>
</pool>

```

NOTA

El subelemento **<auth>** subelemento existe en diferentes ubicaciones dentro de la máquina virtual **<pool>** y **<disk>** Elementos XML. Para un **<pool>**, **<auth>** se especifica dentro del elemento **<source>** ya que describe dónde encontrar las fuentes de pool, ya que la autenticación es una propiedad de algunas fuentes de pool (iSCSI y RBD). Para a **<disk>**, que es un subelemento de un dominio, la autenticación en el disco iSCSI o RBD es una propiedad del disco. Además, el **<auth>** subelemento de un disco difiere del de un pool de almacenamiento.

```

<auth username='redhat'>
  <secret type='iscsi' usage='iscsirhel7secret'/>
</auth>

```

6. Para activar los cambios, active el pool de almacenamiento. Si el pool ya se ha iniciado, detenga y reinicie el pool de almacenamiento:

```

# virsh pool-destroy iscsirhel7pool
# virsh pool-start iscsirhel7pool

```

11.2.2.6. Creación y asignación de almacenamiento basado en LVM para máquinas virtuales utilizando la CLI

A continuación se ofrece información sobre la creación de pools de almacenamiento y volúmenes de almacenamiento basados en LVM y la asignación de volúmenes a máquinas virtuales.

11.2.2.6.1. Creación de pools de almacenamiento basados en LVM mediante la CLI

A continuación se ofrecen instrucciones para crear pools de almacenamiento basados en LVM.

Recomendaciones

Tenga en cuenta lo siguiente antes de crear un pool de almacenamiento basado en LVM:

- Los pools de almacenamiento basados en LVM no proporcionan toda la flexibilidad de LVM.
- **libvirt** admite volúmenes lógicos delgados, pero no ofrece las características de los pools de almacenamiento delgados.
- Los pools de almacenamiento basados en LVM son grupos de volúmenes. Puedes crear grupos de volúmenes utilizando los comandos del Administrador de Volumen Lógico o los comandos de **virsh**. Para gestionar los grupos de volúmenes mediante la interfaz **virsh**, utiliza los comandos

virsh para crear grupos de volúmenes.

Para más información sobre los grupos de volumen, consulte la página web *Red Hat Enterprise Linux Logical Volume Manager Administration Guide*.

- Los pools de almacenamiento basados en LVM requieren una partición de disco completa. Si se activa una nueva partición o dispositivo con estos procedimientos, se formateará la partición y se borrarán todos los datos. Si se utiliza el Grupo de Volumen (VG) existente en el host, no se borrarán nada. Se recomienda hacer una copia de seguridad del dispositivo de almacenamiento antes de empezar.

Requisitos previos

- Asegúrese de que su hipervisor admite grupos de almacenamiento basados en LVM:

```
# virsh pool-capabilities | grep "'logical' supported='yes'"
```

Si el comando muestra alguna salida, los pools basados en LVM son compatibles.

Procedimiento

1. Create a storage pool

Utilice el comando **virsh pool-define-as** para definir y crear un pool de almacenamiento de tipo LVM. Por ejemplo, lo siguiente crea un pool de almacenamiento llamado **guest_images_logical** que utiliza un dispositivo LVM **libvirt_lvm** montado en **/dev/sdc**. El pool de almacenamiento creado se monta como **/dev/libvirt_lvm**.

```
# virsh pool-define-as guest_images_logical logical --source-dev=/dev/sdc --source-name libvirt_lvm --target /dev/libvirt_lvm
Pool guest_images_logical defined
```

Si ya tiene una configuración XML del pool de almacenamiento que desea crear, también puede definir el pool basándose en el XML. Para más detalles, consulte [Sección 11.2.2.6.2, "Parámetros del pool de almacenamiento basado en LVM"](#).

2. Verify that the pool was created

Utilice el comando **virsh pool-list** para verificar que el pool fue creado.

```
# virsh pool-list --all

Name                State  Autostart
-----
default             active yes
guest_images_logical inactive no
```

3. Start the storage pool

Utilice el comando **virsh pool-start** para montar el pool de almacenamiento.

```
# virsh pool-start guest_images_logical
Pool guest_images_logical started
```

**NOTA**

El comando **virsh pool-start** sólo es necesario para los pools de almacenamiento persistente. Los pools de almacenamiento transitorio se inician automáticamente cuando se crean.

4. [Optional] Turn on autostart

Por defecto, un pool de almacenamiento definido con el comando **virsh** no está configurado para iniciarse automáticamente cada vez que libvirtd se inicia. Utilice el comando **virsh pool-autostart** para configurar el pool de almacenamiento para que se inicie automáticamente.

```
# virsh pool-autostart guest_images_logical
Pool guest_images_logical marked as autostarted
```

Verificación

1. Utilice el comando **virsh pool-list** para verificar el **Autostart** estado.

```
# virsh pool-list --all

Name           State  Autostart
-----
default        active yes
guest_images_logical  inactive yes
```

2. Compruebe que el grupo de almacenamiento se ha creado correctamente, que los tamaños indicados son los esperados y que el estado es el siguiente **running**.

```
# virsh pool-info guest_images_logical
Name:      guest_images_logical
UUID:      c7466869-e82a-a66c-2187-dc9d6f0877d0
State:     running
Persistent: yes
Autostart: yes
Capacity:  458.39 GB
Allocation: 197.91 MB
Available: 458.20 GB
```

11.2.2.6.2. Parámetros del pool de almacenamiento basado en LVM

A continuación se ofrece información sobre los parámetros necesarios para un pool de almacenamiento basado en LVM y un ejemplo.

Puede definir un pool de almacenamiento basado en la configuración XML de un archivo específico. Por ejemplo:

```
# virsh pool-define ~/guest_images.xml
Pool defined from guest_images_logical
```

Parámetros

La siguiente tabla proporciona una lista de los parámetros necesarios para el archivo XML para un pool de almacenamiento basado en LVM.

Tabla 11.6. Parámetros del pool de almacenamiento basado en LVM

Descripción	XML
El tipo de pool de almacenamiento	<code><pool type='logical'></code>
El nombre del pool de almacenamiento	<code><name>name</name></code>
La ruta de acceso al dispositivo para el pool de almacenamiento	<code><source> <device path='device_path' /></code>
El nombre del grupo de volúmenes	<code><name>VG-name</name></code>
El formato de grupo virtual	<code><format type='lvm2' /> </source></code>
La trayectoria del objetivo	<code><target> <path=target_path /> </target></code>

**NOTA**

Si el grupo de volúmenes lógicos está formado por varias particiones de disco, puede haber varios dispositivos de origen en la lista. Por ejemplo:

```
<source>
  <device path='/dev/sda1'/>
  <device path='/dev/sdb3'/>
  <device path='/dev/sdc2'/>
  ...
</source>
```

Ejemplo

El siguiente es un ejemplo de un archivo XML para un pool de almacenamiento basado en el LVM especificado:

```
<pool type='logical'>
  <name>guest_images_lvm</name>
  <source>
    <device path='/dev/sdc'/>
    <name>libvirt_lvm</name>
    <format type='lvm2'/>
  </source>
  <target>
    <path>/dev/libvirt_lvm</path>
  </target>
</pool>
```

Recursos adicionales

Para obtener más información sobre la creación de pools de almacenamiento basados en iSCSI, consulte [Sección 11.2.2.6.1, "Creación de pools de almacenamiento basados en LVM mediante la CLI"](#) .

11.2.2.7. Creación y asignación de almacenamiento en red para máquinas virtuales mediante la CLI

A continuación se ofrece información sobre la creación de pools de almacenamiento basados en la red y volúmenes de almacenamiento y la asignación de volúmenes a máquinas virtuales.

Requisitos previos

- Para crear un pool de almacenamiento basado en el Sistema de Archivos de Red (NFS), ya debe estar configurado un Servidor NFS para ser utilizado por la máquina anfitriona. Para más información sobre NFS, consulte la página web *Red Hat Enterprise Linux Storage Administration Guide*.
- Asegúrese de que las utilidades necesarias para el sistema de archivos que se está utilizando están instaladas en el host. Por ejemplo, **cifs-utils** para Common Internet File Systems (CIFS) o **glusterfs.fuse** para GlusterFS.

11.2.2.7.1. Creación de pools de almacenamiento basados en NFS mediante la CLI

A continuación se ofrecen instrucciones para crear pools de almacenamiento basados en el sistema de archivos de red (NFS).

Requisitos previos

- Asegúrese de que su hipervisor admite grupos de almacenamiento basados en NFS:

```
# virsh pool-capabilities | grep "<value>nfs</value>"
```

Si el comando muestra alguna salida, los pools basados en NFS son compatibles.

Procedimiento

1. Create a storage pool

Utilice el comando **virsh pool-define-as** para definir y crear un pool de almacenamiento de tipo NFS. Por ejemplo, para crear un pool de almacenamiento llamado **guest_images_netfs** que utilice un servidor NFS con IP **111.222.111.222** montado en el directorio del servidor **/home/net_mount** utilizando el directorio de destino **/var/lib/libvirt/images/nfspool**:

```
# virsh pool-define-as --name guest_images_netfs --type netfs --source-host='111.222.111.222' source-path='/home/net_mount' --source-format='nfs' --target='/var/lib/libvirt/images/nfspool'
```

Si ya tiene una configuración XML del pool de almacenamiento que desea crear, también puede definir el pool basándose en el XML. Para más detalles, consulte [Sección 11.2.2.7.2, "Parámetros del pool de almacenamiento basado en NFS"](#) .

2. Verify that the pool was created

Utilice el comando **virsh pool-list** para verificar que el pool fue creado.

```
# virsh pool-list --all
```

```

Name           State   Autostart
-----
default        active  yes
guest_images_netfs  inactive no

```

3. Start the storage pool

Utilice el comando **virsh pool-start** para montar el pool de almacenamiento.

```

# virsh pool-start guest_images_netfs
Pool guest_images_netfs started

```



NOTA

El comando **virsh pool-start** sólo es necesario para los pools de almacenamiento persistente. Los pools de almacenamiento transitorio se inician automáticamente cuando se crean.

4. [Optional] Turn on autostart

Por defecto, un pool de almacenamiento definido con el comando **virsh** no está configurado para iniciarse automáticamente cada vez que libvirt se inicia. Utilice el comando **virsh pool-autostart** para configurar el pool de almacenamiento para que se inicie automáticamente.

```

# virsh pool-autostart guest_images_netfs
Pool guest_images_netfs marked as autostarted

```

Verificación

1. Utilice el comando **virsh pool-list** para verificar el **Autostart** estado.

```

# virsh pool-list --all

Name           State   Autostart
-----
default        active  yes
guest_images_netfs  inactive yes

```

2. Compruebe que el grupo de almacenamiento se ha creado correctamente, que los tamaños indicados son los esperados y que el estado es el siguiente **running**. Compruebe que hay un directorio **lost found** en la ruta de destino en el sistema de archivos, lo que indica que el dispositivo está montado.

```

# virsh pool-info guest_images_netfs
Name:         guest_images_netfs
UUID:         c7466869-e82a-a66c-2187-dc9d6f0877d0
State:        running
Persistent:   yes
Autostart:    yes
Capacity:     458.39 GB
Allocation:   197.91 MB
Available:    458.20 GB

```

11.2.2.7.2. Parámetros del pool de almacenamiento basado en NFS

A continuación se ofrece información sobre los parámetros necesarios para un pool de almacenamiento basado en NFS y un ejemplo.

Puede definir un pool de almacenamiento basado en la configuración XML de un archivo específico. Por ejemplo:

```
# virsh pool-define ~/guest_images.xml
Pool defined from guest_images_netfs
```

Parámetros

La siguiente tabla proporciona una lista de los parámetros necesarios para el archivo XML de un pool de almacenamiento basado en NFS.

Tabla 11.7. Parámetros del pool de almacenamiento basado en NFS

Descripción	XML
El tipo de pool de almacenamiento	<code><pool type='netfs'></code>
El nombre del pool de almacenamiento	<code><name>name</name></code>
El nombre del servidor de red donde se encuentra el punto de montaje. Puede ser un nombre de host o una dirección IP.	<code><source></code> <code><host name=hostname /></code>
El formato del pool de almacenamiento	Uno de los siguientes: <code><format type='nfs' /></code> <code><format type='glusterfs' /></code> <code><format type='cifs' /></code>
El directorio utilizado en el servidor de red	<code><dir path=source_path /></code> <code></source></code>
La ruta que especifica el objetivo. Esta será la ruta utilizada para el pool de almacenamiento.	<code><target></code> <code><path>target_path</path></code> <code></target></code>

Ejemplo

El siguiente es un ejemplo de un archivo XML para un pool de almacenamiento basado en el directorio `/home/net_mount` del servidor NFS `file_server`:

```
<pool type='netfs'>
  <name>nfspool</name>
  <source>
    <host name='file_server' />
    <format type='nfs' />
    <dir path='/home/net_mount' />
  </source>
```

```

<target>
  <path>/var/lib/libvirt/images/nfspool</path>
</target>
</pool>

```

Recursos adicionales

Para más información sobre la creación de pools de almacenamiento basados en NFS, consulte [Sección 11.2.2.7.1, “Creación de pools de almacenamiento basados en NFS mediante la CLI”](#) .

11.2.2.8. Creación y asignación de almacenamiento basado en SCSI con dispositivos vHBA para máquinas virtuales utilizando la CLI

A continuación se proporciona información sobre la creación de pools de almacenamiento basados en SCSI y volúmenes de almacenamiento utilizando dispositivos vHBA, así como la asignación de volúmenes a máquinas virtuales (VM).

Recomendaciones

N_Port ID Virtualization (NPIV) es una tecnología de software que permite compartir un único adaptador de bus de host (HBA) Fibre Channel físico. Esto permite que varias máquinas virtuales vean el mismo almacenamiento desde varios hosts físicos y, por tanto, facilita las rutas de migración del almacenamiento. Como resultado, no es necesario que la migración cree o copie el almacenamiento, siempre que se especifique la ruta de almacenamiento correcta.

En la virtualización, el *virtual host bus adapter*, o *vHBA*, controla los Números de Unidad Lógica (LUNs) para las VMs. Para que un host comparta una ruta de dispositivo Fibre Channel entre múltiples VMs, debe crear un vHBA para cada VM. Un solo vHBA no puede ser utilizado por múltiples VMs.

Cada vHBA para NPIV se identifica por su HBA padre y su propio Nombre de Nodo Mundial (WWNN) y Nombre de Puerto Mundial (WWPN). La ruta de acceso al almacenamiento está determinada por los valores WWNN y WWPN. El HBA principal puede definirse como ***scsi_host#*** o como un par WWNN/WWPN.



NOTA

Si se define un HBA padre como ***scsi_host#*** y se añade hardware a la máquina anfitriona, la asignación ***scsi_host#*** asignación puede cambiar. Por lo tanto, se recomienda definir un HBA padre utilizando un par WWNN/WWPN.

Se recomienda definir un pool de almacenamiento **libvirt** basado en el vHBA, porque así se conserva la configuración del vHBA.

El uso de un pool de almacenamiento libvirt tiene dos ventajas principales:

- El código de libvirt puede encontrar fácilmente la ruta del LUN a través de la salida del comando `virsh`.
- Para migrar una VM sólo es necesario definir e iniciar un pool de almacenamiento con el mismo nombre de vHBA en la máquina de destino. Para ello, el LUN de vHBA, el pool de almacenamiento libvirt y el nombre del volumen deben estar especificados en la configuración XML de la VM.



NOTA

Antes de crear un vHBA, se recomienda configurar la zonificación del lado de la matriz de almacenamiento (SAN) en el LUN del host para proporcionar aislamiento entre las máquinas virtuales y evitar la posibilidad de corrupción de datos.

Para crear una configuración de vHBA persistente, primero hay que crear un archivo XML de pool de almacenamiento libvirt `'scsi'`. Para obtener información sobre el archivo XML, consulte [Creación de vHBA](#). Cuando se crea un único vHBA que utiliza un pool de almacenamiento en el mismo HBA físico, se recomienda utilizar una ubicación estable para el valor `<path>`, como una de las `/dev/disk/by-{path/id/uuid/label}` ubicaciones de su sistema.

Cuando se crean múltiples vHBAs que utilizan pools de almacenamiento en el mismo HBA físico, el valor del campo `<path>` debe ser sólo `/dev/`, de lo contrario los volúmenes del pool de almacenamiento son visibles sólo para uno de los vHBAs, y los dispositivos del host no pueden ser expuestos a múltiples VMs con la configuración NPIV.

Para más información sobre `<path>` y los elementos de `<target>`, consulte [la documentación de libvirt](#).

11.2.2.8.1. Creación de vHBAs

A continuación se ofrecen instrucciones para crear un adaptador de bus de host virtual (vHBA).

Procedimiento

1. Localice los HBAs en su sistema anfitrión, utilizando el comando **virsh nodedev-list --cap vports**.

El siguiente ejemplo muestra un host que tiene dos HBAs que soportan vHBA:

```
# virsh nodedev-list --cap vports
scsi_host3
scsi_host4
```

2. Vea los detalles del HBA, utilizando el **virsh nodedev-dumpxml HBA_device** comando.

```
# virsh nodedev-dumpxml scsi_host3
```

La salida del comando muestra los campos `<name>`, `<wwnn>`, y `<wwpn>`, que se utilizan para crear un vHBA. `<max_vports>` muestra el número máximo de vHBAs soportados. Por ejemplo:

```
<device>
  <name>scsi_host3</name>
  <path>/sys/devices/pci0000:00/0000:00:04.0/0000:10:00.0/host3</path>
  <parent>pci_0000_10_00_0</parent>
  <capability type='scsi_host'>
    <host>3</host>
    <unique_id>0</unique_id>
    <capability type='fc_host'>
      <wwnn>20000000c9848140</wwnn>
      <wwpn>10000000c9848140</wwpn>
      <fabric_wwn>2002000573de9a81</fabric_wwn>
    </capability>
  <capability type='vport_ops'>
    <max_vports>127</max_vports>
```



```

    <vports>0</vports>
  </capability>
</capability>
</device>

```

En este ejemplo, el valor **<max_vports>** muestra que hay un total de 127 puertos virtuales disponibles para su uso en la configuración HBA. El valor **<vports>** muestra el número de puertos virtuales que se están utilizando actualmente. Estos valores se actualizan después de crear un vHBA.

3. Cree un archivo XML similar a uno de los siguientes para el host vHBA. En estos ejemplos, el archivo se denomina **vhba_host3.xml**.

Este ejemplo utiliza **scsi_host3** para describir el vHBA padre.

```

<device>
  <parent>scsi_host3</parent>
  <capability type='scsi_host'>
    <capability type='fc_host'>
      </capability>
    </capability>
  </capability>
</device>

```

Este ejemplo utiliza un par WWNN/WWPN para describir el vHBA padre.

```

<device>
  <name>vhba</name>
  <parent wwnn='20000000c9848140' wwpn='10000000c9848140'>
    <capability type='scsi_host'>
      <capability type='fc_host'>
        </capability>
      </capability>
    </capability>
  </capability>
</device>

```



NOTA

Los valores WWNN y WWPN deben coincidir con los de los detalles del HBA vistos en el paso anterior.

El campo **<parent>** especifica el dispositivo HBA a asociar con este dispositivo vHBA. Los detalles de la etiqueta **<device>** se utilizan en el siguiente paso para crear un nuevo dispositivo vHBA para el host. Para más información sobre el formato XML **nodedev**, consulte [las páginas de libvirt upstream](#).



NOTA

El comando **virsh** no permite definir los atributos **parent_wwnn**, **parent_wwpn**, o **parent_fabric_wwn**.

4. Cree un vHBA basado en el archivo XML creado en el paso anterior utilizando el comando **virsh nodedev-create**.

```

# virsh nodedev-create vhba_host3
Node device scsi_host5 created from vhba_host3.xml

```

Verificación

- Verifique los detalles del nuevo vHBA (scsi_host5) utilizando el comando **virsh nodedev-dumpxml**:

```
# virsh nodedev-dumpxml scsi_host5
<device>
  <name>scsi_host5</name>
  <path>/sys/devices/pci0000:00/0000:00:04.0/0000:10:00.0/host3/vport-3:0-0/host5</path>
  <parent>scsi_host3</parent>
  <capability type='scsi_host'>
    <host>5</host>
    <unique_id>2</unique_id>
    <capability type='fc_host'>
      <wwnn>5001a4a93526d0a1</wwnn>
      <wwpn>5001a4ace3ee047d</wwpn>
      <fabric_wwn>2002000573de9a81</fabric_wwn>
    </capability>
  </capability>
</device>
```

11.2.2.8.2. Creación de pools de almacenamiento basados en SCSI con dispositivos vHBA mediante la CLI

A continuación se ofrecen instrucciones para crear grupos de almacenamiento basados en SCSI utilizando dispositivos de adaptador de bus de host virtual (vHBA).

Requisitos previos

- Asegúrese de que su hipervisor admite grupos de almacenamiento basados en SCSI:

```
# virsh pool-capabilities | grep "'scsi' supported='yes'"
```

Si el comando muestra alguna salida, los pools basados en SCSI son compatibles.

- Antes de crear un pool de almacenamiento basado en SCSI con dispositivos vHBA, cree un vHBA. Para obtener más información, consulte [Creación de vHBA](#).

Procedimiento

1. Create a storage pool

Utilice el comando **virsh pool-define-as** para definir y crear un pool de almacenamiento SCSI utilizando un vHBA. Por ejemplo, lo siguiente crea un pool de almacenamiento llamado **guest_images_vhba** que utiliza un vHBA identificado por el adaptador padre **scsi_host3**, el número de puerto mundial **5001a4ace3ee047d**, y el número de nodo mundial **5001a4a93526d0a1**. El pool de almacenamiento se monta en el directorio **/dev/disk/**:

```
# virsh pool-define-as guest_images_vhba scsi --adapter-parent scsi_host3 --adapter-wwnn 5001a4a93526d0a1 --adapter-wwpn 5001a4ace3ee047d --target /dev/disk/
Pool guest_images_vhba defined
```

Si ya tiene una configuración XML del pool de almacenamiento que desea crear, también puede definir el pool basándose en el XML. Para más detalles, consulte [Sección 11.2.2.8.3, "Parámetros para pools de almacenamiento basados en SCSI con dispositivos vHBA"](#).

2. Verify that the pool was created

Utilice el comando **virsh pool-list** para verificar que el pool fue creado.

```
# virsh pool-list --all

Name           State   Autostart
-----
default        active  yes
guest_images_vhba  inactive no
```

3. Start the storage pool

Utilice el comando **virsh pool-start** para montar el pool de almacenamiento.

```
# virsh pool-start guest_images_vhba
Pool guest_images_vhba started
```



NOTA

El comando **virsh pool-start** sólo es necesario para los pools de almacenamiento persistente. Los pools de almacenamiento transitorio se inician automáticamente cuando se crean.

4. [Optional] Turn on autostart

Por defecto, un pool de almacenamiento definido con el comando **virsh** no está configurado para iniciarse automáticamente cada vez que libvirt se inicia. Utilice el comando **virsh pool-autostart** para configurar el pool de almacenamiento para que se inicie automáticamente.

```
# virsh pool-autostart guest_images_vhba
Pool guest_images_vhba marked as autostarted
```

Verificación

1. Utilice el comando **virsh pool-list** para verificar el **Autostart** estado.

```
# virsh pool-list --all

Name           State   Autostart
-----
default        active  yes
guest_images_vhba  inactive yes
```

2. Compruebe que el grupo de almacenamiento se ha creado correctamente, que los tamaños indicados son los esperados y que el estado es el siguiente **running**.

```
# virsh pool-info guest_images_vhba
Name:         guest_images_vhba
UUID:         c7466869-e82a-a66c-2187-dc9d6f0877d0
State:        running
Persistent:   yes
Autostart:    yes
```

```
Capacity: 458.39 GB
Allocation: 197.91 MB
Available: 458.20 GB
```

11.2.2.8.3. Parámetros para pools de almacenamiento basados en SCSI con dispositivos vHBA

A continuación se proporciona información sobre los parámetros necesarios para un grupo de almacenamiento basado en SCSI que utiliza un dispositivo de bus adaptador de host virtual (vHBA).

Puede definir un pool de almacenamiento basado en la configuración XML de un archivo específico. Por ejemplo:

```
# virsh pool-define ~/guest_images.xml
Pool defined from guest_images_vhba
```

Parámetros

La siguiente tabla proporciona una lista de los parámetros necesarios para el archivo XML para un pool de almacenamiento basado en SCSI con vHBA.

Tabla 11.8. Parámetros para pools de almacenamiento basados en SCSI con dispositivos vHBA

Descripción	XML
El tipo de pool de almacenamiento	<code><pool type='scsi'></code>
El nombre del pool de almacenamiento	<code><name>name</name></code>
El identificador del vHBA. El atributo parent es opcional.	<code><source> <adapter type='fc_host' [parent=parent_scsi_device] wwnn='WWNN' wwpn='WWPN' /> </source></code>
La ruta de destino. Esta será la ruta utilizada para el pool de almacenamiento.	<code><target> <path=target_path /> </target></code>



IMPORTANTE

Cuando el campo `<path>` es `/dev/libvirt` genera una ruta de dispositivo corta y única para la ruta de dispositivo del volumen. Por ejemplo, `/dev/sdc`. En caso contrario, se utiliza la ruta del host físico. Por ejemplo, `/dev/disk/by-path/pci-0000:10:00.0-fc-0x5006016044602198-lun-0`. La ruta de dispositivo corta y única permite que el mismo volumen aparezca en varias máquinas virtuales (VM) por parte de varios grupos de almacenamiento. Si la ruta del host físico es utilizada por múltiples VMs, pueden producirse advertencias de tipo de dispositivo duplicado.



NOTA

El atributo **parent** se puede utilizar en el campo **<adapter>** para identificar el HBA físico padre desde el que se pueden utilizar los LUNs NPIV por rutas variables. Este campo, **scsi_hostN**, se combina con los atributos **vports** y **max_vports** para completar la identificación del padre. Los atributos **parent**, **parent_wwnn**, **parent_wwpn**, o **parent_fabric_wwn** proporcionan diversos grados de garantía de que después de que el host se reinicie se utilice el mismo HBA.

- Si no se especifica **parent**, **libvirt** utiliza el primer adaptador de **scsi_hostN** que soporta NPIV.
- Si sólo se especifica el **parent**, pueden surgir problemas si se añaden adaptadores de host SCSI adicionales a la configuración.
- Si se especifica **parent_wwnn** o **parent_wwpn**, después de que el host se reinicie se utilizará el mismo HBA.
- Si se utiliza **parent_fabric_wwn**, después de que el host se reinicie se selecciona un HBA en el mismo tejido, independientemente de la **scsi_hostN** utilizada.

Ejemplos

Los siguientes son ejemplos de archivos XML para pools de almacenamiento basados en SCSI con vHBA.

- Un pool de almacenamiento que es el único pool de almacenamiento en el HBA:

```
<pool type='scsi'>
  <name>vhbapool_host3</name>
  <source>
    <adapter type='fc_host' wwnn='5001a4a93526d0a1' wwpn='5001a4ace3ee047d' />
  </source>
  <target>
    <path>/dev/disk/by-path</path>
  </target>
</pool>
```

- Un pool de almacenamiento que es uno de los varios pools de almacenamiento que utilizan un único vHBA y utiliza el atributo **parent** para identificar el dispositivo host SCSI:

```
<pool type='scsi'>
  <name>vhbapool_host3</name>
  <source>
    <adapter type='fc_host' parent='scsi_host3' wwnn='5001a4a93526d0a1'
    wwpn='5001a4ace3ee047d' />
  </source>
  <target>
    <path>/dev/disk/by-path</path>
  </target>
</pool>
```

Recursos adicionales

Para más información sobre la creación de pools de almacenamiento basados en SCSI con vHBA, consulte [Sección 11.2.2.8.2, “Creación de pools de almacenamiento basados en SCSI con dispositivos vHBA mediante la CLI”](#).

11.2.2.9. Creación y asignación de volúmenes de almacenamiento mediante la CLI

Para obtener una imagen de disco y adjuntarla a una máquina virtual (VM) como disco virtual, cree un volumen de almacenamiento y asigne su configuración XML a la VM.

Requisitos previos

- Hay un pool de almacenamiento con espacio no asignado en el host. Para verificarlo, liste los pools de almacenamiento en el host:

```
# virsh pool-list --details
```

Name	State	Autostart	Persistent	Capacity	Allocation	Available
default	running	yes	yes	48.97 GiB	36.34 GiB	12.63 GiB
Downloads	running	yes	yes	175.92 GiB	121.20 GiB	54.72 GiB
VM-disks	running	yes	yes	175.92 GiB	121.20 GiB	54.72 GiB

Procedimiento

1. Cree un volumen de almacenamiento utilizando el comando **virsh vol-create-as**. Por ejemplo, para crear un volumen qcow2 de 20 GB basado en el pool de almacenamiento **guest-images-fs**:

```
# virsh vol-create-as --pool guest-images-fs --name vm-disk1 --capacity 20 --format qcow2
```

Important: Algunos tipos de pool de almacenamiento no soportan el comando **virsh vol-create-as** y en su lugar requieren procesos específicos para crear volúmenes de almacenamiento:

- **GlusterFS-based** - Utilice el comando **qemu-img** para crear volúmenes de almacenamiento.
 - **iSCSI-based** - Prepare los LUNs iSCSI por adelantado en el servidor iSCSI.
 - **Multipath-based** - Utilice el comando **multipathd** para preparar o gestionar el multipath.
 - **vHBA-based** - Prepare la tarjeta de canal de fibra por adelantado.
2. Cree un archivo XML y añada las siguientes líneas en él. Este archivo se utilizará para añadir el volumen de almacenamiento como un disco a una VM.

```
<disk type='volume' device='disk'>
  <driver name='qemu' type='qcow2'/>
  <source pool='guest-images-fs' volume='vm-disk1'/>
  <target dev='hdk' bus='ide'/>
</disk>
```

Este ejemplo especifica un disco virtual que utiliza el volumen **vm-disk1**, creado en el paso anterior, y configura el volumen como disco **hdk** en un bus **ide**. Modifique los parámetros respectivos según sea apropiado para su entorno.

Important: Con determinados tipos de pool de almacenamiento, debe utilizar diferentes formatos XML para describir un disco de volumen de almacenamiento.

- Para piscinas *GlusterFS-based*:

```
<disk type='network' device='disk'>
  <driver name='qemu' type='raw'/>
  <source protocol='gluster' name='Volume1/Image'>
    <host name='example.org' port='6000'/>
  </source>
  <target dev='vda' bus='virtio'/>
  <address type='pci' domain='0x0000' bus='0x00' slot='0x03' function='0x00'/>
</disk>
```

- Para piscinas *multipath-based*:

```
<disk type='block' device='disk'>
  <driver name='qemu' type='raw'/>
  <source dev='/dev/mapper/mpatha' />
  <target dev='sda' bus='scsi'/>
</disk>
```

- Para piscinas *RBD-based storage*:

```
<disk type='network' device='disk'>
  <driver name='qemu' type='raw'/>
  <source protocol='rbd' name='pool/image'>
    <host name='mon1.example.org' port='6321'/>
  </source>
  <target dev='vdc' bus='virtio'/>
</disk>
```

3. Utilice el archivo XML para asignar el volumen de almacenamiento como disco a una VM. Por ejemplo, para asignar un disco definido en **~/vm-disk1.xml** a la VM **testguest1**:

```
# attach-device --config testguest1 ~/vm-disk1.xml
```

Verificación

- En el sistema operativo invitado de la VM, confirme que la imagen de disco ha quedado disponible como un disco sin formatear y sin asignar.

11.2.3. Eliminación del almacenamiento de las máquinas virtuales mediante la CLI

A continuación se ofrece información sobre la eliminación de grupos de almacenamiento y volúmenes de almacenamiento mediante la CLI.

11.2.3.1. Eliminación de grupos de almacenamiento mediante la CLI

Para eliminar un pool de almacenamiento de su sistema anfitrión, debe detener el pool y eliminar su definición XML.

Procedimiento

1. Liste los pools de almacenamiento definidos utilizando el comando **virsh pool-list**.

```
# virsh pool-list --all
Name           State   Autostart
-----
default        active yes
Downloads      active yes
RHEL8-Storage-Pool active yes
```

2. Detenga el pool de almacenamiento que desea eliminar mediante el comando **virsh pool-destroy**.

```
# virsh pool-destroy Downloads
Pool Downloads destroyed
```

3. **Optional:** Para algunos tipos de pools de almacenamiento, puede eliminar el directorio donde reside el pool de almacenamiento utilizando el comando **virsh pool-delete**. Tenga en cuenta que para hacerlo, el directorio debe estar vacío.

```
# virsh pool-delete Downloads
Pool Downloads deleted
```

4. Elimine la definición del pool de almacenamiento mediante el comando **virsh pool-undefine**.

```
# virsh pool-undefine Downloads
Pool Downloads has been undefined
```

Verificación

- Confirme que el pool de almacenamiento fue eliminado.

```
# virsh pool-list --all
Name           State   Autostart
-----
default        active yes
RHEL8-Storage-Pool active yes
```

11.2.3.2. Eliminación de volúmenes de almacenamiento mediante la CLI

Para eliminar un volumen de almacenamiento de su sistema anfitrión, debe detener el pool y eliminar su definición XML.

Requisitos previos

- Cualquier máquina virtual que utilice el volumen de almacenamiento que desea eliminar se apaga.

Procedimiento

1. Liste los volúmenes de almacenamiento definidos en un pool de almacenamiento utilizando el comando **virsh vol-list**. El comando debe especificar el nombre o la ruta de un pool de almacenamiento.

```
# virsh vol-list --pool RHEL8-Storage-Pool
Name          Path
-----
.bash_history  /home/VirtualMachines/.bash_history
.bash_logout   /home/VirtualMachines/.bash_logout
.bash_profile  /home/VirtualMachines/.bash_profile
.bashrc        /home/VirtualMachines/.bashrc
.git-prompt.sh /home/VirtualMachines/.git-prompt.sh
.gitconfig     /home/VirtualMachines/.gitconfig
RHEL8_Volume.qcow2 /home/VirtualMachines/RHEL8_Volume.qcow2
```

2. Elimine los volúmenes de almacenamiento utilizando el comando **virsh vol-delete**. El comando debe especificar el nombre o la ruta del volumen de almacenamiento y el pool de almacenamiento del que se abstrae el volumen de almacenamiento.

```
# virsh vol-delete --pool RHEL-Storage-Pool RHEL8_Volume.qcow2
Pool RHEL8_Volume.qcow2 deleted
```

Verificación

- Vuelva a enumerar los volúmenes de almacenamiento definidos y compruebe que la salida ya no muestra el volumen eliminado.

```
# virsh vol-list --pool RHEL8-Storage-Pool
Name          Path
-----
.bash_history  /home/VirtualMachines/.bash_history
.bash_logout   /home/VirtualMachines/.bash_logout
.bash_profile  /home/VirtualMachines/.bash_profile
.bashrc        /home/VirtualMachines/.bashrc
.git-prompt.sh /home/VirtualMachines/.git-prompt.sh
.gitconfig     /home/VirtualMachines/.gitconfig
```

11.3. GESTIÓN DEL ALMACENAMIENTO DE LAS MÁQUINAS VIRTUALES MEDIANTE LA CONSOLA WEB

A través de la consola web de RHEL 8, puede gestionar varios aspectos del almacenamiento de una máquina virtual (VM). Puede utilizar la consola web para:

- [Ver la información del pool de almacenamiento](#) .
- [Crear grupos de almacenamiento](#) .
- [Eliminar los grupos de almacenamiento](#) .
- [Desactivar los grupos de almacenamiento](#) .
- [Crear volúmenes de almacenamiento](#) .
- [Eliminar los volúmenes de almacenamiento](#) .

- [Gestionar los discos de las máquinas virtuales](#) .

11.3.1. Visualización de la información del pool de almacenamiento mediante la consola web

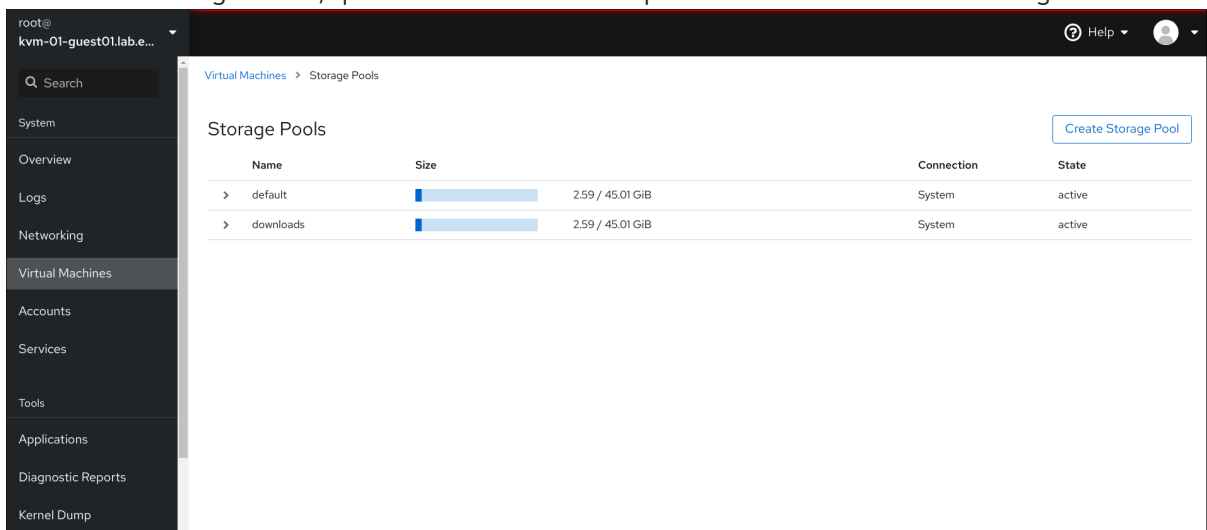
El siguiente procedimiento describe cómo ver información detallada sobre los pools de almacenamiento de las máquinas virtuales (VM) a los que puede acceder la sesión de la consola web.

Requisitos previos

- Para utilizar la consola web para gestionar las máquinas virtuales, [instale el complemento VM de la consola web](#).

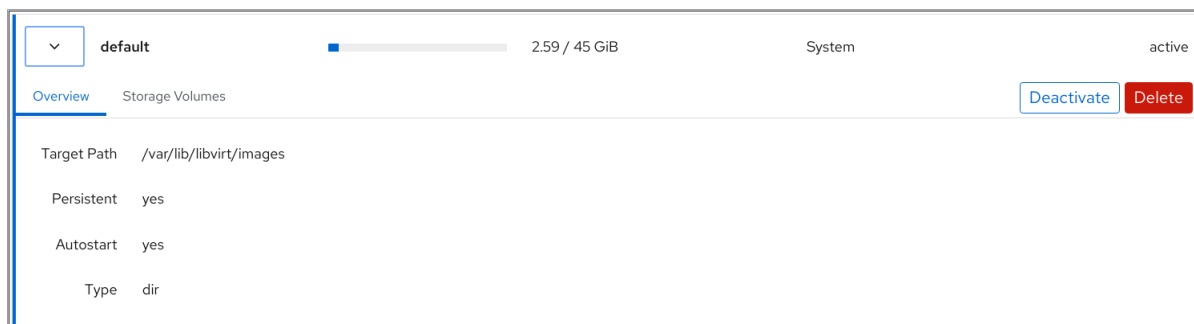
Procedimiento

1. Haga clic en **Storage Pools** en la parte superior de la interfaz de **Máquinas Virtuales**. Aparece la ventana Storage Pools, que muestra una lista de pools de almacenamiento configurados.

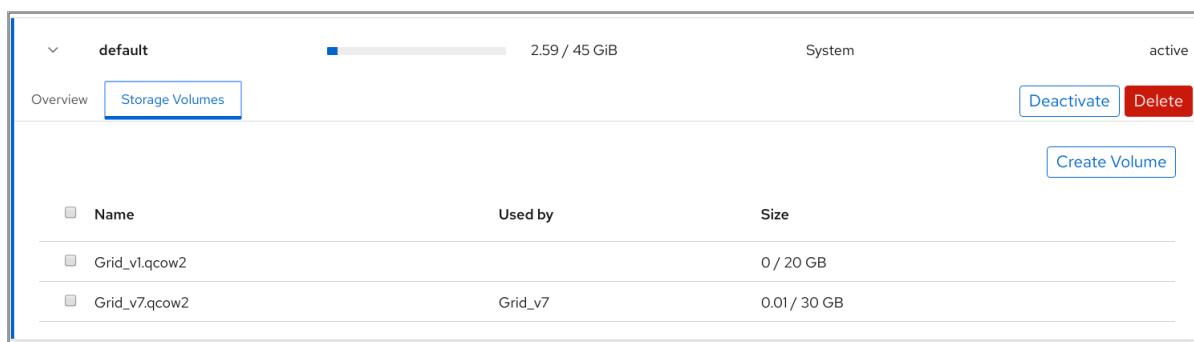


La información incluye lo siguiente:

- **Name** - El nombre del pool de almacenamiento.
 - **Size** - El tamaño del pool de almacenamiento.
 - **Connection** - La conexión utilizada para acceder al pool de almacenamiento.
 - **State** - El estado del pool de almacenamiento.
2. Haga clic en la fila del almacén cuya información desea ver. La fila se expande para revelar el panel de Visión General con la siguiente información sobre el pool de almacenamiento seleccionado:
 - **Path** - La ruta de acceso al pool de almacenamiento.
 - **Persistent** - Si el pool de almacenamiento es persistente o no.
 - **Autostart** - Si el pool de almacenamiento se inicia automáticamente o no.
 - **Type** - El tipo de pool de almacenamiento.



- Para ver una lista de los volúmenes de almacenamiento creados a partir del pool de almacenamiento, haga clic en **Volúmenes de almacenamiento**. Aparece el panel de volúmenes de almacenamiento, que muestra una lista de volúmenes de almacenamiento configurados con sus tamaños y la cantidad de espacio utilizado.



Recursos adicionales

- Para obtener instrucciones sobre cómo ver la información de todas las máquinas virtuales a las que está conectada la sesión de la consola web, consulte [Sección 6.2.1, “Ver un resumen de la virtualización en la consola web”](#).
- Para obtener instrucciones sobre cómo ver la información básica de una VM seleccionada a la que está conectada la sesión de la consola web, consulte [Sección 6.2.3, “Visualización de la información básica de la máquina virtual en la consola web”](#).
- Para obtener instrucciones sobre cómo ver el uso de recursos de una VM seleccionada a la que está conectada la sesión de la consola web, consulte [Sección 6.2.4, “Ver el uso de recursos de la máquina virtual en la consola web”](#).
- Para obtener instrucciones sobre cómo ver la información del disco de una VM seleccionada a la que está conectada la sesión de la consola web, consulte [Sección 6.2.5, “Visualización de la información del disco de la máquina virtual en la consola web”](#).
- Para obtener instrucciones sobre cómo ver la información de la interfaz de red virtual sobre una VM seleccionada a la que está conectada la sesión de la consola web, consulte [Sección 6.2.6, “Ver y editar la información de la interfaz de red virtual en la consola web”](#).

11.3.2. Creación de grupos de almacenamiento mediante la consola web

Una máquina virtual (VM) requiere un archivo, un directorio o un dispositivo de almacenamiento que puede utilizarse para crear volúmenes de almacenamiento para guardar la imagen de la VM o actuar como almacenamiento adicional. Puede crear grupos de almacenamiento a partir de recursos locales o basados en la red que luego puede utilizar para crear los volúmenes de almacenamiento.

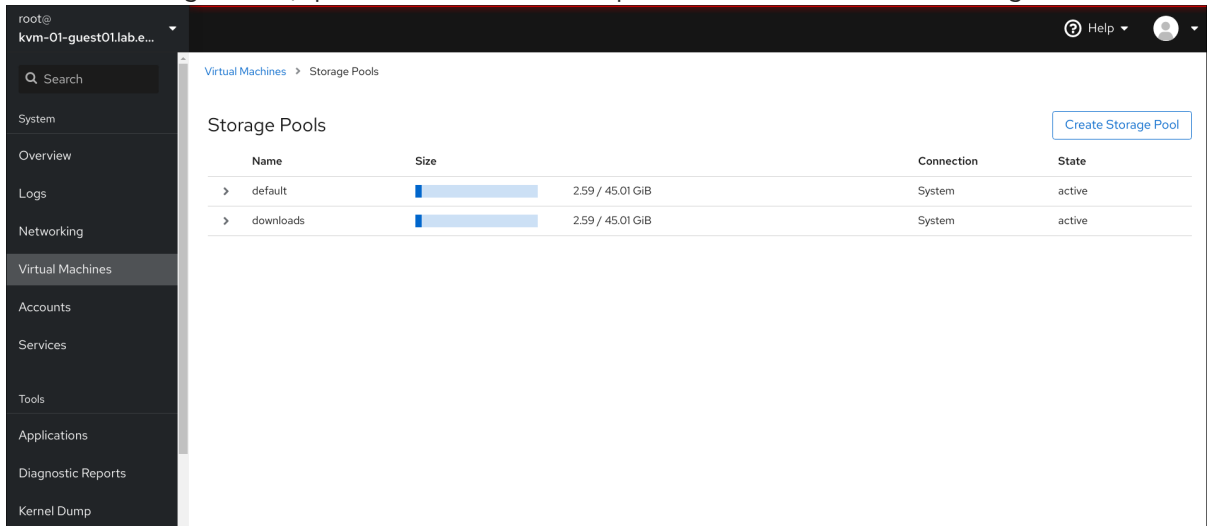
Para crear pools de almacenamiento utilizando la consola web de RHEL, consulte el siguiente procedimiento.

Requisitos previos

- Para utilizar la consola web para gestionar máquinas virtuales (VM), debe [instalar el complemento VM de la consola web](#).

Procedimiento

1. Haga clic en **Storage Pools** en la parte superior de la pestaña Máquinas Virtuales. Aparece la ventana Storage Pools, que muestra una lista de pools de almacenamiento configurados.



2. Haga clic en **Crear** Pool de Almacenamiento. Aparece el cuadro de diálogo Crear Pool de Almacenamiento.

Create Storage Pool ✕

Name

Type

Target Path

Startup Start pool when host boots

3. Introduzca la siguiente información en el cuadro de diálogo Crear grupo de almacenamiento:

- **Name** - El nombre del pool de almacenamiento.
- **Type** - El tipo de pool de almacenamiento. Puede ser un directorio de sistema de archivos, un sistema de archivos de red, un objetivo iSCSI, una unidad de disco física o un grupo de volúmenes LVM.

- **Target Path** - La ruta del pool de almacenamiento en el sistema de archivos del host.
 - **Startup** - Si el pool de almacenamiento se inicia o no cuando el host arranca.
4. Haga clic en **Crear**. El pool de almacenamiento se crea, el cuadro de diálogo Crear pool de almacenamiento se cierra y el nuevo pool de almacenamiento aparece en la lista de pools de almacenamiento.

Recursos adicionales

- Para obtener más información acerca de los grupos de almacenamiento, consulte la sección [sobre los grupos de almacenamiento](#).
- Para obtener instrucciones sobre cómo ver la información de los grupos de almacenamiento mediante la [consola web](#), consulte [Visualización de la información de los grupos de almacenamiento mediante la consola web](#).

11.3.3. Eliminación de grupos de almacenamiento mediante la consola web

Puede eliminar grupos de almacenamiento para liberar recursos en el host o en la red y mejorar el rendimiento del sistema. La eliminación de grupos de almacenamiento también libera recursos que pueden ser utilizados por otras máquinas virtuales (VM).



IMPORTANTE

A menos que se especifique explícitamente, la eliminación de un pool de almacenamiento no elimina simultáneamente los volúmenes de almacenamiento dentro de ese pool.

Para eliminar un pool de almacenamiento mediante la consola web de RHEL, consulte el siguiente procedimiento.



NOTA

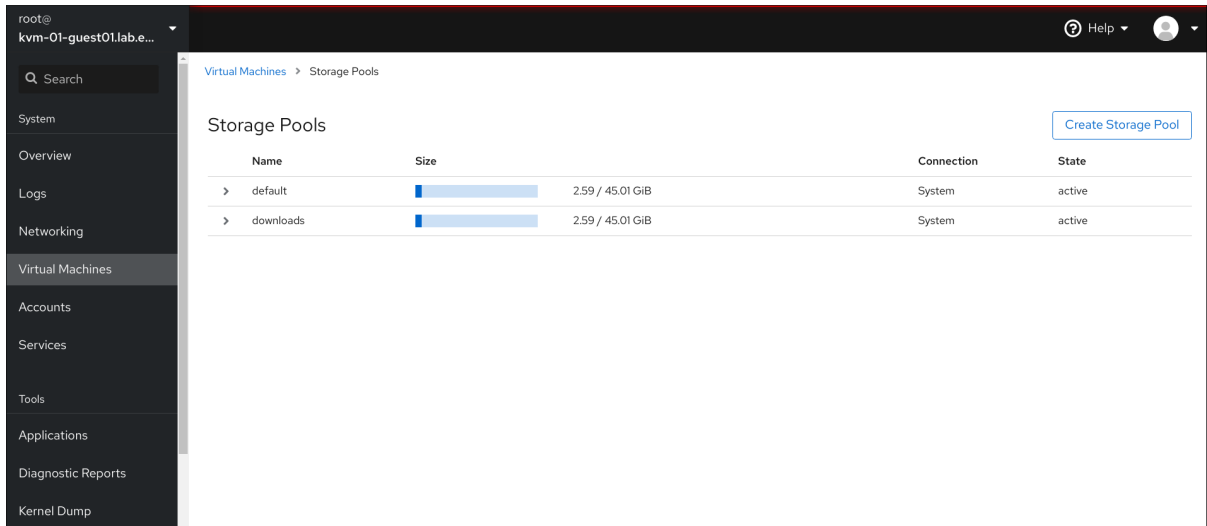
Si desea desactivar temporalmente un grupo de almacenamiento en lugar de eliminarlo, consulte [Desactivación de grupos de almacenamiento mediante la consola web](#)

Requisitos previos

- Para utilizar la consola web para gestionar las máquinas virtuales, debe [instalar el complemento de la consola web para máquinas virtuales](#).
- Si quieres eliminar un volumen de almacenamiento junto con el pool, primero debes [separar el disco](#) de la VM.

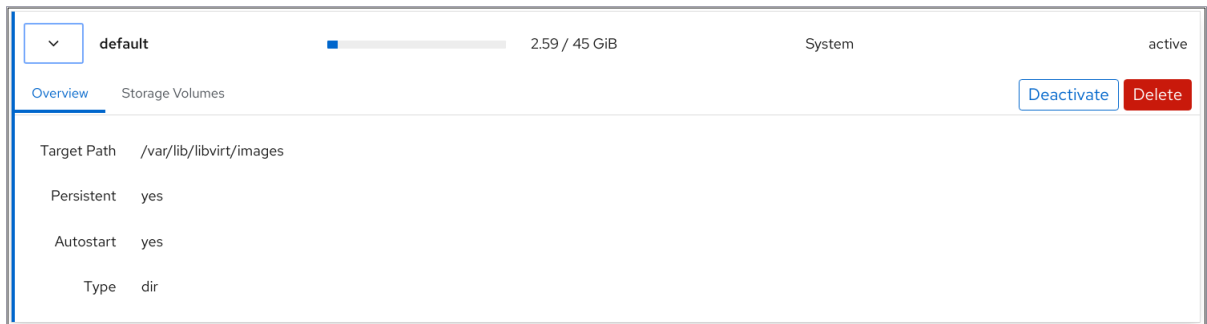
Procedimiento

1. Haga clic en **Storage Pools** en la parte superior de la pestaña Máquinas Virtuales. Aparece la ventana Storage Pools, que muestra una lista de pools de almacenamiento configurados.

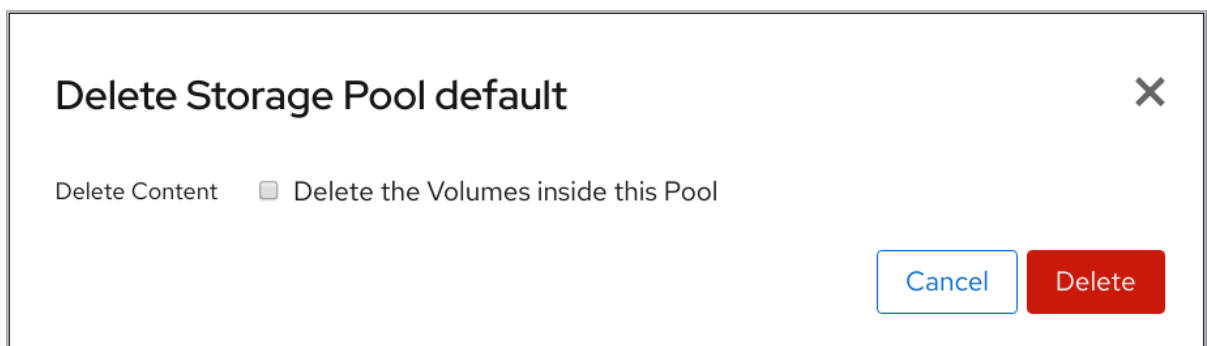


- En la ventana de **Pools de Almacenamiento**, haga clic en el pool de almacenamiento que desea eliminar.

La fila se expande para revelar el panel de Visión General con información básica sobre el pool de almacenamiento seleccionado y los controles para desactivar o eliminar el pool de almacenamiento.



- Haga clic en **Eliminar**. Aparece un diálogo de confirmación.



- Optional:** Para eliminar los volúmenes de almacenamiento dentro del pool, seleccione la casilla de verificación en el diálogo.
- Haga clic en **Eliminar**. Se elimina el pool de almacenamiento. Si ha seleccionado la casilla de verificación en el paso anterior, los volúmenes de almacenamiento asociados también se eliminan.

Recursos adicionales

- Para obtener más información acerca de los grupos de almacenamiento, consulte la sección [sobre los grupos de almacenamiento](#).
- Para obtener instrucciones sobre cómo ver la información de los grupos de almacenamiento mediante la consola web, consulte [cómo ver la información de los grupos de almacenamiento mediante la consola web](#).

11.3.4. Desactivación de grupos de almacenamiento mediante la consola web

Si no desea eliminar permanentemente un grupo de almacenamiento, puede desactivarlo temporalmente.

Cuando se desactiva un pool de almacenamiento, no se pueden crear nuevos volúmenes en ese pool. Sin embargo, cualquier máquina virtual (VM) que tenga volúmenes en ese pool seguirá funcionando. Esto es útil por varias razones, por ejemplo, puede limitar el número de volúmenes que se pueden crear en un pool para aumentar el rendimiento del sistema.

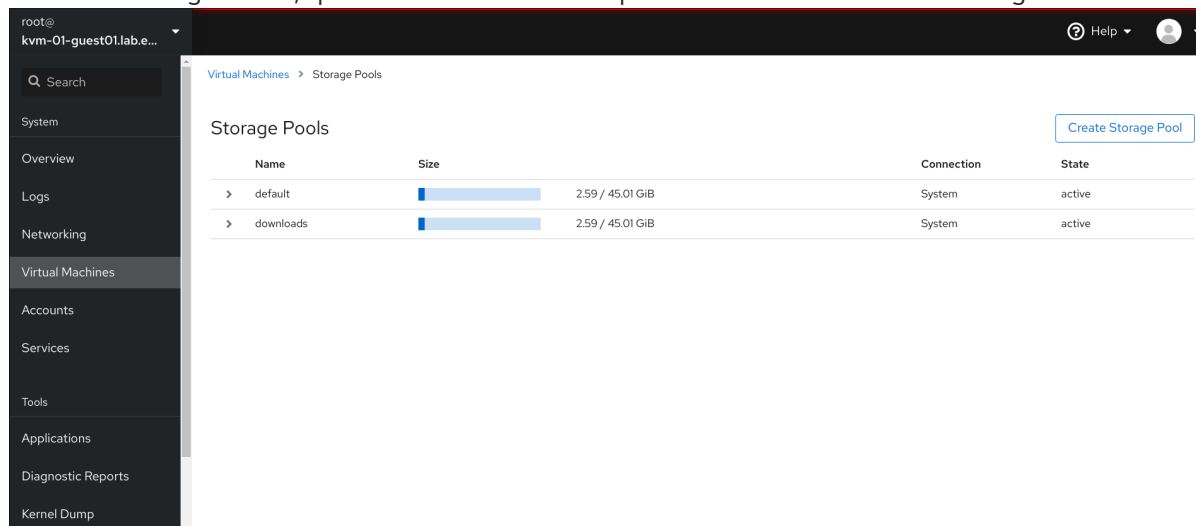
Para desactivar un pool de almacenamiento mediante la consola web de RHEL, consulte el siguiente procedimiento.

Requisitos previos

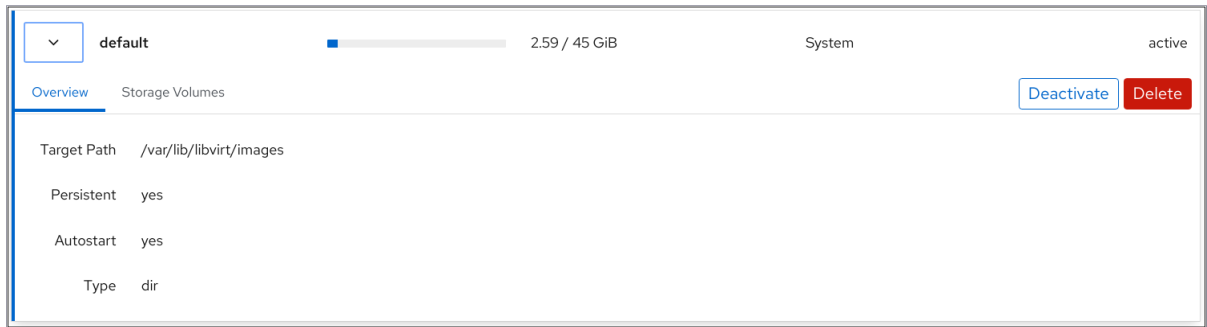
- Para utilizar la consola web para gestionar máquinas virtuales (VM), debe [instalar el complemento VM de la consola web](#).

Procedimiento

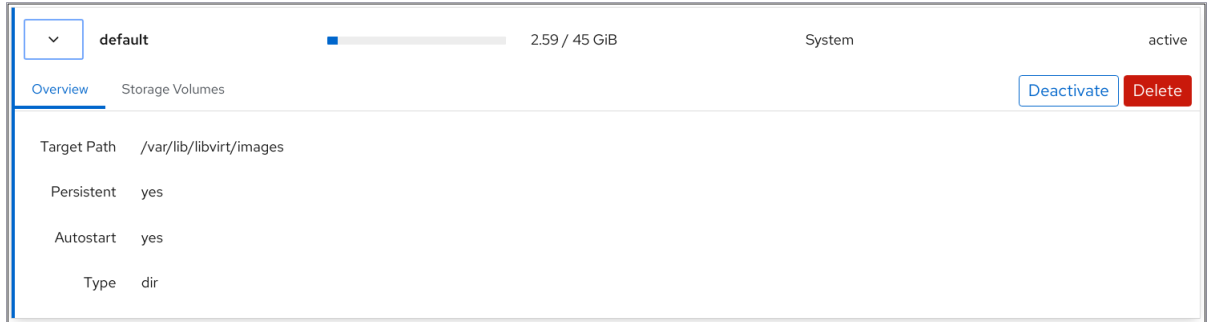
1. Haga clic en **Storage Pools** en la parte superior de la pestaña Máquinas Virtuales. Aparece la ventana Storage Pools, que muestra una lista de pools de almacenamiento configurados.



2. En la ventana de **Pools de Almacenamiento**, haga clic en el pool de almacenamiento que desea desactivar.
La fila se expande para revelar el panel de Visión General con información básica sobre el pool de almacenamiento seleccionado y los controles para desactivar y eliminar la VM.



3. Haga clic en **Desactivar**.



El pool de almacenamiento está desactivado.

Recursos adicionales

- Para obtener más información acerca de los grupos de almacenamiento, consulte la sección [sobre los grupos de almacenamiento](#).
- Para obtener instrucciones sobre cómo ver la información de los grupos de almacenamiento mediante la consola web, consulte [Visualización de la información de los grupos de almacenamiento mediante la consola web](#).

11.3.5. Creación de volúmenes de almacenamiento mediante la consola web

Para crear una máquina virtual (VM) que funcione se necesita un dispositivo de almacenamiento local asignado a la VM que pueda almacenar la imagen de la VM y los datos relacionados con la misma. Puede crear un volumen de almacenamiento en un pool de almacenamiento y asignarlo a una VM como disco de almacenamiento.

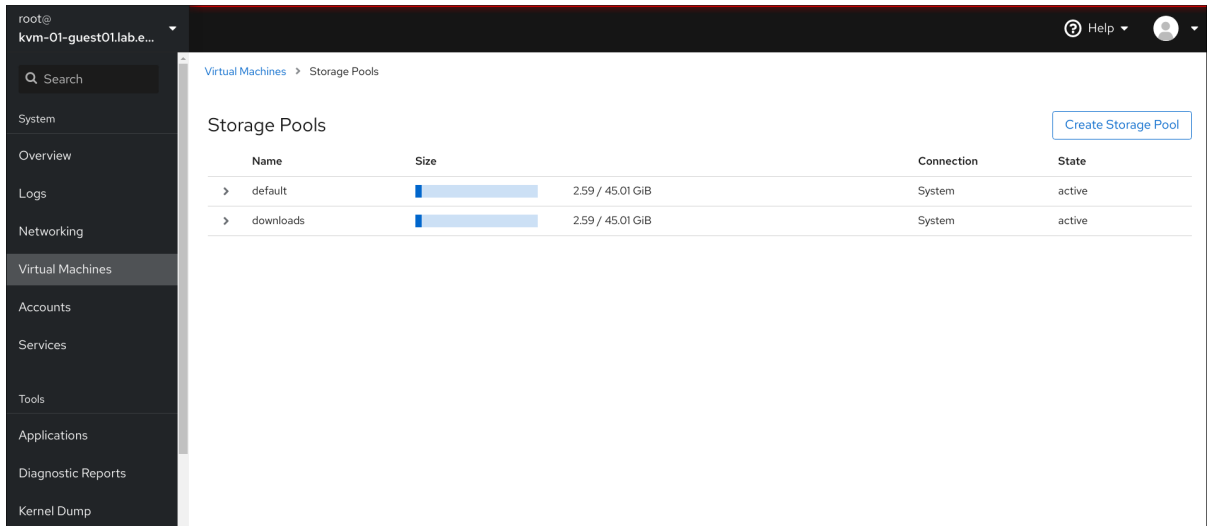
Para crear volúmenes de almacenamiento utilizando la consola web, consulte el siguiente procedimiento.

Requisitos previos

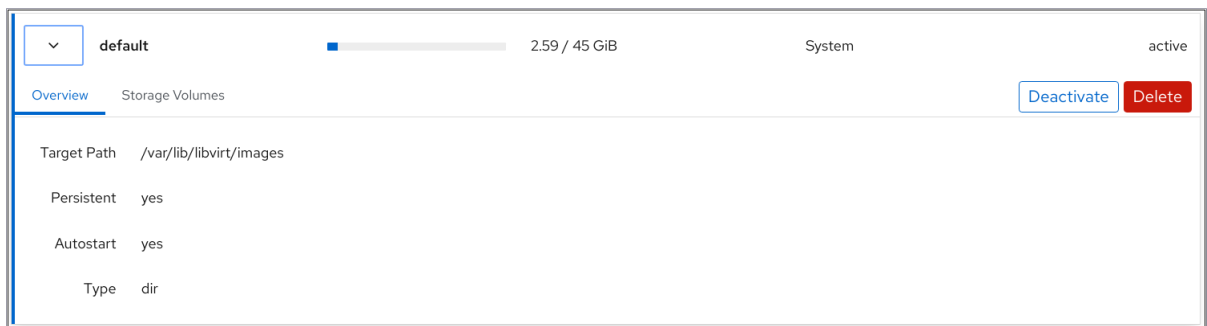
- Para utilizar la consola web para gestionar las máquinas virtuales, debe [instalar el complemento de la consola web para máquinas virtuales](#).

Procedimiento

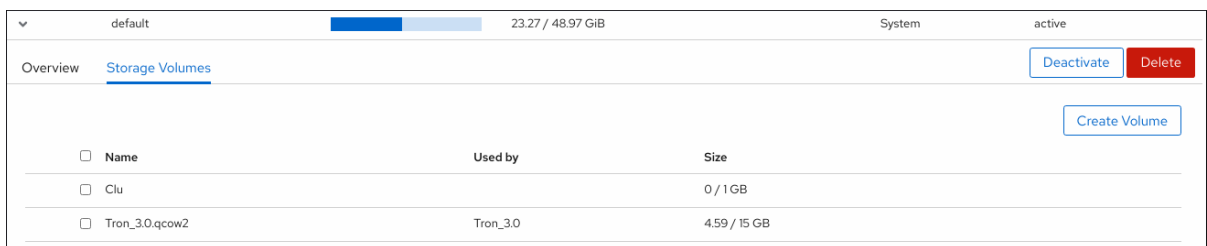
1. Haga clic en **Storage Pools** en la parte superior de la pestaña Máquinas Virtuales. Aparece la ventana Storage Pools, que muestra una lista de pools de almacenamiento configurados.



- En la ventana de **Pools de Almacenamiento**, haz clic en el pool de almacenamiento del que quieres crear un volumen de almacenamiento.
La fila se expande para revelar el panel de Visión General con información básica sobre el pool de almacenamiento seleccionado.



- Haga clic en **Volúmenes de Almacenamiento** junto a la pestaña Resumen en la fila expandida.
Aparece la pestaña Volumen de Almacenamiento con información básica sobre los volúmenes de almacenamiento existentes, si los hay.



- Haz clic en **Crear volumen**.
Aparece el cuadro de diálogo Crear Volumen de Almacenamiento.

5. Introduzca la siguiente información en el cuadro de diálogo Crear volumen de almacenamiento:

- **Name** - El nombre del volumen de almacenamiento.
- **Size** - El tamaño del volumen de almacenamiento en MiB o GiB.
- **Format** - El formato del volumen de almacenamiento. Los tipos admitidos son **qcow2** y **raw**.

6. Haga clic en **Crear**.

Se crea el volumen de almacenamiento, se cierra el diálogo Crear volumen de almacenamiento y el nuevo volumen de almacenamiento aparece en la lista de volúmenes de almacenamiento.

Recursos adicionales

- Para obtener más información sobre los volúmenes de almacenamiento, consulte [Comprender los volúmenes de almacenamiento](#).
- Para obtener información sobre cómo añadir discos a las [máquinas virtuales](#) mediante la [consola web](#), consulte [Añadir nuevos discos a las máquinas virtuales mediante la consola web](#).

11.3.6. Eliminación de volúmenes de almacenamiento mediante la consola web

Puede eliminar volúmenes de almacenamiento para liberar espacio en el pool de almacenamiento, o para eliminar elementos de almacenamiento asociados a máquinas virtuales (VMs) defectuosas.

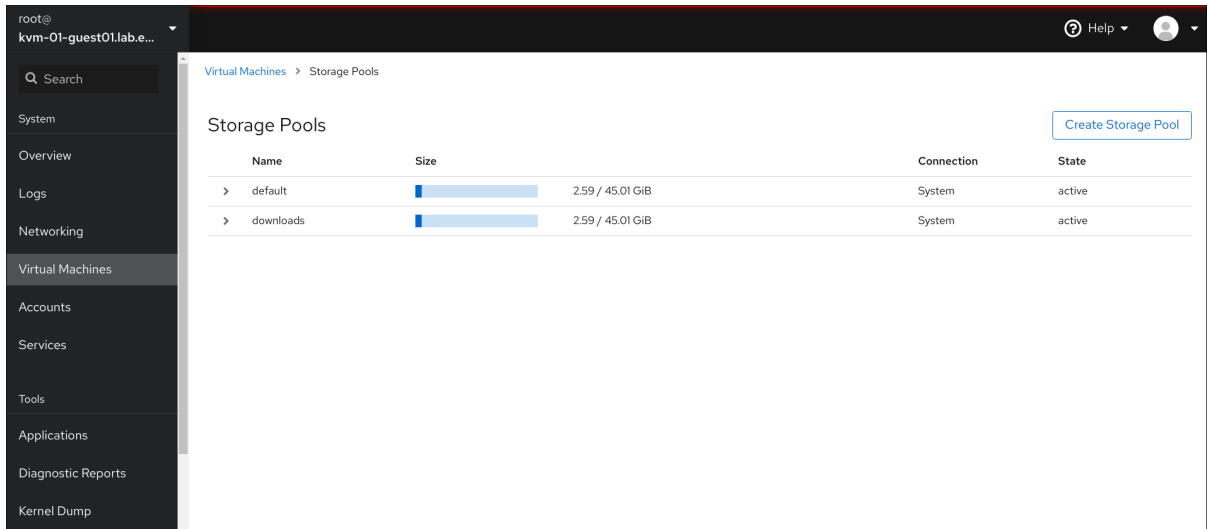
Para eliminar volúmenes de almacenamiento mediante la consola web de RHEL, consulte el siguiente procedimiento.

Requisitos previos

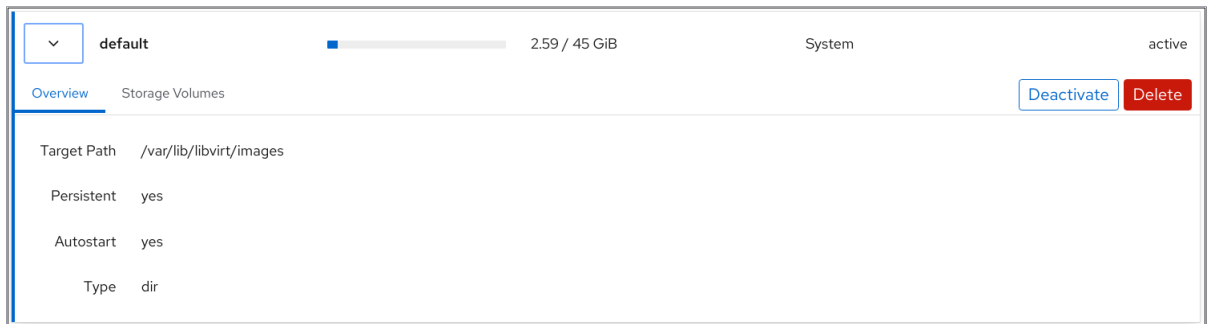
- Para utilizar la consola web para gestionar las máquinas virtuales, debe [instalar el complemento de la consola web para máquinas virtuales](#).
- Debe [separar el volumen](#) de la VM.

Procedimiento

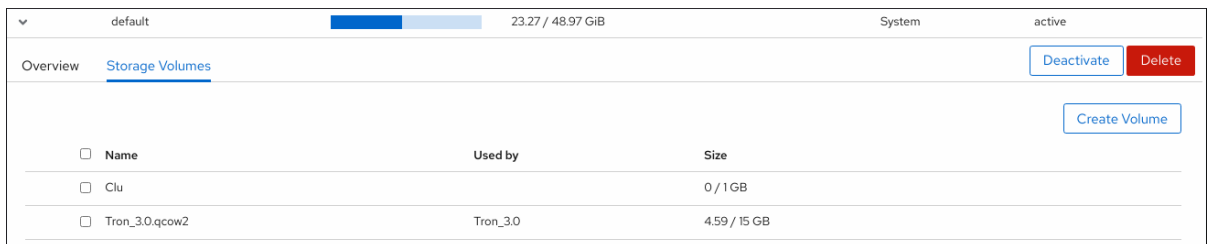
1. Haga clic en **Storage Pools** en la parte superior de la pestaña Máquinas Virtuales. Aparece la ventana Storage Pools, que muestra una lista de pools de almacenamiento configurados.



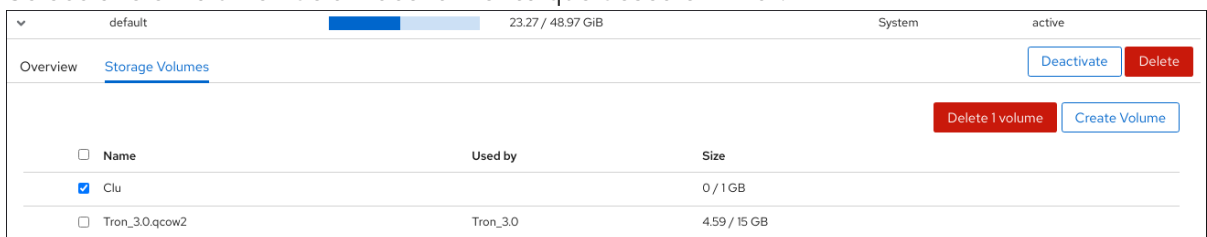
- En la ventana **Pools** de almacenamiento, haga clic en el pool de almacenamiento del que desea eliminar un volumen de almacenamiento.
La fila se expande para revelar el panel de Visión General con información básica sobre el pool de almacenamiento seleccionado.



- Haga clic en **Volúmenes de Almacenamiento** junto a la pestaña Resumen en la fila expandida. Aparece la pestaña Volumen de Almacenamiento con información básica sobre los volúmenes de almacenamiento existentes, si los hay.



- Seleccione el volumen de almacenamiento que desea eliminar.



- Haga clic en **Eliminar 1 Volumen**

Recursos adicionales

- Para obtener más información sobre los volúmenes de almacenamiento, consulte [Comprender los volúmenes de almacenamiento](#).

11.3.7. Gestión de los discos de las máquinas virtuales mediante la consola web

A través de la consola web de RHEL 8, se pueden gestionar los discos configurados para las máquinas virtuales a las que está conectada la consola web.

Puedes hacerlo:

- [Ver información sobre los discos](#).
- [Crear y adjuntar nuevos discos virtuales a las máquinas virtuales](#).
- [Adjunte los discos virtuales existentes a las máquinas virtuales](#).
- [Separar los discos virtuales de las máquinas virtuales](#).

11.3.7.1. Visualización de la información del disco de la máquina virtual en la consola web

El siguiente procedimiento describe cómo ver la información del disco de una máquina virtual (VM) a la que está conectada la sesión de la consola web.

Requisitos previos

Para utilizar la consola web para gestionar las máquinas virtuales, [instale el complemento VM de la consola web](#).

Procedimiento

1. Haga clic en la fila de la máquina virtual cuya información desea ver.
La fila se expande para revelar el panel de Visión General con información básica sobre la VM seleccionada y controles para apagar y eliminar la VM.
2. Haga clic en **Discos**.
Aparece el panel de Discos con información sobre los discos asignados a la VM.

The screenshot shows the 'Virtual Machines' management interface. At the top right, there are 'Create VM' and 'Import VM' buttons. Below the title, there are two buttons: 'Shut Down' and a menu icon. The main content area has tabs for 'Overview', 'Usage', 'Discos' (which is selected), 'Network Interfaces', and 'Consoles'. Under the 'Discos' tab, there is an 'Add Disk' button. Below that is a table with the following data:

Device	Used	Capacity	Bus	Access	Source	
disk	0,00 GiB	9 GiB	virtio	Writeable	File /var/lib/libvirt/images/Oasis_v1.qcow2	Remove

At the bottom right of the table, there is an 'Edit' button.

La información incluye lo siguiente:

- **Device** - El tipo de dispositivo del disco.
- **Used** - La cantidad de disco que se utiliza.
- **Capacity** - El tamaño del disco.

- **Bus** - El tipo de bus del disco.
- **Access** - Si el disco es de escritura o de sólo lectura.
- **Source** - El dispositivo o archivo de disco.

Recursos adicionales

- Para obtener instrucciones sobre cómo ver la información de todas las máquinas virtuales a las que está conectada la sesión de la consola web, consulte [Sección 6.2.1, "Ver un resumen de la virtualización en la consola web"](#).
- Para obtener instrucciones sobre cómo ver la información de los grupos de almacenamiento a los que está conectada la sesión de la consola web, consulte [Sección 6.2.2, "Visualización de la información del pool de almacenamiento mediante la consola web"](#).
- Para obtener instrucciones sobre cómo ver la información básica de una VM seleccionada a la que está conectada la sesión de la consola web, consulte [Sección 6.2.3, "Visualización de la información básica de la máquina virtual en la consola web"](#).
- Para obtener instrucciones sobre cómo ver el uso de recursos de una VM seleccionada a la que está conectada la sesión de la consola web, consulte [Sección 6.2.4, "Ver el uso de recursos de la máquina virtual en la consola web"](#).
- Para obtener instrucciones sobre cómo ver la información de la interfaz de red virtual sobre una VM seleccionada a la que está conectada la sesión de la consola web, consulte [Sección 6.2.6, "Ver y editar la información de la interfaz de red virtual en la consola web"](#).

11.3.7.2. Añadir nuevos discos a las máquinas virtuales mediante la consola web

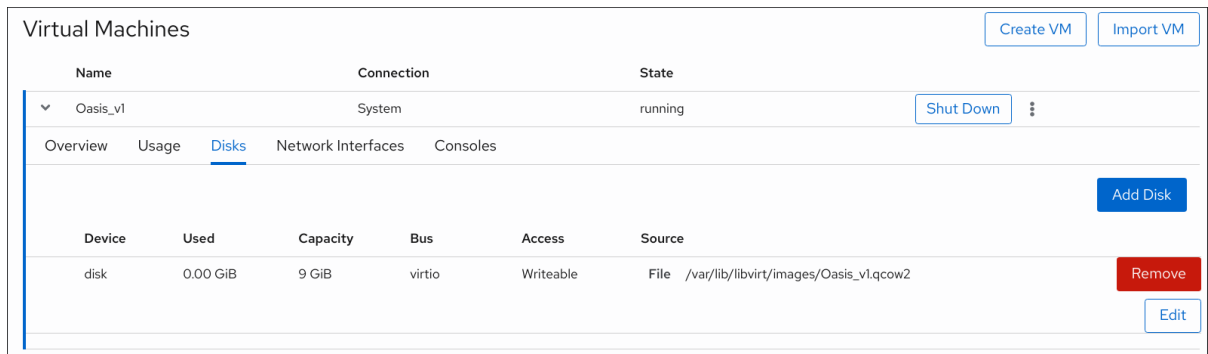
Puede añadir nuevos discos a las máquinas virtuales (VM) creando un nuevo volumen de almacenamiento y adjuntándolo a una VM mediante la consola web de RHEL 8.

Requisitos previos

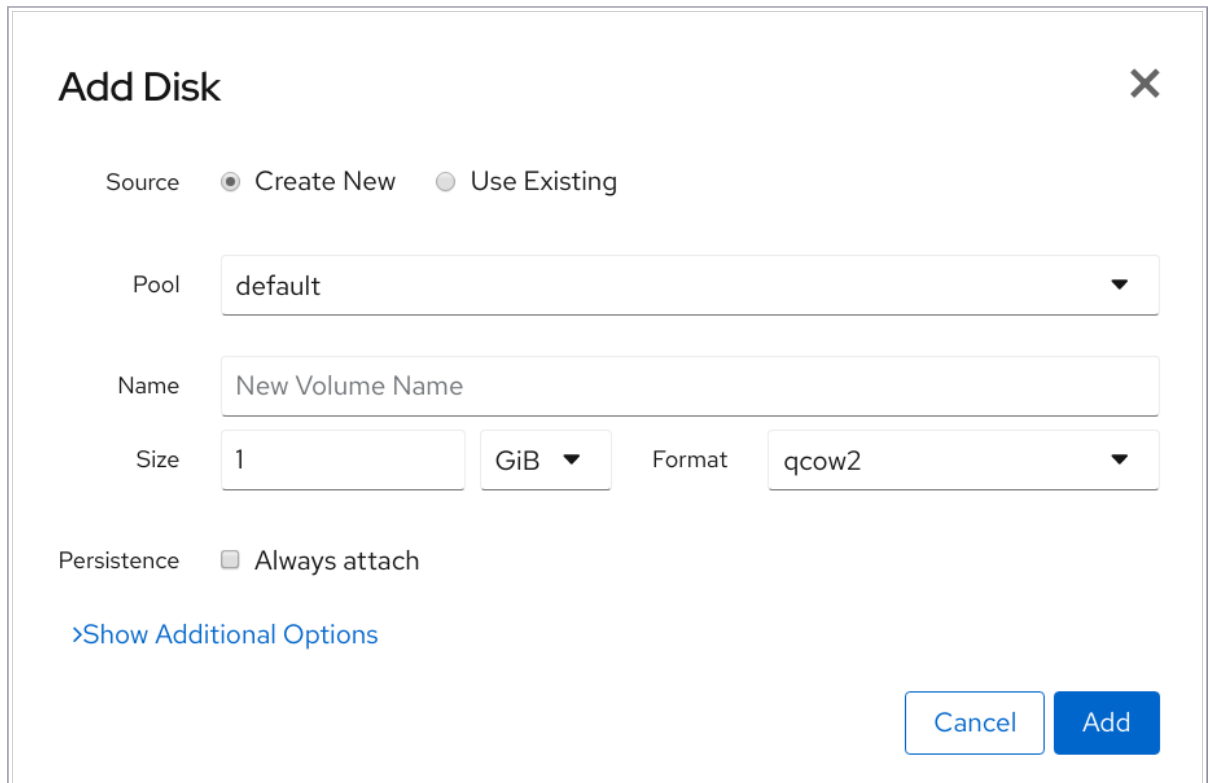
- Para utilizar la consola web para gestionar las máquinas virtuales, [instale el complemento VM de la consola web](#).

Procedimiento

1. En la interfaz de **máquinas virtuales**, haga clic en la fila de la máquina virtual para la que desea crear y adjuntar un nuevo disco.
La fila se expande para revelar el panel de Visión General con información básica sobre la VM seleccionada y controles para apagar y eliminar la VM.
2. Haga clic en **Discos**.
Aparece el panel de Discos con información sobre los discos configurados para la VM.



- Haga clic en **Añadir Disco**. Aparece el cuadro de diálogo Añadir Disco.



- Seleccione la opción *Create New*.
- Configure el nuevo disco.
 - Pool** - Seleccione el pool de almacenamiento del que se creará el disco virtual.
 - Name** - Introduzca un nombre para el disco virtual que se creará.
 - Size** - Introduzca el tamaño y seleccione la unidad (MiB o GiB) del disco virtual que se creará.
 - Format** - Seleccione el formato del disco virtual que se creará. Los tipos soportados son **qcow2** y **raw**.
 - Persistence** - Si se marca, el disco virtual es persistente. Si no se marca, el disco virtual es transitorio.

**NOTA**

Los discos transitorios sólo pueden añadirse a las máquinas virtuales que se están ejecutando.

- **Additional Options** - Establezca configuraciones adicionales para el disco virtual.
 - **Cache** - Seleccione el tipo de caché para el disco virtual.
 - **Bus** - Seleccione el tipo de bus para el disco virtual.
6. Haga clic en **Añadir**.
El disco virtual se crea y se conecta a la VM.

Recursos adicionales

- Para obtener instrucciones sobre cómo ver la información del disco de una VM seleccionada a la que está conectada la sesión de la consola web, consulte [Sección 11.3.7.1, “Visualización de la información del disco de la máquina virtual en la consola web”](#).
- Para obtener información sobre cómo adjuntar discos existentes a las máquinas virtuales, consulte [Sección 11.3.7.3, “Adjuntar discos existentes a máquinas virtuales mediante la consola web”](#).
- Para obtener información sobre cómo separar los discos de las máquinas virtuales, consulte [Sección 11.3.7.4, “Separar los discos de las máquinas virtuales”](#).

11.3.7.3. Adjuntar discos existentes a máquinas virtuales mediante la consola web

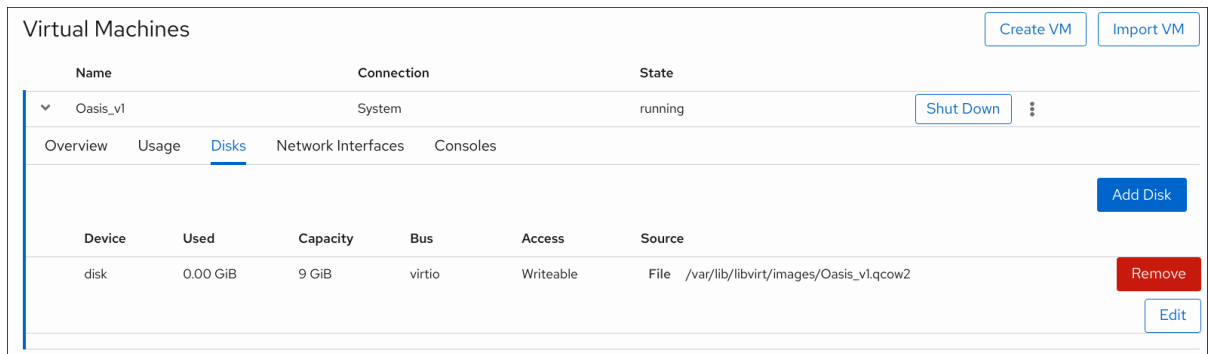
El siguiente procedimiento describe cómo adjuntar volúmenes de almacenamiento existentes como discos a una máquina virtual (VM) utilizando la consola web de RHEL 8.

Requisitos previos

- Para utilizar la consola web para gestionar las máquinas virtuales, [instale el complemento VM de la consola web](#).

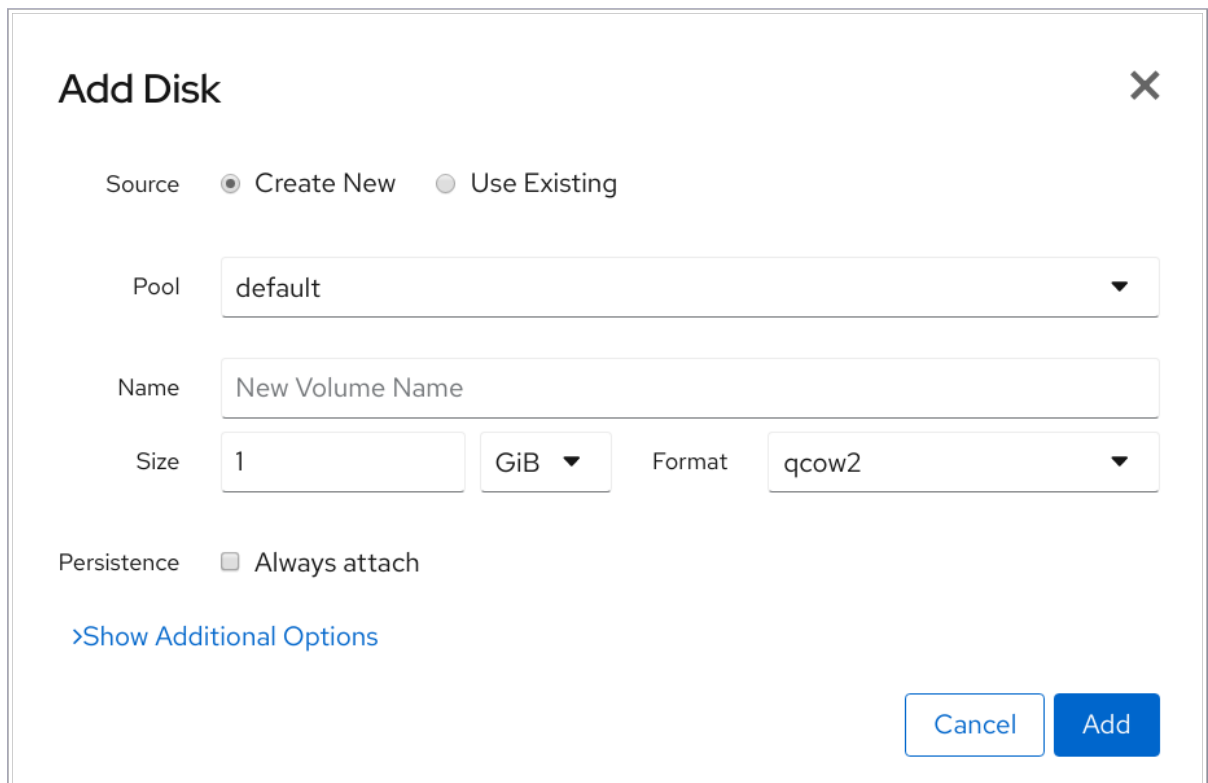
Procedimiento

1. En la interfaz de **máquinas virtuales**, haga clic en la fila de la máquina virtual a la que desea adjuntar un disco existente.
La fila se expande para revelar el panel de Visión General con información básica sobre la VM seleccionada y controles para apagar y eliminar la VM.
2. Haga clic en **Discos**.
Aparece el panel de Discos con información sobre los discos configurados para la VM.



The screenshot shows the 'Virtual Machines' management interface. At the top right, there are buttons for 'Create VM' and 'Import VM'. Below this is a table with columns for 'Name', 'Connection', and 'State'. A VM named 'Oasis_v1' is listed with a 'System' connection and a 'running' state. To the right of the VM name is a 'Shut Down' button and a menu icon. Below the table, there are tabs for 'Overview', 'Usage', 'Disks', 'Network Interfaces', and 'Consoles'. The 'Disks' tab is active, showing a table with columns: 'Device', 'Used', 'Capacity', 'Bus', 'Access', and 'Source'. A single disk is listed with 'disk' as the device, '0.00 GiB' used, '9 GiB' capacity, 'virtio' bus, 'Writeable' access, and 'File /var/lib/libvirt/images/Oasis_v1.qcow2' as the source. To the right of the disk table are buttons for 'Add Disk', 'Remove', and 'Edit'.

3. Haga clic en **Añadir Disco**.
Aparece el cuadro de diálogo Añadir Disco.



The 'Add Disk' dialog box is shown. It has a title bar with 'Add Disk' and a close button (X). The 'Source' section has two radio buttons: 'Create New' (selected) and 'Use Existing'. Below this is a 'Pool' dropdown menu set to 'default'. The 'Name' field contains 'New Volume Name'. The 'Size' field is '1', followed by a 'GiB' dropdown menu. The 'Format' dropdown menu is set to 'qcow2'. The 'Persistence' section has a checkbox for 'Always attach' which is unchecked. At the bottom left, there is a link '>Show Additional Options'. At the bottom right, there are 'Cancel' and 'Add' buttons.

4. Haga clic en el botón de radio **Use Existing**.
Los campos de configuración adecuados aparecen en el cuadro de diálogo Añadir Disco.

5. Configurar el disco para la VM.

- **Pool** - Seleccione el pool de almacenamiento desde el que se adjuntará el disco virtual.
- **Volume** - Seleccione el volumen de almacenamiento que se adjuntará.
- **Persistence** - Marque para que el disco virtual sea persistente. Borrarse para que el disco virtual sea transitorio.
- **Additional Options** - Establezca configuraciones adicionales para el disco virtual.
 - **Cache** - Seleccione el tipo de caché para el disco virtual.
 - **Bus** - Seleccione el tipo de bus para el disco virtual.

6. Haga clic en **Añadir**

El disco virtual seleccionado se adjunta a la VM.

Recursos adicionales

- Para obtener instrucciones sobre cómo ver la información del disco de una VM seleccionada a la que está conectada la sesión de la consola web, consulte [Sección 11.3.7.1, "Visualización de la información del disco de la máquina virtual en la consola web"](#).
- Para obtener información sobre la creación de nuevos discos y su asignación a las máquinas virtuales, consulte [Sección 11.3.7.2, "Añadir nuevos discos a las máquinas virtuales mediante la consola web"](#).
- Para obtener información sobre cómo separar los discos de las máquinas virtuales, consulte [Sección 11.3.7.4, "Separar los discos de las máquinas virtuales"](#).

11.3.7.4. Separar los discos de las máquinas virtuales

A continuación se describe cómo separar los discos de las máquinas virtuales (VM) utilizando la consola web de RHEL 8.

Requisitos previos

- Para utilizar la consola web para gestionar las máquinas virtuales, [instale el complemento VM de la consola web](#).

Procedimiento

1. En la interfaz de **las máquinas virtuales**, haga clic en la fila de la máquina virtual de la que desea separar un disco existente.

La fila se expande para revelar el panel de Visión General con información básica sobre la VM seleccionada y controles para apagar y eliminar la VM.

2. Haga clic en **Discos**.

Aparece el panel de Discos con información sobre los discos configurados para la VM.

The screenshot shows the 'Virtual Machines' interface. At the top right are 'Create VM' and 'Import VM' buttons. Below is a table of VMs with columns 'Name', 'Connection', and 'State'. The first row shows 'Oasis_v1' connected to 'System' and in 'running' state, with a 'Shut Down' button and a menu icon. Below this is a navigation bar with 'Overview', 'Usage', 'Disks' (selected), 'Network Interfaces', and 'Consoles'. The 'Disks' section has an 'Add Disk' button. A table lists the disk configuration:

Device	Used	Capacity	Bus	Access	Source	
disk	0.00 GiB	9 GiB	virtio	Writeable	File /var/lib/libvirt/images/Oasis_v1.qcow2	Remove

At the bottom right of the table is an 'Edit' button.

3. Haga clic en el botón **Eliminar** junto al disco que desea separar de la VM. Aparecerá un diálogo de confirmación en **Remove Disk**.
4. En el diálogo de confirmación, haga clic en **Eliminar**.
El disco virtual se separa de la VM.

Recursos adicionales

- Para obtener instrucciones sobre cómo ver la información del disco de una VM seleccionada a la que está conectada la sesión de la consola web, consulte [Sección 11.3.7.1, "Visualización de la información del disco de la máquina virtual en la consola web"](#).
- Para obtener información sobre la creación de nuevos discos y su asignación a las máquinas virtuales, consulte [Sección 11.3.7.2, "Añadir nuevos discos a las máquinas virtuales mediante la consola web"](#).
- Para obtener información sobre cómo adjuntar discos existentes a las máquinas virtuales, consulte [Sección 11.3.7.3, "Adjuntar discos existentes a máquinas virtuales mediante la consola web"](#).

CAPÍTULO 12. GESTIÓN DE DISPOSITIVOS NVIDIA vGPU

La función vGPU permite dividir un dispositivo de GPU NVIDIA físico en varios dispositivos virtuales, denominados **mediated devices**. Estos dispositivos mediados pueden asignarse entonces a múltiples máquinas virtuales (VM) como GPUs virtuales. Como resultado, estas VMs pueden compartir el rendimiento de una única GPU física.



IMPORTANTE

La asignación de una GPU física a las máquinas virtuales, con o sin el uso de dispositivos mediados, hace que el host no pueda utilizar la GPU.

12.1. CONFIGURACIÓN DE DISPOSITIVOS NVIDIA vGPU

Para configurar la función NVIDIA vGPU, debe descargar los controladores NVIDIA vGPU para su dispositivo GPU, crear dispositivos mediados y asignarlos a las máquinas virtuales previstas. Para obtener instrucciones detalladas, consulte a continuación.

Requisitos previos

- El paquete *mdevctl* está instalado.

```
# yum install mdevctl
```

- Su GPU soporta dispositivos mediados por vGPU. Para obtener una lista actualizada de las GPUs NVIDIA que admiten la creación de vGPUs, consulta [la documentación del software de la GPU NVIDIA](#).
 - Si no sabe qué GPU está utilizando su host, instale el paquete *lshw* y utilice el comando **lshw -C display**. El siguiente ejemplo muestra que el sistema utiliza una GPU NVIDIA Tesla P4, compatible con vGPU.

```
# lshw -C display
```

```
*-display
  description: 3D controller
  product: GP104GL [Tesla P4]
  vendor: NVIDIA Corporation
  physical id: 0
  bus info: pci@0000:01:00.0
  version: a1
  width: 64 bits
  clock: 33MHz
  capabilities: pm msi pciexpress cap_list
  configuration: driver=vfio-pci latency=0
  resources: irq:16 memory:f6000000-f6ffffff memory:e0000000-efffffff
  memory:f0000000-f1ffffff
```

Procedimiento

1. Descargue los controladores NVIDIA vGPU e instálelos en su sistema. Para obtener instrucciones, consulta [la documentación de NVIDIA](#).

- Si el instalador del software de NVIDIA no creó el archivo `/etc/modprobe.d/nvidia-installer-disable-nouveau.conf`, cree un archivo **conf** con cualquier nombre en `/etc/modprobe.d/`, y añada las siguientes líneas en el archivo:

```
blacklist nouveau
options nouveau modeset=0
```

- Regenerar el ramdisk inicial para el kernel actual, luego reiniciar.

```
# dracut --force
# reboot
```

- Compruebe que el kernel ha cargado el módulo `nvidia_vgpu_vfio` y que el servicio `nvidia-vgpu-mgr.service` está funcionando.

```
# lsmod | grep nvidia_vgpu_vfio
nvidia_vgpu_vfio 45011 0
nvidia 14333621 10 nvidia_vgpu_vfio
mdev 20414 2 vfio_mdev,nvidia_vgpu_vfio
vfio 32695 3 vfio_mdev,nvidia_vgpu_vfio,vfio_iommu_type1

# systemctl status nvidia-vgpu-mgr.service
nvidia-vgpu-mgr.service - NVIDIA vGPU Manager Daemon
   Loaded: loaded (/usr/lib/systemd/system/nvidia-vgpu-mgr.service; enabled; vendor preset: disabled)
   Active: active (running) since Fri 2018-03-16 10:17:36 CET; 5h 8min ago
   Main PID: 1553 (nvidia-vgpu-mgr)
   [...]

```

- Generar un UUID de dispositivo.

```
# uuidgen
30820a6f-b1a5-4503-91ca-0c10ba58692a
```

- Cree un dispositivo mediado a partir del hardware de la GPU que ha detectado en los requisitos previos y asigne el UUID generado al dispositivo.

El siguiente ejemplo muestra cómo crear un dispositivo mediado del tipo **nvidia-63** vGPU en una tarjeta NVIDIA Tesla P4 que funciona en el bus PCI 0000:01:00.0:

```
# mdevctl start -u 30820a6f-b1a5-4503-91ca-0c10ba58692a -p 0000:01:00.0 --type nvidia-63
```



NOTA

Para conocer los valores del tipo vGPU para dispositivos GPU específicos, consulte la [documentación del software de la GPU virtual](#).

- Haz que el dispositivo mediado sea persistente:

```
# mdevctl define --auto --uuid 30820a6f-b1a5-4503-91ca-0c10ba58692a
```

- Adjunte el dispositivo mediado a una VM que desee compartir los recursos de la vGPU. Para ello, añada las siguientes líneas, junto con el UUID previamente generado, a las secciones `<devices/>` en la configuración XML de la VM.

```
<hostdev mode='subsystem' type='mdev' managed='no' model='vfio-pci'>
  <source>
    <address uuid='30820a6f-b1a5-4503-91ca-0c10ba58692a'/>
  </source>
</hostdev>
```

Tenga en cuenta que cada UUID sólo puede asignarse a una máquina virtual a la vez.

- Para que la funcionalidad completa de los dispositivos mediados por la vGPU esté disponible en las VMs asignadas, configure el licenciamiento de software invitado de NVIDIA vGPU en las VMs. Para obtener más información e instrucciones, consulte la [Guía del usuario del servidor de licencias](#) de software de la GPU virtual de NVIDIA.

Verificación

- Enumera los dispositivos mediadores activos en tu host. Si la salida muestra un dispositivo definido con el UUID utilizado en el procedimiento, NVIDIA vGPU se ha configurado correctamente. Por ejemplo:

```
# mdevctl list
85006552-1b4b-45ef-ad62-de05be9171df 0000:01:00.0 nvidia-63
30820a6f-b1a5-4503-91ca-0c10ba58692a 0000:01:00.0 nvidia-63 (defined)
```

Recursos adicionales

- Para más información sobre el uso de la utilidad `mdevctl`, utilice `man mdevctl`.

12.2. ELIMINACIÓN DE DISPOSITIVOS NVIDIA vGPU

Para cambiar la configuración de [los dispositivos mediados por vGPU asignados](#), es necesario eliminar los dispositivos existentes de las VMs asignadas. Para obtener instrucciones, consulte a continuación:

Requisitos previos

- El paquete `mdevctl` está instalado.

```
# yum install mdevctl
```

- La VM de la que se quiere eliminar el dispositivo está apagada.

Procedimiento

- Obtenga el UUID del dispositivo mediado que desea eliminar. Para ello, utilice el comando `mdevctl list`:

```
# mdevctl list
85006552-1b4b-45ef-ad62-de05be9171df 0000:01:00.0 nvidia-63 (defined)
30820a6f-b1a5-4503-91ca-0c10ba58692a 0000:01:00.0 nvidia-63 (defined)
```

2. Detenga la instancia en ejecución del dispositivo vGPU mediado. Para ello, utilice el comando **mdevctl stop** con el UUID del dispositivo. Por ejemplo, para detener el dispositivo **30820a6f-b1a5-4503-91ca-0c10ba58692a**:

```
# mdevctl stop -u 30820a6f-b1a5-4503-91ca-0c10ba58692a
```

3. Elimine el dispositivo de la configuración XML de la VM. Para ello, utilice la utilidad **virsh edit** para editar la configuración XML de la VM, y elimine el segmento de configuración de mdev. El segmento tendrá un aspecto similar al siguiente:

```
<hostdev mode='subsystem' type='mdev' managed='no' model='vfio-pci'>
  <source>
    <address uuid='30820a6f-b1a5-4503-91ca-0c10ba58692a'/>
  </source>
</hostdev>
```

Tenga en cuenta que detener y separar el dispositivo mediado no lo elimina, sino que lo mantiene como **defined**. Como tal, puede [reiniciar](#) y [adjuntar](#) el dispositivo a una VM diferente.

4. **Optional:** Para borrar el dispositivo mediado detenido, elimine su definición:

```
# mdevctl undefine -u 30820a6f-b1a5-4503-91ca-0c10ba58692a
```

Verificación

- Si sólo ha detenido y desconectado el dispositivo, enumera los dispositivos mediados activos y los dispositivos mediados definidos.

```
# mdevctl list
85006552-1b4b-45ef-ad62-de05be9171df 0000:01:00.0 nvidia-63 (defined)
# mdevctl list --defined
85006552-1b4b-45ef-ad62-de05be9171df 0000:01:00.0 nvidia-63 auto (active)
30820a6f-b1a5-4503-91ca-0c10ba58692a 0000:01:00.0 nvidia-63 manual
```

Si el primer comando no muestra el dispositivo pero el segundo comando sí, el procedimiento fue exitoso.

- Si también has borrado el dispositivo, el segundo comando no debería mostrar el dispositivo.

```
# mdevctl list
85006552-1b4b-45ef-ad62-de05be9171df 0000:01:00.0 nvidia-63 (defined)
# mdevctl list --defined
85006552-1b4b-45ef-ad62-de05be9171df 0000:01:00.0 nvidia-63 auto (active)
```

Recursos adicionales

- Para más información sobre el uso de la utilidad **mdevctl**, utilice **man mdevctl**.

12.3. OBTENCIÓN DE INFORMACIÓN DE NVIDIA vGPU SOBRE SU SISTEMA

Para evaluar las capacidades de las funciones de la vGPU disponibles, puede obtener información adicional sobre los dispositivos mediados en su sistema, como por ejemplo

- Cuántos dispositivos mediados de un tipo determinado se pueden crear
- Qué dispositivos mediados están ya configurados en su sistema.

Requisitos previos

- El paquete `mdevctl` está instalado.

```
# yum install mdevctl
```

Procedimiento

- Para ver los tipos de vGPU disponibles en su host, utilice el comando **mdevctl types**.
Por ejemplo, a continuación se muestra la información de un sistema que utiliza una tarjeta física Tesla T4 en el bus PCI 0000:41:00.0:

```
# mdevctl types
0000:41:00.0
  nvidia-222
    Available instances: 0
    Device API: vfio-pci
    Name: GRID T4-1B
    Description: num_heads=4, frl_config=45, framebuffer=1024M,
max_resolution=5120x2880, max_instance=16
  nvidia-223
    Available instances: 0
    Device API: vfio-pci
    Name: GRID T4-2B
    Description: num_heads=4, frl_config=45, framebuffer=2048M,
max_resolution=5120x2880, max_instance=8
  nvidia-224
    Available instances: 0
    Device API: vfio-pci
    Name: GRID T4-2B4
    Description: num_heads=4, frl_config=45, framebuffer=2048M,
max_resolution=5120x2880, max_instance=8
  nvidia-225
    Available instances: 0
    Device API: vfio-pci
    Name: GRID T4-1A
    Description: num_heads=1, frl_config=60, framebuffer=1024M,
max_resolution=1280x1024, max_instance=16
  [...]
```

- Para ver los dispositivos vGPU activos en su host, incluyendo sus tipos, UUIDs y buses PCI de los dispositivos padres, utilice el comando **mdevctl list**:

```
# mdevctl list
85006552-1b4b-45ef-ad62-de05be9171df 0000:41:00.0 nvidia-223
83c32df7-d52e-4ec1-9668-1f3c7e4df107 0000:41:00.0 nvidia-223 (defined)
```

Este ejemplo muestra que el dispositivo **85006552-1b4b-45ef-ad62-de05be9171df** está funcionando pero no está definido, y el **83c32df7-d52e-4ec1-9668-1f3c7e4df107** está definido y funcionando.

Recursos adicionales

- Para más información sobre el uso de la utilidad **mdevctl**, utilice **man mdevctl**.

12.4. SERVICIOS DE STREAMING DE ESCRITORIO REMOTO PARA NVIDIA vGPU

Los siguientes servicios de streaming de escritorio remoto han sido probados con éxito para su uso con la función NVIDIA vGPU en los hosts RHEL 8:

- **HP-RGS** - Tenga en cuenta que actualmente no es posible utilizar HP-RGS con máquinas virtuales RHEL 8.
- **Mechdyne TGX** - Tenga en cuenta que actualmente no es posible utilizar Mechdyne TGX con máquinas virtuales de Windows Server 2016.
- **NICE DCV** - Cuando se utiliza este servicio de streaming, Red Hat recomienda utilizar la configuración de resolución fija, ya que el uso de la resolución dinámica en algunos casos resulta en una pantalla negra. Además, actualmente no es posible utilizar NICE DCV con máquinas virtuales RHEL 8.

12.5. INFORMACIÓN RELACIONADA

- Para obtener más información sobre el uso de NVIDIA vGPU en RHEL con KVM, consulte [la documentación del software de la GPU NVIDIA](#).

CAPÍTULO 13. CONFIGURACIÓN DE LAS CONEXIONES DE RED DE LAS MÁQUINAS VIRTUALES

Para que sus máquinas virtuales (VMs) se conecten a través de una red a su anfitrión, a otras VMs en su anfitrión, y a ubicaciones en una red externa, la red de la VM debe ser configurada en consecuencia. Para proporcionar la red de la VM, el hipervisor RHEL 8 y las VMs recién creadas tienen una configuración de red por defecto, que también puede ser modificada. Por ejemplo:

- Puede permitir que las máquinas virtuales de su host sean descubiertas y conectadas por ubicaciones externas al host, como si las máquinas virtuales estuvieran en la misma red que el host.
- Puede aislar parcial o completamente una VM del tráfico de red entrante para aumentar su seguridad y minimizar el riesgo de que cualquier problema con la VM afecte al host.

En las siguientes secciones se explican los distintos tipos de configuración de red de la VM y se ofrecen instrucciones para establecer las configuraciones de red de la VM seleccionadas.

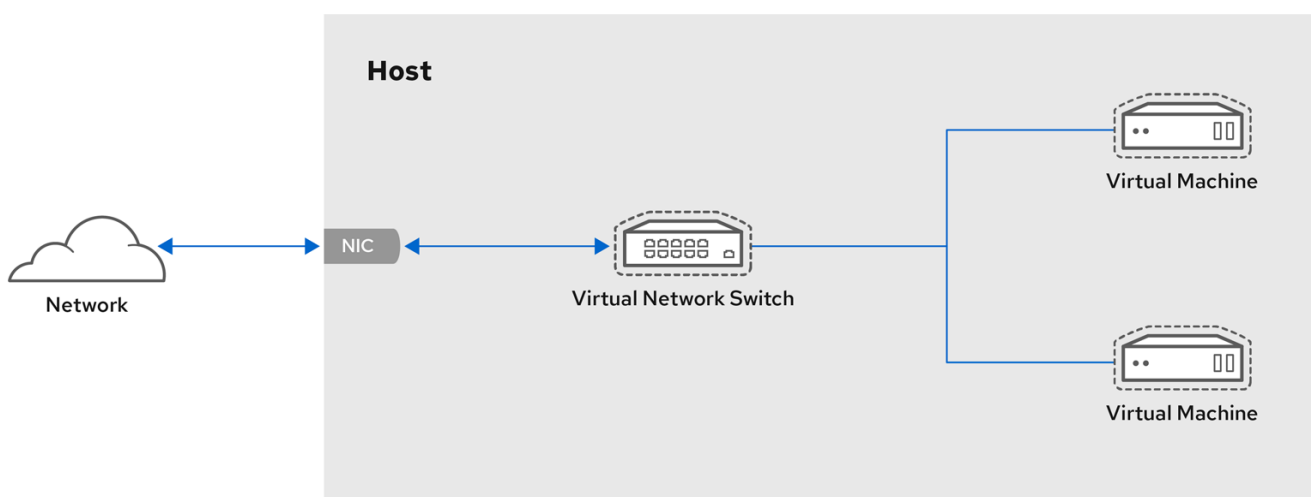
13.1. ENTENDER LA RED VIRTUAL

La conexión de las máquinas virtuales (VMs) a otros dispositivos y ubicaciones en una red tiene que ser facilitada por el hardware del host. Las siguientes secciones explican los mecanismos de las conexiones de red de las VM y describen la configuración de red de las VM por defecto.

13.1.1. Cómo funcionan las redes virtuales

La red virtual utiliza el concepto de conmutador de red virtual. Un conmutador de red virtual es una construcción de software que opera en una máquina anfitriona. Las máquinas virtuales se conectan a la red a través del conmutador de red virtual. En función de la configuración del conmutador virtual, una VM puede utilizar una red virtual existente gestionada por el hipervisor, o un método de conexión de red diferente.

La siguiente figura muestra un conmutador de red virtual que conecta dos máquinas virtuales a la red:



RHEL_52_1219

Desde la perspectiva de un sistema operativo invitado, una conexión de red virtual es lo mismo que una conexión de red física. Las máquinas anfitrionas ven los conmutadores de red virtuales como interfaces de red. Cuando el servicio **libvirt** se instala e inicia por primera vez, crea **virbr0**, la interfaz de red por defecto para las máquinas virtuales.

Para ver información sobre esta interfaz, utilice la utilidad **ip** en el host.

```
$ ip addr show virbr0
3: virbr0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc noqueue state
UNKNOWN link/ether 1b:c4:94:cf:fd:17 brd ff:ff:ff:ff:ff:ff
inet 192.168.122.1/24 brd 192.168.122.255 scope global virbr0
```

Por defecto, todas las máquinas virtuales de un mismo host están conectadas a la misma red virtual de tipo NAT, denominada **default**, que utiliza la interfaz **virbr0**. Para más detalles, consulte [Sección 13.1.2, “Configuración por defecto de la red virtual”](#).

Para el acceso básico a la red de salida desde las máquinas virtuales, no suele ser necesaria ninguna configuración de red adicional, ya que la red por defecto se instala junto con el paquete **libvirt**, y se inicia automáticamente cuando se inicia el servicio **libvirtd**.

Si se necesita una funcionalidad de red de la VM diferente, se pueden crear redes virtuales e interfaces de red adicionales y configurar las VM para que las utilicen. Además del NAT predeterminado, estas redes e interfaces pueden configurarse para utilizar uno de los siguientes modos:

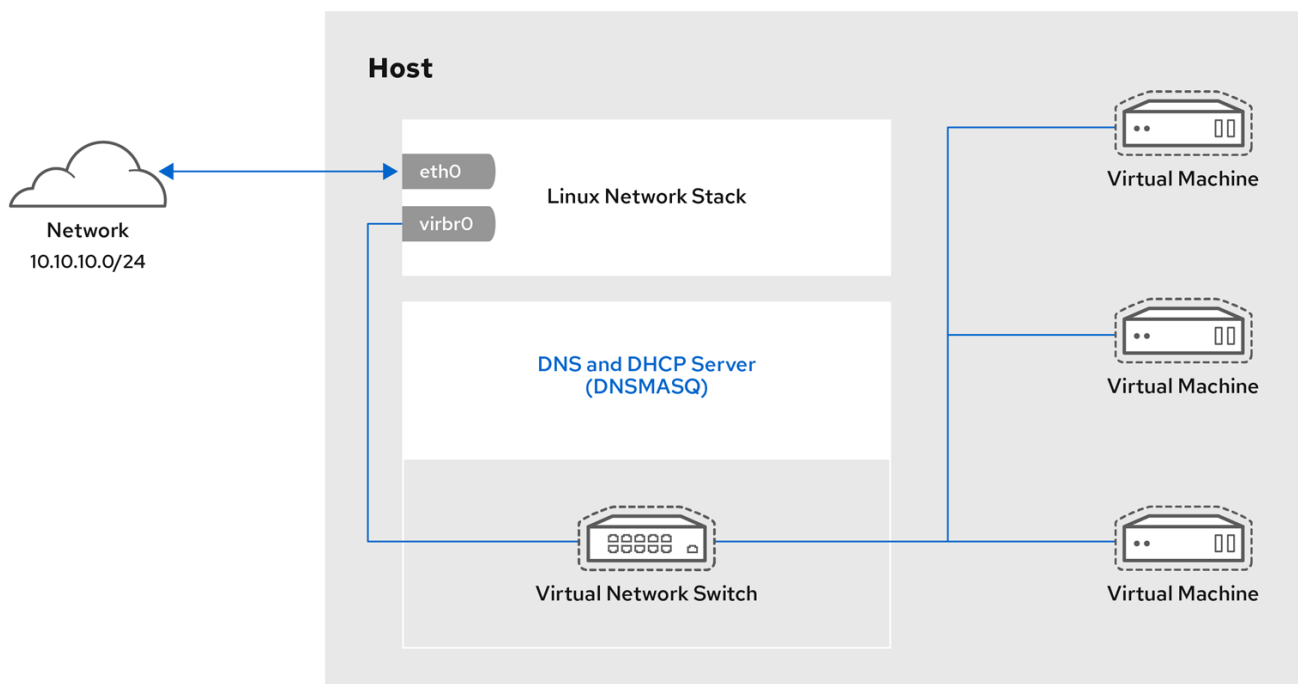
- [Modo de ruta](#)
- [Modo puente](#)
- [Modo aislado](#)
- [Modo abierto](#)

13.1.2. Configuración por defecto de la red virtual

Cuando el servicio **libvirtd** se instala por primera vez en un host de virtualización, contiene una configuración de red virtual inicial en modo de traducción de direcciones de red (NAT). Por defecto, todas las máquinas virtuales del host están conectadas a la misma red virtual **libvirt**, denominada **default**. Las máquinas virtuales en esta red pueden conectarse a ubicaciones tanto en el host como en la red más allá del host, pero con las siguientes limitaciones:

- Las máquinas virtuales en la red son visibles para el anfitrión y otras máquinas virtuales en el anfitrión, pero el tráfico de red se ve afectado por los cortafuegos en la pila de red del sistema operativo invitado y por las reglas de filtrado de red **libvirt** adjuntas a la interfaz del invitado.
- Las máquinas virtuales en la red pueden conectarse no son visibles para los lugares fuera del host. El tráfico saliente se ve afectado por las reglas NAT, así como por el firewall del sistema anfitrión.

El siguiente diagrama ilustra la configuración de red de la VM por defecto:



RHEL_52_1219

13.2. USO DE LA CONSOLA WEB PARA GESTIONAR LAS INTERFACES DE RED DE LAS MÁQUINAS VIRTUALES

Mediante la consola web de RHEL 8, puede gestionar las interfaces de red virtuales de las máquinas virtuales a las que está conectada la consola web. Usted puede:

- [Ver información sobre las interfaces de red y editarlas](#) .
- [Añadir interfaces de red a las máquinas virtuales](#) , y [desconectar o eliminar las interfaces](#).

13.2.1. Ver y editar la información de la interfaz de red virtual en la consola web

Utilizando la consola web de RHEL 8, puede ver y modificar las interfaces de red virtuales en una máquina virtual (VM) seleccionada:

Requisitos previos

- Para utilizar la consola web para gestionar las máquinas virtuales, [instale el complemento VM de la consola web](#).

Procedimiento

1. En la interfaz de **máquinas virtuales**, haga clic en la fila de la máquina virtual cuya información desea ver.
La fila se expande para revelar el panel de Visión General con información básica sobre la VM seleccionada y controles para apagar y eliminar la VM.
2. Haga clic en **Interfaces de red**.
Aparece el panel Interfaces de red con información sobre la interfaz de red virtual configurada para la VM.

Virtual Machines Create VM Import VM

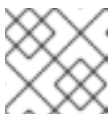
Name	Connection	State
▼ Oasis_v1	System	running Shut Down ⋮

Overview Usage Disks Network Interfaces Consoles Add Network Interface

Type	Model type	MAC Address	IP Address	Source	State	
network	virtio	52:54:00:c0:3e:f5	Unknown	default	up	Unplug Edit Delete

La información incluye lo siguiente:

- **Type** - El tipo de interfaz de red para la VM. Los tipos incluyen red virtual, puente a LAN y conexión directa.



NOTA

La conexión Ethernet genérica no es compatible con RHEL 8.2.

- **Model type** - El modelo de la interfaz de red virtual.
 - **MAC Address** - La dirección MAC de la interfaz de red virtual.
 - **IP Address** - La dirección IP de la interfaz de red virtual.
 - **Source** - El origen de la interfaz de red. Depende del tipo de red.
 - **State** - El estado de la interfaz de red virtual.
3. Para editar la configuración de la interfaz de red virtual, haga clic en **Editar**. Se abre el cuadro de diálogo Configuración de la interfaz de red virtual.

52:54:00:da:e4:97 Virtual Network Interface Settings ✕

Interface Type ▼ Virtual network

Source ▼ default

Model ▼ virtio (Linux, perf)

MAC Address **52:54:00:da:e4:97**

Cancel
Save

4. Cambiar el tipo de interfaz, la fuente o el modelo.
5. Haga clic en **Guardar**. La interfaz de red se modifica.



NOTA

Los cambios en la configuración de la interfaz de red virtual sólo tienen efecto después de reiniciar la VM.

Recursos adicionales

- Para obtener instrucciones sobre cómo ver la información de todas las máquinas virtuales a las que está conectada la sesión de la consola web, consulte [Sección 6.2.1, “Ver un resumen de la virtualización en la consola web”](#).
- Para obtener instrucciones sobre cómo ver la información de los grupos de almacenamiento a los que está conectada la sesión de la consola web, consulte [Sección 6.2.2, “Visualización de la información del pool de almacenamiento mediante la consola web”](#).
- Para obtener instrucciones sobre cómo ver la información básica de una VM seleccionada a la que está conectada la sesión de la consola web, consulte [Sección 6.2.3, “Visualización de la información básica de la máquina virtual en la consola web”](#).
- Para obtener instrucciones sobre cómo ver el uso de recursos de una VM seleccionada a la que está conectada la sesión de la consola web, consulte [Sección 6.2.4, “Ver el uso de recursos de la máquina virtual en la consola web”](#).
- Para obtener instrucciones sobre cómo ver la información del disco de una VM seleccionada a la que está conectada la sesión de la consola web, consulte [Sección 6.2.5, “Visualización de la información del disco de la máquina virtual en la consola web”](#).

13.2.2. Conexión de interfaces de red virtuales en la consola web

Mediante la consola web de RHEL 8, puede volver a conectar la interfaz de red virtual desconectada configurada para una máquina virtual (VM) seleccionada.

Requisitos previos

- Para utilizar la consola web para gestionar las máquinas virtuales, [instale el complemento VM de la consola web](#).

Procedimiento

1. En la interfaz de **máquinas virtuales**, haga clic en la fila de la máquina virtual cuya interfaz de red virtual desea conectar.
La fila se expande para revelar el panel de Visión General con información básica sobre la VM seleccionada y los controles para apagar la VM.
2. Haga clic en **Interfaces de red**.
Aparece el panel de Interfaces de Red con información sobre las interfaces de red virtuales configuradas para la VM.

Virtual Machines

Create VM Import VM

Name	Connection	State
Oasis_v1	System	running

Shut Down

Overview Usage Disks **Network Interfaces** Consoles

Add Network Interface

Type	Model type	MAC Address	IP Address	Source	State
network	virtio	52:54:00:c7:8eb7	Unknown	default	down

Plug Edit Delete

- Haga clic en **Conectar** en la fila de la interfaz de red virtual que desea conectar. La interfaz de red virtual seleccionada se conecta a la VM.

13.2.3. Desconexión de interfaces de red virtuales en la consola web

Mediante la consola web de RHEL 8, puede desconectar las interfaces de red virtuales conectadas a una máquina virtual (VM) seleccionada.

Requisitos previos

- Para utilizar la consola web para gestionar las máquinas virtuales, [instale el complemento VM de la consola web](#).

Procedimiento

- En la interfaz de **máquinas virtuales**, haga clic en la fila de la máquina virtual cuya interfaz de red virtual desea desconectar. La fila se expande para revelar el panel de Visión General con información básica sobre la VM seleccionada y controles para apagar y eliminar la VM.
- Haga clic en **Interfaces de red**. Aparece el panel de Interfaces de Red con información sobre las interfaces de red virtuales configuradas para la VM.

Virtual Machines

Create VM Import VM

Name	Connection	State
Oasis_v1	System	running

Shut Down

Overview Usage Disks **Network Interfaces** Consoles

Add Network Interface

Type	Model type	MAC Address	IP Address	Source	State
network	virtio	52:54:00:c0:3e:f5	Unknown	default	up

Unplug Edit Delete

- Haga clic en **Desenchufar** en la fila de la interfaz de red virtual que desea desconectar. La interfaz de red virtual seleccionada se desconecta de la VM.

13.3. CONFIGURACIONES DE RED DE MÁQUINAS VIRTUALES RECOMENDADAS MEDIANTE LA INTERFAZ DE LÍNEA DE COMANDOS

En muchos escenarios, la configuración de red de la VM por defecto es suficiente. Sin embargo, si es necesario ajustar la configuración, puede utilizar la interfaz de línea de comandos (CLI) para hacerlo. Las siguientes secciones describen configuraciones de red de VM seleccionadas para tales situaciones.

13.3.1. Configuración de máquinas virtuales visibles externamente mediante la interfaz de línea de comandos

Por defecto, una VM recién creada se conecta a una red de tipo NAT que utiliza **virbr0**, el puente virtual por defecto en el host. Esto garantiza que la VM pueda utilizar el controlador de interfaz de red (NIC) del host para conectarse a redes externas, pero la VM no es accesible desde sistemas externos.

Si necesita que una VM aparezca en la misma red externa que el hipervisor, debe utilizar [el modo puente](#) en su lugar. Para ello, adjunte la VM a un dispositivo puente conectado al dispositivo de red físico del hipervisor. Para utilizar la interfaz de línea de comandos para esto, siga las instrucciones siguientes.

Requisitos previos

- Una [VM existente](#) apagada.
- La configuración IP del hipervisor. Esto varía en función de la conexión de red del host. Como ejemplo, este procedimiento utiliza un escenario en el que el host está conectado a la red mediante un cable ethernet, y la dirección MAC del NIC físico del host está asignada a una IP estática en un servidor DHCP. Por lo tanto, la interfaz ethernet es tratada como la IP del hipervisor.

Para obtener la configuración IP de la interfaz ethernet, utilice la utilidad **ip addr**:

```
# ip addr
[...]
enp0s25: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP
group default qlen 1000
    link/ether 54:ee:75:49:dc:46 brd ff:ff:ff:ff:ff:ff
    inet 10.0.0.148/24 brd 10.0.0.255 scope global dynamic noprefixroute enp0s25
```

Procedimiento

1. Cree y configure una conexión de puente para la interfaz física en el host. Para obtener instrucciones, consulte la sección [Configuración de un puente de red](#). Tenga en cuenta que en un escenario en el que se utiliza la asignación de IP estática, debe mover la configuración IPv4 de la interfaz ethernet física a la interfaz de puente.
2. Modifique la red de la VM para que utilice la interfaz puente creada. Por ejemplo, lo siguiente establece que *testguest* utilice *bridge0*.

```
# virt-xml testguest --edit --network bridge=bridge0
Domain 'testguest' defined successfully.
```

3. Inicie la máquina virtual.

```
# virsh start testguest
```

4. En el sistema operativo invitado, ajuste la configuración IP y DHCP de la interfaz de red del sistema como si la VM fuera otro sistema físico en la misma red que el hipervisor.

Los pasos específicos para esto serán diferentes dependiendo del SO huésped utilizado por la VM. Por ejemplo, si el sistema operativo invitado es RHEL 8, consulte [Configuración de una conexión Ethernet](#).

Verificación

1. Asegúrese de que el puente recién creado se está ejecutando y contiene tanto la interfaz física del host como la interfaz de la VM.

```
# ip link show master bridge0
2: enp0s25: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel master
bridge0 state UP mode DEFAULT group default qlen 1000
    link/ether 54:ee:75:49:dc:46 brd ff:ff:ff:ff:ff:ff
10: vnet0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel master
bridge0 state UNKNOWN mode DEFAULT group default qlen 1000
    link/ether fe:54:00:89:15:40 brd ff:ff:ff:ff:ff:ff
```

2. Asegúrese de que la máquina virtual aparece en la misma red externa que el hipervisor:
 - a. En el sistema operativo invitado, obtenga el ID de red del sistema. Por ejemplo, si se trata de un invitado Linux:

```
# ip addr
[...]
enp0s0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state
UP group default qlen 1000
    link/ether 52:54:00:09:15:46 brd ff:ff:ff:ff:ff:ff
    inet 10.0.0.150/24 brd 10.0.0.255 scope global dynamic noprefixroute enp0s0
```

- b. Desde un sistema externo conectado a la red local, conéctese a la VM utilizando el ID obtenido.

```
# ssh root@10.0.0.150
root@10.0.0.150's password:
Last login: Mon Sep 24 12:05:36 2019
root~#*
```

Si la conexión funciona, la red se ha configurado correctamente.

Recursos adicionales

- Para obtener instrucciones sobre la creación de una VM visible desde el exterior mediante la consola web, consulte [Sección 13.4.1, "Configuración de máquinas virtuales visibles externamente mediante la consola web"](#).
- Para más información sobre el modo puente, consulte [Sección 13.5.3, "Red virtual en modo puente"](#).
- En ciertas situaciones, como cuando se utiliza una VPN cliente-sitio mientras la VM está alojada en el cliente, no es posible utilizar el modo puente para hacer que sus VMs estén disponibles en ubicaciones externas.
Para solucionar este problema, puede establecer un NAT de destino para la VM. Para más detalles, consulte el documento [Configuración y gestión de redes](#).

13.3.2. Aislar las máquinas virtuales entre sí mediante la interfaz de línea de comandos

Para evitar que una máquina virtual (VM) se comunice con otras VMs en su host, por ejemplo para evitar compartir datos o para aumentar la seguridad del sistema, puede aislar completamente la VM del tráfico de red del lado del host.

Por defecto, una VM recién creada se conecta a una red de tipo NAT que utiliza **virbr0**, el puente virtual por defecto en el host. Esto asegura que la VM pueda utilizar la NIC del host para conectarse a redes externas, así como a otras VMs en el host. Esta es una conexión generalmente segura, pero en algunos casos, la conectividad con las otras VMs puede ser un peligro para la seguridad o la privacidad de los datos. En tales situaciones, se puede aislar la VM utilizando la conexión directa **macvtap** en modo privado en lugar de la red por defecto.

En el modo privado, la VM es visible para los sistemas externos y puede recibir una IP pública en la subred del host, pero la VM y el host no pueden acceder el uno al otro, y la VM tampoco es visible para otras VMs en el host.

Para obtener instrucciones para configurar el modo privado **macvtap** en su VM utilizando la CLI, consulte a continuación.

Requisitos previos

- Una [VM existente](#) con la configuración NAT por defecto.
- El nombre de la interfaz del host que desea utilizar para la conexión **macvtap**. La interfaz que debes seleccionar variará dependiendo de tu caso de uso y de la configuración de red de tu host. Como ejemplo, este procedimiento utiliza la interfaz ethernet física del host.

Para obtener el nombre de la interfaz de destino:

```
$ ip addr
[...]
2: enp0s31f6: <NO-CARRIER,BROADCAST,MULTICAST,UP> mtu 1500 qdisc fq_codel
state DOWN group default qlen 1000
    link/ether 54:e1:ad:42:70:45 brd ff:ff:ff:ff:ff:ff
[...]
```

Procedimiento

- Utilice la interfaz seleccionada para configurar el modo privado **macvtap** en la VM seleccionada. El siguiente ejemplo configura **macvtap** en modo privado en la interfaz **enp0s31f6** para la VM llamada *panic-room*.

```
# virt-xml panic-room --edit --network
type=direct,source=enp0s31f6,source.mode=private
Domain panic-room XML defined successfully
```

Verificación

1. Inicie la VM actualizada.

```
# virsh start panic-room
Domain panic-room started
```

2. Enumerar las estadísticas de la interfaz para la VM.

```
# virsh domstats panic-room --interface
Domain: 'panic-room'
net.count=1
net.0.name=macvtap0
net.0.rx.bytes=0
net.0.rx.pkts=0
net.0.rx.errs=0
net.0.rx.drop=0
net.0.tx.bytes=0
net.0.tx.pkts=0
net.0.tx.errs=0
net.0.tx.drop=0
```

Si el comando muestra una salida similar, la VM ha sido aislada con éxito.

Recursos adicionales

- Para obtener instrucciones sobre cómo aislar una máquina virtual mediante la consola web, consulte [Sección 13.4.2, "Aislar las máquinas virtuales entre sí mediante la consola web"](#).
- Para obtener más información sobre el modo privado de **macvtap**, consulte [Sección 13.5.6, "Fijación directa del dispositivo de red virtual"](#).
- Para conocer las medidas de seguridad adicionales que puede establecer en una VM, consulte [Capítulo 15, Asegurar las máquinas virtuales](#).

13.4. CONFIGURACIONES DE RED DE MÁQUINAS VIRTUALES RECOMENDADAS MEDIANTE LA CONSOLA WEB

En muchos escenarios, la configuración de red de la VM por defecto es suficiente. Sin embargo, si es necesario ajustar la configuración, puede utilizar la consola web de RHEL 8 para hacerlo. Las siguientes secciones describen configuraciones de red de VM seleccionadas para tales situaciones.

13.4.1. Configuración de máquinas virtuales visibles externamente mediante la consola web

Por defecto, una VM recién creada se conecta a una red de tipo NAT que utiliza **virbr0**, el puente virtual por defecto en el host. Esto garantiza que la VM pueda utilizar el controlador de interfaz de red (NIC) del host para conectarse a redes externas, pero la VM no es accesible desde sistemas externos.

Si necesita que una VM aparezca en la misma red externa que el hipervisor, debe utilizar [el modo puente](#) en su lugar. Para ello, adjunte la VM a un dispositivo puente conectado al dispositivo de red físico del hipervisor. Para utilizar la consola web de RHEL 8 para esto, siga las siguientes instrucciones.

Requisitos previos

- Para utilizar la consola web para gestionar las máquinas virtuales, [instale el complemento VM de la consola web](#).
- Una [VM existente](#) apagada.
- La configuración IP del hipervisor. Esto varía en función de la conexión de red del host. Como

ejemplo, este procedimiento utiliza un escenario en el que el host está conectado a la red mediante un cable ethernet, y la dirección MAC del NIC físico del host está asignada a una IP estática en un servidor DHCP. Por lo tanto, la interfaz ethernet es tratada como la IP del hipervisor.

Para obtener la configuración IP de la interfaz ethernet, vaya a la pestaña **Networking** en la consola web, y vea la sección **Interfaces**.

Interfaces		
Name	IP Address	Sending
enp0s25	10.0.0.148/24, 2a00:1028:83a4:1eda:91c7:667:8845:fa2e/64	2.29 Mbps

Procedimiento

1. Cree y configure una conexión de puente para la interfaz física en el host. Para obtener instrucciones, consulte [Configuración de puentes de red en la consola web](#) . Tenga en cuenta que en un escenario en el que se utiliza la asignación de IP estática, debe mover la configuración IPv4 de la interfaz ethernet física a la interfaz de puente.
2. Modifique la red de la VM para utilizar la interfaz puenteada. En la pestaña [Interfaces de red](#) de la VM:
 - a. Haga clic en **Añadir interfaz de red**
 - b. En el diálogo **Add Virtual Network Interface**, configure:
 - **Interface Type** a **Bridge to LAN**
 - Fuente al puente recién creado, por ejemplo **bridge0**
 - c. Haga clic en **Añadir**
 - d. **Optional:** Haga clic en **Desenchufar** para todas las demás interfaces conectadas a la VM.
3. Haga clic en **Ejecutar** para iniciar la VM.
4. En el sistema operativo invitado, ajuste la configuración IP y DHCP de la interfaz de red del sistema como si la VM fuera otro sistema físico en la misma red que el hipervisor. Los pasos específicos para esto serán diferentes dependiendo del SO huésped utilizado por la VM. Por ejemplo, si el sistema operativo invitado es RHEL 8, consulte [Configuración de una conexión Ethernet](#).

Verificación

1. En la pestaña **Networking** de la consola web del host, haga clic en la fila con el puente recién creado para asegurarse de que se está ejecutando y contiene tanto la interfaz física del host como la interfaz de la VM.

Ports	Sending	Receiving	
enp0s25	2.03 Kbps	2.09 Kbps	<input checked="" type="checkbox"/> -
vnet0	688 bps	624 bps	<input checked="" type="checkbox"/> -

2. Asegúrese de que la máquina virtual aparece en la misma red externa que el hipervisor.
 - a. En el sistema operativo invitado, obtenga el ID de red del sistema. Por ejemplo, si se trata de un invitado Linux:

```
# ip addr
[...]
enp0s0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state
UP group default qlen 1000
    link/ether 52:54:00:09:15:46 brd ff:ff:ff:ff:ff:ff
    inet 10.0.0.150/24 brd 10.0.0.255 scope global dynamic noprefixroute enp0s0
```

- b. Desde un sistema externo conectado a la red local, conéctese a la VM utilizando el ID obtenido.

```
# ssh root@10.0.0.150
root@110.34.5.18's password:
Last login: Mon Sep 24 12:05:36 2019
root~#*
```

Si la conexión funciona, la red se ha configurado correctamente.

Recursos adicionales

- Para obtener instrucciones sobre la creación de una VM visible desde el exterior utilizando la CLI, consulte [Sección 13.3.1, "Configuración de máquinas virtuales visibles externamente mediante la interfaz de línea de comandos"](#).
- Para más información sobre el modo puente, consulte [Sección 13.5.3, "Red virtual en modo puente"](#).
- En ciertas situaciones, como cuando se utiliza una VPN cliente-sitio mientras la VM está alojada en el cliente, no es posible utilizar el modo puente para hacer que sus VMs estén disponibles en ubicaciones externas.
Para solucionar este problema, puede establecer un NAT de destino para la VM. Para más detalles, consulte el documento [Configuración y gestión de redes](#).

13.4.2. Aislar las máquinas virtuales entre sí mediante la consola web

Para evitar que una máquina virtual (VM) se comunice con otras VMs en su host, por ejemplo para evitar compartir datos o para aumentar la seguridad del sistema, puede aislar completamente la VM del tráfico de red del lado del host.

Por defecto, una VM recién creada se conecta a una red de tipo NAT que utiliza **virbr0**, el puente virtual por defecto en el host. Esto asegura que la VM pueda utilizar la NIC del host para conectarse a redes externas, así como a otras VMs en el host. Esta es una conexión generalmente segura, pero en algunos casos, la conectividad con las otras VMs puede ser un peligro para la seguridad o la privacidad de los datos. En tales situaciones, se puede aislar la VM utilizando la conexión directa **macvtap** en modo privado en lugar de la red por defecto.

En el modo privado, la VM es visible para los sistemas externos y puede recibir una IP pública en la subred del host, pero la VM y el host no pueden acceder el uno al otro, y la VM tampoco es visible para otras VMs en el host.

Para obtener instrucciones para configurar el modo privado de **macvtap** en su VM utilizando la consola web, consulte a continuación.

Requisitos previos

- Para utilizar la consola web para gestionar las máquinas virtuales, [instale el complemento VM de la consola web](#).
- Una [VM existente](#) con la configuración NAT por defecto.

Procedimiento

1. En el panel **Virtual Machines**, haga clic en la fila con la máquina virtual que desea aislar. Se abre un panel con la información básica de la VM.
2. Haga clic en la pestaña **Network Interfaces**.
3. Haga clic en **Editar**. Se abre el diálogo **Virtual Machine Interface Settings**.
4. Configure **Interface Type** como **Direct Attachment**
5. Establezca **Source** en la interfaz de host de su elección. Tenga en cuenta que la interfaz que seleccione variará en función de su caso de uso y de la configuración de red de su host.

Verificación

1. Inicie la VM haciendo clic en **Ejecutar**.
2. En el panel **Terminal** de la consola web, liste las estadísticas de la interfaz para la VM. Por ejemplo, para ver el tráfico de la interfaz de red para la VM *panic-room*:

```
# virsh domstats panic-room --interface
Domain: 'panic-room'
net.count=1
net.0.name=macvtap0
net.0.rx.bytes=0
net.0.rx.pkts=0
net.0.rx.errs=0
net.0.rx.drop=0
```

```
net.0.tx.bytes=0
net.0.tx.pkts=0
net.0.tx.errs=0
net.0.tx.drop=0
```

Si el comando muestra una salida similar, la VM ha sido aislada con éxito.

Recursos adicionales

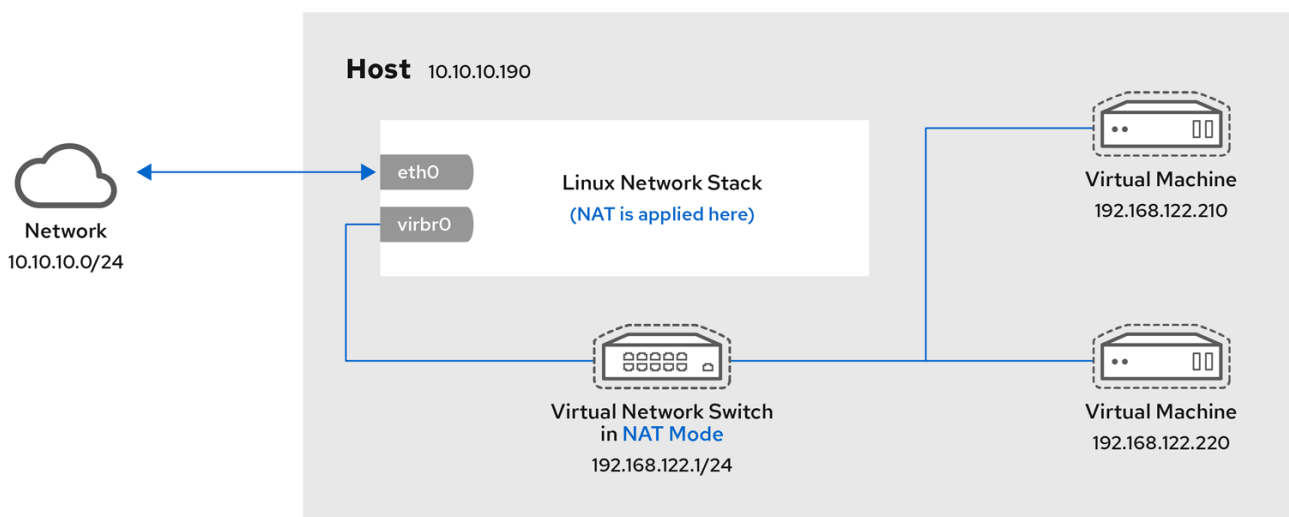
- Para obtener instrucciones sobre cómo aislar una máquina virtual mediante la línea de comandos, consulte [Sección 13.3.2, "Aislar las máquinas virtuales entre sí mediante la interfaz de línea de comandos"](#).
- Para obtener más información sobre el modo privado de **macvtap**, consulte [Sección 13.5.6, "Fijación directa del dispositivo de red virtual"](#).
- Para conocer las medidas de seguridad adicionales que puede establecer en una VM, consulte [Capítulo 15, Asegurar las máquinas virtuales](#).

13.5. TIPOS DE CONEXIONES DE RED DE MÁQUINAS VIRTUALES

Para modificar las propiedades de red y el comportamiento de sus máquinas virtuales, cambie el tipo de red virtual o interfaz que utilizan las máquinas virtuales. Las siguientes secciones describen los tipos de conexión disponibles para las VM en RHEL 8.

13.5.1. Red virtual con traducción de direcciones de red

Por defecto, los conmutadores de red virtuales funcionan en modo de traducción de direcciones de red (NAT). Utilizan el enmascaramiento de IP en lugar de Source-NAT (SNAT) o Destination-NAT (DNAT). El enmascaramiento de IP permite a las máquinas virtuales conectadas utilizar la dirección IP de la máquina anfitriona para comunicarse con cualquier red externa. Cuando el conmutador de red virtual funciona en modo NAT, los ordenadores externos al host no pueden comunicarse con las máquinas virtuales dentro del host.



RHEL_52_1219



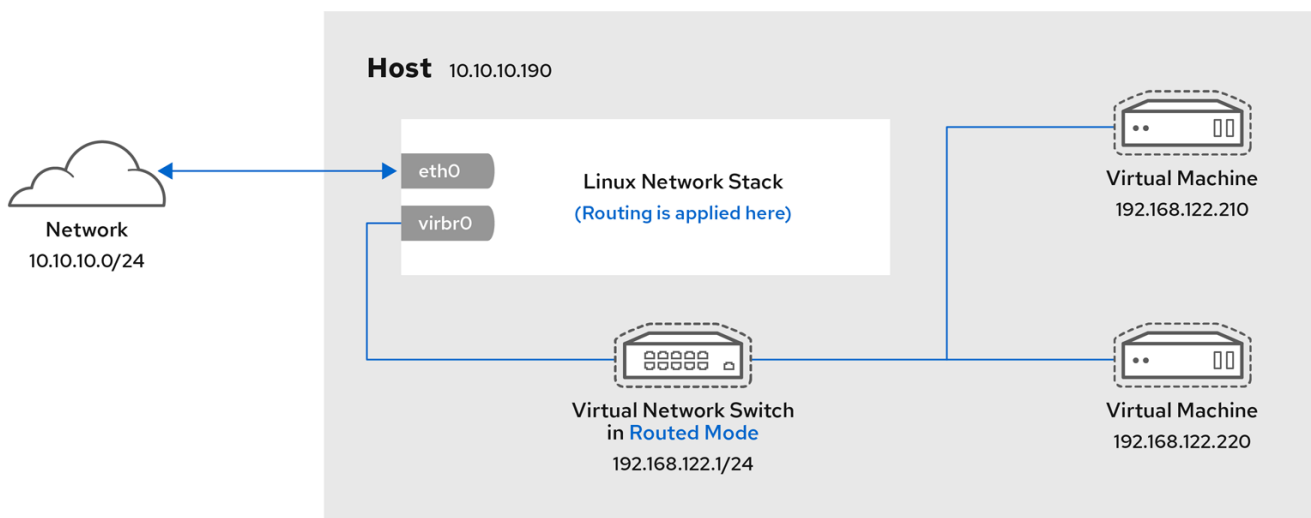
AVISO

Los conmutadores de red virtuales utilizan NAT configurado por reglas iptables. No se recomienda editar estas reglas mientras el conmutador está en funcionamiento, ya que las reglas incorrectas pueden hacer que el conmutador no pueda comunicarse.

13.5.2. Red virtual en modo rotado

Cuando se utiliza el modo *Routed*, el conmutador virtual se conecta a la LAN física conectada a la máquina anfitriona, pasando el tráfico de ida y vuelta sin el uso de NAT. El conmutador virtual puede examinar todo el tráfico y utilizar la información contenida en los paquetes de red para tomar decisiones de enrutamiento. Cuando se utiliza este modo, las máquinas virtuales (VM) están todas en una única subred, separada de la máquina anfitriona. La subred de la VM se enruta a través de un conmutador virtual, que existe en la máquina anfitriona. Esto permite las conexiones entrantes, pero requiere entradas adicionales en la tabla de enrutamiento para los sistemas de la red externa.

El modo enrutado utiliza el enrutamiento basado en la dirección IP:

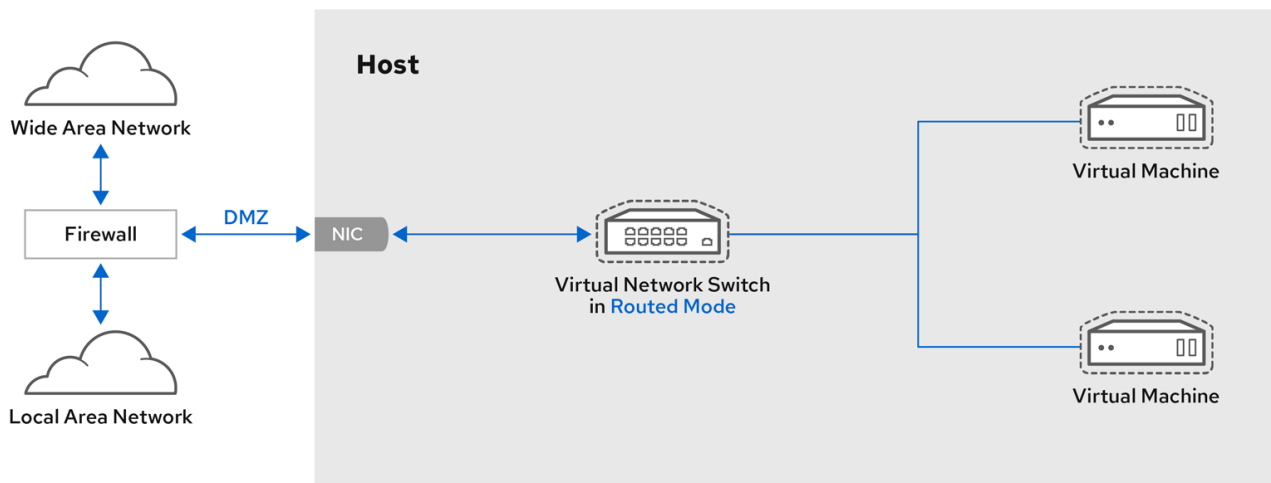


RHEL_52_1219

Las topologías comunes que utilizan el modo enrutado incluyen las DMZ y el alojamiento de servidores virtuales.

DMZ

Puedes crear una red en la que uno o varios nodos se sitúen en una subred controlada por motivos de seguridad. Dicha subred se conoce como zona desmilitarizada (DMZ).

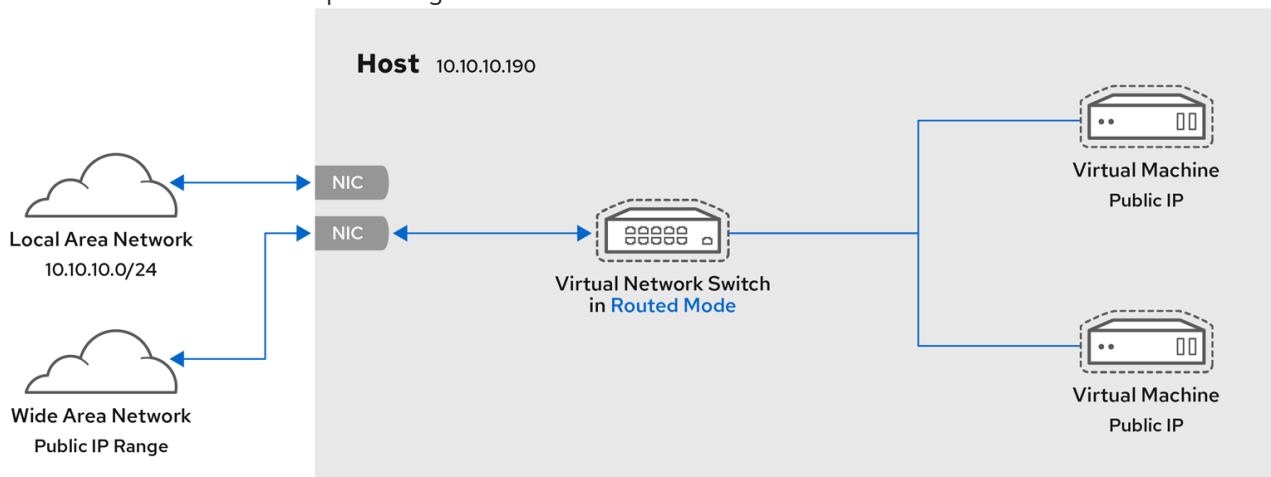


RHEL_52_1219

Las máquinas anfitrionas en una DMZ normalmente proporcionan servicios a máquinas anfitrionas WAN (externas) así como a máquinas anfitrionas LAN (internas). Dado que esto requiere que sean accesibles desde múltiples ubicaciones, y teniendo en cuenta que estas ubicaciones se controlan y operan de diferentes maneras en función de su nivel de seguridad y confianza, el modo enrutado es la mejor configuración para este entorno.

Alojamiento de servidores virtuales

Un proveedor de alojamiento de servidores virtuales puede tener varias máquinas anfitrionas, cada una con dos conexiones de red físicas. Una interfaz se utiliza para la gestión y la contabilidad, y la otra para que las máquinas virtuales se conecten a través de ella. Cada VM tiene su propia dirección IP pública, pero las máquinas anfitrionas utilizan direcciones IP privadas para que sólo los administradores internos puedan gestionar las VM.

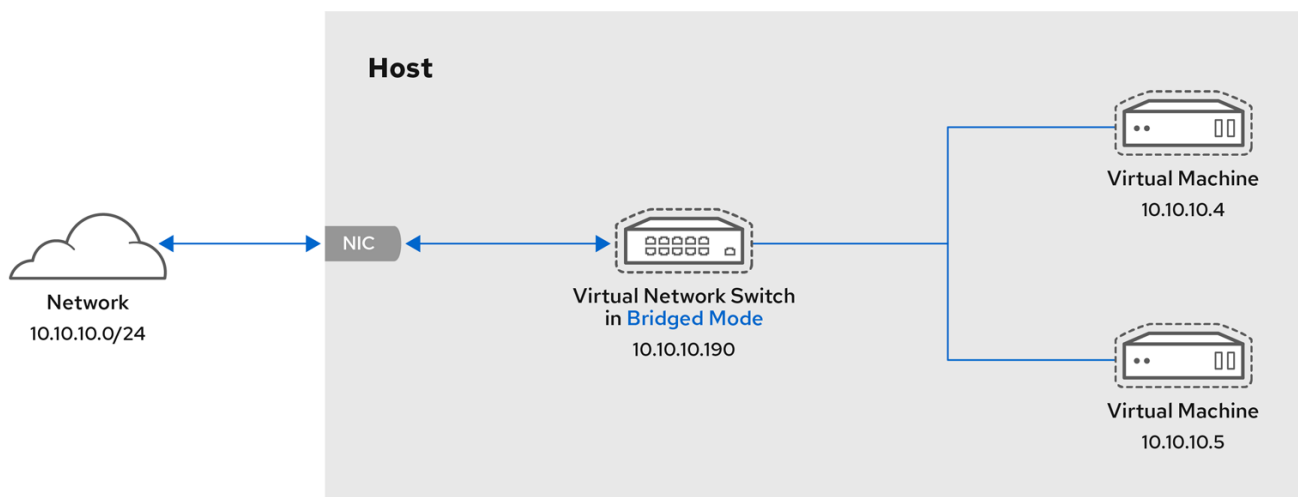


93_RHEL_0520

13.5.3. Red virtual en modo puente

En la mayoría de los modos de red de las VM, éstas crean y se conectan automáticamente al puente virtual **virbr0**. En cambio, en el modo *bridged*, la VM se conecta a un puente Linux existente en el host. Como resultado, la VM es directamente visible en la red física. Esto permite las conexiones entrantes, pero no requiere ninguna entrada extra en la tabla de enrutamiento.

El modo puente utiliza la conmutación de la conexión basada en la dirección MAC:



RHEL_52_1219

En modo puente, la VM aparece dentro de la misma subred que la máquina anfitriona. Todas las demás máquinas físicas de la misma red física pueden detectar la VM y acceder a ella.

Unión de redes en puente

Es posible utilizar múltiples interfaces físicas de puente en el hipervisor uniéndolas con un enlace. El enlace puede entonces añadirse a un puente, tras lo cual las VMs pueden añadirse también al puente. Sin embargo, el controlador de unión tiene varios modos de funcionamiento, y no todos estos modos funcionan con un puente donde las VMs están en uso.

Se pueden utilizar los siguientes [modos de unión](#):

- modo 1
- modo 2
- modo 4

Por el contrario, si se utilizan los modos 0, 3, 5 o 6 es probable que la conexión falle. También hay que tener en cuenta que la monitorización de la interfaz independiente del medio (MII) debe utilizarse para monitorizar los modos de enlace, ya que la monitorización del Protocolo de Resolución de Direcciones (ARP) no funciona correctamente.

Para más información sobre los modos de enlace, consulte la [base de conocimientos de Red Hat](#).

Escenarios comunes

Los casos de uso más comunes para el modo puente incluyen:

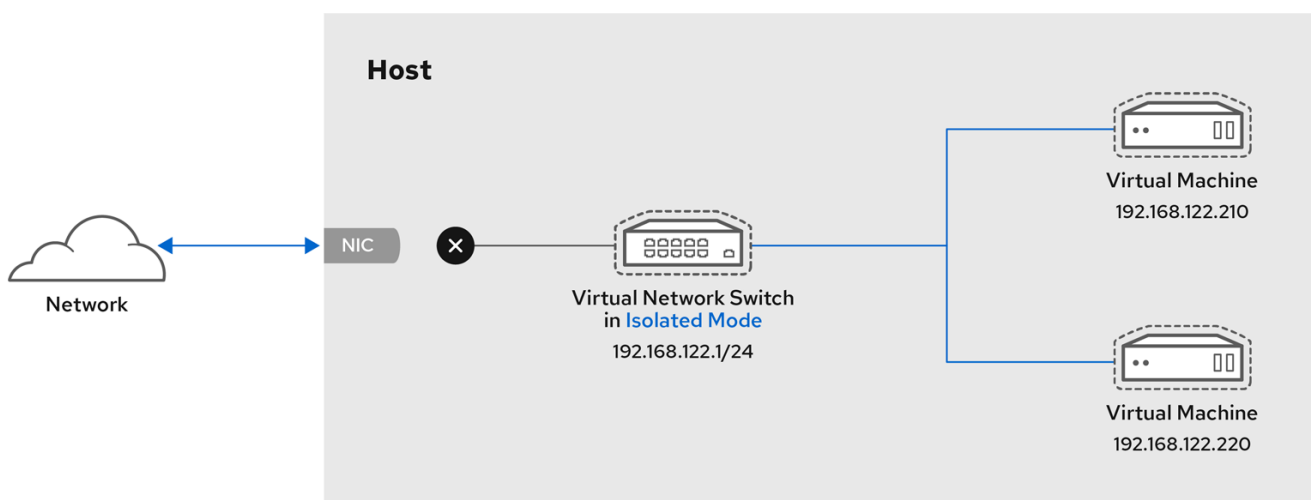
- Desplegar las máquinas virtuales en una red existente junto a las máquinas anfitrionas, haciendo que la diferencia entre máquinas virtuales y físicas sea invisible para el usuario final.
- Despliegue de máquinas virtuales sin realizar ningún cambio en la configuración de la red física existente.
- Desplegar VMs que deben ser fácilmente accesibles a una red física existente. Colocación de máquinas virtuales en una red física en la que deben acceder a servicios DHCP.
- Conectar las máquinas virtuales a una red existente en la que se utilicen LANs virtuales (VLANs).

Recursos adicionales

- Para obtener instrucciones sobre la configuración de sus máquinas virtuales para utilizar el modo puente, consulte [Sección 13.3.1, “Configuración de máquinas virtuales visibles externamente mediante la interfaz de línea de comandos”](#) o [Sección 13.4.1, “Configuración de máquinas virtuales visibles externamente mediante la consola web”](#).
- Para una explicación detallada de los parámetros de **bridge_opts**, utilizados para configurar el modo de red en puente, consulte el [Manual de administración de Red Hat Virtualization](#) .

13.5.4. Red virtual en modo aislado

Cuando se utiliza el modo *isolated*, las máquinas virtuales conectadas al conmutador virtual pueden comunicarse entre sí y con la máquina anfitriona, pero su tráfico no pasará fuera de la máquina anfitriona, y no pueden recibir tráfico desde fuera de la máquina anfitriona. El uso de **dnsmasq** en este modo es necesario para la funcionalidad básica como el DHCP.



RHEL_52_1219

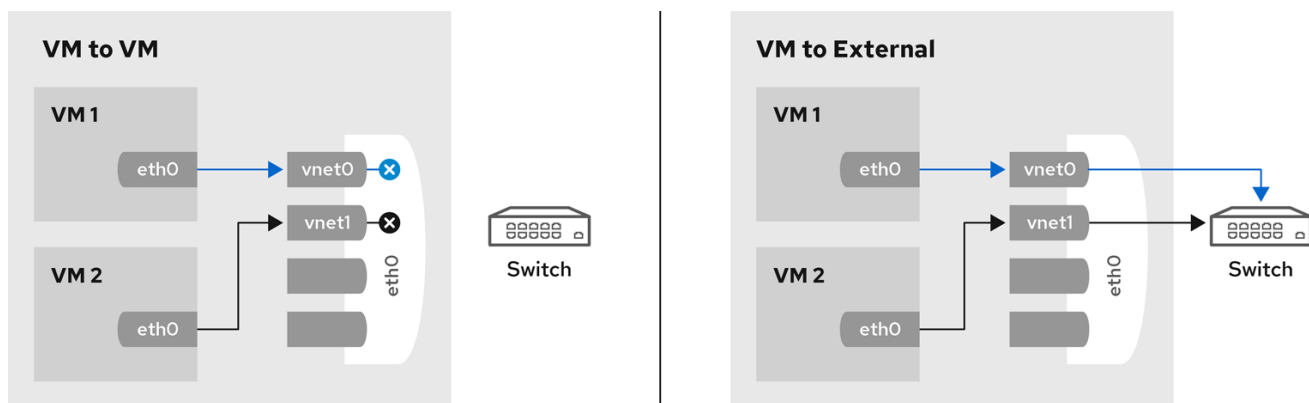
13.5.5. Red virtual en modo abierto

Cuando se utiliza el modo *open* para la red, **libvirt** no genera ninguna regla **iptables** para la red. Como resultado, **libvirt** no sobrescribe las reglas de **iptables** proporcionadas por el host, y el usuario puede por tanto gestionar manualmente las reglas de la VM **iptables**.

13.5.6. Fijación directa del dispositivo de red virtual

Puede utilizar el controlador **macvtap** para conectar la NIC de una máquina virtual directamente a una interfaz física específica de la máquina anfitriona. La conexión **macvtap** tiene varios modos, incluyendo **private mode**.

En este modo, todos los paquetes se envían al conmutador externo y sólo se entregarán a una VM de destino en la misma máquina anfitriona si se envían a través de un enrutador o puerta de enlace externa y éstos los envían de vuelta al anfitrión. El modo privado puede utilizarse para evitar que las máquinas virtuales individuales de un mismo host se comuniquen entre sí.



RHEL_52_1219

Recursos adicionales

- Para obtener instrucciones sobre la configuración de sus máquinas virtuales para utilizar **macvtap** en modo privado, consulte [Sección 13.3.2, “Aislar las máquinas virtuales entre sí mediante la interfaz de línea de comandos”](#) o [Sección 13.4.2, “Aislar las máquinas virtuales entre sí mediante la consola web”](#).

13.5.7. Comparación de los tipos de conexión de las máquinas virtuales

La siguiente tabla proporciona información sobre las ubicaciones a las que pueden conectarse los tipos seleccionados de configuraciones de red de máquinas virtuales (VM) y a las que son visibles.

Tabla 13.1. Tipos de conexión de máquinas virtuales

	Conexión con el host	Conexión con otras máquinas virtuales en el host	Conexión con el exterior	Visible a lugares exteriores
Bridged mode	SI	SI	SI	SI
NAT	SI	SI	SI	<i>no</i>
Routed mode	SI	SI	SI	SI
Isolated mode	SI	SI	<i>no</i>	<i>no</i>
Private mode	<i>no</i>	<i>no</i>	SI	SI
Open mode	<i>Depends on the host's iptables rules</i>			

13.6. RECURSOS ADICIONALES

- Para obtener información adicional sobre la configuración de redes en RHEL 8, consulte el documento [Configuración y gestión de redes](#).

- Las tarjetas de interfaz de red específicas pueden adjuntarse a las máquinas virtuales como dispositivos SR-IOV, lo que aumenta su rendimiento. Para más detalles, consulte [Sección 10.8, "Gestión de dispositivos SR-IOV"](#).

CAPÍTULO 14. COMPARTIR ARCHIVOS ENTRE EL HOST Y SUS MÁQUINAS VIRTUALES

Es posible que con frecuencia necesite compartir datos entre su sistema anfitrión y las máquinas virtuales (VM) que ejecuta. Para hacerlo de forma rápida y eficiente, puede configurar recursos compartidos de archivos NFS o Samba en su sistema.

14.1. COMPARTIR ARCHIVOS ENTRE EL HOST Y LAS MÁQUINAS VIRTUALES LINUX

Para compartir archivos de manera eficiente entre su sistema anfitrión y las máquinas virtuales Linux a las que está conectado, puede exportar un recurso compartido NFS que sus máquinas virtuales puedan montar y acceder.

Requisitos previos

- El paquete **nfs-utils** está instalado en el host.
- Un directorio que quieras compartir con tus máquinas virtuales. Si no quiere compartir ninguno de sus directorios existentes, cree uno nuevo, por ejemplo llamado *shared-files*.

```
# mkdir shared-files
```

- El host es visible y alcanzable a través de una red para la VM. Esto es generalmente el caso si la VM está conectada usando el tipo de redes virtuales *NAT* y *bridge*. Sin embargo, para la conexión *macvtap*, primero debe configurar la función *macvlan* en el host. Para ello:
 1. Cree un archivo de dispositivo de red, por ejemplo, llamado **vm-macvlan.netdev** en el directorio **/etc/systemd/network/** del host.

```
# touch /etc/systemd/network/vm-macvlan.netdev
```

2. Edite el archivo del dispositivo de red para que tenga el siguiente contenido. Puede sustituir **vm-macvlan** por el nombre que haya elegido para su dispositivo de red.

```
[NetDev]
Name=vm-macvlan
Kind=macvlan

[MACVLAN]
Mode=bridge
```

3. Cree un archivo de configuración de red para su dispositivo de red macvlan, por ejemplo **vm-macvlan.network**.

```
# touch /etc/systemd/network/vm-macvlan.network
```

4. Edite el archivo de configuración de red para que tenga el siguiente contenido. Puede sustituir **vm-macvlan** por el nombre que haya elegido para su dispositivo de red.

```
[Match]
Name=_vm-macvlan_
```

```
[Network]
IPForward=yes
Address=192.168.250.33/24
Gateway=192.168.250.1
DNS=192.168.250.1
```

5. Cree un archivo de configuración de red para su interfaz de red física. Por ejemplo, si su interfaz es **enp4s0**:

```
# touch /etc/systemd/network/enp4s0.network
```

Si no está seguro de qué nombre de interfaz utilizar, puede utilizar el comando **ifconfig** en su host para obtener la lista de interfaces de red activas.

6. Edite el archivo de configuración de la red física para que la red física forme parte de la interfaz macvlan, en este caso *vm-macvlan*:

```
[Match]
Name=enp4s0

[Network]
MACVLAN=vm-macvlan
```

7. Reinicie su host.

- **Optional:** Para mejorar la seguridad, asegúrese de que sus máquinas virtuales son compatibles con la versión 4 o posterior de NFS.

Procedimiento

1. En el host, exporte un directorio con los archivos que desea compartir como sistema de archivos de red (NFS).
 - a. Obtenga la dirección IP de cada máquina virtual con la que desee compartir archivos. El siguiente ejemplo obtiene las IP de *testquest1* y *testquest2*.

```
# virsh domifaddr testquest1
Name      MAC address      Protocol  Address
-----
vnet0     52:53:00:84:57:90  ipv4     192.168.124.220/24

# virsh domifaddr testquest2
Name      MAC address      Protocol  Address
-----
vnet1     52:53:00:65:29:21  ipv4     192.168.124.17/24
```

- b. Edite el archivo **/etc/exports** en el host y añada una línea que incluya el directorio que desea compartir, las IP de las máquinas virtuales con las que desea compartir y las opciones de compartición.

```
Shared directory VM1-IP(options) VM2-IP(options) [...]
```

Por ejemplo, lo siguiente comparte el directorio **/usr/local/shared-files** en el host con *testquest1* y *testquest2*, y permite a las máquinas virtuales editar el contenido del directorio:

```
| /usr/local/shared-files/ 192.168.124.220(rw,sync) 192.168.124.17(rw,sync)
```

- c. Exporta el sistema de archivos actualizado.

```
| # exportfs -a
```

- d. Asegúrese de que el proceso NFS se inicia:

```
| # systemctl start nfs-server
```

- e. Obtenga la dirección IP del sistema anfitrión. Se utilizará para montar el directorio compartido en las máquinas virtuales más adelante.

```
| # ip addr
| [...]
| 5: virbr0: [BROADCAST,MULTICAST,UP,LOWER_UP] mtu 1500 qdisc noqueue state
| UP group default qlen 1000
|    link/ether 52:54:00:32:ff:a5 brd ff:ff:ff:ff:ff:ff
|    inet 192.168.124.1/24 brd 192.168.124.255 scope global virbr0
|        valid_lft forever preferred_lft forever
| [...]
```

Tenga en cuenta que la red relevante es la que se utiliza para la conexión al host por parte de las VMs con las que quiere compartir archivos. Normalmente, esta es **virbr0**.

2. En el sistema operativo invitado de una máquina virtual especificada en el archivo **/etc/exports**, monte el sistema de archivos exportado.
 - a. Cree un directorio que desee utilizar como punto de montaje para el sistema de archivos compartido, por ejemplo **/mnt/host-share**:

```
| # mkdir /mnt/host-share
```

- b. Monta el directorio exportado por el host en el punto de montaje. Este ejemplo monta el directorio **/usr/local/shared-files** exportado por el host **192.168.124.1** en **/mnt/host-share** en el invitado:

```
| # mount 192.168.124.1:/usr/local/shared-files /mnt/host-share
```

- c. Para verificar que el montaje se ha realizado con éxito, accede y explora el directorio compartido en el punto de montaje:

```
| # cd /mnt/host-share
| # ls
| shared-file1 shared-file2 shared-file3
```

14.2. COMPARTIR ARCHIVOS ENTRE EL HOST Y LAS MÁQUINAS VIRTUALES DE WINDOWS

Para compartir eficazmente los archivos entre su sistema anfitrión y las máquinas virtuales de Windows a las que está conectado, puede preparar un servidor Samba al que sus máquinas virtuales puedan acceder.

Requisitos previos

- Los paquetes de **samba** están instalados en su host. Si no lo están:

```
# yum install samba
```

- El host es visible y alcanzable a través de una red para la VM. Esto es generalmente el caso si la VM está conectada usando el tipo de redes virtuales *NAT* y *bridge*. Sin embargo, para la conexión *macvtap*, primero debe configurar la función *macvlan* en el host. Para ello:
 1. Cree un archivo de dispositivo de red, por ejemplo, llamado **vm-macvlan.netdev** en el directorio **/etc/systemd/network/** del host.

```
# touch /etc/systemd/network/vm-macvlan.netdev
```

2. Edite el archivo del dispositivo de red para que tenga el siguiente contenido. Puede sustituir **vm-macvlan** por el nombre que haya elegido para su dispositivo de red.

```
[NetDev]
Name=vm-macvlan
Kind=macvlan

[MACVLAN]
Mode=bridge
```

3. Cree un archivo de configuración de red para su dispositivo de red macvlan, por ejemplo **vm-macvlan.network**.

```
# touch /etc/systemd/network/vm-macvlan.network
```

4. Edite el archivo de configuración de red para que tenga el siguiente contenido. Puede sustituir **vm-macvlan** por el nombre que haya elegido para su dispositivo de red.

```
[Match]
Name=_vm-macvlan_

[Network]
IPForward=yes
Address=192.168.250.33/24
Gateway=192.168.250.1
DNS=192.168.250.1
```

5. Cree un archivo de configuración de red para su interfaz de red física. Por ejemplo, si su interfaz es **enp4s0**:

```
# touch /etc/systemd/network/enp4s0.network
```

Si no está seguro de qué interfaz utilizar, puede utilizar el comando **ifconfig** en su host para obtener la lista de interfaces de red activas.

6. Edite el archivo de configuración de la red física para que la red física forme parte de la interfaz macvlan, en este caso *vm-macvlan*:

```
[Match]
```



```
Name=enp4s0
[Network]
MACVLAN=vm-macvlan
```

7. Reinicie su host.

Procedimiento

1. En el host, cree un recurso compartido Samba y hágalo accesible para los sistemas externos.
 - a. Añadir permisos de firewall para Samba.

```
# firewall-cmd --permanent --zone=public --add-service=samba
success
# firewall-cmd --reload
success
```

- b. Edite el archivo `/etc/samba/smb.conf`:
 - i. Añade lo siguiente a la sección **[global]**:

```
map to guest = Bad User
```

- ii. Añade lo siguiente al final del archivo:

```
#=== Share Definitions ===
[VM-share]
path = /samba/VM-share
browsable = yes
guest ok = yes
read only = no
hosts allow = 192.168.122.0/24
```

Tenga en cuenta que la línea **hosts allow** restringe la accesibilidad del recurso compartido sólo a los hosts de la red de la máquina virtual. Si desea que cualquiera pueda acceder al recurso compartido, elimine la línea.

- c. Cree el directorio `/samba/VM-share`.

```
# mkdir -p /samba/VM-share
```

- d. Habilite el servicio Samba.

```
# systemctl enable smb.service
Created symlink /etc/systemd/system/multi-user.target.wants/smb.service →
/usr/lib/systemd/system/smb.service.
```

- e. Reinicie el servicio Samba.

```
# systemctl restart smb.service
```

- f. Permitir que el directorio **VM-share** sea accesible y modificable para las máquinas virtuales.

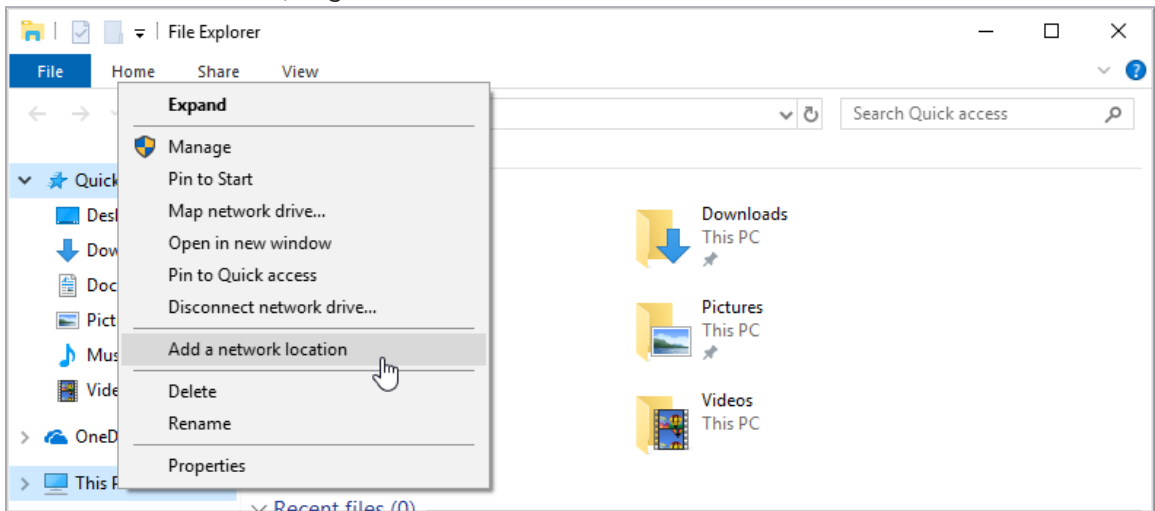
```
# chmod -R 0755 /samba/VM-share/
# chown -R nobody:nobody /samba/VM-share/
```

- g. Añade la etiqueta de compartición SELinux Samba a `/etc/samba/VM-share/`

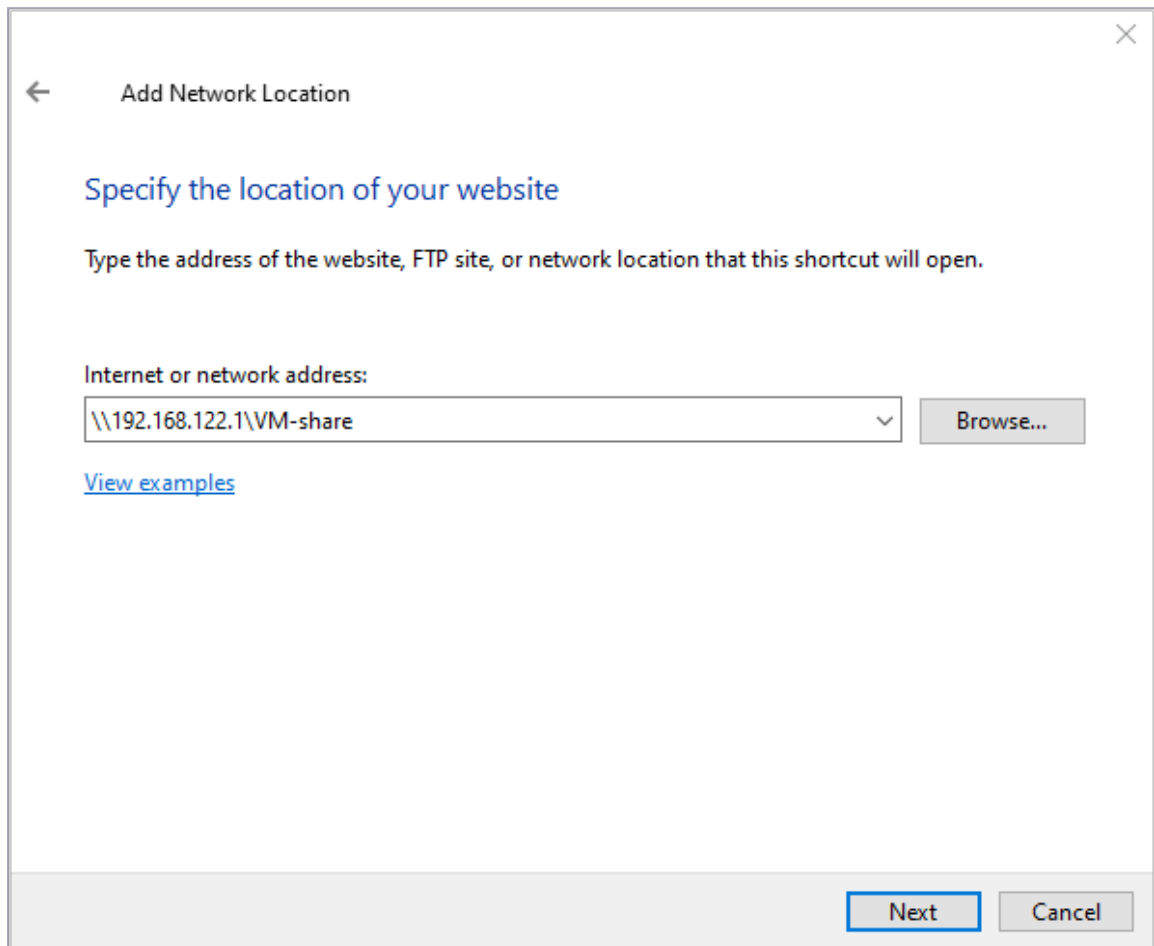
```
# chcon -t samba_share_t /samba/VM-share/
```

2. En el sistema operativo invitado Windows, adjunte el recurso compartido Samba como una ubicación de red.

- a. Abra el Explorador de Archivos y haga clic con el botón derecho en "Este PC".
- b. En el menú contextual, haga clic en **Add a network location**.



- c. En el asistente *Add Network Location* que se abre, seleccione "Elegir una ubicación de red personalizada" y haga clic en **Siguiente**.
- d. En el campo "Dirección de Internet o de red", escriba `host-IP/VM-share`, donde `host-IP` es la dirección IP del host. Normalmente, la IP del host es la puerta de enlace por defecto de la VM. A continuación, haga clic en **Siguiente**.



- e. Cuando el asistente le pregunte si desea cambiar el nombre del directorio compartido, mantenga el nombre por defecto. Esto asegura la consistencia de la configuración de la compartición de archivos en la VM y en el huésped. Haga clic en **Siguiente**.
- f. Si el acceso a la ubicación de red fue exitoso, ahora puede hacer clic en **Finalizar** y abrir el directorio compartido.

CAPÍTULO 15. ASEGURAR LAS MÁQUINAS VIRTUALES

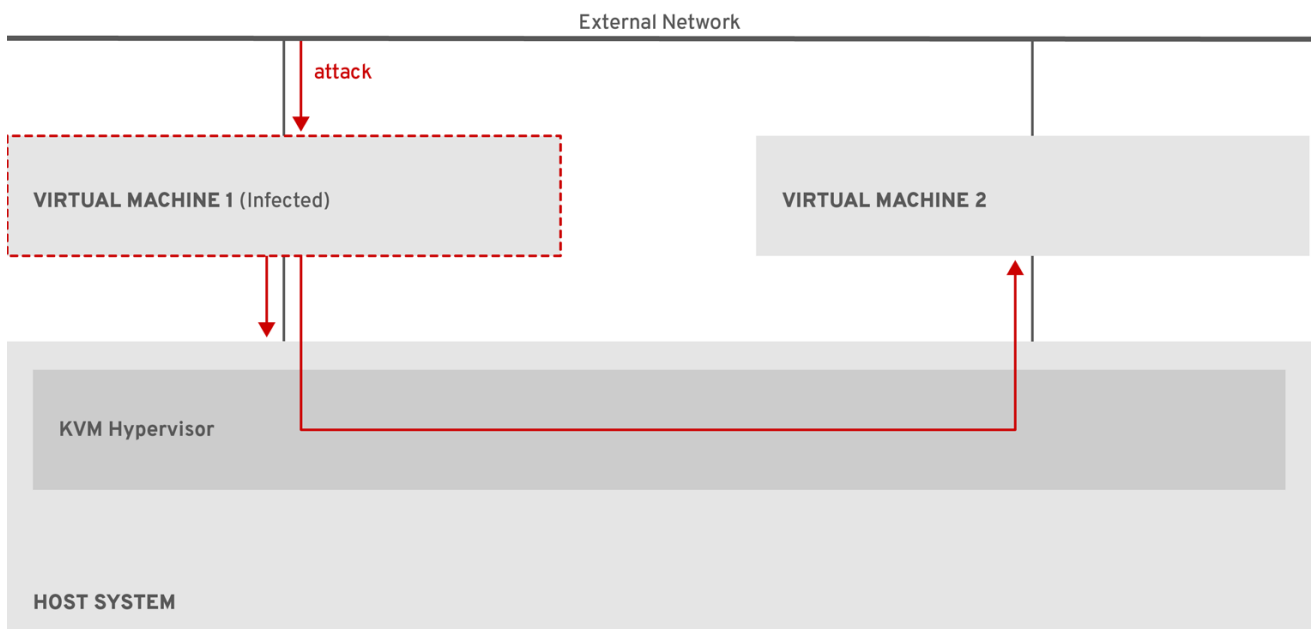
Como administrador de un sistema RHEL 8 con máquinas virtuales (VM), asegurarse de que sus VM son lo más seguras posible reduce significativamente el riesgo de que sus sistemas operativos invitados y anfitriones sean infectados por software malicioso.

Este documento describe la [mecánica para asegurar las máquinas virtuales](#) en un host RHEL 8 y proporciona [una lista de métodos](#) para aumentar la seguridad de sus máquinas virtuales.

15.1. CÓMO FUNCIONA LA SEGURIDAD EN LAS MÁQUINAS VIRTUALES

Cuando se utilizan máquinas virtuales (VM), se pueden alojar varios sistemas operativos dentro de una única máquina anfitriona. Estos sistemas están conectados con el host a través del hipervisor, y normalmente también a través de una red virtual. Como consecuencia, cada VM puede ser utilizada como un vector para atacar al host con software malicioso, y el host puede ser utilizado como un vector para atacar a cualquiera de las VMs.

Figura 15.1. Un potencial vector de ataque de malware en un host de virtualización

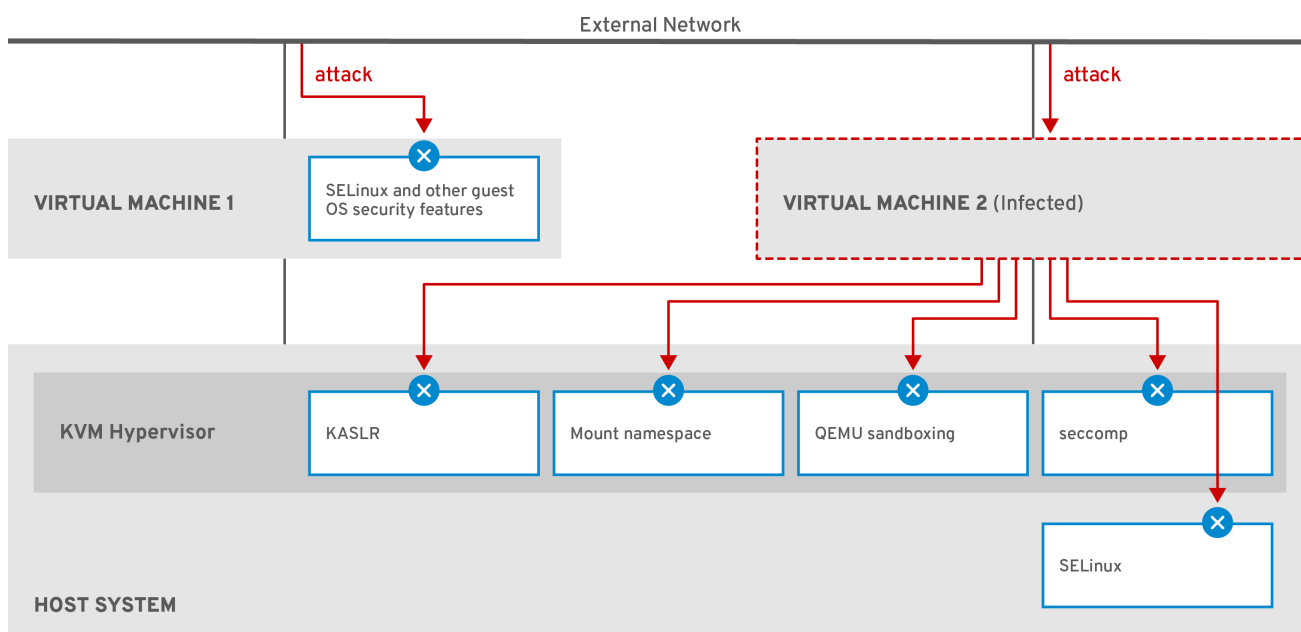


RHEL_7_0319

Dado que el hipervisor utiliza el kernel del host para gestionar las máquinas virtuales, los servicios que se ejecutan en el sistema operativo de la máquina virtual se utilizan con frecuencia para inyectar código malicioso en el sistema del host. Sin embargo, puedes proteger tu sistema contra estas amenazas de seguridad utilizando [una serie de funciones de seguridad](#) en tu sistema anfitrión y en el sistema invitado.

Estas características, como SELinux o el sandboxing de QEMU, proporcionan diversas medidas que dificultan el ataque de código malicioso al hipervisor y la transferencia entre el host y las máquinas virtuales.

Figura 15.2. Prevención de ataques de malware en un host de virtualización



RHEL_7_0319

Muchas de las características que RHEL 8 proporciona para la seguridad de las máquinas virtuales están siempre activas y no tienen que ser activadas o configuradas. Para más detalles, consulte [Sección 15.4, “Funciones automáticas para la seguridad de las máquinas virtuales”](#).

Además, puede adherirse a una variedad de mejores prácticas para minimizar la vulnerabilidad de sus máquinas virtuales y su hipervisor. Para obtener más información, consulte [Sección 15.2, “Mejores prácticas para asegurar las máquinas virtuales”](#).

15.2. MEJORES PRÁCTICAS PARA ASEGURAR LAS MÁQUINAS VIRTUALES

Siguiendo las siguientes instrucciones se reduce significativamente el riesgo de que sus máquinas virtuales se infecten con código malicioso y se utilicen como vectores de ataque para infectar su sistema anfitrión.

On the guest side:

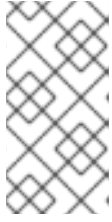
- Asegure la máquina virtual como si fuera una máquina física. Los métodos específicos disponibles para mejorar la seguridad dependen del SO invitado. Si su máquina virtual está ejecutando RHEL 8, consulte [Configuración y gestión de la seguridad en RHEL 8](#) para obtener instrucciones detalladas sobre cómo mejorar la seguridad de su sistema invitado.

On the host side:

- Cuando gestione las VMs de forma remota, utilice utilidades criptográficas como **SSH** y protocolos de red como **SSL** para conectarse a las VMs.
- Asegúrese de que SELinux está en modo de aplicación:

```
# getenforce
Enforcing
```

Si SELinux está deshabilitado o en modo *Permissive*, consulte el documento [Uso de SELinux](#) para obtener instrucciones sobre cómo activar el modo Enforcing.



NOTA

El modo SELinux Enforcing también habilita la función sVirt RHEL 8. Se trata de un conjunto de booleanos especializados en SELinux para la virtualización, que pueden [ajustarse manualmente](#) para la gestión de la seguridad de las máquinas virtuales.

- Utilice VMs con *SecureBoot*:

SecureBoot es una función que garantiza que su máquina virtual ejecute un sistema operativo firmado criptográficamente. Esto evita que las máquinas virtuales cuyo sistema operativo haya sido alterado por un ataque de malware puedan arrancar.

SecureBoot sólo puede aplicarse cuando se instala una VM Linux que utiliza el firmware OVMF. Para obtener instrucciones, consulte [Sección 15.3, "Creación de una máquina virtual SecureBoot"](#).

- No utilice los comandos **qemu-***, como **qemu-img**. QEMU es un componente esencial de la arquitectura de virtualización en RHEL 8, pero es difícil de gestionar manualmente, y las configuraciones inadecuadas de QEMU pueden causar vulnerabilidades de seguridad. Por lo tanto, el uso de los comandos de **qemu-*** no está soportado por Red Hat. En su lugar, se recomienda encarecidamente interactuar con QEMU utilizando las utilidades *libvirt*, como **virsh**, **virt-install**, y **virt-xml**, ya que éstas orquestan QEMU de acuerdo con las mejores prácticas.

Recursos adicionales

- Para obtener información detallada sobre la modificación de los booleanos de virtualización, consulte [Sección 15.5, "Booleanos de virtualización"](#).

15.3. CREACIÓN DE UNA MÁQUINA VIRTUAL SECUREBOOT

A continuación se ofrecen instrucciones para crear una máquina virtual (VM) Linux que utilice la función *SecureBoot*, que garantiza que su VM ejecute un SO firmado criptográficamente. Si el SO invitado de una VM ha sido alterado por un malware, SecureBoot impide que la VM arranque, lo que detiene la posible propagación del malware a su máquina anfitriona.

Requisitos previos

- La VM está utilizando el tipo de máquina Q35.
- El paquete **edk2-OVMF** está instalado:

```
# yum install edk2-ovmf
```

- Una fuente de instalación del sistema operativo (SO) está disponible localmente o en una red. Puede ser uno de los siguientes formatos:
 - Una imagen ISO de un medio de instalación
 - Una imagen de disco de una instalación VM existente

- Opcional: Se puede proporcionar un archivo Kickstart para una configuración más rápida y sencilla de la instalación.

Procedimiento

1. Utilice el comando **virt-install** para crear una VM como se detalla en [Sección 2.2.1, “Creación de máquinas virtuales mediante la interfaz de línea de comandos”](#). Para la opción **--boot**, utilice el valor **uefi,nvram_template=/usr/share/OVMF/OVMF_VARS.secboot.fd**. Esto utiliza los archivos **OVMF_VARS.secboot.fd** y **OVMF_CODE.secboot.fd** como plantillas para la configuración de la RAM no volátil (NVRAM) de la VM, lo que permite la función SecureBoot. Por ejemplo:

```
# virt-install --name rhel8sb --memory 4096 --vcpus 4 --os-variant rhel8.0 --boot
uefi,nvram_template=/usr/share/OVMF/OVMF_VARS.secboot.fd --disk
boot_order=2,size=10 --disk
boot_order=1,device=cdrom,bus=scsi,path=/images/RHEL-8.0-installation.iso
```

2. Siga el procedimiento de instalación del sistema operativo según las instrucciones que aparecen en la pantalla.
3. Una vez instalado el SO huésped, acceda a la línea de comandos de la VM abriendo el terminal en [la consola gráfica del huésped](#) o conectándose al SO huésped [mediante SSH](#).
4. Compruebe que SecureBoot está activado mediante el comando **mokutil --sb-state**:

```
# mokutil --sb-state
SecureBoot enabled
```

15.4. FUNCIONES AUTOMÁTICAS PARA LA SEGURIDAD DE LAS MÁQUINAS VIRTUALES

Además de los medios manuales para mejorar la seguridad de sus máquinas virtuales que se enumeran en [Sección 15.2, “Mejores prácticas para asegurar las máquinas virtuales”](#), el paquete de software **libvirt** proporciona una serie de funciones de seguridad que se activan automáticamente al utilizar la virtualización en RHEL 8. Entre ellas se incluyen:

Sistema y sesiones de usuario

Para acceder a todas las utilidades disponibles para la gestión de máquinas virtuales en RHEL 8, es necesario utilizar el *system session* de libvirt. Para ello, debe tener privilegios de root en el sistema o formar parte del grupo de usuarios *libvirt*.

Los usuarios que no son root y que no están en el grupo libvirt sólo pueden acceder a un *user session* de libvirt, que tiene que respetar los derechos de acceso del usuario local cuando accede a los recursos. Por ejemplo, en la sesión de usuario, no puede detectar o acceder a las VMs creadas en la sesión del sistema o por otros usuarios. Además, las opciones de configuración de redes de VM disponibles son significativamente limitadas.



NOTA

La documentación de RHEL 8 asume que tiene privilegios de sesión del sistema libvirt.

Separación de máquinas virtuales

Las máquinas virtuales individuales se ejecutan como procesos aislados en el host, y dependen de la seguridad impuesta por el kernel del host. Por lo tanto, una máquina virtual no puede leer o acceder a la memoria o al almacenamiento de otras máquinas virtuales en el mismo host.

Sandboxing de QEMU

Una función que impide que el código de QEMU ejecute llamadas al sistema que puedan comprometer la seguridad del host.

Aleatorización del espacio de direcciones del núcleo (KASLR)

Permite aleatorizar las direcciones físicas y virtuales en las que se descomprime la imagen del kernel. De este modo, KASLR evita que se produzcan explotaciones de seguridad de los huéspedes basadas en la ubicación de los objetos del kernel.

15.5. BOOLEANOS DE VIRTUALIZACIÓN

Para una configuración detallada de la seguridad de las máquinas virtuales en un sistema RHEL 8, puede configurar booleanos de SELinux en el host para garantizar que el hipervisor actúe de una manera específica.

Para listar todos los booleanos relacionados con la virtualización y sus estados, utilice el comando **getsebool -a | grep virt**:

```
$ getsebool -a | grep virt
[...]
virt_sandbox_use_netlink --> off
virt_sandbox_use_sys_admin --> off
virt_transition_userdomain --> off
virt_use_commm --> off
virt_use_execmem --> off
virt_use_fusefs --> off
[...]
```

Para activar un booleano específico, utilice el comando **setsebool -P *boolean_name* on** como root. Para desactivar un booleano, utilice **setsebool -P *boolean_name* off**.

La siguiente tabla enumera los booleanos relacionados con la virtualización disponibles en RHEL 8 y lo que hacen cuando están activados:

Tabla 15.1. Booleanos de virtualización de SELinux

Booleano de SELinux	Descripción
staff_use_svirt	Permite a los usuarios que no son root crear y pasar las máquinas virtuales a sVirt.
unprivuser_use_svirt	Permite a los usuarios sin privilegios crear y pasar las máquinas virtuales a sVirt.
virt_sandbox_use_audit	Permite a los contenedores de la caja de arena enviar mensajes de auditoría.
virt_sandbox_use_netlink	Permite que los contenedores de la caja de arena utilicen las llamadas del sistema netlink.

Booleano de SELinux	Descripción
virt_sandbox_use_sys_admin	Permite que los contenedores sandbox utilicen las llamadas al sistema sys_admin, como el montaje.
virt_transition_userdomain	Permite que los procesos virtuales se ejecuten como dominios de usuario.
virt_use_comm	Permite a virt utilizar puertos de comunicación serie/paralelo.
virt_use_execmem	Permite que los huéspedes virtuales confinados utilicen memoria ejecutable y pila ejecutable.
virt_use_fusefs	Permite a virt leer los archivos montados en FUSE.
virt_use_nfs	Permite a virt gestionar los archivos montados en NFS.
virt_use_rawip	Permite a virt interactuar con sockets rawip.
virt_use_samba	Permite a virt gestionar archivos montados en CIFS.
virt_use_sanlock	Permite que los invitados virtuales confinados interactúen con el sanlock.
virt_use_usb	Permite a virt utilizar dispositivos USB.
virt_use_xserver	Permite a la máquina virtual interactuar con el sistema X Window.

15.6. CONFIGURACIÓN DE IBM SECURE EXECUTION EN IBM Z

Si utiliza hardware IBM Z para ejecutar un host RHEL 8, puede mejorar la seguridad de sus máquinas virtuales (VM) configurando IBM Secure Execution para las VM.

IBM Secure Execution, también conocida como Virtualización Protegida, impide que el sistema anfitrión acceda al estado y al contenido de la memoria de una máquina virtual. Como resultado, incluso si el host está comprometido, no puede ser utilizado como un vector para atacar el sistema operativo invitado. Además, la Ejecución Segura puede utilizarse para evitar que los hosts no confiables obtengan información sensible de la VM.

El siguiente procedimiento describe cómo convertir una VM existente en un host IBM Z en una VM segura.

Requisitos previos

- El hardware del sistema es uno de los siguientes:

- IBM z15 o posterior
- IBM LinuxONE III o posterior
- La función de ejecución segura está activada en su sistema. Para comprobarlo, utilice:

```
# grep facilities /proc/cpuinfo | grep 158
```

Si este comando muestra alguna salida, su CPU es compatible con la ejecución segura.

- El kernel incluye soporte para la Ejecución Segura. Para confirmarlo, utilice:

```
# ls /sys/firmware | grep uv
```

Si el comando genera alguna salida, su kernel soporta la Ejecución Segura.

- El modelo de CPU del host contiene la función **unpack**. Para confirmarlo, utilice:

```
# virsh domcapabilities | grep unpack
<feature policy='require' name='unpack'/>
```

Si el comando genera la salida anterior, su modelo de CPU es compatible con la ejecución segura.

- El modo de CPU de la VM está configurado en **host-model**. Para confirmarlo, utilice lo siguiente y sustituya **vm-name** por el nombre de su VM.

```
# virsh dumpxml vm-name | grep "<cpu mode='host-model'/>"
```

Si el comando genera alguna salida, el modo de CPU de la VM está configurado correctamente.

- Ha obtenido y verificado el documento de clave de host de IBM Z. Para obtener instrucciones para hacerlo, consulte [Verificación del documento de clave de host](#) en la documentación de IBM.

Procedimiento

1. Añade el parámetro del kernel **prot_virt=1** a la [configuración de arranque](#) del host.

```
# # grubby --update-kernel=ALL --args="prot_virt=1"
```

2. Cree un archivo de parámetros para la máquina virtual que desea proteger. Por ejemplo:

```
# touch ~/secure-parameters
```

3. En el directorio **/boot/loader/entries** del host, identifique la entrada del gestor de arranque con la última versión:

```
# ls /boot/loader/entries -l
[...]
-rw-r--r--. 1 root root 281 Oct 9 15:51 3ab27a195c2849429927b00679db15c1-4.18.0-240.el8.s390x.conf
```

4. Recupera la línea de opciones del kernel de la entrada del gestor de arranque:

```
# cat /boot/loader/entries/3ab27a195c2849429927b00679db15c1-4.18.0-240.el8.s390x.conf | grep options
options root=/dev/mapper/rhel-root crashkernel=auto rd.lvm.lv=rhel/root rd.lvm.lv=rhel/swap
```

- Añade el contenido de la línea de opciones y **swiotlb=262144** al archivo de parámetros creado.

```
# echo "root=/dev/mapper/rhel-root crashkernel=auto rd.lvm.lv=rhel/root rd.lvm.lv=rhel/swap swiotlb=262144" > ~/secure-parameters
```

- Generar una imagen de IBM Secure Execution para la VM seleccionada. Por ejemplo, lo siguiente crea una imagen segura **/boot/secure-image** basada en la imagen **/boot/vmlinuz-4.18.0-240.el8.s390x**, utilizando el archivo **secure-parameters**, el archivo de disco RAM inicial **/boot/initramfs-4.18.0-240.el8.s390x.img** y el documento de clave de host **HKD-8651-000201C048.crt**.

```
# genprotimg -i /boot/vmlinuz-4.18.0-240.el8.s390x -r /boot/initramfs-4.18.0-240.el8.s390x.img -p ~/secure-parameters -k HKD-8651-00020089A8.crt -o /boot/secure-image
```

Utilizando la utilidad **genprotimg** se crea la imagen segura, que contiene los parámetros del kernel, el disco RAM inicial y la imagen de arranque.

- En el sistema operativo invitado de la VM, actualice el menú de arranque de la VM para arrancar desde la imagen segura. Además, elimine las líneas que comienzan con **initrd** y **options**, ya que no son necesarias.

Por ejemplo, en una VM RHEL 8.3, el menú de arranque se puede editar en el directorio **/boot/loader/entries/**:

```
# cat /boot/loader/entries/3ab27a195c2849429927b00679db15c1-4.18.0-240.el8.s390x.conf
title Red Hat Enterprise Linux 8.3
version 4.18.0-240.el8.s390x
linux /boot/secure-image
[...]
```

- Habilita los dispositivos virtio para que utilicen buffers compartidos. Para ello, utilice **virsh edit** para modificar la configuración XML de la VM, y añada **iommu='on'** a la línea **<driver>** de todos los dispositivos que lo tengan. Por ejemplo:

```
<interface type='network'>
  <source network='default'/>
  <model type='virtio'/>
  <driver name='vhost' iommu='on'/>
</interface>
```

Si la configuración de un dispositivo no contiene ninguna línea **<driver>**, añada en su lugar **<driver iommu='on'>**.

- Desactivar el memory ballooning en la VM, ya que esta característica no es compatible con la Ejecución Segura. Para ello, añada la siguiente línea a la configuración XML de la máquina virtual.

```
<memballoon model='none'/>
```

10. Crear la imagen de disco de arranque

```
# zipl -V
```

11. Eliminar de forma segura los archivos originales no protegidos. Por ejemplo:

```
# shred /boot/vmlinuz-4.18.0-240.el8.s390x
# shred /boot/initramfs-4.18.0-240.el8.s390x.img
# shred secure-parameters
```

La imagen de arranque original, la imagen de RAM inicial y el archivo de parámetros del kernel están desprotegidos y, si no se eliminan, las máquinas virtuales con la ejecución segura activada pueden seguir siendo vulnerables a los intentos de piratería o a la extracción de datos sensibles.

Verificación

- En el host, utilice la utilidad **virsh dumpxml** para confirmar la configuración XML de la máquina virtual protegida. La configuración debe incluir los elementos **<driver iommu='on'/>** y **<memballoon model='none'/>**.

```
# virsh dumpxml vm-name
[...]
<cpu mode='host-model'/>
<devices>
  <disk type='file' device='disk'>
    <driver name='qemu' type='qcow2' cache='none' io='native' iommu='on'>
    <source file='/var/lib/libvirt/images/secure-guest.qcow2'/>
    <target dev='vda' bus='virtio'/>
  </disk>
  <interface type='network'>
    <driver iommu='on'/>
    <source network='default'/>
    <model type='virtio'/>
  </interface>
  <console type='pty'/>
  <memballoon model='none'/>
</devices>
</domain>
```

Recursos adicionales

- Para obtener instrucciones adicionales sobre cómo modificar la configuración de arranque del host, consulte [Configuración de los parámetros de la línea de comandos del kernel](#) .
- Para más información sobre la utilidad **genprotimg**, consulte [la documentación de IBM](#) .

15.7. ADJUNTAR COPROCESADORES CRIPTOGRÁFICOS A MÁQUINAS VIRTUALES EN IBM Z

Para utilizar el cifrado de hardware en su máquina virtual (VM) en un host IBM Z, cree dispositivos mediados de un dispositivo de coprocesador criptográfico y asígnelos a las VM previstas. Para obtener instrucciones detalladas, consulte a continuación.

Requisitos previos

- Su host se ejecuta en hardware IBM Z.
- El coprocesador criptográfico es compatible con la asignación de dispositivos. Para confirmarlo, asegúrese de que la dirección **type** de su coprocesador figura como **CEX4** o posterior.

```
# lszcrypt -V
```

```
CARD.DOMAIN TYPE MODE STATUS REQUESTS PENDING HWTYPE QDEPTH
FUNCTIONS DRIVER
-----
05 CEX5C CCA-Coproc online 1 0 11 08 S--D--N-- cex4card
05.0004 CEX5C CCA-Coproc online 1 0 11 08 S--D--N-- cex4queue
05.00ab CEX5C CCA-Coproc online 1 0 11 08 S--D--N-- cex4queue
```

- El paquete *mdevctl* está instalado.
- El módulo del kernel **vfio_ap** está cargado. Para verificarlo, utilice:

```
# lsmod | grep vfio_ap
vfio_ap      24576 0
[...]
```

Para cargar el módulo, utilice:

```
# modprobe vfio_ap
```

Procedimiento

1. En el host, reasigne su dispositivo criptográfico a los controladores **vfio-ap**. El siguiente ejemplo asigna dos dispositivos criptográficos con ID de máscara de bits (**0x05**, **0x0004**) y (**0x05**, **0x00ab**) a **vfio-ap**.

```
# echo -0x05 > /sys/bus/ap/apmask
# echo -0x0004, -0x00ab > /sys/bus/ap/aqmask
```

Para obtener información sobre la identificación de los valores de ID de la máscara de bits, consulte [Preparación de los dispositivos de paso para los recursos del adaptador criptográfico](#) en el documento KVM Virtual Server Management de IBM.

2. Compruebe que los dispositivos criptográficos se han reasignado correctamente.

```
# lszcrypt -V
```

```
CARD.DOMAIN TYPE MODE STATUS REQUESTS PENDING HWTYPE QDEPTH
FUNCTIONS DRIVER
-----
05 CEX5C CCA-Coproc - 1 0 11 08 S--D--N-- cex4card
05.0004 CEX5C CCA-Coproc - 1 0 11 08 S--D--N-- vfio_ap
05.00ab CEX5C CCA-Coproc - 1 0 11 08 S--D--N-- vfio_ap
```

Si los valores de DRIVER de las colas del dominio cambiaron a **vfio_ap**, la reasignación tuvo éxito.

3. Generar un UUID de dispositivo.

```
# uuidgen
669d9b23-fe1b-4ecb-be08-a2fabca99b71
```

En los siguientes pasos de este procedimiento, sustituya **669d9b23-fe1b-4ecb-be08-a2fabca99b71** por su UUID generado.

4. Utilizando el UUID, cree un nuevo dispositivo **vfio_ap**.

El siguiente ejemplo muestra la creación de un dispositivo mediado persistente y la asignación de colas al mismo. Por ejemplo, los siguientes comandos asignan el adaptador de dominio **0x05** y las colas de dominio **0x0004** y **0x00ab** al dispositivo **669d9b23-fe1b-4ecb-be08-a2fabca99b71**.

```
# mdevctl define --uuid 669d9b23-fe1b-4ecb-be08-a2fabca99b71 --parent matrix --type
vfio_ap-passthrough
# mdevctl modify --uuid 669d9b23-fe1b-4ecb-be08-a2fabca99b71 --
addattr=assign_adapter --value=0x05
# mdevctl modify --uuid 669d9b23-fe1b-4ecb-be08-a2fabca99b71 --
addattr=assign_domain --value=0x0004
# mdevctl modify --uuid 669d9b23-fe1b-4ecb-be08-a2fabca99b71 --
addattr=assign_domain --value=0x00ab
```

5. Inicie el dispositivo mediado.

```
# mdevctl start --uuid 669d9b23-fe1b-4ecb-be08-a2fabca99b71
```

6. Compruebe que la configuración se ha aplicado correctamente

```
# cat /sys/devices/vfio_ap/matrix/mdev_supported_types/vfio_ap-
passthrough/devices/669d9b23-fe1b-4ecb-be08-a2fabca99b71
05.0004
05.00ab
```

Si la salida contiene los valores numéricos de las colas que ha asignado previamente a **vfio-ap**, el proceso se ha realizado con éxito.

7. Utilice el comando **virsh edit** para abrir la configuración XML de la VM en la que desea utilizar los dispositivos criptográficos.

```
# virsh edit vm-name
```

8. Añade las siguientes líneas a la sección **<devices>** en la configuración XML, y guárdala.

```
<hostdev mode='subsystem' type='mdev' managed='no' model='vfio-ap'>
  <source>
    <address uuid='669d9b23-fe1b-4ecb-be08-a2fabca99b71' />
  </source>
</hostdev>
```

Tenga en cuenta que cada UUID sólo puede asignarse a una máquina virtual a la vez.

Verificación

1. Inicie la VM a la que asignó el dispositivo mediado.
2. Después de que el sistema operativo (SO) huésped arranque, asegúrese de que detecta los dispositivos criptográficos asignados.

```
# lszycrypt -V
```

```
CARD.DOMAIN TYPE MODE STATUS REQUESTS PENDING HWTYPE QDEPTH
FUNCTIONS DRIVER
-----
05 CEX5C CCA-Coproc online 1 0 11 08 S--D--N-- cex4card
05.0004 CEX5C CCA-Coproc online 1 0 11 08 S--D--N-- cex4queue
05.00ab CEX5C CCA-Coproc online 1 0 11 08 S--D--N-- cex4queue
```

La salida de este comando en el SO huésped será idéntica a la de una partición lógica del host con los mismos dispositivos de coprocesador criptográfico disponibles.

15.8. HABILITACIÓN DE LA SEGURIDAD DE HARDWARE ESTÁNDAR EN LAS MÁQUINAS VIRTUALES DE WINDOWS

Para proteger las máquinas virtuales (VM) de Windows, puede activar la seguridad de nivel básico utilizando las capacidades de hardware estándar del dispositivo de Windows.

Requisitos previos

- Asegúrese de haber instalado los últimos controladores VirtIO con certificación WHQL.
- Asegúrese de que el firmware de la máquina virtual soporta el arranque UEFI.
- Instale el paquete **edk2-OVMF** en su máquina anfitriona.

```
# yum install edk2-ovmf
```

- Instale los paquetes de **vTPM** en su máquina anfitriona.

```
# yum install swtpm libtpms
```

- Asegúrese de que la VM está utilizando la arquitectura de máquina Q35.
- Asegúrate de que tienes los medios de instalación de Windows.

Procedimiento

1. Habilite el TPM 2.0 añadiendo los siguientes parámetros a la sección **<devices>** en la configuración XML de la máquina virtual.

```
<devices>
[...
  <tpm model='tpm-crb'
    <backend type='emulator' version='2.0'>
  </tpm>
[...
</devices>
```

2. Instale Windows en modo UEFI. Para obtener más información sobre cómo hacerlo, consulte [Creación de una máquina virtual SecureBoot](#).
3. Instale los controladores VirtIO en la VM de Windows. Para obtener más información sobre cómo hacerlo, consulte [Instalación de los controladores de VirtIO en un invitado de Windows](#).
4. En UEFI, active el arranque seguro. Para más información sobre cómo hacerlo, consulta [Secure Boot](#).

Verificación

- Asegúrese de que la página **Device Security** de su máquina Windows muestra el siguiente mensaje:

Your device meets the requirements for standard hardware security.

CAPÍTULO 16. OPTIMIZACIÓN DEL RENDIMIENTO DE LAS MÁQUINAS VIRTUALES

Las máquinas virtuales (VM) siempre experimentan cierto grado de deterioro del rendimiento en comparación con el host. Las siguientes secciones explican las razones de este deterioro y proporcionan instrucciones sobre cómo minimizar el impacto de la virtualización en el rendimiento en RHEL 8, para que los recursos de su infraestructura de hardware puedan ser utilizados de la manera más eficiente posible.

16.1. QUÉ INFLUYE EN EL RENDIMIENTO DE LAS MÁQUINAS VIRTUALES

Las máquinas virtuales se ejecutan como procesos de espacio de usuario en el host. Por lo tanto, el hipervisor tiene que convertir los recursos del sistema del host para que las máquinas virtuales puedan utilizarlos. Como consecuencia, una parte de los recursos es consumida por la conversión, por lo que la VM no puede alcanzar la misma eficiencia de rendimiento que el host.

El impacto de la virtualización en el rendimiento del sistema

Entre las razones más específicas de la pérdida de rendimiento de las máquinas virtuales se encuentran:

- Las CPUs virtuales (vCPUs) se implementan como hilos en el host, manejados por el planificador de Linux.
- Las máquinas virtuales no heredan automáticamente las características de optimización, como NUMA o las páginas enormes, del núcleo anfitrión.
- Los ajustes de E/S del disco y de la red del host pueden tener un impacto significativo en el rendimiento de la máquina virtual.
- El tráfico de red suele viajar a una VM a través de un puente basado en software.
- Dependiendo de los dispositivos anfitriones y sus modelos, puede haber una sobrecarga significativa debido a la emulación de un hardware particular.

La gravedad del impacto de la virtualización en el rendimiento de las máquinas virtuales se ve influida por una serie de factores, entre los que se incluyen:

- El número de máquinas virtuales que se ejecutan simultáneamente.
- La cantidad de dispositivos virtuales utilizados por cada VM.
- Los tipos de dispositivos utilizados por las máquinas virtuales.

Reducción de la pérdida de rendimiento de las máquinas virtuales

RHEL 8 ofrece una serie de funciones que puede utilizar para reducir los efectos negativos de la virtualización sobre el rendimiento. En particular:

- [El servicio tuned](#) puede optimizar automáticamente la distribución de recursos y el rendimiento de sus máquinas virtuales.
- [El ajuste de E/S en bloque](#) puede mejorar el rendimiento de los dispositivos de bloque de la máquina virtual, como los discos.
- [El ajuste de NUMA](#) puede aumentar el rendimiento de las vCPU.

- [Lared virtual](#) puede optimizarse de varias maneras.



IMPORTANTE

El ajuste del rendimiento de la VM puede tener efectos adversos en otras funciones de virtualización. Por ejemplo, puede dificultar la migración de la VM modificada.

16.2. OPTIMIZACIÓN DEL RENDIMIENTO DE LAS MÁQUINAS VIRTUALES MEDIANTE EL AJUSTE

La utilidad **tuned** es un mecanismo de entrega de perfiles de ajuste que adapta RHEL a determinadas características de la carga de trabajo, como los requisitos de las tareas intensivas de la CPU o la capacidad de respuesta del rendimiento de la red de almacenamiento. Proporciona una serie de perfiles de ajuste preconfigurados para mejorar el rendimiento y reducir el consumo de energía en una serie de casos de uso específicos. Puede editar estos perfiles o crear otros nuevos para crear soluciones de rendimiento adaptadas a su entorno, incluidos los entornos virtualizados.

Red Hat recomienda utilizar los siguientes perfiles cuando se utiliza la virtualización en RHEL 8:

- Para las máquinas virtuales RHEL 8, utilice el perfil **virtual-guest**. Se basa en el perfil de aplicación general **throughput-performance** pero también disminuye el intercambio de la memoria virtual.
- Para los hosts de virtualización RHEL 8, utilice el perfil **virtual-host**. Esto permite una escritura más agresiva de las páginas de memoria sucias, lo que beneficia el rendimiento del host.

Requisitos previos

- El servicio **tuned** está [instalado y habilitado](#).

Procedimiento

Para activar un perfil específico de **tuned**:

1. Enumera los perfiles disponibles en **tuned**.

```
# tuned-adm list
```

```
Available profiles:
```

```
- balanced          - General non-specialized tuned profile
- desktop           - Optimize for the desktop use-case
[...]
- virtual-guest     - Optimize for running inside a virtual guest
- virtual-host      - Optimize for running KVM guests
Current active profile: balanced
```

2. **Optional:** Crear un nuevo perfil **tuned** o editar un perfil existente **tuned**.
Para más información, consulte la sección de [personalización de los perfiles ajustados](#).
3. Activar un perfil **tuned**.

```
# tuned-adm profile selected-profile
```

- Para optimizar un host de virtualización, utilice el perfil *virtual-host*.

tuned-adm profile virtual-host

- En un sistema operativo invitado RHEL, utilice el perfil *virtual-guest*.

tuned-adm profile virtual-guest

Recursos adicionales

- Para más información sobre los perfiles de **tuned** y **tuned**, consulte [Supervisión y gestión del estado y el rendimiento del sistema](#).

16.3. CONFIGURACIÓN DE LA MEMORIA DE LA MÁQUINA VIRTUAL

Para mejorar el rendimiento de una máquina virtual (VM), puede asignar más memoria RAM del host a la VM. Del mismo modo, puede disminuir la cantidad de memoria asignada a una VM para que la memoria del host pueda asignarse a otras VM o tareas.

Para realizar estas acciones, puede utilizar [la consola web](#) o [la interfaz de línea de comandos](#).

16.3.1. Añadir y eliminar la memoria de la máquina virtual mediante la consola web

Para mejorar el rendimiento de una máquina virtual (VM) o para liberar los recursos del host que está utilizando, puede utilizar la consola web para ajustar la cantidad de memoria asignada a la VM.

Requisitos previos

- El sistema operativo invitado está ejecutando los controladores del globo de memoria. Para verificar que este es el caso:
 1. Asegúrese de que la configuración de la máquina virtual incluye el dispositivo **memballoon**:


```
# virsh dumpxml testguest | grep memballoon
<memballoon model='virtio'>
  </memballoon>
```

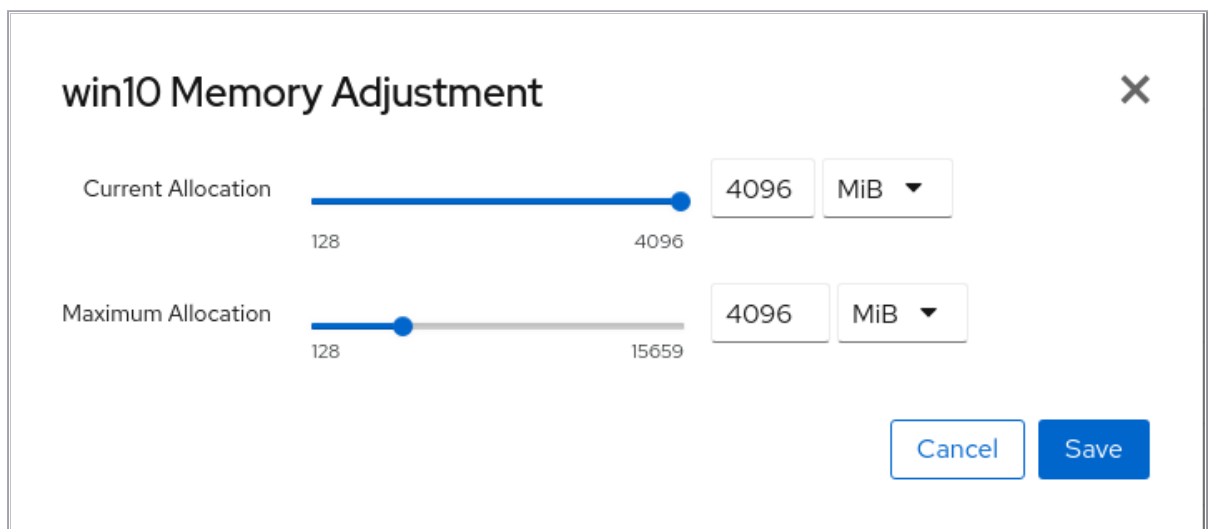
Si este comando muestra alguna salida y el modelo no está configurado en **none**, el dispositivo **memballoon** está presente.
 2. Asegúrese de que los controladores de balones se ejecutan en el sistema operativo invitado.
 - En los huéspedes de Windows, los controladores se instalan como parte del paquete de controladores virtio-win. Para obtener instrucciones, consulte [Sección 17.2.1, "Instalación de controladores paravirtuales KVM para máquinas virtuales Windows"](#).
 - En los huéspedes de Linux, los controladores suelen estar incluidos por defecto y se activan cuando el dispositivo **memballoon** está presente.
- Para utilizar la consola web para gestionar las máquinas virtuales, [instale el complemento VM de la consola web](#).

Procedimiento

1. **Optional:** Obtenga la información sobre la memoria máxima y la memoria actualmente utilizada para una VM. Esto servirá como línea de base para sus cambios, y también para la verificación.

```
# virsh dominfo testquest
Max memory: 2097152 KiB
Used memory: 2097152 KiB
```

2. En la interfaz de **máquinas virtuales**, haga clic en una fila con el nombre de las máquinas virtuales para las que desea ver y ajustar la memoria asignada. La fila se expande para revelar el panel de Visión General con información básica sobre las máquinas virtuales seleccionadas.
3. Haga clic en el valor de la línea **Memory** en el panel de visión general. Aparece el diálogo **Memory Adjustment**.



4. Configurar las CPUs virtuales para la VM seleccionada.
 - **Maximum allocation** - Establece la cantidad máxima de memoria del host que la VM puede utilizar para sus procesos. El aumento de este valor mejora el potencial de rendimiento de la VM, y la reducción del valor disminuye la huella de rendimiento que la VM tiene en su host. El ajuste de la asignación máxima de memoria sólo es posible en una máquina virtual apagada.
 - **Current allocation** - Establece la cantidad real de memoria asignada a la VM. Puede ajustar el valor para regular la memoria disponible para la VM para sus procesos. Este valor no puede superar el valor máximo de asignación.
5. Haga clic en **Guardar**. Se ajusta la asignación de memoria de la VM.

Recursos adicionales

- Para obtener instrucciones para ajustar la configuración de la memoria de la VM mediante la interfaz de línea de comandos, consulte [Sección 16.3.2, "Añadir y eliminar la memoria de la máquina virtual mediante la interfaz de línea de comandos"](#).
- Para optimizar la forma en que la VM utiliza la memoria asignada, puede modificar su configuración de vCPU. Para más información, consulte [Sección 16.5, "Optimización del rendimiento de la CPU de la máquina virtual"](#).

16.3.2. Añadir y eliminar la memoria de la máquina virtual mediante la interfaz de línea de comandos

Para mejorar el rendimiento de una máquina virtual (VM) o para liberar los recursos del host que está utilizando, puede utilizar la CLI para ajustar la cantidad de memoria asignada a la VM.

Requisitos previos

- El sistema operativo invitado está ejecutando los controladores del globo de memoria. Para verificar que este es el caso:
 1. Asegúrese de que la configuración de la máquina virtual incluye el dispositivo **memballoon**:

```
# virsh dumpxml testguest | grep memballoon
<memballoon model='virtio'>
  </memballoon>
```

Si este comando muestra alguna salida y el modelo no está configurado en **none**, el dispositivo **memballoon** está presente.

2. Asegúrese de que los controladores de balones se ejecutan en el sistema operativo invitado.
 - En los huéspedes de Windows, los controladores se instalan como parte del paquete de controladores virtio-win. Para obtener instrucciones, consulte [Sección 17.2.1, "Instalación de controladores paravirtuales KVM para máquinas virtuales Windows"](#).
 - En los huéspedes de Linux, los controladores suelen estar incluidos por defecto y se activan cuando el dispositivo **memballoon** está presente.

Procedimiento

1. **Optional:** Obtenga la información sobre la memoria máxima y la memoria actualmente utilizada para una VM. Esto servirá como línea de base para sus cambios, y también para la verificación.

```
# virsh dominfo testguest
Max memory: 2097152 KiB
Used memory: 2097152 KiB
```

2. Ajuste la memoria máxima asignada a una VM. Aumentar este valor mejora el potencial de rendimiento de la VM, y reducir el valor disminuye la huella de rendimiento que la VM tiene en su host. Tenga en cuenta que este cambio sólo puede realizarse en una VM apagada, por lo que ajustar una VM en funcionamiento requiere un reinicio para que tenga efecto. Por ejemplo, para cambiar la memoria máxima que la VM *testguest* puede utilizar a 4096 MiB:

```
# virt-xml testguest --edit --memory memory=4096,currentMemory=4096
Domain 'testguest' defined successfully.
Changes will take effect after the domain is fully powered off.
```

1. **Optional:** También puede ajustar la memoria utilizada actualmente por la VM, hasta la asignación máxima. Esto regula la carga de memoria que la VM tiene en el host hasta el próximo reinicio, sin cambiar la asignación máxima de la VM.

```
# virsh setmem testguest --current 2048
```

Verificación

1. Confirme que la memoria utilizada por la VM ha sido actualizada:

```
# virsh dominfo testguest
Max memory: 4194304 KiB
Used memory: 2097152 KiB
```

2. **Optional:** Si ha ajustado la memoria actual de la VM, puede obtener las estadísticas del globo de memoria de la VM para evaluar la eficacia con la que regula el uso de la memoria.

```
# virsh domstats --balloon testguest
Domain: 'testguest'
balloon.current=365624
balloon.maximum=4194304
balloon.swap_in=0
balloon.swap_out=0
balloon.major_fault=306
balloon.minor_fault=156117
balloon.unused=3834448
balloon.available=4035008
balloon.usable=3746340
balloon.last-update=1587971682
balloon.disk_caches=75444
balloon.hugetlb_pgalloc=0
balloon.hugetlb_pgfail=0
balloon.rss=1005456
```

Recursos adicionales

- Para obtener instrucciones para ajustar la configuración de la memoria de la VM mediante la consola web, consulte [Sección 16.3.1, “Añadir y eliminar la memoria de la máquina virtual mediante la consola web”](#).
- Para optimizar la forma en que la VM utiliza la memoria asignada, puede modificar su configuración de vCPU. Para más información, consulte [Sección 16.5, “Optimización del rendimiento de la CPU de la máquina virtual”](#).

16.3.3. Recursos adicionales

- Para aumentar la memoria máxima de una VM en ejecución, puede adjuntar un dispositivo de memoria a la VM. Esto también se conoce como **memory hot plug**. Para más detalles, consulte [Sección 10.2, “Adjuntar dispositivos a las máquinas virtuales”](#). Tenga en cuenta que la eliminación de un dispositivo de memoria de una VM, también conocida como **memory hot unplug**, no está soportada en RHEL 8 y Red Hat desaconseja su uso.

16.4. OPTIMIZACIÓN DEL RENDIMIENTO DE E/S DE LAS MÁQUINAS VIRTUALES

Las capacidades de entrada y salida (E/S) de una máquina virtual (VM) pueden limitar significativamente la eficiencia general de la VM. Para solucionar esto, puede optimizar la E/S de una VM configurando los parámetros de E/S en bloque.

16.4.1. Ajuste de la E/S en bloque en las máquinas virtuales

Cuando una o varias máquinas virtuales utilizan varios dispositivos de bloque, puede ser importante ajustar la prioridad de E/S de determinados dispositivos virtuales modificando su *I/O weights*.

Aumentar el peso de E/S de un dispositivo aumenta su prioridad para el ancho de banda de E/S, y por lo tanto le proporciona más recursos del host. Del mismo modo, reducir el peso de un dispositivo hace que consuma menos recursos del host.



NOTA

El valor de **weight** de cada dispositivo debe estar dentro del rango de **100 a 1000**. Como alternativa, el valor puede ser **0**, lo que elimina ese dispositivo de los listados por dispositivo.

Procedimiento

Para visualizar y configurar los parámetros de E/S de bloque de una VM:

1. Muestra los parámetros actuales de **<blkio>** para una VM:

```
# virsh dumpxml VM-name
```

```
<domain>
[...]
<blkio>
  <weight>800</weight>
  <device>
    <path>/dev/sda</path>
    <weight>1000</weight>
  </device>
  <device>
    <path>/dev/sdb</path>
    <weight>500</weight>
  </device>
</blkio>
[...]
</domain>
```

2. Editar el peso de E/S de un dispositivo especificado:

```
# virsh blkio VM-name --device-weights device, I/O-weight
```

Por ejemplo, lo siguiente cambia el peso del dispositivo */dev/sda* en la VM *liftrul* a 500.

```
# virsh blkio liftrul --device-weights /dev/sda, 500
```

16.4.2. Estrangulamiento de E/S de disco en máquinas virtuales

Cuando varias máquinas virtuales se ejecutan simultáneamente, pueden interferir con el rendimiento del sistema mediante el uso excesivo de E/S de disco. El estrangulamiento de E/S de disco en la virtualización KVM proporciona la capacidad de establecer un límite en las solicitudes de E/S de disco enviadas desde las VMs a la máquina anfitriona. Esto puede evitar que una máquina virtual sobreutilice los recursos compartidos y afecte al rendimiento de otras máquinas virtuales.

Para activar el estrangulamiento de E/S de disco, establezca un límite en las solicitudes de E/S de disco enviadas desde cada dispositivo de bloque conectado a las máquinas virtuales a la máquina anfitriona.

Procedimiento

1. Utilice el comando **virsh domblklist** para listar los nombres de todos los dispositivos de disco en una VM especificada.

```
# virsh domblklist rollin-coal
Target  Source
-----
vda     /var/lib/libvirt/images/rollin-coal.qcow2
sda     -
sdb     /home/horridly-demanding-processes.iso
```

2. Busque el dispositivo de bloque del host en el que está montado el disco virtual que desea estrangular.

Por ejemplo, si quiere acelerar el disco virtual **sdb** del paso anterior, la siguiente salida muestra que el disco está montado en la partición **/dev/nvme0n1p3**.

```
$ lsblk
NAME                                MAJ:MIN RM  SIZE RO TYPE  MOUNTPOINT
zram0                               252:0  0    4G  0 disk [SWAP]
nvme0n1                             259:0  0 238.5G  0 disk
├─nvme0n1p1                         259:1  0   600M  0 part /boot/efi
├─nvme0n1p2                         259:2  0    1G  0 part /boot
├─nvme0n1p3                         259:3  0 236.9G  0 part
└─luks-a1123911-6f37-463c-b4eb-fxzy1ac12fea 253:0  0 236.9G  0 crypt /home
```

3. Establezca los límites de E/S para el dispositivo de bloque utilizando el comando **virsh blkiotune**.

```
# virsh blkiotune VM-name --parameter device,limit
```

El siguiente ejemplo regula el disco **sdb** en la VM **rollin-coal** a 1000 operaciones de E/S de lectura y escritura por segundo y a 50 MB por segundo de rendimiento de lectura y escritura.

```
# virsh blkiotune rollin-coal --device-read-iops-sec /dev/nvme0n1p3,1000 --device-
write-iops-sec /dev/nvme0n1p3,1000 --device-write-bytes-sec
/dev/nvme0n1p3,52428800 --device-read-bytes-sec /dev/nvme0n1p3,52428800
```

Información adicional

- El estrangulamiento de E/S de disco puede ser útil en varias situaciones, por ejemplo, cuando se ejecutan máquinas virtuales pertenecientes a diferentes clientes en el mismo host, o cuando se dan garantías de calidad de servicio para diferentes máquinas virtuales. El estrangulamiento de E/S de disco también puede utilizarse para simular discos más lentos.
- El estrangulamiento de E/S puede aplicarse de forma independiente a cada dispositivo de bloque conectado a una máquina virtual y admite límites de rendimiento y operaciones de E/S.
- Red Hat no soporta el uso del comando **virsh blkdeviotune** para configurar el estrangulamiento de E/S en las VMs. Para más información sobre las características no soportadas cuando se utiliza RHEL 8 como anfitrión de VM, consulte [Sección 20.3](#),

“Características no soportadas en la virtualización de RHEL 8”.

16.4.3. Activación de la cola múltiple virtio-scsi

Al utilizar dispositivos de almacenamiento **virtio-scsi** en sus máquinas virtuales (VM), la función *multi-queue virtio-scsi* proporciona un mejor rendimiento y escalabilidad del almacenamiento. Permite que cada CPU virtual (vCPU) tenga una cola y una interrupción separadas para usar sin afectar a otras vCPU.

Procedimiento

- Para habilitar el soporte de múltiples colas virtio-scsi para una VM específica, añada lo siguiente a la configuración XML de la VM, donde *N* es el número total de colas vCPU:

```
<controller type='scsi' index='0' model='virtio-scsi'>
  <driver queues='N' />
</controller>
```

16.5. OPTIMIZACIÓN DEL RENDIMIENTO DE LA CPU DE LA MÁQUINA VIRTUAL

Al igual que las CPUs físicas en las máquinas anfitrionas, las vCPUs son críticas para el rendimiento de las máquinas virtuales (VM). Como resultado, la optimización de las vCPUs puede tener un impacto significativo en la eficiencia de los recursos de sus VMs. Para optimizar su vCPU:

1. Ajuste cuántas CPUs de host se asignan a la VM. Puede hacerlo mediante [la CLI](#) o [la consola web](#).
2. Asegúrese de que el modelo de vCPU está alineado con el modelo de CPU del host. Por ejemplo, para configurar la VM *testguest1* para que utilice el modelo de CPU del host:

```
# virt-xml testguest1 --edit --cpu host-model
```

3. [Desactivar la fusión de páginas del núcleo \(KSM\)](#) .
4. Si su máquina anfitriona utiliza acceso no uniforme a la memoria (NUMA), también puede **configure NUMA** para sus máquinas virtuales. Esto mapea los procesos de la CPU y la memoria del host en los procesos de la CPU y la memoria de la VM lo más cerca posible. En efecto, el ajuste NUMA proporciona a la vCPU un acceso más ágil a la memoria del sistema asignada a la VM, lo que puede mejorar la eficacia del procesamiento de la vCPU. Para más detalles, consulte [Sección 16.5.3, “Configuración de NUMA en una máquina virtual”](#) y [Sección 16.5.4, “Ejemplo de escenario de ajuste de rendimiento de vCPU”](#) .

16.5.1. Añadir y eliminar CPUs virtuales mediante la interfaz de línea de comandos

Para aumentar u optimizar el rendimiento de la CPU de una máquina virtual (VM), puede añadir o eliminar CPUs virtuales (vCPUs) asignadas a la VM.

Cuando se realiza en una VM en ejecución, esto también se conoce como vCPU hot plugging y hot unplugging. Sin embargo, tenga en cuenta que la desconexión en caliente de vCPU no está soportada en RHEL 8 y Red Hat desaconseja su uso.

Requisitos previos

- **Optional:** Ver el estado actual de las vCPUs en la VM objetivo. Por ejemplo, para mostrar el número de vCPUs en la VM *testguest*:

```
# virsh vcpucount testguest
maximum  config  4
maximum  live    2
current  config  2
current  live    1
```

Esta salida indica que *testguest* está usando actualmente 1 vCPU, y se puede conectar en caliente 1 vCPU más para aumentar el rendimiento de la VM. Sin embargo, después de reiniciar, el número de vCPUs que utiliza *testguest* cambiará a 2, y será posible conectar en caliente 2 vCPUs más.

Procedimiento

1. Ajusta el número máximo de vCPUs que se pueden adjuntar a una VM, que entra en vigor en el siguiente arranque de la VM.

Por ejemplo, para aumentar el número máximo de vCPU para la VM *testguest* a 8:

```
# virsh setvcpus testguest 8 --maximum --config
```

Tenga en cuenta que el máximo puede estar limitado por la topología de la CPU, el hardware del host, el hipervisor y otros factores.

2. Ajuste el número actual de vCPUs adjuntas a una VM, hasta el máximo configurado en el paso anterior. Por ejemplo:

- Para aumentar el número de vCPUs adjuntas a la VM *testguest* en ejecución a 4:

```
# virsh setvcpus testguest 4 --live
```

Esto aumenta el rendimiento de la VM y la huella de carga del host de *testguest* hasta el siguiente arranque de la VM.

- Para disminuir permanentemente el número de vCPUs adjuntas a la VM *testguest* a 1:

```
# virsh setvcpus testguest 1 --config
```

Esto disminuye el rendimiento de la VM y la huella de carga del host de *testguest* después del siguiente arranque de la VM. Sin embargo, si es necesario, se pueden conectar en caliente vCPUs adicionales a la VM para aumentar temporalmente su rendimiento.

Verificación

- Confirme que el estado actual de vCPU para la VM refleja sus cambios.

```
# virsh vcpucount testguest
maximum  config  8
maximum  live    4
current  config  1
current  live    4
```

Recursos adicionales

- Para obtener información sobre cómo añadir y eliminar vCPUs mediante la consola web, consulte [Sección 16.5.2, “Gestión de las CPUs virtuales mediante la consola web”](#).

16.5.2. Gestión de las CPUs virtuales mediante la consola web

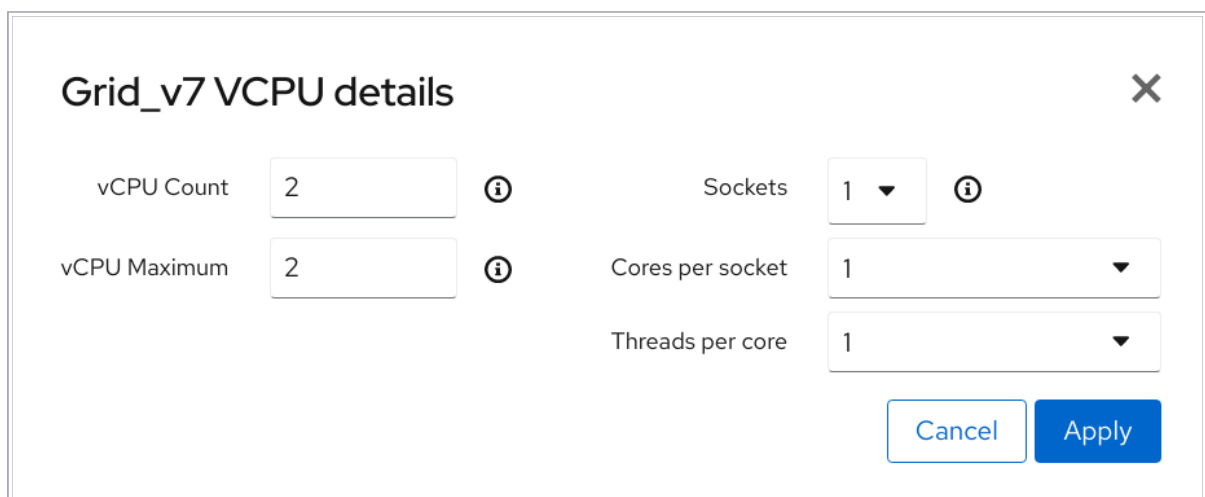
Utilizando la consola web de RHEL 8, puede revisar y configurar las CPUs virtuales utilizadas por las máquinas virtuales (VMs) a las que está conectada la consola web.

Requisitos previos

- Para utilizar la consola web para gestionar las máquinas virtuales, [instale el complemento VM de la consola web](#).

Procedimiento

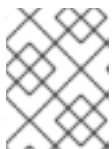
1. En la interfaz de **máquinas virtuales**, haga clic en una fila con el nombre de las máquinas virtuales para las que desea ver y configurar los parámetros de la CPU virtual. La fila se expande para revelar el panel de Visión General con información básica sobre las VMs seleccionadas, incluyendo el número de CPUs virtuales, y los controles para apagar y eliminar la VM.
2. Haga clic en el número de vCPUs en el panel de visión general. Aparece el diálogo de detalles de la vCPU.



The screenshot shows a dialog box titled "Grid_v7 VCPU details" with a close button (X) in the top right corner. The dialog contains several configuration fields:

- vCPU Count:** A text input field containing the number "2", with an information icon (i) to its right.
- vCPU Maximum:** A text input field containing the number "2", with an information icon (i) to its right.
- Sockets:** A dropdown menu showing "1", with an information icon (i) to its right.
- Cores per socket:** A dropdown menu showing "1".
- Threads per core:** A dropdown menu showing "1".

At the bottom right of the dialog, there are two buttons: "Cancel" (white with blue border) and "Apply" (solid blue).

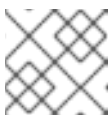


NOTA

La advertencia en el diálogo de detalles de la vCPU sólo aparece después de cambiar la configuración de la CPU virtual.

3. Configurar las CPUs virtuales para la VM seleccionada.

- **vCPU Count** - El número de vCPUs actualmente en uso.



NOTA

El recuento de vCPU no puede ser mayor que el máximo de vCPU.

- **vCPU Maximum** - El número máximo de CPUs virtuales que se pueden configurar para la VM. Si este valor es superior a **vCPU Count**, se pueden adjuntar vCPUs adicionales a la VM.

- **Sockets** - El número de sockets a exponer a la VM.
- **Cores per socket**- El número de núcleos para cada socket a exponer a la VM.
- **Threads per core** - El número de hilos para cada núcleo que se exponen a la VM.
Tenga en cuenta que las opciones **Sockets**, **Cores per socket** y **Threads per core** ajustan la topología de la CPU de la VM. Esto puede ser beneficioso para el rendimiento de la vCPU y puede impactar la funcionalidad de cierto software en el SO huésped. Si su implementación no requiere una configuración diferente, Red Hat recomienda mantener los valores por defecto.

4. Haga clic en **Aplicar**.

Se configuran las CPUs virtuales para la VM.



NOTA

Los cambios en la configuración de la CPU virtual sólo tienen efecto después de reiniciar la VM.

Recursos adicionales:

- Para obtener información sobre la gestión de las vCPUs mediante la interfaz de línea de comandos, consulte [Sección 16.5.1, "Añadir y eliminar CPUs virtuales mediante la interfaz de línea de comandos"](#).

16.5.3. Configuración de NUMA en una máquina virtual

Los siguientes métodos se pueden utilizar para configurar los ajustes de acceso no uniforme a la memoria (NUMA) de una máquina virtual (VM) en un host RHEL 8.

Requisitos previos

- El host es una máquina compatible con NUMA. Para detectar si es así, utilice el comando **virsh nodeinfo** y consulte la línea **NUMA cell(s)**:

```
# virsh nodeinfo
CPU model:      x86_64
CPU(s):         48
CPU frequency:  1200 MHz
CPU socket(s):  1
Core(s) per socket: 12
Thread(s) per core: 2
NUMA cell(s):  2
Memory size:    67012964 KiB
```

Si el valor de la línea es 2 o mayor, el host es compatible con NUMA.

Procedimiento

Para facilitar su uso, puede establecer la configuración NUMA de una máquina virtual utilizando utilidades y servicios automatizados. Sin embargo, es más probable que la configuración manual de NUMA produzca una mejora significativa del rendimiento.

Automatic methods

- Establezca la política NUMA de la máquina virtual en **Preferred**. Por ejemplo, para hacerlo para la VM *testguest5*:

```
# virt-xml testguest5 --edit --vcpus placement=auto
# virt-xml testguest5 --edit --numatune mode=preferred
```

- Activar el equilibrio automático de NUMA en el host:

```
# echo 1 > /proc/sys/kernel/numa_balancing
```

- Utilice el comando **numad** para alinear automáticamente la CPU de la VM con los recursos de memoria.

```
# numad
```

Manual methods

1. Asigne hilos específicos de vCPU a una CPU de host específica o a un rango de CPUs. Esto también es posible en hosts y máquinas virtuales que no son NUMA, y se recomienda como método seguro para mejorar el rendimiento de las vCPU.

Por ejemplo, los siguientes comandos conectan los hilos vCPU 0 a 5 de la VM *testguest6* a las CPUs del host 1, 3, 5, 7, 9 y 11, respectivamente:

```
# virsh vcpupin testguest6 0 1
# virsh vcpupin testguest6 1 3
# virsh vcpupin testguest6 2 5
# virsh vcpupin testguest6 3 7
# virsh vcpupin testguest6 4 9
# virsh vcpupin testguest6 5 11
```

A continuación, puede comprobar si se ha realizado con éxito:

```
# virsh vcpupin testguest6
VCPU  CPU Affinity
-----
0     1
1     3
2     5
3     7
4     9
5    11
```

2. Después de fijar los hilos de vCPU, también puede fijar los hilos de proceso de QEMU asociados con una VM específica a una CPU de host específica o a un rango de CPUs. Por ejemplo, los siguientes comandos fijan el hilo de proceso QEMU de *testguest6* a las CPUs 13 y 15, y verifican que esto se ha realizado con éxito:

```
# virsh emulatorpin testguest6 13,15
# virsh emulatorpin testguest6
emulator: CPU Affinity
-----
*: 13,15
```

- Por último, también puede especificar qué nodos NUMA del host se asignarán específicamente a una determinada VM. Esto puede mejorar el uso de la memoria del host por parte de la vCPU de la VM. Por ejemplo, los siguientes comandos configuran `testquest6` para que utilice los nodos NUMA del host 3 a 5, y verifican que esto fue exitoso:

```
# virsh numatune testquest6 --nodeset 3-5
# virsh numatune testquest6
```

Recursos adicionales

- Tenga en cuenta que para obtener los mejores resultados de rendimiento, se recomienda utilizar todos los métodos de ajuste manual indicados anteriormente. Para ver un ejemplo de esta configuración, consulte [Sección 16.5.4, “Ejemplo de escenario de ajuste de rendimiento de vCPU”](#).
- Para ver la configuración NUMA actual de tu sistema, puedes utilizar la utilidad **numastat**. Para más detalles sobre el uso de **numastat**, consulte [Sección 16.7, “Herramientas de supervisión del rendimiento de las máquinas virtuales”](#).
- Actualmente no es posible realizar el ajuste de NUMA en los hosts IBM Z. Para más información, consulte [Sección 4.2, “En qué se diferencia la virtualización en IBM Z de la de AMD64 e Intel 64”](#).

16.5.4. Ejemplo de escenario de ajuste de rendimiento de vCPU

Para obtener el mejor rendimiento posible de la vCPU, Red Hat recomienda utilizar los ajustes manuales **vcpupin**, **emulatorpin**, y **numatune** juntos, por ejemplo como en el siguiente escenario.

Escenario de partida

- Su host tiene las siguientes características de hardware:
 - 2 nodos NUMA
 - 3 núcleos de CPU en cada nodo
 - 2 hilos en cada núcleo

La salida de **virsh nodeinfo** de una máquina de este tipo sería similar:

```
# virsh nodeinfo
CPU model:      x86_64
CPU(s):         12
CPU frequency:  3661 MHz
CPU socket(s):  2
Core(s) per socket: 3
Thread(s) per core: 2
NUMA cell(s):  2
Memory size:    31248692 KiB
```

- Usted pretende modificar una VM existente para que tenga 8 vCPUs, lo que significa que no cabrá en un solo nodo NUMA. Por lo tanto, debes distribuir 4 vCPUs en cada nodo NUMA y hacer que la topología de las vCPUs se parezca lo más posible a la topología del host. Esto significa que las vCPUs que se ejecutan como hilos hermanos de una determinada CPU física deben estar vinculadas a los hilos

del host en el mismo núcleo. Para más detalles, consulte la página web *Solution*:

Solución

1. Obtenga la información sobre la topología del host:

virsh capabilities

La salida debe incluir una sección similar a la siguiente:

```
<topology>
  <cells num="2">
    <cell id="0">
      <memory unit="KiB">15624346</memory>
      <pages unit="KiB" size="4">3906086</pages>
      <pages unit="KiB" size="2048">0</pages>
      <pages unit="KiB" size="1048576">0</pages>
      <distances>
        <sibling id="0" value="10" />
        <sibling id="1" value="21" />
      </distances>
      <cpus num="6">
        <cpu id="0" socket_id="0" core_id="0" siblings="0,3" />
        <cpu id="1" socket_id="0" core_id="1" siblings="1,4" />
        <cpu id="2" socket_id="0" core_id="2" siblings="2,5" />
        <cpu id="3" socket_id="0" core_id="0" siblings="0,3" />
        <cpu id="4" socket_id="0" core_id="1" siblings="1,4" />
        <cpu id="5" socket_id="0" core_id="2" siblings="2,5" />
      </cpus>
    </cell>
    <cell id="1">
      <memory unit="KiB">15624346</memory>
      <pages unit="KiB" size="4">3906086</pages>
      <pages unit="KiB" size="2048">0</pages>
      <pages unit="KiB" size="1048576">0</pages>
      <distances>
        <sibling id="0" value="21" />
        <sibling id="1" value="10" />
      </distances>
      <cpus num="6">
        <cpu id="6" socket_id="1" core_id="3" siblings="6,9" />
        <cpu id="7" socket_id="1" core_id="4" siblings="7,10" />
        <cpu id="8" socket_id="1" core_id="5" siblings="8,11" />
        <cpu id="9" socket_id="1" core_id="3" siblings="6,9" />
        <cpu id="10" socket_id="1" core_id="4" siblings="7,10" />
        <cpu id="11" socket_id="1" core_id="5" siblings="8,11" />
      </cpus>
    </cell>
  </cells>
</topology>
```

2. **Optional:** Pruebe el rendimiento de la máquina virtual utilizando [las herramientas y utilidades aplicables](#).
3. Configurar y montar páginas enormes de 1 GiB en el host:

- a. Añade la siguiente línea a la línea de comandos del kernel del host:

```
default_hugepagesz=1G hugepagesz=1G
```

- b. Cree el archivo **/etc/systemd/system/hugetlb-gigantic-pages.service** con el siguiente contenido:

```
[Unit]
Description=HugeTLB Gigantic Pages Reservation
DefaultDependencies=no
Before=dev-hugepages.mount
ConditionPathExists=/sys/devices/system/node
ConditionKernelCommandLine=hugepagesz=1G

[Service]
Type=oneshot
RemainAfterExit=yes
ExecStart=/etc/systemd/hugetlb-reserve-pages.sh

[Install]
WantedBy=sysinit.target
```

- c. Cree el archivo **/etc/systemd/hugetlb-reserve-pages.sh** con el siguiente contenido:

```
#!/bin/sh

nodes_path=/sys/devices/system/node/
if [ ! -d $nodes_path ]; then
    echo "ERROR: $nodes_path does not exist"
    exit 1
fi

reserve_pages()
{
    echo $1 > $nodes_path/$2/hugepages/hugepages-1048576kB/nr_hugepages
}

reserve_pages 4 node1
reserve_pages 4 node2
```

Esto reserva cuatro páginas enormes de 1GiB de *node1* y cuatro páginas enormes de 1GiB de *node2*.

- d. Haga ejecutable el script creado en el paso anterior:

```
# chmod x /etc/systemd/hugetlb-reserve-pages.sh
```

- e. Habilitar la reserva de página enorme en el arranque:

```
# systemctl enable hugetlb-gigantic-pages
```

4. Utilice el comando **virsh edit** para editar la configuración XML de la VM que desea optimizar, en este ejemplo *super-VM*:

virsh edit super-vm

5. Ajuste la configuración XML de la VM de la siguiente manera:

- a. Configure la VM para utilizar 8 vCPUs estáticas. Utilice el elemento `<vcpu/>` para hacerlo.
- b. Conecte cada uno de los hilos de la vCPU a los correspondientes hilos de la CPU del host que refleja en la topología. Para ello, utilice los elementos de `<vcpupin/>` en la sección `<cputune>`.

Tenga en cuenta que, como se muestra en la utilidad **virsh capabilities** anterior, los hilos de la CPU del host no están ordenados secuencialmente en sus respectivos núcleos. Además, los hilos de la vCPU deben fijarse en el conjunto más alto de núcleos de host disponibles en el mismo nodo NUMA. Para ver una ilustración de la tabla, consulte la sección **Additional Resources** a continuación.

La configuración XML para los pasos a. y b. puede ser similar a

```
<cputune>
  <vcpupin vcpu='0' cpuset='1' />
  <vcpupin vcpu='1' cpuset='4' />
  <vcpupin vcpu='2' cpuset='2' />
  <vcpupin vcpu='3' cpuset='5' />
  <vcpupin vcpu='4' cpuset='7' />
  <vcpupin vcpu='5' cpuset='10' />
  <vcpupin vcpu='6' cpuset='8' />
  <vcpupin vcpu='7' cpuset='11' />
  <emulatorpin cpuset='6,9' />
</cputune>
```

- c. Configura la VM para que utilice páginas enormes de 1 GiB:

```
<memoryBacking>
  <hugepages>
    <page size='1' unit='GiB' />
  </hugepages>
</memoryBacking>
```

- d. Configure los nodos NUMA de la máquina virtual para que utilicen la memoria de los nodos NUMA correspondientes del host. Para ello, utilice los elementos de `<memnode/>` en la sección `<numatune/>`:

```
<numatune>
  <memory mode="preferred" nodeset="1" />
  <memnode cellid="0" mode="strict" nodeset="0" />
  <memnode cellid="1" mode="strict" nodeset="1" />
</numatune>
```

- e. Asegúrese de que el modo de la CPU está configurado en **host-passthrough**, y que la CPU utiliza la caché en el modo **passthrough**:

```
<cpu mode="host-passthrough">
  <topology sockets="2" cores="2" threads="2" />
  <cache mode="passthrough" />
```

6. La configuración XML resultante de la VM debe incluir una sección similar a la siguiente:

```
[...]
<memoryBacking>
  <hugepages>
    <page size='1' unit='GiB'/>
  </hugepages>
</memoryBacking>
<vcpu placement='static'>8</vcpu>
<cpupin>
  <vcpupin vcpu='0' cpuset='1'/>
  <vcpupin vcpu='1' cpuset='4'/>
  <vcpupin vcpu='2' cpuset='2'/>
  <vcpupin vcpu='3' cpuset='5'/>
  <vcpupin vcpu='4' cpuset='7'/>
  <vcpupin vcpu='5' cpuset='10'/>
  <vcpupin vcpu='6' cpuset='8'/>
  <vcpupin vcpu='7' cpuset='11'/>
  <emulatorpin cpuset='6,9'/>
</cpupin>
<numatune>
  <memory mode="preferred" nodeset="1"/>
  <memnode cellid="0" mode="strict" nodeset="0"/>
  <memnode cellid="1" mode="strict" nodeset="1"/>
</numatune>
<cpu mode="host-passthrough">
  <topology sockets="2" cores="2" threads="2"/>
  <cache mode="passthrough"/>
  <numa>
    <cell id="0" cpus="0-3" memory="2" unit="GiB">
      <distances>
        <sibling id="0" value="10"/>
        <sibling id="1" value="21"/>
      </distances>
    </cell>
    <cell id="1" cpus="4-7" memory="2" unit="GiB">
      <distances>
        <sibling id="0" value="21"/>
        <sibling id="1" value="10"/>
      </distances>
    </cell>
  </numa>
</cpu>
</domain>
```

7. **Optional:** Pruebe el rendimiento de la VM utilizando [las herramientas y utilidades aplicables](#) para evaluar el impacto de la optimización de la VM.

Recursos adicionales

- Las siguientes tablas ilustran las conexiones entre las vCPUs y las CPUs anfitrionas a las que deben ser fijadas:

Tabla 16.1. Topología de host

CPU threads	0	3	1	4	2	5	6	9	7	10	8	11
Cores	0		1		2		3		4		5	
Sockets	0						1					
NUMA nodes	0						1					

Tabla 16.2. Topología VM

vCPU threads	0	1	2	3	4	5	6	7
Cores	0		1		2		3	
Sockets	0						1	
NUMA nodes	0						1	

Tabla 16.3. Topología combinada de host y VM

vCPU threads		0	1	2	3		4	5	6	7		
Host CPU threads	0	3	1	4	2	5	6	9	7	10	8	11
Cores	0		1		2		3		4		5	
Sockets	0						1					
NUMA nodes	0						1					

En este escenario, hay 2 nodos NUMA y 8 vCPUs. Por lo tanto, 4 hilos de vCPU deben ser fijados a cada nodo.

Además, Red Hat recomienda dejar al menos un hilo de CPU disponible en cada nodo para las operaciones del sistema anfitrión.

Como en este ejemplo, cada nodo NUMA alberga 3 núcleos, cada uno con 2 hilos de la CPU anfitriona, el conjunto para el nodo 0 se traduce como sigue:

```
<vcpupin vcpu='0' cpuset='1'/>
<vcpupin vcpu='1' cpuset='4'/>
<vcpupin vcpu='2' cpuset='2'/>
<vcpupin vcpu='3' cpuset='5'/>
```

16.5.5. Desactivación de la fusión de páginas en el núcleo

Aunque la fusión de la misma página del núcleo (KSM) mejora la densidad de la memoria, aumenta la utilización de la CPU y puede afectar negativamente al rendimiento general dependiendo de la carga de trabajo. En estos casos, puede mejorar el rendimiento de la máquina virtual (VM) desactivando KSM.

Dependiendo de sus necesidades, puede desactivar KSM para una sola sesión o de forma persistente.

Procedimiento

- Para desactivar KSM para una sola sesión, utilice la utilidad **systemctl** para detener los servicios **ksm** y **ksmtuned**.

```
# systemctl stop ksm
# systemctl stop ksmtuned
```

- Para desactivar KSM de forma persistente, utilice la utilidad **systemctl** para desactivar los servicios **ksm** y **ksmtuned**.

```
# systemctl disable ksm
Removed /etc/systemd/system/multi-user.target.wants/ksm.service.
# systemctl disable ksmtuned
Removed /etc/systemd/system/multi-user.target.wants/ksmtuned.service.
```



NOTA

Las páginas de memoria compartidas entre máquinas virtuales antes de desactivar el KSM seguirán siendo compartidas. Para dejar de compartir, elimine todas las páginas de **PageKSM** en el sistema utilizando el siguiente comando:

```
# echo 2 > /sys/kernel/mm/ksm/run
```

Después de que las páginas anónimas sustituyan a las páginas KSM, el servicio del kernel **khugepaged** reconstruirá las hugepages transparentes en la memoria física de la máquina virtual.

16.6. OPTIMIZACIÓN DEL RENDIMIENTO DE LA RED DE MÁQUINAS VIRTUALES

Debido a la naturaleza virtual de la tarjeta de interfaz de red (NIC) de una VM, la VM pierde una parte de su ancho de banda de red de host asignado, lo que puede reducir la eficiencia general de la carga de trabajo de la VM. Los siguientes consejos pueden minimizar el impacto negativo de la virtualización en el rendimiento de la NIC virtual (vNIC).

Procedimiento

Utilice cualquiera de los siguientes métodos y observe si tiene un efecto beneficioso en el rendimiento de su red VM:

Habilitar el módulo **vhost_net**

En el host, asegúrese de que la función del kernel **vhost_net** está activada:

```
# lsmod | grep vhost
vhost_net      32768  1
vhost          53248  1 vhost_net
```

```
tap          24576  1 vhost_net
tun          57344  6 vhost_net
```

Si la salida de este comando está en blanco, active el módulo del kernel **vhost_net**:

```
# modprobe vhost_net
```

Configurar la red virtio de colas múltiples

Para configurar la función *multi-queue virtio-net* para una VM, utilice el comando **virsh edit** para editar la configuración XML de la VM. En el XML, añada lo siguiente a la sección **<devices>**, y sustituya **N** por el número de vCPUs de la VM, hasta 16:

```
<interface type='network'>
  <source network='default'/>
  <model type='virtio'/>
  <driver name='vhost' queues='N'/>
</interface>
```

Si la máquina virtual está funcionando, reiníciela para que los cambios surtan efecto.

Procesamiento de paquetes de red

En las configuraciones de máquinas virtuales de Linux con una ruta de transmisión larga, la agrupación de paquetes por lotes antes de enviarlos al kernel puede mejorar la utilización de la caché. Para configurar la agrupación de paquetes, utilice el siguiente comando en el host y sustituya *tap0* por el nombre de la interfaz de red que utilizan las máquinas virtuales:

```
# ethtool -C tap0 rx-frames 128
```

SR-IOV

Si su NIC de host admite SR-IOV, utilice la asignación de dispositivos SR-IOV para sus vNIC. Para obtener más información, consulte [Sección 10.8, "Gestión de dispositivos SR-IOV"](#).

Recursos adicionales

- Para obtener información adicional sobre los tipos de conexión de red virtual y consejos de uso, consulte [Sección 13.1, "Entender la red virtual"](#).

16.7. HERRAMIENTAS DE SUPERVISIÓN DEL RENDIMIENTO DE LAS MÁQUINAS VIRTUALES

Para identificar qué es lo que consume más recursos de la VM y qué aspecto del rendimiento de la VM necesita ser optimizado, se pueden utilizar herramientas de diagnóstico de rendimiento, tanto generales como específicas para la VM.

Herramientas de supervisión del rendimiento del sistema operativo por defecto

Para la evaluación estándar del rendimiento, puede utilizar las utilidades proporcionadas por defecto por sus sistemas operativos anfitrión y huésped:

- En su host RHEL 8, como root, utilice la utilidad **top** o la aplicación **system monitor**, y busque **qemu** y **virt** en la salida. Esto muestra la cantidad de recursos del sistema anfitrión que sus máquinas virtuales están consumiendo.

- Si la herramienta de monitorización muestra que alguno de los procesos de **qemu** o **virt** consume una gran parte de la capacidad de la CPU o de la memoria del host, utilice la utilidad **perf** para investigar. Para más detalles, véase más abajo.
- Además, si un proceso de hilo de **vhost_net**, llamado por ejemplo *vhost_net-1234*, se muestra como consumiendo una cantidad excesiva de capacidad de la CPU del host, considere el uso de [las características de optimización de la red virtual](#), como **multi-queue virtio-net**.
- En el sistema operativo invitado, utilice las utilidades de rendimiento y las aplicaciones disponibles en el sistema para evaluar qué procesos consumen más recursos del sistema.
 - En los sistemas Linux, puede utilizar la utilidad **top**.
 - En los sistemas Windows, puede utilizar la aplicación **Task Manager**.

perf kvm

Puede utilizar la utilidad **perf** para recopilar y analizar estadísticas específicas de la virtualización sobre el rendimiento de su host RHEL 8. Para ello:

1. En el host, instale el paquete *perf*:

```
# yum install perf
```

2. Utilice el comando **perf kvm stat** para mostrar las estadísticas de perfeccionamiento de su host de virtualización:
 - Para la monitorización en tiempo real de su hipervisor, utilice el comando **perf kvm stat live**.
 - Para registrar los datos de perf de su hipervisor durante un periodo de tiempo, active el registro mediante el comando **perf kvm stat record**. Una vez cancelado o interrumpido el comando, los datos se guardan en el archivo **perf.data.guest**, que puede analizarse mediante el comando **perf kvm stat report**.
3. Analice la salida de **perf** para ver los tipos de eventos de **VM-EXIT** y su distribución. Por ejemplo, los eventos **PAUSE_INSTRUCTION** deberían ser poco frecuentes, pero en la siguiente salida, la alta ocurrencia de este evento sugiere que las CPUs del host no están manejando bien las vCPUs en funcionamiento. En tal escenario, considere apagar algunas de sus VMs activas, remover vCPUs de estas VMs, o [afinar el rendimiento de las vCPUs](#).

```
# perf kvm stat report
```

Analyze events for all VMs, all VCPUs:

VM-EXIT	Samples	Samples%	Time%	Min Time	Max Time	Avg time
EXTERNAL_INTERRUPT	365634	31.59%	18.04%	0.42us	58780.59us	204.08us (+- 0.99%)
MSR_WRITE	293428	25.35%	0.13%	0.59us	17873.02us	1.80us (+- 4.63%)
PREEMPTION_TIMER	276162	23.86%	0.23%	0.51us	21396.03us	3.38us (+- 5.19%)
PAUSE_INSTRUCTION	189375	16.36%	11.75%	0.72us	29655.25us	256.77us (+- 0.70%)

```

)          HLT      20440   1.77%  69.83%   0.62us 79319.41us 14134.56us (+- 0.79%
)          VMCALL   12426   1.07%   0.03%   1.02us 5416.25us   8.77us (+- 7.36%
)          EXCEPTION_NMI      27   0.00%   0.00%   0.69us   1.34us   0.98us (+-
3.50%)
          EPT_MISCONFIG      5   0.00%   0.00%   5.15us  10.85us   7.88us (+-
11.67%)

Total Samples:1157497, Total events handled time:413728274.66us.

```

Otros tipos de eventos que pueden señalar problemas en la salida de **perf kvm stat** incluyen:

- **INSN_EMULATION** - sugiere una [configuración](#) subóptima [de E/S de la máquina virtual](#) .

Para obtener más información sobre el uso de **perf** para supervisar el rendimiento de la virtualización, consulte la página de manual **perf-kvm**.

numastat

Para ver la configuración NUMA actual de tu sistema, puedes utilizar la utilidad **numastat**, que se proporciona al instalar el paquete **numactl**.

A continuación se muestra un host con 4 VMs en ejecución, cada una de las cuales obtiene memoria de múltiples nodos NUMA. Esto no es óptimo para el rendimiento de las vCPUs, y [justifica un ajuste](#) :

numastat -c qemu-kvm

Per-node process memory usage (in MBs)

PID	Node 0	Node 1	Node 2	Node 3	Node 4	Node 5	Node 6	Node 7	Total
51722 (qemu-kvm)	68	16	357	6936	2	3	147	598	8128
51747 (qemu-kvm)	245	11	5	18	5172	2532	1	92	8076
53736 (qemu-kvm)	62	432	1661	506	4851	136	22	445	8116
53773 (qemu-kvm)	1393	3	1	2	12	0	0	6702	8114
Total	1769	463	2024	7462	10037	2672	169	7837	32434

Por el contrario, a continuación se muestra la memoria proporcionada a cada VM por un único nodo, lo que es significativamente más eficiente.

numastat -c qemu-kvm

Per-node process memory usage (in MBs)

PID	Node 0	Node 1	Node 2	Node 3	Node 4	Node 5	Node 6	Node 7	Total
51747 (qemu-kvm)	0	0	7	0	8072	0	1	0	8080
53736 (qemu-kvm)	0	0	7	0	0	0	8113	0	8120
53773 (qemu-kvm)	0	0	7	0	0	0	1	8110	8118
59065 (qemu-kvm)	0	0	8050	0	0	0	0	0	8051
Total	0	0	8072	0	8072	0	8114	8110	32368

16.8. INFORMACIÓN RELACIONADA

- Cuando utilice Windows como sistema operativo invitado de su VM, Red Hat recomienda aplicar medidas de optimización adicionales. Para más detalles, consulte [Sección 17.2, “Optimización de las máquinas virtuales de Windows”](#).

CAPÍTULO 17. INSTALACIÓN Y GESTIÓN DE MÁQUINAS VIRTUALES WINDOWS

Para utilizar Microsoft Windows como sistema operativo invitado en sus máquinas virtuales (VMs) en un host RHEL 8, Red Hat recomienda tomar medidas adicionales para asegurar que estas VMs se ejecuten correctamente.

Para ello, las siguientes secciones proporcionan información sobre la instalación y optimización de las VMs de Windows en el host, así como la instalación y configuración de los controladores en estas VMs.

17.1. INSTALACIÓN DE MÁQUINAS VIRTUALES WINDOWS

A continuación se proporciona información sobre cómo crear una máquina Windows totalmente virtualizada en un host RHEL 8, lanzar el instalador gráfico de Windows dentro de la máquina virtual (VM) y optimizar el sistema operativo invitado (SO) Windows instalado.

Puede crear una VM e instalarla utilizando el comando **virt-install** o la consola web de RHEL 8.

Requisitos previos

- Una fuente de instalación del sistema operativo Windows, que puede ser una de las siguientes, y estar disponible localmente o en una red:
 - Una imagen ISO de un medio de instalación
 - Una imagen de disco de una instalación VM existente
- Un medio de almacenamiento con los controladores KVM **virtio**.
Para crear este medio, consulte [Sección 17.2.1.2, "Preparación de los medios de instalación del controlador virtio en una máquina anfitriona"](#).

Procedimiento

1. Cree la VM. Para obtener instrucciones, consulte [Sección 2.2, "Creación de máquinas virtuales"](#).

- Si utiliza la utilidad **virt-install** para crear la VM, añada las siguientes opciones al comando:
 - El medio de almacenamiento con los controladores KVM **virtio**. Por ejemplo:

```
--disk path=/usr/share/virtio-win/virtio-win.iso,device=disk,bus=virtio
```

- La versión de Windows que va a instalar. Por ejemplo, para Windows 10:

```
--os-variante win10
```

Para obtener una lista de las versiones de Windows disponibles y la opción adecuada, utilice el siguiente comando:

```
# osinfo-query os
```

- Si utiliza la consola web para crear la VM, especifique su versión de Windows en el campo **Operating System** de la ventana **Create New Virtual Machine**. Una vez creada la VM e instalado el SO invitado, conecte el medio de almacenamiento con los controladores virtio a

la VM utilizando la interfaz **Disks**. Para obtener instrucciones, consulte [Sección 11.3.7.3, “Adjuntar discos existentes a máquinas virtuales mediante la consola web”](#).

2. Instale el sistema operativo Windows en la VM.
Para obtener información sobre cómo instalar un sistema operativo Windows, consulte la documentación de instalación de Microsoft correspondiente.
3. Configure los controladores KVM **virtio** en el sistema operativo invitado Windows. Para más detalles, consulte [Sección 17.2.1, “Instalación de controladores paravirtuales KVM para máquinas virtuales Windows”](#).

Recursos adicionales

- Para obtener información sobre cómo optimizar aún más las máquinas virtuales de Windows, consulte [Sección 17.2, “Optimización de las máquinas virtuales de Windows”](#).

17.2. OPTIMIZACIÓN DE LAS MÁQUINAS VIRTUALES DE WINDOWS

Cuando se utiliza Microsoft Windows como sistema operativo invitado en una máquina virtual (VM) alojada en RHEL 8, el rendimiento del invitado puede verse afectado negativamente.

Por lo tanto, Red Hat recomienda optimizar sus VMs de Windows haciendo cualquier combinación de lo siguiente:

- Uso de controladores paravirtualizados. Para más información, consulte [Sección 17.2.1, “Instalación de controladores paravirtuales KVM para máquinas virtuales Windows”](#).
- Habilitación de las iluminaciones de Hyper-V. Para más información, consulte [Sección 17.2.2, “Habilitación de las iluminaciones de Hyper-V”](#).
- Configuración de los parámetros del controlador NetKVM. Para más información, consulte [Sección 17.2.3, “Configuración de los parámetros del controlador NetKVM”](#).
- Optimizar o desactivar los procesos de fondo de Windows. Para más información, consulte [Sección 17.2.5, “Optimización de los procesos de fondo en las máquinas virtuales de Windows”](#).

17.2.1. Instalación de controladores paravirtuales KVM para máquinas virtuales Windows

El principal método para mejorar el rendimiento de sus máquinas virtuales (VM) de Windows es instalar controladores KVM paravirtualizados (**virtio**) para Windows en el sistema operativo (SO) invitado.

Para ello:

1. Prepare el medio de instalación en la máquina anfitriona. Para más información, consulte [Sección 17.2.1.2, “Preparación de los medios de instalación del controlador virtio en una máquina anfitriona”](#).
2. Adjunte el medio de instalación a una VM de Windows existente, o adjúntelo al [crear una nueva VM de Windows](#).
3. Instale los controladores **virtio** en el sistema operativo invitado Windows. Para más información, consulte [Sección 17.2.1.3, “Instalación de los controladores virtio en un invitado de Windows”](#).

17.2.1.1. Cómo funcionan los controladores de Windows virtio

Los controladores paravirtualizados mejoran el rendimiento de las máquinas virtuales (VMs) disminuyendo la latencia de E/S y aumentando el rendimiento a niveles casi bare-metal. Red Hat recomienda el uso de controladores paravirtualizados para las máquinas virtuales que ejecutan tareas y aplicaciones de E/S intensas.

los controladores **virtio** son los controladores de dispositivos paravirtualizados de KVM, disponibles para las máquinas virtuales de Windows que se ejecutan en hosts KVM. Estos controladores son proporcionados por el paquete **virtio-win**, que incluye controladores para:

- Dispositivos (de almacenamiento) en bloque
- Controladores de interfaz de red
- Controladores de vídeo
- Dispositivo de ampliación de la memoria
- Dispositivo de puerto serie paravirtual
- Dispositivo fuente de entropía
- Dispositivo de pánico paravirtual
- Dispositivos de entrada, como ratones, teclados o tabletas
- Un pequeño conjunto de dispositivos emulados



NOTA

Para obtener información adicional sobre los dispositivos emulados, **virtio**, y asignados, consulte [Capítulo 10, Gestión de dispositivos virtuales](#).

Utilizando los controladores KVM virtio, se espera que las siguientes versiones de Microsoft Windows se ejecuten de forma similar a los sistemas físicos:

- Versiones de Windows Server: Consulte [Sistemas operativos invitados certificados para Red Hat Enterprise Linux con KVM](#) en la Base de conocimientos de Red Hat.
- Versiones de Windows Desktop (no servidor):
 - Windows 7 (versiones de 32 y 64 bits)
 - Windows 8 (versiones de 32 y 64 bits)
 - Windows 8.1 (versiones de 32 y 64 bits)
 - Windows 10 (versiones de 32 y 64 bits)

17.2.1.2. Preparación de los medios de instalación del controlador virtio en una máquina anfitriona

Para instalar los controladores virtio de KVM en una máquina virtual (VM) de Windows, primero debe preparar el medio de instalación del controlador virtio en la máquina anfitriona. Para ello, instale el paquete **virtio-win** en la máquina anfitriona y utilice el archivo **.iso** que proporciona como almacenamiento para la VM.

Requisitos previos

Requisitos previos

- Asegúrese de que la virtualización está [activada](#) en su sistema host RHEL 8.

Procedimiento

1. Descargue los controladores
 - a. Navegue para [descargar Red Hat Enterprise Linux](#) .
 - b. Seleccione la dirección **Product Variant** correspondiente a la arquitectura de su sistema. Por ejemplo, para Intel 64 y AMD64, seleccione **Red Hat Enterprise Linux for x86_64**
 - c. Asegúrese de que el **Version** es **8**.
 - d. En la página web **Packages**, busque **virtio-win**.
 - e. Haga clic en **Download Latest**.
El archivo **RPM** se descarga.
2. Instale el paquete **virtio-win** desde el directorio de descarga. Por ejemplo:

```
# yum install ~/Downloads/virtio-win-1.9.9-3.el8.noarch.rpm
[...]  
Installed:  
virtio-win-1.9.9-3.el8.noarch
```

Si la instalación tiene éxito, los archivos del controlador **virtio-win** se preparan en el directorio **/usr/share/virtio-win/**. Estos incluyen los archivos **ISO** y un directorio **drivers** con los archivos del controlador en directorios, uno para cada arquitectura y versión de Windows soportada.

```
# ls /usr/share/virtio-win/  
drivers/ guest-agent/ virtio-win-1.9.9.iso virtio-win.iso
```

3. Adjunte el archivo **virtio-win.iso** a la VM de Windows. Para ello, realice una de las siguientes acciones:
 - Utiliza el archivo como disco al [crear una nueva VM de Windows](#) .
 - Añada el archivo como un CD-ROM a una máquina virtual de Windows existente. Por ejemplo:

```
# virt-xml WindowsVM --add-device --disk virtio-win.iso,device=cdrom  
Domain 'WindowsVM' defined successfully.
```

Recursos adicionales

- Cuando **virtio-win.iso** está conectado a la VM de Windows, puede proceder a instalar el controlador de virtio en el sistema operativo invitado de Windows. Para obtener instrucciones, consulte [Sección 17.2.1.3, "Instalación de los controladores virtio en un invitado de Windows"](#) .

17.2.1.3. Instalación de los controladores virtio en un invitado de Windows

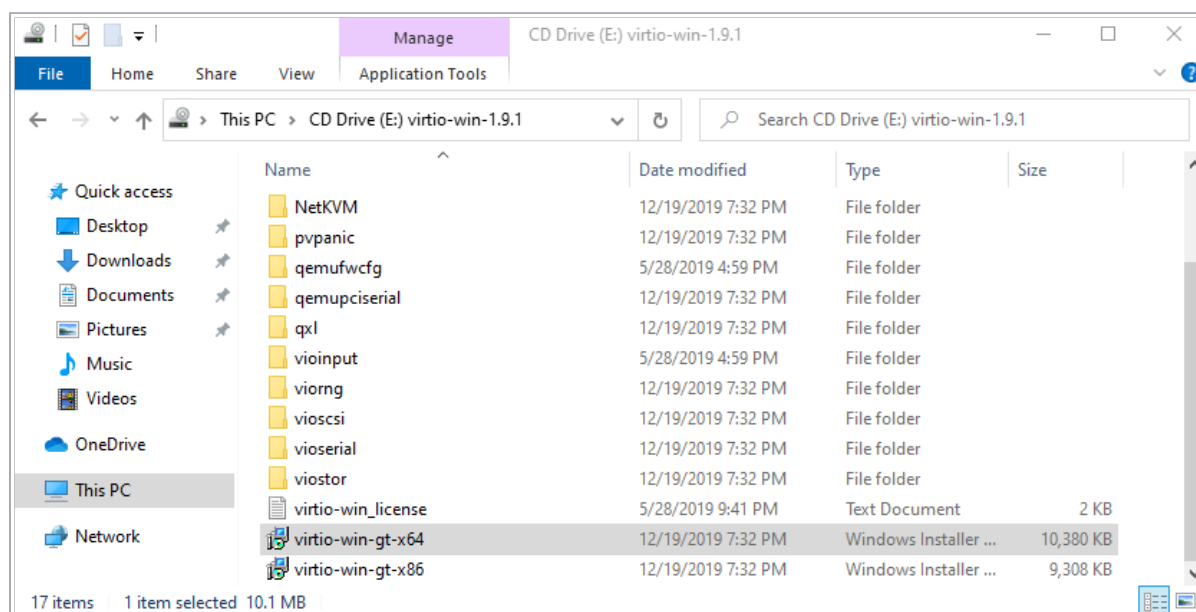
Para instalar los controladores de KVM **virtio** en un sistema operativo (SO) invitado Windows, debe añadir un dispositivo de almacenamiento que contenga los controladores - ya sea al crear la máquina virtual (VM) o después - e instalar los controladores en el SO invitado Windows.

Requisitos previos

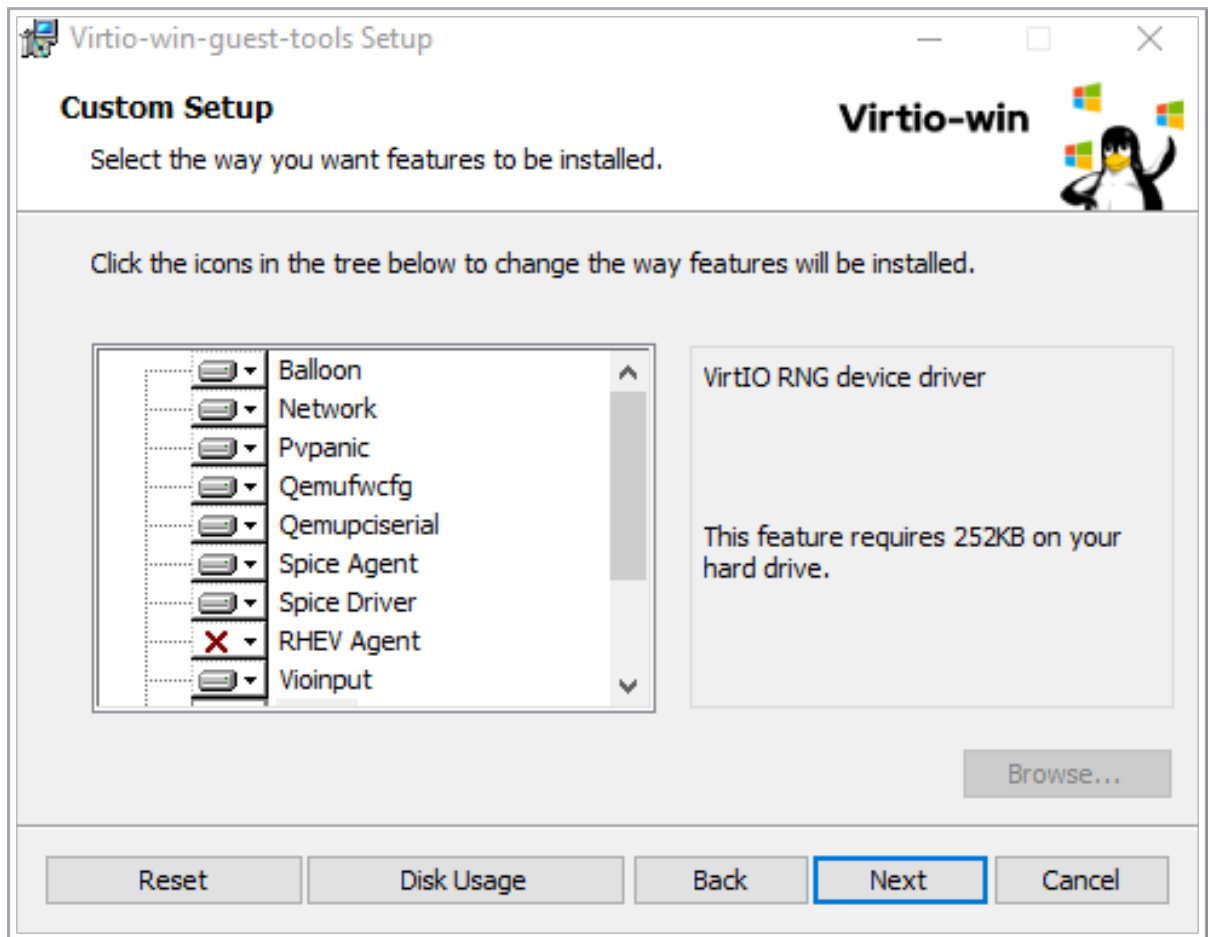
- Debe adjuntarse a la VM un medio de instalación con los controladores KVM **virtio**. Para obtener instrucciones sobre la preparación del medio, consulte [Sección 17.2.1.2, "Preparación de los medios de instalación del controlador virtio en una máquina anfitriona"](#).

Procedimiento

1. En el sistema operativo invitado Windows, abra la aplicación **File Explorer**.
2. Haga clic en **This PC**.
3. En el panel **Devices and drives**, abra el medio **virtio-win**.
4. Basándose en la arquitectura de la vCPU de la VM, ejecute uno de los instaladores en el medio.
 - Si utiliza una vCPU de 32 bits, ejecute el instalador **virtio-win-gt-x86**.
 - Si utiliza una vCPU de 64 bits, ejecute el instalador **virtio-win-gt-x64**.



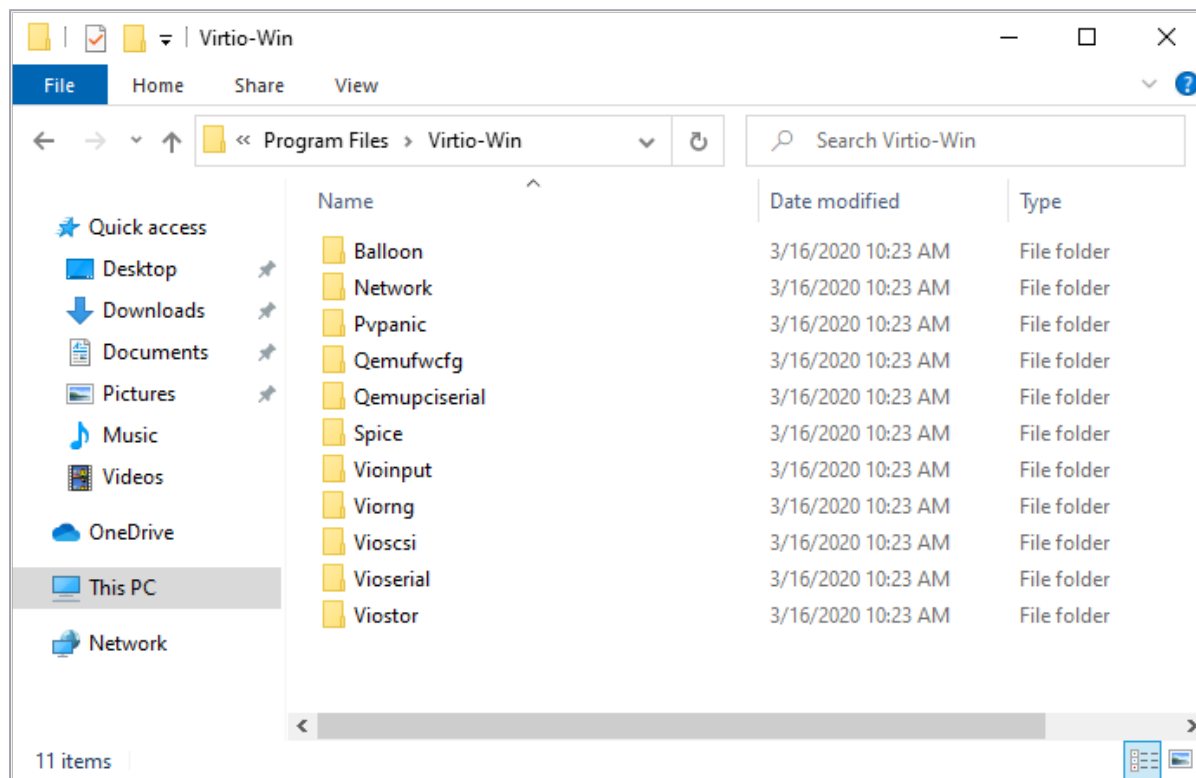
5. En el asistente de configuración de **Virtio-win-guest-tools** que se abre, siga las instrucciones mostradas hasta llegar al paso **Custom Setup**.



6. En la ventana de configuración personalizada, seleccione los controladores de dispositivo que desea instalar. El conjunto de controladores recomendado se selecciona automáticamente, y las descripciones de los controladores se muestran a la derecha de la lista.
7. Haga clic en **siguiente** y luego en **Instalar**.
8. Una vez finalizada la instalación, haga clic en **Finalizar**.
9. Reinicie la máquina virtual para completar la instalación del controlador.

Verificación

1. En **This PC**, abra el disco del sistema. Esto es típicamente **(C:)**.
2. En el directorio **Program Files**, abra el directorio **Virtio-Win**.
Si el directorio **Virtio-Win** está presente y contiene un subdirectorio para cada uno de los controladores seleccionados, la instalación fue exitosa.



Recursos adicionales

- Puede utilizar la interfaz de línea de comandos (CLI) de Microsoft Windows Installer (MSI) en lugar de la interfaz gráfica para instalar los controladores. Para obtener más información sobre MSI, consulte la [documentación de Microsoft](#).
- Si instala el controlador NetKVM, es posible que también tenga que configurar los parámetros de red del invitado de Windows. Para obtener instrucciones, consulte [Sección 17.2.3, “Configuración de los parámetros del controlador NetKVM”](#).

17.2.2. Habilitación de las iluminaciones de Hyper-V

Las iluminaciones de Hyper-V proporcionan un método para que KVM emule el hipervisor Microsoft Hyper-V. Esto mejora el rendimiento de las máquinas virtuales de Windows.

Las siguientes secciones proporcionan información sobre las iluminaciones de Hyper-V soportadas y cómo habilitarlas.

17.2.2.1. Habilitación de las iluminaciones de Hyper-V en una máquina virtual de Windows

Las iluminaciones de Hyper-V proporcionan un mejor rendimiento en una máquina virtual (VM) Windows que se ejecuta en un host RHEL 8. Para obtener instrucciones sobre cómo habilitarlas, consulte lo siguiente.

Procedimiento

1. Edite la configuración XML de la VM, añadiendo las iluminaciones de Hyper-V. En los siguientes comandos, sustituya `$VMNAME` por el nombre de la VM de Windows.

```
# virt-xml $VMNAME --edit --features
hyperv_relaxed=on,hyperv_vapic=on,hyperv_spinlocks=on,hyperv_spinlocks_retries=
8191,hyperv_vpindex=on,hyperv_runtime=on,hyperv_synic=on,hyperv_stimer=on,hyp
```

```
erv_frequencies=on
```

```
# virt-xml $VMNAME --edit --clock hypervclock_present=yes
```

2. Reiniciar la VM

Verificación

- Utilice el comando **virsh dumpxml** para mostrar la configuración XML de la VM modificada. Si incluye los siguientes segmentos, las iluminaciones de Hyper-V están habilitadas en la VM.

```
<hyperv>
  <relaxed state='on'/>
  <vapic state='on'/>
  <spinlocks state='on' retries='8191'/>
  <vpindex state='on'/>
  <runtime state='on' />
  <synic state='on'/>
  <stimer state='on'/>
  <frequencies state='on'/>
</hyperv>

<clock offset='localtime'>
  <timer name='hypervclock' present='yes'/>
</clock>
```

17.2.2.2. Luces configurables de Hyper-V

Puede configurar ciertas características de Hyper-V para optimizar las VM de Windows. La siguiente tabla proporciona información sobre estas características configurables de Hyper-V y sus valores.

Tabla 17.1. Funciones configurables de Hyper-V

Ilustración	Descripción	Valores
emcs	Implementa el protocolo paravirtualizado entre los hipervisores LO (KVM) y L1 (Hyper-V), que permite salidas L2 más rápidas al hipervisor. Esta función es exclusiva de los procesadores Intel.	encendido, apagado
frecuencias	Activa los registros específicos de la máquina de frecuencia Hyper-V (MSR).	encendido, apagado
ipi	Activa el soporte de interrupciones paravirtuales entre procesadores (IPI).	encendido, apagado

Ilustración	Descripción	Valores
no-nonarch-coresharing	Notifica al sistema operativo invitado que los procesadores virtuales nunca compartirán un núcleo físico a menos que sean reportados como hilos SMT hermanos. Esta información es requerida por los huéspedes de Windows y Hyper-V para mitigar adecuadamente las vulnerabilidades de la CPU relacionadas con el multihilo simultáneo (SMT).	on, off, auto
reencuentro con la luz	Notifica cuando hay un cambio de frecuencia del contador de tiempo (TSC) que sólo se produce durante la migración. También permite que el huésped siga utilizando la frecuencia antigua hasta que esté listo para cambiar a la nueva.	encendido, apagado
relajado	Desactiva un chequeo de sanidad de Windows que comúnmente resulta en un BSOD cuando la VM está corriendo en un host muy cargado. Esto es similar a la opción del kernel de Linux <code>no_timer_check</code> , que se activa automáticamente cuando Linux se ejecuta en KVM.	encendido, apagado
reiniciar	Activa el reinicio de Hyper-V.	encendido, apagado
tiempo de ejecución	Establece el tiempo que el procesador dedica a la ejecución del código del huésped, y en nombre del código del huésped.	encendido, apagado

Ilustración	Descripción	Valores
spinlock	<ul style="list-style-type: none"> ● Utilizado por el sistema operativo de una VM para notificar a Hyper-V que el procesador virtual que llama está intentando adquirir un recurso que potencialmente está en manos de otro procesador virtual dentro de la misma partición. ● Utilizado por Hyper-V para indicar al sistema operativo de la máquina virtual el número de veces que se debe intentar una adquisición de spinlock antes de indicar una situación de spin excesivo a Hyper-V. 	encendido, apagado
stimer	Activa los temporizadores sintéticos para los procesadores virtuales. Tenga en cuenta que algunas versiones de Windows vuelven a utilizar HPET (o incluso RTC cuando HPET no está disponible) cuando no se proporciona esta iluminación, lo que puede conducir a un consumo significativo de la CPU, incluso cuando la CPU virtual está inactiva.	encendido, apagado
stimer-direct	Habilita los temporizadores sintéticos cuando un evento de expiración se entrega a través de una interrupción normal.	encendido, apagado.
sincronización	Junto con el estimulador, activa el temporizador sintético. Windows 8 utiliza esta función en modo periódico.	encendido, apagado

Ilustración	Descripción	Valores
tiempo	Activa las siguientes fuentes de reloj específicas de Hyper-V disponibles para la VM, <ul style="list-style-type: none"> ● Fuente de reloj 82 Hyper-V basada en MSR (HV_X64_MSR_TIME_REFERENCE_COUNT, 0x40000020) ● Página de referencia TSC 83 que se habilita mediante MSR (HV_X64_MSR_REFERENCE_TSC, 0x40000021) 	encendido, apagado
tlbflush	Vacía el TLB de los procesadores virtuales.	encendido, apagado
vapic	Activa el APIC virtual, que proporciona un acceso MSR acelerado a los registros del Controlador de Interrupciones Programable Avanzado (APIC) de alto uso y mapeados en memoria.	encendido, apagado
vendor_id	Establece el identificador de proveedor de Hyper-V.	<ul style="list-style-type: none"> ● encendido, apagado ● Valor de identificación - cadena de hasta 12 caracteres
vpindex	Activa el índice del procesador virtual.	encendido, apagado

17.2.3. Configuración de los parámetros del controlador NetKVM

Una vez instalado el controlador NetKVM, puede configurarlo para que se adapte mejor a su entorno. Los parámetros enumerados en esta sección pueden configurarse mediante el Administrador de dispositivos de Windows (devmgmt.msc).



IMPORTANTE


La modificación de los parámetros del controlador hace que Windows recargue dicho controlador. Esto interrumpe la actividad de red existente.

Requisitos previos

- El controlador NetKVM está instalado en la máquina virtual.

Para más información, consulte [Sección 17.2.1, “Instalación de controladores paravirtuales KVM para máquinas virtuales Windows”](#).

Procedimiento


1. Abra el Administrador de dispositivos de Windows.
Para obtener información sobre cómo abrir el Administrador de dispositivos, consulte la documentación de Windows.
2. Localice el adaptador Ethernet Red Hat VirtIO.
 - a. En la ventana del Administrador de dispositivos, haga clic en  junto a Adaptadores de red.
 - b. En la lista de adaptadores de red, haga doble clic en **Red Hat VirtIO Ethernet Adapter**. Se abre la ventana de propiedades del dispositivo.
3. Ver los parámetros del dispositivo.
En la ventana de Propiedades, haga clic en la pestaña **Advanced**.
4. Modificar los parámetros del dispositivo.
 - a. Haga clic en el parámetro que desea modificar. Se muestran las opciones de ese parámetro.
 - b. Modifique las opciones según sea necesario.
Para obtener información sobre las opciones de los parámetros de NetKVM, consulte [Sección 17.2.4, “Parámetros del controlador NetKVM”](#).
 - c. Haga clic en **Aceptar** para guardar los cambios.

17.2.4. Parámetros del controlador NetKVM

La siguiente tabla proporciona información sobre los parámetros configurables de registro del controlador NetKVM.

Tabla 17.2. Parámetros de registro

Parámetro	Descripción 2
Activación del registro	Un valor booleano que determina si el registro está habilitado. El valor por defecto es Enabled.

Parámetro	Descripción 2
Nivel de registro	<p>Un entero que define el nivel de registro. A medida que el entero aumenta, también lo hace la verbosidad del registro.</p> <ul style="list-style-type: none"> ● El valor por defecto es 0 (sólo errores). ● 1-2 añade mensajes de configuración. ● 3-4 añade información sobre el flujo de paquetes. ● 5-6 agrega información de rastreo a nivel de interrupción y DPC. <p> NOTA</p> <p>Los niveles de registro elevados ralentizarán su máquina virtual.</p>

La siguiente tabla proporciona información sobre los parámetros iniciales configurables del controlador NetKVM.

Tabla 17.3. Parámetros iniciales

Parámetro	Descripción
Asignar MAC	Una cadena que define la dirección MAC administrada localmente para la NIC paravirtualizada. No se establece por defecto.
Init.ConnectionRate(Mb)	Un número entero que representa la velocidad de conexión en megabits por segundo. El valor por defecto para Windows 2008 y posteriores es 10G (10.000 megabits por segundo).
Init.Do802.IPQ	Un valor booleano que habilita el soporte de la población y eliminación de etiquetas de prioridad/VLAN. El valor por defecto es Enabled.
Init.MTUSize	Un número entero que define la unidad de transmisión máxima (MTU). El valor por defecto es 1500. Cualquier valor entre 500 y 65500 es aceptable.

Parámetro	Descripción
Init.MaxTxBuffers	<p>Un número entero que representa el número de descriptores de anillo TX que se asignarán.</p> <p>El valor por defecto es 1024.</p> <p>Los valores válidos son: 16, 32, 64, 128, 256, 512 y 1024.</p>
Init.MaxRxBuffers	<p>Un número entero que representa el número de descriptores de anillo RX que se asignarán.</p> <p>El valor por defecto es 256.</p> <p>Los valores válidos son: 16, 32, 64, 128, 256, 512 y 1024.</p>
Descarga.Tx.Checksum	<p>Especifica el modo de descarga de la suma de comprobación de TX.</p> <p>En Red Hat Enterprise Linux 8, los valores válidos para este parámetro son:</p> <ul style="list-style-type: none"> * All (por defecto), que permite la descarga de sumas de comprobación IP, TCP y UDP tanto para IPv4 como para IPv6 * TCP/UDP(v4,v6) que permite la descarga de sumas de comprobación TCP y UDP tanto para IPv4 como para IPv6 * TCP/UDP(v4) que permite la descarga de sumas de comprobación TCP y UDP sólo para IPv4 * TCP(v4), que permite la descarga de sumas de comprobación TCP sólo para IPv4

17.2.5. Optimización de los procesos de fondo en las máquinas virtuales de Windows

Para optimizar el rendimiento de una máquina virtual (VM) que ejecuta un sistema operativo Windows, puede configurar o desactivar una serie de procesos de Windows.



AVISO

Algunos procesos pueden no funcionar como se espera si se cambia su configuración.

Procedimiento

Puede optimizar sus máquinas virtuales de Windows realizando cualquier combinación de lo siguiente:

- Retire los dispositivos no utilizados, como los USB o los CD-ROM, y desactive los puertos.
- Desactive la actualización automática de Windows Update. Para obtener más información sobre cómo hacerlo, consulte [Configuración de la directiva de grupo para actualizaciones automáticas](#) o [Configuración de Windows Update para empresas](#).
Tenga en cuenta que Windows Update es esencial para instalar las últimas actualizaciones y hotfixes de Microsoft. Por ello, Red Hat no recomienda desactivar Windows Update
- Desactive los servicios en segundo plano, como SuperFetch y Windows Search. Para obtener más información sobre cómo detener los servicios, consulte [Desactivar los servicios del sistema](#) o [Detener el servicio](#).
- Desactive **useplatformclock**. Para ello, ejecute el siguiente comando,

```
# bcdedit /set useplatformclock No
```

- Revise y desactive las tareas programadas innecesarias, como la desfragmentación programada del disco. Para obtener más información sobre cómo hacerlo, consulte [Desactivar tareas programadas](#).
- Asegúrate de que los discos no están encriptados.
- Reducir la actividad periódica de las aplicaciones del servidor. Puede hacerlo editando los temporizadores respectivos. Para más información, consulte [Temporizadores multimedia](#).
- Cierre la aplicación Server Manager en la VM.
- Desactive el software antivirus. Tenga en cuenta que desactivar el antivirus podría comprometer la seguridad de la máquina virtual.
- Desactivar el protector de pantalla.
- Mantenga el sistema operativo Windows en la pantalla de inicio de sesión cuando no lo utilice.

17.3. INFORMACIÓN RELACIONADA

- Para utilizar las utilidades para acceder, editar y crear discos de máquinas virtuales u otras imágenes de disco para una VM de Windows, debe instalar los paquetes **libguestfs-tools** y **libguestfs-winsupport** en la máquina anfitriona.

```
$ sudo yum install libguestfs-tools libguestfs-winsupport
```

CAPÍTULO 18. CREACIÓN DE MÁQUINAS VIRTUALES ANIDADAS

RHEL 8.2 y posteriores proporcionan soporte completo para la función KVM *nested virtualization* en hosts Intel. Esto hace posible que una máquina virtual (también denominada **level 1**, o **L1**) que se ejecuta en un host físico RHEL 8 (**level 0**, o **L0**) actúe como hipervisor y cree sus propias máquinas virtuales (**level 2** o **L2**).

En otras palabras, un host RHEL 8 puede ejecutar máquinas virtuales (VM) L1, y cada una de estas VM puede albergar VM L2 anidadas.

La virtualización anidada no se recomienda en los entornos de usuario de producción, ya que está sujeta a diversas [limitaciones de funcionalidad](#). En su lugar, la virtualización anidada está pensada principalmente para el desarrollo y las pruebas, para lo cual puede ser útil en una variedad de escenarios, tales como:

- Depuración de hipervisores en un entorno restringido
- Pruebas de despliegues virtuales más grandes en una cantidad limitada de recursos físicos

Es posible crear VMs anidadas en múltiples arquitecturas, pero Red Hat actualmente soporta VMs anidadas sólo [en sistemas Intel](#). Por el contrario, la virtualización anidada en sistemas [AMD](#), [IBM POWER9](#) e [IBM Z](#) sólo se proporciona como una [Muestra de Tecnología](#) y, por lo tanto, no es compatible.

18.1. CREACIÓN DE UNA MÁQUINA VIRTUAL ANIDADADA EN INTEL

Siga los pasos siguientes para activar y configurar la virtualización anidada en un host Intel.

Requisitos previos

- Un host RHEL8 L0 que ejecuta una máquina virtual (VM) L1.
- La CPU del hipervisor debe soportar la virtualización anidada. Para comprobarlo, utilice el comando **cat /proc/cpuinfo** en el hipervisor L0. Si la salida del comando incluye las banderas **vmx** y **ept**, la creación de VMs L2 es posible. Este suele ser el caso de los núcleos Intel Xeon v3 y posteriores.
- Asegúrese de que la virtualización anidada está activada en el host L0:

```
# cat /sys/module/kvm_intel/parameters/nested
```

- Si el comando devuelve **1**, la función está activada, y puede iniciar el procedimiento siguiente..
- Si el comando devuelve **0** o **N** pero su sistema admite la virtualización anidada, siga los siguientes pasos para activar la función.
 - i. Descargue el módulo **kvm_intel**:

```
# modprobe -r kvm_intel
```

- ii. Activar la función de anidamiento:


```
# modprobe kvm_intel nested=1
```

- iii. La función de anidamiento está ahora habilitada, pero sólo hasta el próximo reinicio del host L0. Para habilitarla permanentemente, añada la siguiente línea al archivo `/etc/modprobe.d/kvm.conf`:

```
options kvm_intel nested=1
```

Procedimiento

1. Configure su VM L1 para la virtualización anidada.
 - a. Abra la configuración XML de la VM. El siguiente ejemplo abre la configuración de la VM *Intel-L1*:

```
# virsh edit Intel-L1
```

- b. Añade la siguiente línea a la configuración:

```
<cpu mode='host-passthrough'/>
```

Si el archivo de configuración XML de la máquina virtual ya contiene un elemento `<cpu>`, vuelva a escribirlo.

2. Cree una VM L2 dentro de la VM L1. Para ello, siga el mismo procedimiento que al [crear la VM L1](#).

18.2. CREACIÓN DE UNA MÁQUINA VIRTUAL ANIDADADA EN AMD

Siga los pasos siguientes para activar y configurar la virtualización anidada en un host AMD.



AVISO

La virtualización anidada se proporciona actualmente sólo como una [Muestra de Tecnología](#) en la arquitectura AMD64, y por lo tanto no está soportada.

Requisitos previos

- Un host RHEL8 L0 que ejecuta una máquina virtual (VM) L1.
- La CPU del hipervisor debe soportar la virtualización anidada. Para comprobarlo, utilice el comando `cat /proc/cpuinfo` en el hipervisor L0. Si la salida del comando incluye las banderas `svm` y `npt`, la creación de VMs L2 es posible. Este suele ser el caso de los núcleos AMD EPYC y posteriores.
- Asegúrese de que la virtualización anidada está activada en el host L0:

```
# cat /sys/module/kvm_amd/parameters/nested
```

- Si el comando devuelve **Y** o **1**, la función está activada, y puede iniciar el procedimiento siguiente..
- Si el comando devuelve **O** o **N**, siga los siguientes pasos para activar la función.
 - i. Detenga todas las máquinas virtuales en ejecución en el host L0.

- ii. Descargue el módulo **kvm_amd**:

```
# modprobe -r kvm_amd
```

- iii. Activar la función de anidamiento:

```
# modprobe kvm_amd nested=1
```

- iv. La función de anidamiento está ahora habilitada, pero sólo hasta el próximo reinicio del host L0. Para habilitarla permanentemente, añada lo siguiente al archivo **/etc/modprobe.d/kvm.conf**:

```
options kvm_amd nested=1
```

Procedimiento

1. Configure su VM L1 para la virtualización anidada.
 - a. Abra la configuración XML de la VM. El siguiente ejemplo abre la configuración de la VM *AMD-L1*:

```
# virsh edit AMD-L1
```

- b. Configure la CPU de la máquina virtual para que utilice el modo **host-passthrough**.

```
<cpu mode='host-passthrough'/>
```

Si necesita que la VM utilice una CPU específica en lugar de **host-passthrough**, añada una línea **<feature policy='require' name='vmx'/>** a la configuración de la CPU. Por ejemplo:

```
<cpu mode='custom' match='exact' check='partial'>
<model fallback='allow'>Haswell-noTSX</model>
<feature policy='require' name='vmx'/>
```

2. Cree una VM L2 dentro de la VM L1. Para ello, siga el mismo procedimiento que al [crear la VM L1](#).

18.3. CREACIÓN DE UNA MÁQUINA VIRTUAL ANIDADA EN IBM Z

Siga los pasos siguientes para habilitar y configurar la virtualización anidada en un host IBM Z.



AVISO

La virtualización anidada se proporciona actualmente sólo como una [Muestra de Tecnología](#) en la arquitectura IBM Z, y por lo tanto no está soportada.

Requisitos previos

- Un host RHEL8 L0 que ejecuta una máquina virtual (VM) L1.
- La CPU del hipervisor debe soportar la virtualización anidada. Para comprobarlo, utilice el comando **cat /proc/cpuinfo** en el hipervisor L0. Si la salida del comando incluye la bandera **svm**, la creación de VMs L2 es posible.
- Asegúrese de que la virtualización anidada está activada en el host L0:

```
# cat /sys/module/kvm/parameters/nested
```

- Si el comando devuelve **Y** o **1**, la función está activada, y puede iniciar el procedimiento siguiente..
- Si el comando devuelve **O** o **N**, siga los siguientes pasos para activar la función.
 - i. Detenga todas las máquinas virtuales en ejecución en el host L0.

- ii. Descargue el módulo **kvm**:

```
# modprobe -r kvm
```

- iii. Activar la función de anidamiento:

```
# modprobe kvm nested=1
```

- iv. La función de anidamiento está ahora habilitada, pero sólo hasta el próximo reinicio del host L0. Para habilitarla permanentemente, añada la siguiente línea al archivo **/etc/modprobe.d/kvm.conf**:

```
opciones kvm nested=1
```

Procedimiento

- Cree una VM L2 dentro de la VM L1. Para ello, siga el mismo procedimiento que al [crear la VM L1](#).

18.4. CREACIÓN DE UNA MÁQUINA VIRTUAL ANIDADADA EN IBM POWER9

Siga los pasos siguientes para habilitar y configurar la virtualización anidada en un host IBM POWER9.



AVISO

La virtualización anidada se proporciona actualmente sólo como una [Muestra de Tecnología](#) en la arquitectura IBM POWER9, y por lo tanto no es compatible. Además, la creación de máquinas virtuales (VM) anidadas no es posible en versiones anteriores de los sistemas IBM POWER, como IBM POWER8.

Requisitos previos

- Un host L0 RHEL8 está ejecutando una VM L1. La VM L1 está utilizando RHEL 8 como sistema operativo invitado.
- La virtualización anidada está habilitada en el host L0:

```
# cat /sys/module/kvm_hv/parameters/nested
```

- Si el comando devuelve **Y** o **1**, la función está activada y puede iniciar el procedimiento que se indica a continuación.
- Si el comando devuelve **O** o **N**, siga los siguientes pasos para activar la función:

- i. Detenga todas las máquinas virtuales en ejecución en el host L0.

- ii. Descargue el módulo **kvm**:

```
# modprobe -r kvm_hv
```

- iii. Activar la función de anidamiento:

```
# modprobe kvm_hv nested=1
```

- iv. La función de anidamiento está ahora habilitada, pero sólo hasta el próximo reinicio del host L0. Para habilitarla permanentemente, añada la siguiente línea al archivo **/etc/modprobe.d/kvm.conf**:

```
opciones kvm_hv nested=1
```

Procedimiento

1. Para garantizar que la L1 VM pueda crear L2 VMs, añada el parámetro **cap-nested-hv** al tipo de máquina de la L1 VM. Para ello, utilice el comando **virsh edit** para modificar la configuración XML de la L1 VM, y la siguiente línea a la sección **<features>**:

```
<nested-hv state='on'/>
```

2. Cree una VM L2 dentro de la VM L1. Para ello, siga el mismo procedimiento que al [crear la VM L1](#). Para mejorar significativamente el rendimiento de las VMs L2, Red Hat recomienda añadir el parámetro `cap-nested-hv`` a las configuraciones XML de las VMs L2 también. Para obtener instrucciones, consulte el paso anterior.

Información adicional

- Tenga en cuenta que el uso de IBM POWER8 como arquitectura para la L2 VM actualmente no funciona.

18.5. RESTRICCIONES Y LIMITACIONES DE LA VIRTUALIZACIÓN ANIDADADA

Tenga en cuenta las siguientes restricciones cuando utilice la virtualización anidada.

Supported architectures

- El host L0 debe ser un sistema Intel, AMD, IBM POWER9 o IBM Z. La virtualización anidada no funciona actualmente en otras arquitecturas. Además, Red Hat actualmente sólo soporta Intel como host para máquinas virtuales anidadas (VMs), y todas las demás arquitecturas se proporcionan sólo como [Technology Previews](#).

Supported guest operating systems

- Para que la virtualización anidada sea compatible, debe utilizar los siguientes sistemas operativos invitados (OS):
 - En **L0 host** - RHEL 8.2 y posterior
 - En **L1 VMs** - RHEL 7.8 y posterior, o RHEL 8.2 y posterior



NOTA

Este soporte no se aplica al uso de ofertas de virtualización basadas en RHEL 7 y RHEL 8 en VMs L1. Estas incluyen:

- Virtualización de Red Hat
 - Red Hat OpenStack Platform
 - Virtualización de OpenShift
- En la página web **L2 VMs** - debe utilizar uno de los siguientes sistemas operativos:
 - RHEL 7.8 y posteriores
 - RHEL 8.2 y posterior
 - Microsoft Windows Server 2016
 - Microsoft Windows Server 2019
- Además, en IBM POWER9, la virtualización anidada actualmente sólo funciona bajo las siguientes circunstancias:
 - Tanto el host L0 como la VM L1 utilizan RHEL 8
 - La L2 VM utiliza RHEL 8, o RHEL 7 con un kernel **rhel-alt**.
 - La L1 VM y la L2 VM no se ejecutan en modo de compatibilidad con POWER8.

Non-KVM hypervisors

- Cuando se utiliza una VM RHEL 8 L1 en un hipervisor no KVM L0, como VMware ESXi o Amazon Web Services (AWS), la creación de VMs L2 en el SO huésped RHEL 8 puede funcionar, pero no es compatible.

Feature limitations

- El uso de VMs L2 como hipervisores y la creación de invitados L3 no se ha probado adecuadamente y no se espera que funcione.
- Actualmente, la migración de máquinas virtuales no funciona en los sistemas AMD si se ha activado la virtualización anidada en el host L0.
- En un sistema IBM Z, el almacenamiento de respaldo de páginas enormes y la virtualización anidada no pueden utilizarse al mismo tiempo.

```
# modprobe kvm hpage=1 nested=1
modprobe: ERROR: could not insert 'kvm': Invalid argument
# dmesg |tail -1
[90226.508366] kvm-s390: A KVM host that supports nesting cannot back its KVM guests
with huge pages
```

- Algunas funciones disponibles en el host L0 pueden no estar disponibles para el hipervisor L1. Por ejemplo, en el hardware IBM POWER 9, el motor de virtualización de interrupciones externas (XIVE) no funciona. Sin embargo, las máquinas virtuales L1 pueden utilizar el controlador de interrupción XIVE emulado para lanzar máquinas virtuales L2.

CAPÍTULO 19. DIAGNÓSTICO DE LOS PROBLEMAS DE LAS MÁQUINAS VIRTUALES

Al trabajar con máquinas virtuales (VM), puede encontrar problemas con diferentes niveles de gravedad. Algunos problemas pueden tener una solución rápida y fácil, mientras que para otros, es posible que tenga que capturar datos y registros relacionados con las VM para informar o diagnosticar los problemas.

Las siguientes secciones proporcionan información detallada sobre la generación de registros y el diagnóstico de algunos problemas comunes de las máquinas virtuales, así como sobre la notificación de estos problemas.

19.1. GENERACIÓN DE REGISTROS DE DEPURACIÓN DE MÁQUINAS VIRTUALES

Para diagnosticar los problemas de la máquina virtual (VM), es útil generar y revisar los registros de depuración. Adjuntar los registros de depuración también es útil a la hora de solicitar soporte para resolver problemas relacionados con la VM.

Las siguientes secciones explican [qué son los registros de depuración](#), cómo se pueden [configurar para que sean persistentes, habilitarlos durante el tiempo de ejecución](#) y [adjuntarlos](#) al informar de problemas.

19.1.1. Entender los registros de depuración de la máquina virtual

Los registros de depuración son archivos de texto que contienen datos sobre eventos que ocurren durante el tiempo de ejecución de la máquina virtual (VM). Los registros proporcionan información sobre funcionalidades fundamentales del lado del servidor, como las bibliotecas del host y el servicio **libvirtd**. Los archivos de registro también contienen la salida de error estándar (**stderr**) de todas las VMs en ejecución.

El registro de depuración no está habilitado por defecto y tiene que ser habilitado cuando libvirt se inicia. Puede habilitar el registro para una sola sesión o [de forma persistente](#). También puede habilitar el registro cuando una sesión del demonio **libvirtd** ya se está ejecutando [modificando la configuración del tiempo de ejecución del demonio](#).

[Adjuntar los registros de depuración de libvirt](#) también es útil cuando se solicita apoyo con un problema de VM.

19.1.2. Activación de la configuración persistente para los registros de depuración de la máquina virtual

Puede configurar el registro de depuración de la máquina virtual (VM) para que se active automáticamente cada vez que se inicie libvirt editando el archivo de configuración **libvirtd.conf** que se encuentra en el directorio **/etc/libvirt**.

Procedimiento

1. Abra el archivo **libvirtd.conf** en un editor.
2. Sustituya o ajuste los filtros según sus necesidades.
Fijar el valor del filtro en:
 - **1**: registra todos los mensajes generados por libvirt.

- **2:** registra toda la información no relacionada con la depuración.
- **3:** registra todos los mensajes de advertencia y error. Este es el valor por defecto.
- **4:** registra sólo los mensajes de error.

Por ejemplo, el siguiente comando:

- Registra todos los mensajes de error y advertencia de las capas **remote**, **util.json**, y **rpc**
- Registra sólo los mensajes de error de la capa **event**.
- Guarda los registros filtrados en **/var/log/libvirt/libvirtd.log**

```
log_filters="3:remote 4:event 3:util.json 3:rpc"
log_outputs="1:file:/var/log/libvirt/libvirtd.log"
```

3. Guardar y salir.
4. Reinicie el servicio **libvirtd**.

```
$ systemctl restart libvirtd.service
```

19.1.3. Activación de los registros de depuración de la máquina virtual durante el tiempo de ejecución

Puede modificar la configuración de tiempo de ejecución del demonio libvirt para habilitar los registros de depuración y guardarlos en un archivo de salida.

Esto es útil cuando no es posible reiniciar **libvirtd** porque el reinicio soluciona el problema, o porque hay otro proceso, como la migración o la copia de seguridad, ejecutándose al mismo tiempo. Modificar la configuración en tiempo de ejecución también es útil si quieres probar un comando sin editar los archivos de configuración o reiniciar el demonio.

Requisitos previos

- Asegúrese de que el paquete **libvirt-admin** está instalado.

Procedimiento

1. **Optional:** Realiza una copia de seguridad del conjunto de filtros de registro activos.

```
# virt-admin daemon-log-filters >> virt-filters-backup
```



NOTA

Se recomienda hacer una copia de seguridad del conjunto de filtros activos para poder restaurarlos después de generar los registros. Si no restaura los filtros, los mensajes seguirán registrándose, lo que puede afectar al rendimiento del sistema.

2. Utilice la utilidad **virt-admin** para activar la depuración y establecer los filtros según sus necesidades.

Fijar el valor del filtro en:

- **1:** registra todos los mensajes generados por libvirt.
- **2:** registra toda la información no relacionada con la depuración.
- **3:** registra todos los mensajes de advertencia y error. Este es el valor por defecto.
- **4:** registra sólo los mensajes de error.

Por ejemplo, el siguiente comando:

- Registra todos los mensajes de error y advertencia de las capas **remote**, **util.json**, y **rpc**
- Registra sólo los mensajes de error de la capa **event**.

```
# virt-admin daemon-log-filters "3:remote 4:event 3:util.json 3:rpc"
```

3. Utilice la utilidad **virt-admin** para guardar los registros en un archivo o directorio específico. Por ejemplo, el siguiente comando guarda la salida del registro en el archivo **libvirt.log** en el directorio **/var/log/libvirt/**.

```
# virt-admin daemon-log-outputs "1:file:/var/log/libvirt/libvirtd.log"
```

4. **Optional:** También puede eliminar los filtros para generar un archivo de registro que contenga toda la información relacionada con la VM. Sin embargo, no se recomienda ya que este archivo puede contener una gran cantidad de información redundante producida por los módulos de libvirt.
 - Utilice la utilidad **virt-admin** para especificar un conjunto vacío de filtros.

```
# virt-admin daemon-log-filters
Logging filters:
```

5. **Optional:** Restaurar los filtros a su estado original utilizando el archivo de copia de seguridad. Realice el segundo paso con los valores guardados para restaurar los filtros.

19.1.4. Adjuntar registros de depuración de máquinas virtuales a las solicitudes de asistencia

Es posible que tenga que solicitar soporte adicional para diagnosticar y resolver los problemas de la máquina virtual (VM). Se recomienda encarecidamente adjuntar los registros de depuración a la solicitud de soporte para garantizar que el equipo de soporte tenga acceso a toda la información que necesita para proporcionar una rápida resolución del problema relacionado con la VM.

Procedimiento

- Para informar de un problema y solicitar asistencia, [abra un caso de asistencia](#) .
- Basándose en los problemas encontrados, adjunte los siguientes registros junto con su informe:
 - En caso de problemas con el servicio libvirt, adjunte el archivo **/var/log/libvirt/libvirtd.log** del host.
 - Para problemas con una VM específica, adjunte su respectivo archivo de registro.

Por ejemplo, para la máquina virtual *testguest1*, adjunte el archivo **testguest1.log**, que se encuentra en `/var/log/libvirt/qemu/testguest1.log`.

Recursos adicionales

- Para más información sobre cómo adjuntar archivos de registro, consulte [¿Cómo proporcionar archivos al Soporte de Red Hat?](#)

19.2. DUMPING DE UN NÚCLEO DE MÁQUINA VIRTUAL

Para analizar por qué una máquina virtual (VM) se bloqueó o funcionó mal, puede volcar el núcleo de la VM a un archivo en el disco para su posterior análisis y diagnóstico.

Esta sección proporciona una breve [introducción al volcado de núcleos](#) y explica cómo se puede [volcar un núcleo de VM](#) a un archivo específico.

19.2.1. Cómo funciona el volcado del núcleo de la máquina virtual

Una máquina virtual (VM) requiere numerosos procesos en ejecución para funcionar de forma precisa y eficiente. En algunos casos, una VM en ejecución puede terminar inesperadamente o funcionar mal mientras se está utilizando. El reinicio de la VM puede hacer que los datos se restablezcan o se pierdan, lo que dificulta el diagnóstico del problema exacto que ha provocado el fallo de la VM.

En estos casos, puede utilizar la utilidad **virsh dump** para guardar (o *dump*) el núcleo de una VM en un archivo antes de reiniciar la VM. El archivo de volcado del núcleo contiene una imagen de memoria física sin procesar de la VM que contiene información detallada sobre la VM. Esta información puede ser utilizada para diagnosticar problemas de la VM, ya sea manualmente, o utilizando una herramienta como la utilidad **crash**.

Recursos adicionales

- Para obtener información sobre el uso de la utilidad **crash**, consulte la [página man de crash](#) y la [página principal de la utilidad crash](#).

19.2.2. Creación de un archivo de volcado del núcleo de la máquina virtual

Un volcado de núcleo de máquina virtual (VM) contiene información detallada sobre el estado de una VM en un momento dado. Esta información, esencialmente una instantánea de la VM, es extremadamente útil para detectar problemas en caso de un mal funcionamiento de la VM o un cierre repentino de la misma.

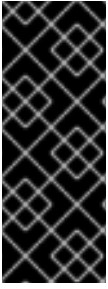
Requisitos previos

- Asegúrese de que tiene suficiente espacio en el disco para guardar el archivo. Tenga en cuenta que el espacio ocupado por la VM depende de la cantidad de RAM asignada a la VM.

Procedimiento

- Utilice la utilidad **virsh dump**.
Por ejemplo, el siguiente comando vuelca los núcleos de la VM **lander1**, su memoria y el archivo de registro común de la CPU a **gargantua.file** en el directorio **/core/file**.

```
# virsh dump lander1 /core/file/gargantua.file --memory-only
Domain lander1 dumped to /core/file/gargantua.file
```



IMPORTANTE

La utilidad **crash** ya no soporta el formato de archivo por defecto del comando `virsh dump`. Para analizar un archivo de volcado del núcleo utilizando **crash**, debe crear el archivo utilizando la opción **--memory-only**.

Además, debe utilizar la opción **--memory-only** cuando cree un archivo de volcado de núcleo para adjuntarlo a un Red Hat Support Case.

Recursos adicionales

- Para otros argumentos de **virsh dump**, utilice **virsh dump --help** o consulte la página de manual **virsh**.
- Para obtener información sobre cómo abrir un caso de asistencia, consulte [Abrir un caso de asistencia](#)

19.3. RASTREO DE PROCESOS DE MÁQUINAS VIRTUALES

Cuando un proceso relacionado con una máquina virtual (VM) funciona mal, se puede utilizar el comando **gstack** junto con el identificador de proceso (PID) para generar un seguimiento de la pila de ejecución del proceso que funciona mal. Si el proceso forma parte de un grupo de hilos, también se rastrean todos los hilos.

Requisitos previos

- Asegúrese de que el paquete **GDB** está instalado.
Para más detalles sobre la instalación de **GDB** y los componentes disponibles, vea [Instalación del depurador GNU](#).
- Asegúrese de conocer el PID de los procesos que desea rastrear.
Puede encontrar el PID utilizando el comando **pgrep** seguido del nombre del proceso. Por ejemplo:

```
# pgrep libvirt
22014
22025
```

Procedimiento

- Utilice la utilidad **gstack** seguida del PID del proceso que desea rastrear.
Por ejemplo, el siguiente comando rastrea el proceso libvirt con el PID 22014.

```
# gstack 22014
Thread 3 (Thread 0x7f33edaf7700 (LWP 22017)):
#0 0x00007f33f81aef21 in poll () from /lib64/libc.so.6
#1 0x00007f33f89059b6 in g_main_context_iterate.isra () from /lib64/libglib-2.0.so.0
#2 0x00007f33f8905d72 in g_main_loop_run () from /lib64/libglib-2.0.so.0
...
```

Recursos adicionales

- Para otros argumentos de **gstack**, consulte la página de manual **gstack**.

- Para más información sobre **GDB**, véase [GNU Debugger](#).

19.4. RECURSOS ADICIONALES PARA INFORMAR DE LOS PROBLEMAS DE LAS MÁQUINAS VIRTUALES Y PROPORCIONAR REGISTROS

Para solicitar ayuda y apoyo adicionales, puede:

- Lance una solicitud de servicio utilizando la opción de línea de comandos **redhat-support-tool**, la interfaz de usuario del Portal de Red Hat, o varios métodos diferentes utilizando FTP.
 - Para informar de problemas y solicitar asistencia, consulte [Abrir un caso de asistencia](#) .
- Cargue el informe SOS y los archivos de registro cuando envíe una solicitud de servicio. Esto asegura que el ingeniero de soporte de Red Hat tenga toda la información de diagnóstico necesaria para su referencia.
 - Para más información sobre los informes SOS, consulte [¿Qué es un informe SOS y cómo crear uno en Red Hat Enterprise Linux?](#)
 - Para obtener información sobre cómo adjuntar archivos de registro, consulte [¿Cómo proporcionar archivos al Soporte de Red Hat?](#)

CAPÍTULO 20. COMPATIBILIDAD Y LIMITACIONES DE LAS FUNCIONES EN LA VIRTUALIZACIÓN DE RHEL 8

Este documento proporciona información sobre el soporte y las restricciones de las características de la virtualización de Red Hat Enterprise Linux 8 (RHEL 8).

20.1. CÓMO FUNCIONA EL SOPORTE DE VIRTUALIZACIÓN DE RHEL 8

Un conjunto de limitaciones de soporte se aplica a la virtualización en Red Hat Enterprise Linux 8 (RHEL 8). Esto significa que cuando se utilizan ciertas características o se excede una cierta cantidad de recursos asignados cuando se utilizan máquinas virtuales en RHEL 8, Red Hat no dará soporte a estos huéspedes a menos que tenga un plan de suscripción específico.

Las características listadas en [Sección 20.2, "Características recomendadas en la virtualización de RHEL 8"](#) han sido probadas y certificadas por Red Hat para funcionar con el hipervisor KVM en un sistema RHEL 8. Por lo tanto, son totalmente compatibles y se recomienda su uso en la virtualización en RHEL 8.

Las funcionalidades listadas en [Sección 20.3, "Características no soportadas en la virtualización de RHEL 8"](#) pueden funcionar, pero no están soportadas y no están pensadas para su uso en RHEL 8. Por lo tanto, Red Hat recomienda encarecidamente no utilizar estas funcionalidades en RHEL 8 con KVM.

[Sección 20.4, "Límites de asignación de recursos en la virtualización de RHEL 8"](#) enumera la cantidad máxima de recursos específicos soportados en un huésped KVM en RHEL 8. Los huéspedes que exceden estos límites no son soportados por Red Hat.

Además, a menos que se indique lo contrario, todas las características y soluciones utilizadas por la documentación para la virtualización de RHEL 8 son compatibles. Sin embargo, algunas de ellas no han sido completamente probadas y, por tanto, pueden no estar totalmente optimizadas.



IMPORTANTE

Muchas de estas limitaciones no se aplican a otras soluciones de virtualización proporcionadas por Red Hat, como Red Hat Virtualization (RHV), OpenShift Virtualization o Red Hat OpenStack Platform (RHOSP).

20.2. CARACTERÍSTICAS RECOMENDADAS EN LA VIRTUALIZACIÓN DE RHEL 8

Las siguientes funciones se recomiendan para su uso con el hipervisor KVM incluido en Red Hat Enterprise Linux 8 (RHEL 8):

Arquitecturas de sistemas de acogida

Red Hat Enterprise Linux con KVM sólo es compatible con las siguientes arquitecturas de host:

- AMD64 e Intel 64
- IBM Z - Sistemas IBM z13 y posteriores
- IBM POWER8
- IBM POWER9



NOTA

La documentación de RHEL 8 describe principalmente las características y el uso de AMD64 e Intel 64. Para obtener información sobre la especificidad del uso de la virtualización de RHEL 8 en diferentes arquitecturas, consulte:

- [Capítulo 3, Introducción a la virtualización en IBM POWER](#)
- [Capítulo 4, Introducción a la virtualización en IBM Z](#).

Sistemas operativos invitados

Red Hat soporta máquinas virtuales KVM que utilizan los siguientes sistemas operativos:

- Red Hat Enterprise Linux 6 y posteriores
- Microsoft Windows 10 y posteriores
- Microsoft Windows Server 2016 y posterior

Q35 invitados

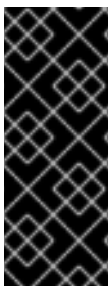
El tipo de máquina recomendado para las máquinas virtuales KVM es QEMU Q35, que emula el chipset ICH9.

Recursos adicionales

- Para obtener información sobre los tipos de sistemas operativos invitados y las funciones no admitidas en la virtualización de RHEL 8, consulte [Sección 20.3, "Características no soportadas en la virtualización de RHEL 8"](#).
- Para obtener información sobre las cantidades máximas admitidas de recursos que pueden asignarse a una máquina virtual, consulte [Sección 20.4, "Límites de asignación de recursos en la virtualización de RHEL 8"](#).

20.3. CARACTERÍSTICAS NO SOPORTADAS EN LA VIRTUALIZACIÓN DE RHEL 8

Las siguientes funciones no son compatibles con el hipervisor KVM incluido en Red Hat Enterprise Linux 8 (RHEL 8):



IMPORTANTE

Muchas de estas limitaciones pueden no aplicarse a otras soluciones de virtualización proporcionadas por Red Hat, como Red Hat Virtualization (RHV), OpenShift Virtualization o Red Hat OpenStack Platform (RHOSP).

Las características soportadas por RHV 4.2 y posteriores, o RHOSP 13 y posteriores, se describen como tales en los siguientes párrafos.

Sistemas operativos invitados

Las máquinas virtuales (VM) KVM que utilizan los siguientes sistemas operativos invitados en un host RHEL 8 no son compatibles:

- Microsoft Windows 8.1 y anteriores

- Microsoft Windows Server 2012 y anteriores
- macOS
- Solaris para sistemas x86
- Cualquier sistema operativo anterior a 2009

Para una lista de sistemas operativos huéspedes soportados en hosts RHEL, vea [Sistemas operativos huéspedes certificados para Red Hat Enterprise Linux con KVM](#).

Para ver una lista de sistemas operativos invitados compatibles con otras soluciones de virtualización proporcionadas por Red Hat, consulte [Sistemas operativos invitados certificados en Red Hat OpenStack Platform y Red Hat Virtualization](#).

Para obtener una lista de los sistemas operativos invitados admitidos específicamente por RHV, consulte [Sistemas operativos invitados admitidos en RHV](#).

vCPU hot unplug

La eliminación de una CPU virtual (vCPU) de una VM en ejecución, también conocida como vCPU hot unplug, no está soportada en RHEL 8.

Tenga en cuenta que las desconexiones en caliente de vCPUs están soportadas en RHV. Para obtener más detalles, consulte [Conexión en caliente de VCPUs](#).

Desenchufe en caliente de la memoria

La eliminación de un dispositivo de memoria conectado a una máquina virtual en ejecución, también conocida como desconexión en caliente de la memoria, no está soportada en RHEL 8.

Tenga en cuenta que la desconexión en caliente de la memoria está soportada en RHV, pero sólo en las máquinas virtuales invitadas que ejecutan RHEL con configuraciones específicas de invitado. Para más detalles, consulte [Desenchufar en caliente la memoria virtual](#).

Estrangulamiento de E/S del lado de QEMU

El uso de la utilidad **virsh blkdeviotune** para configurar los niveles máximos de entrada y salida para las operaciones en el disco virtual, también conocido como estrangulamiento de E/S del lado de QEMU, no es compatible con RHEL 8.

Para configurar el estrangulamiento de E/S en RHEL 8, utilice **virsh blkiotune**. Esto también se conoce como estrangulamiento de E/S del lado de libvirt. Para obtener instrucciones, consulte [Sección 16.4.2, "Estrangulamiento de E/S de disco en máquinas virtuales"](#).

Tenga en cuenta que el estrangulamiento de E/S del lado de QEMU está soportado en RHV. Para más detalles, consulte [Calidad de servicio de almacenamiento](#).

El estrangulamiento de E/S del lado de QEMU también está soportado en RHOSP. Para obtener más detalles, consulte la sección [Configuración de la limitación de recursos en el disco](#) y **Use Quality-of-Service Specifications** en la [Guía de almacenamiento](#) de RHOSP.

Además, OpenShift Virtualization también admite el estrangulamiento de E/S del lado de QEMU.

Migración en vivo del almacenamiento

La migración de una imagen de disco de una VM en ejecución entre hosts no está soportada en RHEL 8.

Tenga en cuenta que la migración de almacenamiento en vivo es compatible con RHV. Para obtener más detalles, consulte [Visión general de la migración de almacenamiento en vivo](#).

La migración en vivo del almacenamiento también es compatible con RHOSP, pero con algunas limitaciones. Para más detalles, consulte [Migrar un volumen](#).

Instantáneas en directo

La creación o carga de una instantánea de una máquina virtual en funcionamiento, también denominada instantánea en vivo, no está soportada en RHEL 8.

Además, tenga en cuenta que las instantáneas de VM no vivas están obsoletas en RHEL 8. Por lo tanto, la creación o carga de una instantánea de una VM apagada es compatible, pero Red Hat recomienda no utilizarla.

Tenga en cuenta que las instantáneas en vivo están soportadas en RHV. Para más detalles, consulte [Instantáneas en vivo en Red Hat Virtualization](#).

Las instantáneas en vivo también son compatibles con RHOSP. Para obtener más detalles, consulte [Importar máquinas virtuales a la nube externa](#).

vhost-user

RHEL 8 no soporta la implementación de una interfaz vHost de espacio de usuario.

Ten en cuenta que vhost-user está soportado en RHOSP, pero sólo para las interfaces **virtio-net**. Para más detalles, véase la [implementación de virtio-net](#) y los [puertos de usuario vhost](#).

Estados de potencia del sistema S3 y S4

La suspensión de una VM a los estados de energía del sistema **Suspend to RAM (S3)** o **Suspend to disk (S4)** no está soportada. Tenga en cuenta que estas funciones están deshabilitadas por defecto, y habilitarlas hará que su VM no sea soportada por Red Hat.

Tenga en cuenta que los estados S3 y S4 tampoco son compatibles actualmente con RHV y RHOSP.

S3-PR en un vDisk multipathed

La reserva persistente SCSI3 (S3-PR) en un vDisk multipathed no está soportada en RHEL 8. En consecuencia, Windows Cluster no es compatible con RHEL 8.

Tenga en cuenta que S3-PR en un vDisk multipathed está soportado en RHV. Por lo tanto, si usted requiere soporte para Windows Cluster, Red Hat recomienda usar RHV como su solución de virtualización. Para más detalles, vea Soporte de cluster [en huéspedes RHV](#).

virtio-crypto

Los controladores para el dispositivo *virtio-crypto* están disponibles en el kernel de RHEL 8.0, por lo que el dispositivo puede ser habilitado en un hipervisor KVM bajo ciertas circunstancias. Sin embargo, el uso del dispositivo *virtio-crypto* en RHEL 8 no está soportado y, por tanto, se desaconseja su uso.

Tenga en cuenta que los dispositivos *virtio-crypto* tampoco son compatibles con RHV o RHOSP.

Copia de seguridad incremental en vivo

La configuración de una copia de seguridad de VM que sólo guarda los cambios de VM desde la última copia de seguridad, también conocida como copia de seguridad incremental en vivo, no está soportada en RHEL 8, y Red Hat desaconseja su uso.

Tenga en cuenta que la copia de seguridad incremental en vivo se proporciona como un Technology Preview en RHV 4.4 y posteriores.

net_failover

El uso del controlador **net_failover** para configurar un mecanismo automatizado de conmutación por error de los dispositivos de red no es compatible con RHEL 8.

Tenga en cuenta que actualmente **net_failover** tampoco es compatible con RHV y RHOSP.

vTPM

No se admite la conexión de dispositivos virtuales Trusted Platform Module (vTPM) a máquinas virtuales alojadas en un sistema RHEL 8.

Tenga en cuenta que actualmente vTPM tampoco es compatible con RHV y RHOSP.

Migración multi-FD

La migración de máquinas virtuales utilizando múltiples descriptores de archivos (FD), también conocida como migración multi-FD, no está soportada en RHEL 8.

Tenga en cuenta que las migraciones multi-FD tampoco están soportadas actualmente en RHV o RHOSP.

virtiofs

Compartir archivos entre el host y sus máquinas virtuales utilizando el sistema de archivos **virtiofs** no es compatible con RHEL8.

Tenga en cuenta que el uso de **virtiofs** tampoco está soportado actualmente en RHV o RHOSP.

Dispositivos NVMe

No se admite la conexión de dispositivos de memoria no volátil exprés (NVMe) a las máquinas virtuales instaladas en RHEL 8.

Tenga en cuenta que adjuntar dispositivos **NVMe** a las máquinas virtuales tampoco está soportado actualmente en RHV o RHOSP.

TCG

QEMU y libvirt incluyen un modo de traducción dinámica que utiliza el Tiny Code Generator (TCG) de QEMU. Este modo no requiere soporte de virtualización de hardware. Sin embargo, el TCG no está soportado por Red Hat.

Los huéspedes basados en TCG pueden ser reconocidos examinando su configuración XML, por ejemplo usando el comando "virsh dumpxml".

- El archivo de configuración de un huésped TCG contiene la siguiente línea:

```
<dominio type='qemu'>
```

- El archivo de configuración de un huésped KVM contiene la siguiente línea:

```
<dominio type='kvm'>
```

Recursos adicionales

- Para obtener información sobre los tipos de SO invitados compatibles y las características recomendadas en la virtualización de RHEL 8, consulte [Sección 20.2, "Características recomendadas en la virtualización de RHEL 8"](#).
- Para obtener información sobre las cantidades máximas admitidas de recursos que pueden asignarse a una máquina virtual, consulte [Sección 20.4, "Límites de asignación de recursos en la virtualización de RHEL 8"](#).

20.4. LÍMITES DE ASIGNACIÓN DE RECURSOS EN LA VIRTUALIZACIÓN DE RHEL 8

Los siguientes límites se aplican a los recursos virtualizados que pueden ser asignados a una sola máquina virtual KVM (VM) en un host Red Hat Enterprise Linux 8 (RHEL 8).



IMPORTANTE

Muchas de estas limitaciones no se aplican a otras soluciones de virtualización proporcionadas por Red Hat, como Red Hat Virtualization (RHV), OpenShift Virtualization o Red Hat OpenStack Platform (RHOSP).

Máximo de máquinas virtuales por host

Un único host RHEL 8 admite hasta **4** VMs ejecutadas al mismo tiempo. Este límite no se aplica a las máquinas virtuales en RHV y RHOSP.

Máximo de vCPUs por VM

RHEL 8 soporta hasta **384** vCPUs asignadas a una sola VM.

Dispositivos PCI por VM

RHEL 8 soporta **32** ranuras de dispositivos PCI por bus VM, y **8** funciones PCI por ranura de dispositivo. Esto da un máximo teórico de 256 funciones PCI por bus cuando se habilitan las capacidades multifunción en la VM, y no se utilizan puentes PCI.

Cada puente PCI añade un nuevo bus, permitiendo potencialmente otras 256 direcciones de dispositivos. Sin embargo, algunos buses no ponen a disposición del usuario las 256 direcciones de dispositivo; por ejemplo, el bus raíz tiene varios dispositivos incorporados que ocupan ranuras.

Dispositivos IDE virtualizados

KVM está limitado a un máximo de **4** dispositivos IDE virtualizados por VM.

20.5. UNA VISIÓN GENERAL DE LA COMPATIBILIDAD CON LAS FUNCIONES DE VIRTUALIZACIÓN

Las siguientes tablas proporcionan información comparativa sobre el estado de soporte de las características de virtualización seleccionadas en RHEL 8 a través de las arquitecturas de sistema soportadas.

Tabla 20.1. Conexión y desconexión en caliente del dispositivo

	Intel 64 y AMD64	IBM Z	IBM POWER
CPU hot plug	Apoyado	Apoyado	Apoyado
CPU hot unplug	<i>UNSUPPORTED</i>	<i>UNSUPPORTED</i>	<i>UNSUPPORTED</i>
Memory hot plug	Apoyado	<i>UNSUPPORTED</i>	Apoyado
Memory hot unplug	<i>UNSUPPORTED</i>	<i>UNSUPPORTED</i>	<i>UNSUPPORTED</i>
PCI hot plug	Apoyado	Soportado [a]	Apoyado
PCI hot unplug	Apoyado	Apoyado [a]	Apoyado
[a] Requiere el uso de virtio-<i>*ccw</i> dispositivos en lugar de virtio-<i>*pci</i>			

Tabla 20.2. Otras características seleccionadas

	Intel 64 y AMD64	IBM Z	IBM POWER
NUMA tuning	Apoyado	<i>UNSUPPORTED</i>	Apoyado
SR-IOV devices	Apoyado	<i>UNSUPPORTED</i>	Apoyado
virt-v2v and p2v	Apoyado	<i>UNSUPPORTED</i>	<i>UNSUPPORTED</i>

Fuentes adicionales

- Para obtener una lista completa de las características no compatibles con las máquinas virtuales en RHEL 8, consulte [Sección 20.3, “Características no soportadas en la virtualización de RHEL 8”](#).
- Para conocer los detalles de la virtualización en la arquitectura IBM Z, consulte [Sección 4.2, “En qué se diferencia la virtualización en IBM Z de la de AMD64 e Intel 64”](#).
- Para conocer los detalles de la virtualización en la arquitectura IBM POWER, consulte [Sección 3.2, “En qué se diferencia la virtualización en IBM POWER de la de AMD64 e Intel 64”](#).