



Red Hat Enterprise Linux 8

Configuración y gestión de clusters de alta disponibilidad

Configuración y gestión del complemento de alta disponibilidad de Red Hat

Red Hat Enterprise Linux 8 Configuración y gestión de clusters de alta disponibilidad

Configuración y gestión del complemento de alta disponibilidad de Red Hat

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_high_availability_clusters.ent file | This material may only be distributed subject to the terms and conditions set forth in the GNU Free Documentation License (GFDL), V1.2 or later (the latest version is presently available at <http://www.gnu.org/licenses/fdl.txt>).

Resumen

Este manual proporciona información sobre la instalación, configuración y gestión del complemento de alta disponibilidad de Red Hat para Red Hat Enterprise Linux 8.

Table of Contents

HACER QUE EL CÓDIGO ABIERTO SEA MÁS INCLUSIVO	8
PROPORCIONAR COMENTARIOS SOBRE LA DOCUMENTACIÓN DE RED HAT	9
CAPÍTULO 1. DESCRIPCIÓN DEL COMPLEMENTO DE ALTA DISPONIBILIDAD	10
1.1. COMPONENTES DEL COMPLEMENTO DE ALTA DISPONIBILIDAD	10
1.2. VISIÓN GENERAL DEL MARCAPASOS	10
1.2.1. Componentes de la arquitectura del marcapasos	10
1.2.2. Herramientas de configuración y gestión	11
1.2.3. Los archivos de configuración del clúster y del marcapasos	11
1.3. VISIÓN GENERAL DE LA ESGRIMA	12
1.4. RESUMEN DEL QUÓRUM	12
1.5. RESUMEN DE RECURSOS	13
1.6. VOLÚMENES LÓGICOS LVM EN UN CLUSTER DE ALTA DISPONIBILIDAD DE RED HAT	13
1.6.1. Elección de HA-LVM o volúmenes compartidos	13
1.6.2. Configuración de volúmenes LVM en un clúster	14
CAPÍTULO 2. CÓMO EMPEZAR CON PACEMAKER	15
2.1. APRENDER A USAR EL MARCAPASOS	15
2.2. APRENDER A CONFIGURAR LA CONMUTACIÓN POR ERROR	19
CAPÍTULO 3. LA INTERFAZ DE LÍNEA DE COMANDOS DE PCS	24
3.1. PANTALLA DE AYUDA DE PCS	24
3.2. VISUALIZACIÓN DE LA CONFIGURACIÓN BRUTA DEL CLÚSTER	24
3.3. GUARDAR UN CAMBIO DE CONFIGURACIÓN EN UN ARCHIVO DE TRABAJO	24
3.4. VISUALIZACIÓN DEL ESTADO DEL CLÚSTER	25
3.5. VISUALIZACIÓN DE LA CONFIGURACIÓN COMPLETA DEL CLÚSTER	26
CAPÍTULO 4. CREACIÓN DE UN CLUSTER DE ALTA DISPONIBILIDAD DE RED HAT CON PACEMAKER	27
4.1. INSTALACIÓN DEL SOFTWARE DEL CLÚSTER	27
4.2. INSTALACIÓN DEL PAQUETE PCP-ZEROCONF (RECOMENDADO)	29
4.3. CREACIÓN DE UN CLÚSTER DE ALTA DISPONIBILIDAD	29
4.4. CREACIÓN DE UN CLUSTER DE ALTA DISPONIBILIDAD CON MÚLTIPLES ENLACES	30
4.5. CONFIGURACIÓN DE LAS VALLAS	31
4.6. COPIA DE SEGURIDAD Y RESTAURACIÓN DE LA CONFIGURACIÓN DE UN CLÚSTER	32
4.7. HABILITACIÓN DE PUERTOS PARA EL COMPLEMENTO DE ALTA DISPONIBILIDAD	33
CAPÍTULO 5. CONFIGURACIÓN DE UN SERVIDOR HTTP APACHE ACTIVO/PASIVO EN UN CLUSTER DE ALTA DISPONIBILIDAD DE RED HAT	36
5.1. CONFIGURACIÓN DE UN VOLUMEN LVM CON UN SISTEMA DE ARCHIVOS EXT4 EN UN CLUSTER PACEMAKER	37
5.2. CONFIGURACIÓN DE UN SERVIDOR HTTP APACHE	38
5.3. CREACIÓN DE LOS RECURSOS Y GRUPOS DE RECURSOS	39
5.4. PROBAR LA CONFIGURACIÓN DE LOS RECURSOS	41
CAPÍTULO 6. CONFIGURACIÓN DE UN SERVIDOR NFS ACTIVO/PASIVO EN UN CLUSTER DE ALTA DISPONIBILIDAD DE RED HAT	43
6.1. REQUISITOS PREVIOS	43
6.2. RESUMEN DEL PROCEDIMIENTO	43
6.3. CONFIGURACIÓN DE UN VOLUMEN LVM CON UN SISTEMA DE ARCHIVOS EXT4 EN UN CLUSTER PACEMAKER	43
6.4. CONFIGURACIÓN DE UN RECURSO COMPARTIDO NFS	45
6.5. CONFIGURACIÓN DE LOS RECURSOS Y DEL GRUPO DE RECURSOS PARA UN SERVIDOR NFS EN UN CLÚSTER	45

6.6. PRUEBA DE LA CONFIGURACIÓN DE LOS RECURSOS NFS	49
6.6.1. Prueba de la exportación NFS	49
6.6.2. Pruebas de conmutación por error	50
CAPÍTULO 7. SISTEMAS DE ARCHIVOS GFS2 EN UN CLÚSTER	52
7.1. CONFIGURACIÓN DE UN SISTEMA DE ARCHIVOS GFS2 EN UN CLÚSTER	52
7.2. MIGRACIÓN DE UN SISTEMA DE ARCHIVOS GFS2 DE RHEL7 A RHEL8	57
CAPÍTULO 8. INTRODUCCIÓN A LA INTERFAZ WEB DE PCSD	59
8.1. INSTALACIÓN DEL SOFTWARE DEL CLÚSTER	59
8.2. CONFIGURACIÓN DE LA INTERFAZ WEB DE PCSD	60
8.3. CREACIÓN DE UN CLÚSTER CON LA INTERFAZ WEB DE PCSD	61
8.3.1. Configuración de las opciones avanzadas de configuración del clúster con la interfaz web de pcsd	61
8.3.2. Establecer los permisos de gestión del clúster	62
8.4. CONFIGURACIÓN DE LOS COMPONENTES DEL CLÚSTER CON LA INTERFAZ WEB DE PCSD	62
8.4.1. Configuración de los nodos del clúster con la interfaz web de pcsd	63
8.4.2. Configuración de los recursos del clúster con la interfaz web de pcsd	63
8.4.3. Configuración de los dispositivos de la valla con la interfaz web de pcsd	64
8.4.4. Configuración de ACLs con la Web UI de pcsd	64
8.4.5. Configuración de las propiedades del clúster con la interfaz web de pcsd	64
8.5. CONFIGURACIÓN DE UNA INTERFAZ WEB DE ALTA DISPONIBILIDAD PCSD	65
CAPÍTULO 9. CONFIGURACIÓN DEL CERCADO EN UN CLUSTER DE ALTA DISPONIBILIDAD DE RED HAT .	66
9.1. VISUALIZACIÓN DE LOS AGENTES DE LA VALLA DISPONIBLES Y SUS OPCIONES	66
9.2. CREACIÓN DE UN DISPOSITIVO DE VALLAS	67
9.3. PROPIEDADES GENERALES DE LOS DISPOSITIVOS DE CERCADO	67
9.4. OPCIONES AVANZADAS DE CONFIGURACIÓN DEL CERCADO	69
9.5. PRUEBA DE UN DISPOSITIVO DE VALLA	81
9.6. CONFIGURACIÓN DE LOS NIVELES DE VALLADO	84
9.7. CONFIGURAR EL CERCADO PARA LAS FUENTES DE ALIMENTACIÓN REDUNDANTES	85
9.8. VISUALIZACIÓN DE LOS DISPOSITIVOS DE VALLAS CONFIGURADOS	86
9.9. MODIFICACIÓN Y SUPRESIÓN DE LOS DISPOSITIVOS DE LA VALLA	86
9.10. CERCAR MANUALMENTE UN NODO DEL CLÚSTER	86
9.11. DESACTIVACIÓN DE UN DISPOSITIVO DE VALLA	87
9.12. IMPEDIR QUE UN NODO UTILICE UN DISPOSITIVO DE VALLA	87
9.13. CONFIGURACIÓN DE ACPI PARA SU USO CON DISPOSITIVOS DE VALLA INTEGRADOS	87
9.13.1. Desactivación de ACPI Soft-Off con la BIOS	88
9.13.2. Desactivación de ACPI Soft-Off en el archivo logind.conf	89
9.13.3. Desactivar completamente ACPI en el archivo GRUB 2	90
CAPÍTULO 10. CONFIGURACIÓN DE LOS RECURSOS DEL CLÚSTER	91
Ejemplos de creación de recursos	91
Borrar un recurso configurado	91
10.1. IDENTIFICADORES DE AGENTES DE RECURSOS	91
10.2. VISUALIZACIÓN DE LOS PARÁMETROS ESPECÍFICOS DE LOS RECURSOS	93
10.3. CONFIGURACIÓN DE LAS OPCIONES META DE LOS RECURSOS	93
10.3.1. Cambiar el valor por defecto de una opción de recurso	138
10.3.2. Cambio del valor por defecto de una opción de recurso para conjuntos de recursos (RHEL 8.3 y posteriores)	138
10.3.3. Visualización de los valores predeterminados de los recursos configurados actualmente	139
10.3.4. Configuración de las opciones meta en la creación de recursos	139
10.4. CONFIGURACIÓN DE LOS GRUPOS DE RECURSOS	140
10.4.1. Creación de un grupo de recursos	140

10.4.2. Eliminar un grupo de recursos	141
10.4.3. Visualización de los grupos de recursos	141
10.4.4. Opciones de grupo	141
10.4.5. Adherencia al grupo	141
10.5. DETERMINACIÓN DEL COMPORTAMIENTO DE LOS RECURSOS	141
CAPÍTULO 11. DETERMINACIÓN DE LOS NODOS EN LOS QUE PUEDE EJECUTARSE UN RECURSO	143
11.1. CONFIGURAR LAS RESTRICCIONES DE UBICACIÓN	143
11.2. LIMITACIÓN DEL DESCUBRIMIENTO DE RECURSOS A UN SUBCONJUNTO DE NODOS	144
11.3. CONFIGURAR UNA ESTRATEGIA DE RESTRICCIÓN DE UBICACIÓN	147
11.3.1. Configuración de un clúster "Opt-In"	147
11.3.2. Configuración de un clúster "Opt-Out"	147
11.4. CONFIGURAR UN RECURSO PARA QUE PREFIERA SU NODO ACTUAL	148
CAPÍTULO 12. DETERMINACIÓN DEL ORDEN DE EJECUCIÓN DE LOS RECURSOS DEL CLÚSTER	149
12.1. CONFIGURACIÓN DE LA ORDENACIÓN OBLIGATORIA	150
12.2. CONFIGURAR LA ORDENACIÓN DE LOS AVISOS	151
12.3. CONFIGURACIÓN DE CONJUNTOS DE RECURSOS ORDENADOS	151
12.4. CONFIGURACIÓN DEL ORDEN DE INICIO PARA LAS DEPENDENCIAS DE RECURSOS NO GESTIONADAS POR PACEMAKER	152
CAPÍTULO 13. COLOCACIÓN DE LOS RECURSOS DEL CLÚSTER	154
13.1. ESPECIFICACIÓN DE LA UBICACIÓN OBLIGATORIA DE LOS RECURSOS	154
13.2. ESPECIFICACIÓN DE LA COLOCACIÓN DE LOS RECURSOS CON FINES DE ASESORAMIENTO	155
13.3. COLOCACIÓN DE CONJUNTOS DE RECURSOS	155
13.4. ELIMINACIÓN DE LAS RESTRICCIONES DE COLOCACIÓN	156
CAPÍTULO 14. VISUALIZACIÓN DE LAS LIMITACIONES DE RECURSOS	157
14.1. VISUALIZACIÓN DE TODAS LAS RESTRICCIONES CONFIGURADAS	157
14.2. VISUALIZACIÓN DE LAS RESTRICCIONES DE UBICACIÓN	157
14.3. VISUALIZACIÓN DE LAS RESTRICCIONES DE ORDENACIÓN	157
14.4. VISUALIZACIÓN DE LAS RESTRICCIONES DE COLOCACIÓN	157
14.5. VISUALIZACIÓN DE LAS RESTRICCIONES ESPECÍFICAS DE LOS RECURSOS	157
14.6. VISUALIZACIÓN DE LAS DEPENDENCIAS DE RECURSOS (RED HAT ENTERPRISE LINUX 8.2 Y POSTERIORES)	158
CAPÍTULO 15. DETERMINACIÓN DE LA UBICACIÓN DE LOS RECURSOS CON REGLAS	160
15.1. REGLAS DEL MARCAPASOS	160
15.1.1. Expresiones de atributos de los nodos	160
15.1.2. Expresiones basadas en la fecha y la hora	162
15.1.3. Especificaciones de fecha	163
15.2. CONFIGURACIÓN DE UNA RESTRICCIÓN DE UBICACIÓN DE MARCAPASOS MEDIANTE REGLAS	164
CAPÍTULO 16. GESTIÓN DE LOS RECURSOS DEL CLÚSTER	166
16.1. VISUALIZACIÓN DE LOS RECURSOS CONFIGURADOS	166
16.2. MODIFICACIÓN DE LOS PARÁMETROS DE LOS RECURSOS	166
16.3. BORRAR EL ESTADO DE FALLO DE LOS RECURSOS DEL CLÚSTER	167
16.4. DESPLAZAMIENTO DE RECURSOS EN UN CLÚSTER	167
16.4.1. Desplazamiento de recursos por avería	168
16.4.2. Desplazamiento de recursos por cambios de conectividad	168
16.5. DESACTIVACIÓN DE UNA OPERACIÓN DE MONITORIZACIÓN	169
16.6. CONFIGURACIÓN Y GESTIÓN DE LAS ETIQUETAS DE RECURSOS DEL CLÚSTER (RHEL 8.3 Y POSTERIOR)	170
16.6.1. Etiquetado de los recursos del clúster para su administración por categorías	170
16.6.2. Eliminar un recurso de clúster etiquetado	171

CAPÍTULO 17. CREACIÓN DE RECURSOS DE CLUSTER ACTIVOS EN VARIOS NODOS (RECURSOS CLONADOS)	172
17.1. CREACIÓN Y ELIMINACIÓN DE UN RECURSO CLONADO	172
17.2. CONFIGURAR LAS LIMITACIONES DE RECURSOS DE LOS CLONES	174
17.3. CREACIÓN DE RECURSOS CLONADOS PROMOCIONABLES	175
17.3.1. Crear un recurso promocionable	175
17.3.2. Configuración de las limitaciones de los recursos promocionables	176
17.4. DESPLAZAMIENTO DE UN RECURSO PROMOCIONADO EN CASO DE FALLO (RHEL 8.3 Y POSTERIORES)	176
CAPÍTULO 18. GESTIÓN DE LOS NODOS DEL CLÚSTER	178
18.1. DETENCIÓN DE LOS SERVICIOS DEL CLÚSTER	178
18.2. ACTIVACIÓN Y DESACTIVACIÓN DE LOS SERVICIOS DEL CLÚSTER	178
18.3. AÑADIR NODOS DEL CLÚSTER	178
18.4. ELIMINACIÓN DE LOS NODOS DEL CLÚSTER	180
18.5. AÑADIR UN NODO A UN CLUSTER CON MÚLTIPLES ENLACES	180
18.6. AÑADIR Y MODIFICAR ENLACES EN UN CLÚSTER EXISTENTE (RHEL 8.1 Y POSTERIOR)	180
18.6.1. Añadir y eliminar enlaces en un clúster existente	180
18.6.2. Modificación de un enlace en un cluster con múltiples enlaces	181
18.6.3. Modificación de las direcciones de enlace en un clúster con un solo enlace	181
18.6.4. Modificación de las opciones de enlace para un enlace en un cluster con un solo enlace	182
18.6.5. No es posible modificar un enlace cuando se añade uno nuevo	183
18.7. CONFIGURACIÓN DE UN GRAN CLÚSTER CON MUCHOS RECURSOS	183
CAPÍTULO 19. CONFIGURACIÓN DE LOS PERMISOS DE LOS USUARIOS PARA UN CLÚSTER PACEMAKER .	185
19.1. ESTABLECER PERMISOS PARA EL ACCESO A LOS NODOS A TRAVÉS DE UNA RED	185
19.2. ESTABLECIMIENTO DE PERMISOS LOCALES MEDIANTE ACLS	185
CAPÍTULO 20. OPERACIONES DE CONTROL DE RECURSOS	187
20.1. CONFIGURACIÓN DE LAS OPERACIONES DE SUPERVISIÓN DE RECURSOS	188
20.2. CONFIGURACIÓN DE LOS VALORES PREDETERMINADOS DE LAS OPERACIONES DE RECURSOS GLOBALES	189
20.2.1. Anulación de los valores de las operaciones específicas de los recursos	189
20.2.2. Cambio del valor por defecto de una operación de recursos para conjuntos de recursos (RHEL 8.3 y posteriores)	190
20.2.3. Visualización de los valores por defecto de las operaciones de los recursos actualmente configurados	190
20.3. CONFIGURACIÓN DE MÚLTIPLES OPERACIONES DE SUPERVISIÓN	191
CAPÍTULO 21. PROPIEDADES DE LA AGRUPACIÓN DE MARCAPASOS	192
21.1. RESUMEN DE LAS PROPIEDADES Y OPCIONES DEL CLÚSTER	192
21.2. ESTABLECER Y ELIMINAR LAS PROPIEDADES DEL CLÚSTER	197
21.3. CONSULTA DE LA CONFIGURACIÓN DE LAS PROPIEDADES DEL CLÚSTER	198
CAPÍTULO 22. CONFIGURAR LOS RECURSOS PARA QUE PERMANEZCAN DETENIDOS AL APAGAR UN NODO LIMPIO (RHEL 8.2 Y POSTERIORES)	199
22.1. PROPIEDADES DEL CLÚSTER PARA CONFIGURAR QUE LOS RECURSOS PERMANEZCAN PARADOS AL APAGAR UN NODO LIMPIO	199
22.2. ESTABLECIMIENTO DE LA PROPIEDAD DE CLUSTER SHUTDOWN-LOCK	200
CAPÍTULO 23. CONFIGURAR UNA ESTRATEGIA DE COLOCACIÓN DE NODOS	202
23.1. ATRIBUTOS DE UTILIZACIÓN Y ESTRATEGIA DE COLOCACIÓN	202
23.1.1. Configuración de la capacidad de nodos y recursos	202
23.1.2. Configuración de la estrategia de colocación	203
23.2. ASIGNACIÓN DE RECURSOS DEL MARCAPASOS	203

23.2.1. Preferencia de nodo	203
23.2.2. Capacidad del nodo	204
23.2.3. Preferencia de asignación de recursos	204
23.3. DIRECTRICES DE LA ESTRATEGIA DE COLOCACIÓN DE RECURSOS	204
23.4. EL AGENTE DE RECURSOS NODEUTILIZATION	205
CAPÍTULO 24. CONFIGURACIÓN DE UN DOMINIO VIRTUAL COMO RECURSO	206
24.1. OPCIONES DE RECURSOS DEL DOMINIO VIRTUAL	206
24.2. CREACIÓN DEL RECURSO DE DOMINIO VIRTUAL	244
CAPÍTULO 25. QUÓRUM DEL CLÚSTER	245
25.1. CONFIGURACIÓN DE LAS OPCIONES DE QUÓRUM	245
25.2. MODIFICACIÓN DE LAS OPCIONES DE QUÓRUM	246
25.3. VISUALIZACIÓN DE LA CONFIGURACIÓN Y EL ESTADO DEL QUÓRUM	247
25.4. EJECUCIÓN DE CLÚSTERES INQUEMADOS	247
25.5. DISPOSITIVOS DE QUÓRUM	248
25.5.1. Instalación de paquetes de dispositivos de quórum	248
25.5.2. Configuración de un dispositivo de quórum	249
25.5.3. Gestión del servicio de dispositivos de quórum	253
25.5.4. Gestión de la configuración del dispositivo de quórum en un clúster	253
25.5.4.1. Cambiar la configuración del dispositivo de quórum	253
25.5.4.2. Eliminación de un dispositivo de quórum	254
25.5.4.3. Destrucción de un dispositivo de quórum	254
CAPÍTULO 26. SCRIPTS DE ACTIVACIÓN DE EVENTOS DE CLUSTER	255
26.1. INSTALACIÓN Y CONFIGURACIÓN DE AGENTES DE ALERTA DE MUESTRA	255
26.2. CREACIÓN DE UNA ALERTA DE CLÚSTER	256
26.3. VISUALIZACIÓN, MODIFICACIÓN Y ELIMINACIÓN DE ALERTAS DE CLUSTER	257
26.4. CONFIGURAR LOS DESTINATARIOS DE LAS ALERTAS	257
26.5. OPCIONES DE META-ALERTA	258
26.6. EJEMPLOS DE COMANDOS DE CONFIGURACIÓN DE ALERTAS	258
26.7. ESCRIBIR UN AGENTE DE ALERTA	260
CAPÍTULO 27. CONFIGURACIÓN DE CLUSTERS MULTISITIO CON PACEMAKER	263
27.1. VISIÓN GENERAL DEL GESTOR DE TICKETS DEL CLÚSTER BOOTH	263
27.2. CONFIGURACIÓN DE CLUSTERS MULTISITIO CON PACEMAKER	263
CAPÍTULO 28. INTEGRACIÓN DE NODOS NO SINCRONIZADOS EN UN CLÚSTER: EL SERVICIO PACEMAKER_REMOTE	267
28.1. AUTENTICACIÓN DE HOST Y GUEST DE LOS NODOS PACEMAKER_REMOTE	268
28.2. CONFIGURACIÓN DE LOS NODOS INVITADOS KVM	268
28.2.1. Opciones de recursos del nodo invitado	268
28.2.2. Integración de una máquina virtual como nodo invitado	269
28.3. CONFIGURACIÓN DE LOS NODOS REMOTOS DE PACEMAKER	270
28.3.1. Opciones de recursos del nodo remoto	270
28.3.2. Resumen de la configuración del nodo remoto	279
28.4. CAMBIAR LA UBICACIÓN DEL PUERTO POR DEFECTO	280
28.5. ACTUALIZACIÓN DE SISTEMAS CON NODOS PACEMAKER_REMOTE	281
CAPÍTULO 29. REALIZAR EL MANTENIMIENTO DEL CLÚSTER	282
29.1. PONER UN NODO EN MODO DE ESPERA	282
29.2. MOVER MANUALMENTE LOS RECURSOS DEL CLÚSTER	283
29.2.1. Mover un recurso de su nodo actual	283
29.2.2. Mover un recurso a su nodo preferido	284
29.3. DESACTIVACIÓN, ACTIVACIÓN Y PROHIBICIÓN DE LOS RECURSOS DEL CLÚSTER	285

Desactivación de un recurso del clúster	285
Habilitación de un recurso de clúster	285
Impedir que un recurso se ejecute en un nodo concreto	285
Forzar el inicio de un recurso en el nodo actual	286
29.4. PONER UN RECURSO EN MODO NO GESTIONADO	286
29.5. PONER UN CLÚSTER EN MODO DE MANTENIMIENTO	286
29.6. ACTUALIZACIÓN DE UN CLÚSTER DE ALTA DISPONIBILIDAD RHEL	287
29.7. ACTUALIZACIÓN DE NODOS REMOTOS Y NODOS INVITADOS	287
29.8. MIGRACIÓN DE MÁQUINAS VIRTUALES EN UN CLÚSTER RHEL	288
CAPÍTULO 30. CONFIGURACIÓN DE CLUSTERS DE RECUPERACIÓN DE DESASTRES	290
30.1. CONSIDERACIONES SOBRE LOS CLUSTERS DE RECUPERACIÓN DE DESASTRES	290
30.2. VISUALIZACIÓN DEL ESTADO DE LOS CLUSTERS DE RECUPERACIÓN (RHEL 8.2 Y POSTERIORES)	290

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. DESCRIPCIÓN DEL COMPLEMENTO DE ALTA DISPONIBILIDAD

El complemento de alta disponibilidad es un sistema en clúster que proporciona fiabilidad, escalabilidad y disponibilidad a los servicios de producción críticos.

Un clúster son dos o más ordenadores (llamados *nodes* o *members*) que trabajan juntos para realizar una tarea. Los clústeres pueden utilizarse para proporcionar servicios o recursos de alta disponibilidad. La redundancia de múltiples máquinas se utiliza para protegerse de fallos de muchos tipos.

Los clústeres de alta disponibilidad proporcionan servicios de alta disponibilidad eliminando los puntos únicos de fallo y pasando los servicios de un nodo del clúster a otro en caso de que un nodo quede inoperativo. Normalmente, los servicios de un clúster de alta disponibilidad leen y escriben datos (mediante sistemas de archivos montados de lectura y escritura). Por lo tanto, un cluster de alta disponibilidad debe mantener la integridad de los datos cuando un nodo del cluster toma el control de un servicio de otro nodo del cluster. Los fallos de los nodos de un clúster de alta disponibilidad no son visibles para los clientes externos al clúster. (Los clústeres de alta disponibilidad se denominan a veces clústeres de conmutación por error). El complemento de alta disponibilidad proporciona clústeres de alta disponibilidad a través de su componente de gestión de servicios de alta disponibilidad, **Pacemaker**.

1.1. COMPONENTES DEL COMPLEMENTO DE ALTA DISPONIBILIDAD

El complemento de alta disponibilidad consta de los siguientes componentes principales:

- Infraestructura de clústeres
- Gestión de servicios de alta disponibilidad
- Herramientas de administración de clústeres

Puede complementar el complemento de alta disponibilidad con los siguientes componentes:

- Red Hat GFS2 (Global File System 2)
- Demonio de bloqueo LVM (**lvmlockd**)
- Complemento del equilibrador de carga

1.2. VISIÓN GENERAL DEL MARCAPASOS

Pacemaker es un gestor de recursos de clúster. Consigue la máxima disponibilidad para los servicios y recursos de su clúster haciendo uso de las capacidades de mensajería y afiliación de la infraestructura del clúster para disuadir y recuperarse de los fallos a nivel de nodos y recursos.

1.2.1. Componentes de la arquitectura del marcapasos

Un clúster configurado con Pacemaker está compuesto por demonios de componentes separados que supervisan la pertenencia al clúster, scripts que gestionan los servicios y subsistemas de gestión de recursos que supervisan los distintos recursos.

Los siguientes componentes forman la arquitectura de Pacemaker:

Base de información de clusters (CIB)

El demonio de información de Pacemaker, que utiliza XML internamente para distribuir y sincronizar la configuración actual y la información de estado del Coordinador Designado (DC)

Demonio de gestión de recursos del clúster (CRMd)

Las acciones de los recursos del cluster Pacemaker se enrutan a través de este demonio. Los recursos gestionados por CRMd pueden ser consultados por los sistemas cliente, movidos, instanciados y modificados cuando sea necesario.

Cada nodo del clúster también incluye un demonio gestor de recursos local (LRMd) que actúa como interfaz entre CRMd y los recursos. LRMd pasa comandos de CRMd a los agentes, como el arranque y la parada y la transmisión de información de estado.

Disparar al otro nodo en la cabeza (STONITH)

STONITH es la implementación de cercado de Pacemaker. Actúa como un recurso de clúster en Pacemaker que procesa las solicitudes de vallado, apagando a la fuerza los nodos y retirándolos del clúster para garantizar la integridad de los datos. STONITH está configurado en el CIB y puede ser monitorizado como un recurso de cluster normal. Para una visión general de la esgrima, ver [Sección 1.3, "Visión general de la esgrima"](#).

corosync

corosync es el componente - y un demonio del mismo nombre - que sirve a las necesidades centrales de membresía y comunicación de los miembros para los clusters de alta disponibilidad. Es necesario para que el complemento de alta disponibilidad funcione.

Además de esas funciones de afiliación y mensajería, **corosync** también:

- Gestiona las normas de quórum y su determinación.
- Proporciona capacidades de mensajería para las aplicaciones que coordinan u operan a través de múltiples miembros del clúster y, por lo tanto, deben comunicar información de estado o de otro tipo entre las instancias.
- Utiliza la librería **kronosnet** como transporte de red para proporcionar múltiples enlaces redundantes y conmutación automática por error.

1.2.2. Herramientas de configuración y gestión

El complemento de alta disponibilidad cuenta con dos herramientas de configuración para el despliegue, la supervisión y la gestión del clúster.

pcs

La interfaz de línea de comandos **pcs** controla y configura Pacemaker y el demonio **corosync** heartbeat. Un programa basado en la línea de comandos, **pcs** puede realizar las siguientes tareas de gestión del clúster:

- Crear y configurar un clúster Pacemaker/Corosync
- Modificar la configuración del clúster mientras se está ejecutando
- Configurar remotamente tanto Pacemaker como Corosync, así como iniciar, detener y mostrar la información de estado del clúster

pcsd Web UI

Una interfaz gráfica de usuario para crear y configurar clusters Pacemaker/Corosync.

1.2.3. Los archivos de configuración del clúster y del marcapasos

Los archivos de configuración para el complemento de alta disponibilidad de Red Hat son **corosync.conf** y **cib.xml**.

El archivo **corosync.conf** proporciona los parámetros de cluster utilizados por **corosync**, el gestor de cluster en el que se basa Pacemaker. En general, no se debe editar el **corosync.conf** directamente sino, en su lugar, utilizar la interfaz **pcs** o **pcsctl**.

El archivo **cib.xml** es un archivo XML que representa tanto la configuración del cluster como el estado actual de todos los recursos del cluster. Este archivo es utilizado por la Base de Información del Cluster (CIB) de Pacemaker. El contenido de la CIB se mantiene automáticamente sincronizado en todo el cluster. No edite el archivo **cib.xml** directamente; utilice la interfaz **pcs** o **pcsctl** en su lugar.

1.3. VISIÓN GENERAL DE LA ESGRIMA

Si la comunicación con un solo nodo del clúster falla, los demás nodos del clúster deben ser capaces de restringir o liberar el acceso a los recursos a los que el nodo del clúster que ha fallado pueda tener acceso. Esto no puede lograrse contactando con el propio nodo del clúster, ya que éste puede no responder. En su lugar, debe proporcionar un método externo, que se denomina cercado con un agente de cercado. Un agente de valla es un dispositivo externo que puede ser utilizado por el clúster para restringir el acceso a los recursos compartidos por un nodo errante, o para emitir un reinicio duro en el nodo del clúster.

Sin un dispositivo de valla configurado, no tiene forma de saber que los recursos utilizados previamente por el nodo de clúster desconectado han sido liberados, y esto podría impedir que los servicios se ejecuten en cualquiera de los otros nodos de clúster. A la inversa, el sistema puede suponer erróneamente que el nodo del clúster ha liberado sus recursos y esto puede llevar a la corrupción y pérdida de datos. Sin un dispositivo de valla configurado no se puede garantizar la integridad de los datos y la configuración del clúster no será compatible.

Cuando el cercado está en curso, no se permite la ejecución de ninguna otra operación del clúster. El funcionamiento normal del clúster no se puede reanudar hasta que se haya completado el cercado o hasta que el nodo del clúster se reincorpore al clúster después de reiniciar el nodo del clúster.

Para más información sobre el cercado, consulte [Cerrado en un cluster de alta disponibilidad de Red Hat](#).

1.4. RESUMEN DEL QUÓRUM

Para mantener la integridad y la disponibilidad del clúster, los sistemas de clúster utilizan un concepto conocido como *quorum* para evitar la corrupción y la pérdida de datos. Un clúster tiene quórum cuando más de la mitad de los nodos del clúster están en línea. Para mitigar la posibilidad de corrupción de datos debido a un fallo, Pacemaker detiene por defecto todos los recursos si el clúster no tiene quórum.

El quórum se establece mediante un sistema de votación. Cuando un nodo del clúster no funciona como debería o pierde la comunicación con el resto del clúster, la mayoría de los nodos que funcionan pueden votar para aislar y, si es necesario, cercar el nodo para su mantenimiento.

Por ejemplo, en un cluster de 6 nodos, el quórum se establece cuando al menos 4 nodos del cluster están funcionando. Si la mayoría de los nodos se desconectan o dejan de estar disponibles, el clúster deja de tener quórum y Pacemaker detiene los servicios en clúster.

Las características de quórum en Pacemaker previenen lo que también se conoce como *split-brain*, un fenómeno en el que el cluster se separa de la comunicación pero cada parte continúa trabajando como clusters separados, potencialmente escribiendo en los mismos datos y posiblemente causando

corrupción o pérdida. Para más información sobre lo que significa estar en un estado de cerebro dividido, y sobre los conceptos de quórum en general, vea [Explorando los conceptos de los clusters de alta disponibilidad de RHEL - Quórum](#).

Un cluster de Red Hat Enterprise Linux High Availability Add-On utiliza el servicio **votequorum**, en conjunto con fencing, para evitar situaciones de split brain. Se asigna un número de votos a cada sistema en el cluster, y las operaciones del cluster se permiten sólo cuando hay una mayoría de votos.

1.5. RESUMEN DE RECURSOS

Un *cluster resource* es una instancia de programa, datos o aplicación que debe ser gestionada por el servicio de cluster. Estos recursos se abstraen mediante *agents* que proporciona una interfaz estándar para gestionar el recurso en un entorno de clúster.

Para asegurarse de que los recursos se mantienen en buen estado, puede añadir una operación de supervisión a la definición de un recurso. Si no se especifica una operación de supervisión para un recurso, se añade una por defecto.

Puede determinar el comportamiento de un recurso en un cluster configurando *constraints*. Puede configurar las siguientes categorías de restricciones:

- limitaciones de ubicación
- restricciones de ordenación
- limitaciones de colocación

Uno de los elementos más comunes de un clúster es un conjunto de recursos que deben ubicarse juntos, iniciarse secuencialmente y detenerse en el orden inverso. Para simplificar esta configuración, Pacemaker admite el concepto de *groups*.

1.6. VOLÚMENES LÓGICOS LVM EN UN CLUSTER DE ALTA DISPONIBILIDAD DE RED HAT

El complemento de alta disponibilidad de Red Hat proporciona soporte para volúmenes LVM en dos configuraciones de cluster distintas:

- Volúmenes LVM de alta disponibilidad (HA-LVM) en configuraciones de conmutación por error activas/pasivas en las que sólo un único nodo del clúster accede al almacenamiento en todo momento.
- Volúmenes LVM que utilizan el demonio **lvmlockd** para gestionar los dispositivos de almacenamiento en configuraciones activas/activas en las que más de un nodo del clúster requiere acceso al almacenamiento al mismo tiempo. El demonio **lvmlockd** forma parte del complemento de almacenamiento resistente.

1.6.1. Elección de HA-LVM o volúmenes compartidos

Cuándo utilizar HA-LVM o volúmenes lógicos compartidos gestionados por el demonio **lvmlockd** debe basarse en las necesidades de las aplicaciones o servicios que se están implementando.

- Si varios nodos del cluster requieren acceso simultáneo de lectura/escritura a los volúmenes LVM en un sistema activo/activo, entonces debe utilizar el demonio **lvmlockd** y configurar sus volúmenes como volúmenes compartidos. El demonio **lvmlockd** proporciona un sistema para coordinar la activación y los cambios de los volúmenes LVM en todos los nodos de un clúster de

forma simultánea. El servicio de bloqueo del demonio **lvmlockd** proporciona protección a los metadatos LVM cuando varios nodos del cluster interactúan con los volúmenes y realizan cambios en su disposición. Esta protección depende de la configuración de cualquier grupo de volúmenes que se active simultáneamente en varios nodos del clúster como un volumen compartido.

- Si el clúster de alta disponibilidad está configurado para gestionar los recursos compartidos de forma activa/pasiva con un solo miembro que necesita acceder a un volumen LVM determinado a la vez, entonces puede utilizar HA-LVM sin el servicio de bloqueo **lvmlockd**.

La mayoría de las aplicaciones se ejecutarán mejor en una configuración activa/pasiva, ya que no están diseñadas u optimizadas para ejecutarse simultáneamente con otras instancias. La elección de ejecutar una aplicación que no sea compatible con el clúster en volúmenes lógicos compartidos puede resultar en un rendimiento degradado. Esto se debe a que hay una sobrecarga de comunicación de clúster para los propios volúmenes lógicos en estas instancias. Una aplicación con conciencia de clúster debe ser capaz de lograr ganancias de rendimiento por encima de las pérdidas de rendimiento introducidas por los sistemas de archivos de clúster y los volúmenes lógicos con conciencia de clúster. Esto se puede lograr para algunas aplicaciones y cargas de trabajo más fácilmente que para otras. Determinar cuáles son los requisitos del clúster y si el esfuerzo extra hacia la optimización para un clúster activo/activo dará sus frutos es la forma de elegir entre las dos variantes de LVM. La mayoría de los usuarios conseguirán los mejores resultados de HA utilizando HA-LVM.

HA-LVM y los volúmenes lógicos compartidos que utilizan **lvmlockd** son similares en el hecho de que evitan la corrupción de los metadatos de LVM y sus volúmenes lógicos, que de otro modo podría ocurrir si se permite que varias máquinas realicen cambios superpuestos. HA-LVM impone la restricción de que un volumen lógico sólo puede ser activado de forma exclusiva; es decir, activo en una sola máquina a la vez. Esto significa que sólo se utilizan implementaciones locales (no agrupadas) de los controladores de almacenamiento. Evitar la sobrecarga de coordinación del clúster de esta manera aumenta el rendimiento. Un volumen compartido que utiliza **lvmlockd** no impone estas restricciones y un usuario es libre de activar un volumen lógico en todas las máquinas de un clúster; esto obliga a utilizar controladores de almacenamiento compatibles con el clúster, lo que permite poner encima sistemas de archivos y aplicaciones compatibles con el clúster.

1.6.2. Configuración de volúmenes LVM en un clúster

En Red Hat Enterprise Linux 8, los clusters se gestionan a través de Pacemaker. Tanto HA-LVM como los volúmenes lógicos compartidos son soportados sólo en conjunto con los clusters de Pacemaker, y deben ser configurados como recursos de cluster.

- Para ver ejemplos de procedimientos para configurar un volumen HA-LVM como parte de un cluster Pacemaker, consulte Configuración de un [servidor HTTP Apache activo/pasivo](#) en un cluster de alta disponibilidad de Red Hat y [Configuración de un servidor NFS activo/pasivo en un cluster de alta disponibilidad de Red Hat](#).
Tenga en cuenta que estos procedimientos incluyen los siguientes pasos:
 - Garantizar que sólo el clúster es capaz de activar el grupo de volúmenes
 - Configuración de un volumen lógico LVM
 - Configuración del volumen LVM como recurso de cluster
- Para conocer el procedimiento de configuración de volúmenes LVM compartidos que utilizan el demonio **lvmlockd** para gestionar dispositivos de almacenamiento en configuraciones activas/activas, consulte [Configuración de un sistema de archivos GFS2 en un clúster](#)

CAPÍTULO 2. CÓMO EMPEZAR CON PACEMAKER

Los siguientes procedimientos proporcionan una introducción a las herramientas y procesos que se utilizan para crear un cluster Pacemaker. Están destinados a los usuarios que están interesados en ver el aspecto del software de clúster y cómo se administra, sin necesidad de configurar un clúster en funcionamiento.



NOTA

Estos procedimientos no crean un cluster Red Hat compatible, que requiere al menos dos nodos y la configuración de un dispositivo de cercado.

2.1. APRENDER A USAR EL MARCAPASOS

Este ejemplo requiere un único nodo que ejecute RHEL 8 y requiere una dirección IP flotante que resida en la misma red que una de las direcciones IP asignadas estáticamente al nodo.

- El nodo utilizado en este ejemplo es **z1.example.com**.
- La dirección IP flotante utilizada en este ejemplo es 192.168.122.120.



NOTA

Asegúrese de que el nombre del nodo en el que se está ejecutando está en su archivo **/etc/hosts**.

Mediante este procedimiento, aprenderá a utilizar Pacemaker para configurar un clúster, a mostrar el estado del clúster y a configurar un servicio de clúster. Este ejemplo crea un servidor Apache HTTP como recurso de clúster y muestra cómo responde el clúster cuando el recurso falla.

1. Instale los paquetes de software de Red Hat High Availability Add-On desde el canal de Alta Disponibilidad, e inicie y habilite el servicio **pcsd**.

```
# yum install pcs pacemaker fence-agents-all
...
# systemctl start pcsd.service
# systemctl enable pcsd.service
```

Si está ejecutando el demonio **firewalld**, habilite los puertos requeridos por el complemento de alta disponibilidad de Red Hat.

```
# firewall-cmd --permanent --add-service=high-availability
# firewall-cmd --reload
```

2. Establezca una contraseña para el usuario **hacluster** en cada nodo del clúster y autentique el usuario **hacluster** para cada nodo del clúster en el nodo desde el que ejecutará los comandos **pcs**. Este ejemplo está utilizando sólo un nodo, el nodo desde el cual está ejecutando los comandos, pero este paso se incluye aquí ya que es un paso necesario en la configuración de un cluster de alta disponibilidad de Red Hat soportado.

```
# passwd hacluster
...
# pcs host auth z1.example.com
```

3. Crear un clúster llamado **my_cluster** con un miembro y comprobar el estado del clúster. Este comando crea e inicia el clúster en un solo paso.

```
# pcs cluster setup my_cluster --start z1.example.com
...
# pcs cluster status
Cluster Status:
Stack: corosync
Current DC: z1.example.com (version 2.0.0-10.el8-b67d8d0de9) - partition with quorum
Last updated: Thu Oct 11 16:11:18 2018
Last change: Thu Oct 11 16:11:00 2018 by hacluster via crmd on z1.example.com
1 node configured
0 resources configured

PCSD Status:
z1.example.com: Online
```

4. Un cluster de Alta Disponibilidad de Red Hat requiere que se configure el cercado para el cluster. Las razones de este requisito se describen en [Esgrima en un cluster de alta disponibilidad de Red Hat](#). Para esta introducción, sin embargo, que pretende mostrar sólo cómo usar los comandos básicos de Pacemaker, desactive el cercado estableciendo la opción de cluster **stonith-enabled** a **false**.



AVISO

El uso de **stonith-enabled=false** es completamente inapropiado para un cluster de producción. Le dice al clúster que simplemente finja que los nodos que fallan están cercados de forma segura.

```
# pcs property set stonith-enabled=false
```

5. Configure un navegador web en su sistema y cree una página web para mostrar un simple mensaje de texto. Si está ejecutando el demonio **firewalld**, habilite los puertos que requiere **httpd**.



NOTA

No utilice **systemctl enable** para habilitar cualquier servicio que vaya a ser gestionado por el clúster para que se inicie en el arranque del sistema.

```
# yum install -y httpd wget
...
# firewall-cmd --permanent --add-service=http
# firewall-cmd --reload

# cat <<-END >/var/www/html/index.html
<html>
```

```
<body>My Test Site - $(hostname)</body>
</html>
END
```

Para que el agente de recursos de Apache obtenga el estado de Apache, cree la siguiente adición a la configuración existente para habilitar la URL del servidor de estado.

```
# cat <<-END > /etc/httpd/conf.d/status.conf
<Location /server-status>
SetHandler server-status
Order deny,allow
Deny from all
Allow from 127.0.0.1
Allow from ::1
</Location>
END
```

6. Cree los recursos **IPAddr2** y **apache** para que el clúster los gestione. El recurso 'IPAddr2' es una dirección IP flotante que no debe ser una ya asociada a un nodo físico. Si no se especifica el dispositivo NIC del recurso 'IPAddr2', la IP flotante debe residir en la misma red que la dirección IP asignada estáticamente y utilizada por el nodo.

Puede mostrar una lista de todos los tipos de recursos disponibles con el comando **pcs resource list**. Puede utilizar el comando **pcs resource describe resourcetype** para mostrar los parámetros que puede establecer para el tipo de recurso especificado. Por ejemplo, el siguiente comando muestra los parámetros que puede establecer para un recurso del tipo **apache**:

```
# pcs resource describe apache
...
```

En este ejemplo, el recurso dirección IP y el recurso apache están configurados como parte de un grupo llamado **apachegroup**, lo que asegura que los recursos se mantienen juntos para ejecutarse en el mismo nodo cuando se configura un clúster multinodal en funcionamiento.

```
# pcs resource create ClusterIP ocf:heartbeat:IPAddr2 ip=192.168.122.120 --group
apachegroup

# pcs resource create WebSite ocf:heartbeat:apache
configfile=/etc/httpd/conf/httpd.conf statusurl="http://localhost/server-status" --group
apachegroup

# pcs status
Cluster name: my_cluster
Stack: corosync
Current DC: z1.example.com (version 2.0.0-10.el8-b67d8d0de9) - partition with quorum
Last updated: Fri Oct 12 09:54:33 2018
Last change: Fri Oct 12 09:54:30 2018 by root via cibadmin on z1.example.com

1 node configured
2 resources configured

Online: [ z1.example.com ]

Full list of resources:

Resource Group: apachegroup
```

```
ClusterIP (ocf::heartbeat:IPaddr2): Started z1.example.com
WebSite (ocf::heartbeat:apache): Started z1.example.com
```

```
PCSD Status:
z1.example.com: Online
...
```

Después de haber configurado un recurso de clúster, puede utilizar el comando **pcs resource config** para mostrar las opciones configuradas para ese recurso.

pcs resource config WebSite

```
Resource: WebSite (class=ocf provider=heartbeat type=apache)
Attributes: configfile=/etc/httpd/conf/httpd.conf statusurl=http://localhost/server-status
Operations: start interval=0s timeout=40s (WebSite-start-interval-0s)
            stop interval=0s timeout=60s (WebSite-stop-interval-0s)
            monitor interval=1min (WebSite-monitor-interval-1min)
```

- Dirija su navegador al sitio web que ha creado utilizando la dirección IP flotante que ha configurado. Esto debería mostrar el mensaje de texto que definiste.
- Detenga el servicio web de apache y compruebe el estado del clúster. El uso de **killall -9** simula un fallo a nivel de aplicación.

killall -9 httpd

Compruebe el estado del clúster. Debería ver que la detención del servicio web provocó una acción fallida, pero que el software del clúster reinició el servicio y debería seguir pudiendo acceder al sitio web.

pcs status

```
Cluster name: my_cluster
...
Current DC: z1.example.com (version 1.1.13-10.el7-44eb2dd) - partition with quorum
1 node and 2 resources configured

Online: [ z1.example.com ]

Full list of resources:

Resource Group: apachegroup
ClusterIP (ocf::heartbeat:IPaddr2): Started z1.example.com
WebSite (ocf::heartbeat:apache): Started z1.example.com

Failed Resource Actions:
* WebSite_monitor_60000 on z1.example.com 'not running' (7): call=13, status=complete,
exitreason='none',
last-rc-change='Thu Oct 11 23:45:50 2016', queued=0ms, exec=0ms

PCSD Status:
z1.example.com: Online
```

Puede borrar el estado de fallo en el recurso que ha fallado una vez que el servicio esté en funcionamiento de nuevo y el aviso de acción fallida ya no aparecerá cuando vea el estado del clúster.

```
# pcs resource cleanup WebSite
```

9. Cuando haya terminado de ver el clúster y el estado del mismo, detenga los servicios del clúster en el nodo. Aunque sólo haya iniciado los servicios en un nodo para esta introducción, se incluye el parámetro **--all** ya que detendría los servicios de clúster en todos los nodos de un clúster real de varios nodos.

```
# pcs cluster stop --all
```

2.2. APRENDER A CONFIGURAR LA CONMUTACIÓN POR ERROR

Este procedimiento proporciona una introducción a la creación de un cluster Pacemaker que ejecuta un servicio que fallará de un nodo a otro cuando el nodo en el que se ejecuta el servicio deje de estar disponible. Mediante este procedimiento, podrá aprender a crear un servicio en un clúster de dos nodos y podrá observar lo que ocurre con ese servicio cuando falla en el nodo en el que se está ejecutando.

Este procedimiento de ejemplo configura un cluster Pacemaker de dos nodos que ejecuta un servidor HTTP Apache. A continuación, puede detener el servicio Apache en un nodo para ver cómo el servicio sigue estando disponible.

Este procedimiento requiere como prerequisite que tenga dos nodos ejecutando Red Hat Enterprise Linux 8 que puedan comunicarse entre sí, y requiere una dirección IP flotante que resida en la misma red que una de las direcciones IP asignadas estáticamente del nodo.

- Los nodos utilizados en este ejemplo son **z1.example.com** y **z2.example.com**.
- La dirección IP flotante utilizada en este ejemplo es 192.168.122.120.



NOTA

Asegúrese de que los nombres de los nodos que está utilizando están en el archivo **/etc/hosts** en cada nodo.

1. En ambos nodos, instale los paquetes de software Red Hat High Availability Add-On desde el canal de Alta Disponibilidad, e inicie y habilite el servicio **pcsd**.

```
# yum install pcs pacemaker fence-agents-all
```

```
...
```

```
# systemctl start pcsd.service
```

```
# systemctl enable pcsd.service
```

Si está ejecutando el demonio **firewalld**, en ambos nodos habilite los puertos requeridos por el complemento de alta disponibilidad de Red Hat.

```
# firewall-cmd --permanent --add-service=high-availability
```

```
# firewall-cmd --reload
```

2. En ambos nodos del clúster, establezca una contraseña para el usuario **hacluster**.

```
# passwd hacluster
```

3. Autentifique el usuario **hacluster** para cada nodo del clúster en el nodo desde el que va a ejecutar los comandos **pcs**.

```
# pcs host auth z1.example.com z2.example.com
```

4. Cree un clúster llamado **my_cluster** con ambos nodos como miembros del clúster. Este comando crea e inicia el cluster en un solo paso. Sólo es necesario ejecutarlo desde un nodo del clúster porque los comandos de configuración de **pcs** tienen efecto para todo el clúster. En un nodo del clúster, ejecute el siguiente comando.

```
# pcs cluster setup my_cluster --start z1.example.com z2.example.com
```

5. Un cluster de Alta Disponibilidad de Red Hat requiere que se configure el cercado para el cluster. Las razones de este requisito se describen en [Esgrima en un cluster de alta disponibilidad de Red Hat](#). Para esta introducción, sin embargo, para mostrar solamente cómo funciona la conmutación por error en esta configuración, desactive el cercado estableciendo la opción de cluster **stonith-enabled** a **false**

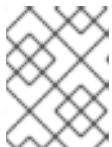


AVISO

El uso de **stonith-enabled=false** es completamente inapropiado para un cluster de producción. Le dice al clúster que simplemente finja que los nodos que fallan están cercados de forma segura.

```
# pcs property set stonith-enabled=false
```

6. Después de crear un clúster y desactivar el cercado, compruebe el estado del clúster.



NOTA

Cuando ejecute el comando **pcs cluster status**, es posible que la salida difiera temporalmente de los ejemplos mientras se inician los componentes del sistema.

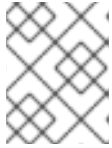
```
# pcs cluster status
```

```
Cluster Status:
Stack: corosync
Current DC: z1.example.com (version 2.0.0-10.el8-b67d8d0de9) - partition with quorum
Last updated: Thu Oct 11 16:11:18 2018
Last change: Thu Oct 11 16:11:00 2018 by hacluster via crmd on z1.example.com
2 nodes configured
0 resources configured
```

```
PCSD Status:
```

```
z1.example.com: Online
z2.example.com: Online
```

7. En ambos nodos, configure un navegador web y cree una página web para mostrar un simple mensaje de texto. Si está ejecutando el demonio **firewalld**, habilite los puertos que requiere **httpd**.



NOTA

No utilice **systemctl enable** para habilitar cualquier servicio que vaya a ser gestionado por el clúster para que se inicie en el arranque del sistema.

```
# yum install -y httpd wget
...
# firewall-cmd --permanent --add-service=http
# firewall-cmd --reload

# cat <<-END >/var/www/html/index.html
<html>
<body>My Test Site - $(hostname)</body>
</html>
END
```

Para que el agente de recursos de Apache obtenga el estado de Apache, en cada nodo del clúster cree la siguiente adición a la configuración existente para habilitar la URL del servidor de estado.

```
# cat <<-END > /etc/httpd/conf.d/status.conf
<Location /server-status>
SetHandler server-status
Order deny,allow
Deny from all
Allow from 127.0.0.1
Allow from ::1
</Location>
END
```

8. Cree los recursos **IPAddr2** y **apache** para que el clúster los gestione. El recurso 'IPAddr2' es una dirección IP flotante que no debe ser una ya asociada a un nodo físico. Si no se especifica el dispositivo NIC del recurso 'IPAddr2', la IP flotante debe residir en la misma red que la dirección IP asignada estáticamente y utilizada por el nodo.

Puede mostrar una lista de todos los tipos de recursos disponibles con el comando **pcs resource list**. Puede utilizar el comando **pcs resource describe resourcetype** para mostrar los parámetros que puede establecer para el tipo de recurso especificado. Por ejemplo, el siguiente comando muestra los parámetros que puede establecer para un recurso del tipo **apache**:

```
# pcs resource describe apache
...
```

En este ejemplo, el recurso dirección IP y el recurso apache están configurados como parte de un grupo llamado **apachegroup**, lo que asegura que los recursos se mantengan juntos para ejecutarse en el mismo nodo.

Ejecute los siguientes comandos desde un nodo del clúster:

```
# pcs resource create ClusterIP ocf:heartbeat:IPAddr2 ip=192.168.122.120 --group
apachegroup

# pcs resource create WebSite ocf:heartbeat:apache
configfile=/etc/httpd/conf/httpd.conf statusurl="http://localhost/server-status" --group
apachegroup
```

```

# pcs status
Cluster name: my_cluster
Stack: corosync
Current DC: z1.example.com (version 2.0.0-10.el8-b67d8d0de9) - partition with quorum
Last updated: Fri Oct 12 09:54:33 2018
Last change: Fri Oct 12 09:54:30 2018 by root via cibadmin on z1.example.com

2 nodes configured
2 resources configured

Online: [ z1.example.com z2.example.com ]

Full list of resources:

Resource Group: apachegroup
  ClusterIP (ocf::heartbeat:IPAddr2):   Started z1.example.com
  WebSite (ocf::heartbeat:apache):     Started z1.example.com

PCSD Status:
z1.example.com: Online
z2.example.com: Online
...

```

Tenga en cuenta que en este caso, el servicio **apachegroup** se está ejecutando en el nodo z1.ejemplo.com.

9. Acceda al sitio web que ha creado, detenga el servicio en el nodo en el que se está ejecutando y observe cómo el servicio falla en el segundo nodo.
 - a. Dirija un navegador al sitio web que ha creado utilizando la dirección IP flotante que ha configurado. Esto debería mostrar el mensaje de texto que definió, mostrando el nombre del nodo en el que se está ejecutando el sitio web.
 - b. Detener el servicio web de apache. El uso de **killall -9** simula un fallo a nivel de aplicación.

```
# killall -9 httpd
```

Compruebe el estado del clúster. Debería ver que la detención del servicio web provocó una acción fallida, pero que el software del clúster reinició el servicio en el nodo en el que se había estado ejecutando y debería seguir pudiendo acceder al navegador web.

```

# pcs status
Cluster name: my_cluster
Stack: corosync
Current DC: z1.example.com (version 2.0.0-10.el8-b67d8d0de9) - partition with quorum
Last updated: Fri Oct 12 09:54:33 2018
Last change: Fri Oct 12 09:54:30 2018 by root via cibadmin on z1.example.com

2 nodes configured
2 resources configured

Online: [ z1.example.com z2.example.com ]

Full list of resources:

```

```
Resource Group: apachegroup
  ClusterIP (ocf::heartbeat:IPAddr2):    Started z1.example.com
  WebSite (ocf::heartbeat:apache):      Started z1.example.com
```

Failed Resource Actions:

```
* WebSite_monitor_60000 on z1.example.com 'not running' (7): call=31,
status=complete, exitreason='none',
last-rc-change='Fri Feb 5 21:01:41 2016', queued=0ms, exec=0ms
```

Borre el estado de fallo una vez que el servicio esté de nuevo en funcionamiento.

```
# pcs resource cleanup WebSite
```

- c. Ponga el nodo en el que se está ejecutando el servicio en modo de espera. Tenga en cuenta que, dado que hemos desactivado el cercado, no podemos simular eficazmente un fallo a nivel de nodo (como tirar de un cable de alimentación) porque el cercado es necesario para que el clúster se recupere de tales situaciones.

```
# pcs node standby z1.example.com
```

- d. Compruebe el estado del clúster y anote dónde se está ejecutando el servicio.

```
# pcs status
```

```
Cluster name: my_cluster
```

```
Stack: corosync
```

```
Current DC: z1.example.com (version 2.0.0-10.el8-b67d8d0de9) - partition with quorum
```

```
Last updated: Fri Oct 12 09:54:33 2018
```

```
Last change: Fri Oct 12 09:54:30 2018 by root via cibadmin on z1.example.com
```

```
2 nodes configured
```

```
2 resources configured
```

```
Node z1.example.com: standby
```

```
Online: [ z2.example.com ]
```

```
Full list of resources:
```

```
Resource Group: apachegroup
```

```
  ClusterIP (ocf::heartbeat:IPAddr2):    Started z2.example.com
```

```
  WebSite (ocf::heartbeat:apache):      Started z2.example.com
```

- e. Acceda al sitio web. No debería haber pérdida de servicio, aunque el mensaje de la pantalla debería indicar el nodo en el que se está ejecutando el servicio.

10. Para restaurar los servicios del clúster en el primer nodo, saque el nodo del modo de espera. Esto no necesariamente moverá el servicio de nuevo a ese nodo.

```
# pcs node unstandby z1.example.com
```

11. Para la limpieza final, detenga los servicios del clúster en ambos nodos.

```
# pcs cluster stop --all
```

CAPÍTULO 3. LA INTERFAZ DE LÍNEA DE COMANDOS DE PCS

La interfaz de línea de comandos **pcs** controla y configura servicios de clúster como **corosync**, **pacemaker**, **booth** y **sbd** proporcionando una interfaz más sencilla para sus archivos de configuración.

Tenga en cuenta que no debe editar directamente el archivo de configuración **cib.xml**. En la mayoría de los casos, Pacemaker rechazará un archivo **cib.xml** modificado directamente.

3.1. PANTALLA DE AYUDA DE PCS

Puede utilizar la opción **-h** de **pcs** para mostrar los parámetros de un comando **pcs** y una descripción de dichos parámetros. Por ejemplo, el siguiente comando muestra los parámetros del comando **pcs resource**. Sólo se muestra una parte de la salida.

```
# pcs resource -h
```

3.2. VISUALIZACIÓN DE LA CONFIGURACIÓN BRUTA DEL CLÚSTER

Aunque no debe editar el archivo de configuración del clúster directamente, puede ver la configuración del clúster en bruto con el comando **pcs cluster cib**.

Puede guardar la configuración bruta del clúster en un archivo específico con el comando **pcs cluster cib filename** comando. Si ha configurado previamente un clúster y ya hay un CIB activo, utilice el siguiente comando para guardar el archivo xml sin procesar.

```
pcs cluster cib filename
```

Por ejemplo, el siguiente comando guarda el xml crudo del CIB en un archivo llamado **testfile**.

```
pcs cluster cib testfile
```

3.3. GUARDAR UN CAMBIO DE CONFIGURACIÓN EN UN ARCHIVO DE TRABAJO

Al configurar un clúster, puede guardar los cambios de configuración en un archivo específico sin que ello afecte al CIB activo. Esto le permite especificar las actualizaciones de configuración sin actualizar inmediatamente la configuración del clúster que se está ejecutando actualmente con cada actualización individual.

Para obtener información sobre cómo guardar el CIB en un archivo, consulte [Visualización de la configuración bruta del clúster](#). Una vez que haya creado ese archivo, puede guardar los cambios de configuración en ese archivo en lugar de en el CIB activo utilizando la opción **-f** del comando **pcs**. Cuando haya completado los cambios y esté listo para actualizar el archivo CIB activo, puede empujar esas actualizaciones del archivo con el comando **pcs cluster cib-push**.

El siguiente es el procedimiento recomendado para introducir cambios en el archivo CIB. Este procedimiento crea una copia del archivo CIB original guardado y realiza cambios en esa copia. Al transferir esos cambios al CIB activo, este procedimiento especifica la opción **diff-against** del comando **pcs cluster cib-push** para que sólo se transfieran al CIB los cambios entre el archivo original y el archivo actualizado. Esto permite a los usuarios realizar cambios en paralelo que no se sobrescriben entre sí, y reduce la carga de Pacemaker, que no necesita analizar todo el archivo de configuración.

1. Guardar el CIB activo en un archivo. Este ejemplo guarda el CIB en un archivo llamado **original.xml**.

```
# pcs cluster cib original.xml
```

2. Copie el archivo guardado en el archivo de trabajo que utilizará para las actualizaciones de configuración.

```
# cp original.xml updated.xml
```

3. Actualice su configuración según sea necesario. El siguiente comando crea un recurso en el archivo **updated.xml** pero no añade ese recurso a la configuración del clúster que se está ejecutando actualmente.

```
# pcs -f updated.xml resource create VirtualIP ocf:heartbeat:IPaddr2 ip=192.168.0.120
op monitor interval=30s
```

4. Empuje el archivo actualizado al CIB activo, especificando que está empujando sólo los cambios que ha hecho en el archivo original.

```
# pcs cluster cib-push updated.xml diff-against=original.xml
```

Alternativamente, puede empujar todo el contenido actual de un archivo CIB con el siguiente comando.

```
pcs cluster cib-push filename
```

Al empujar el archivo CIB completo, Pacemaker comprueba la versión y no permite empujar un archivo CIB que sea más antiguo que el que ya está en un cluster. Si necesita actualizar todo el archivo CIB con una versión más antigua que la que se encuentra actualmente en el clúster, puede utilizar la opción **--config** del comando **pcs cluster cib-push**.

```
pcs cluster cib-push --config filename
```

3.4. VISUALIZACIÓN DEL ESTADO DEL CLÚSTER

Puede mostrar el estado del clúster y de los recursos del clúster con el siguiente comando.

```
estado de los pcs
```

Puede mostrar el estado de un componente concreto del clúster con el parámetro *commands* del comando **pcs status**, especificando **resources**, **cluster**, **nodes** o **pcsd**.

```
estado de los pcs commands
```

Por ejemplo, el siguiente comando muestra el estado de los recursos del cluster.

```
recursos del estado de los pcs
```

El siguiente comando muestra el estado del clúster, pero no los recursos del mismo.

```
estado del cluster pcs
```

3.5. VISUALIZACIÓN DE LA CONFIGURACIÓN COMPLETA DEL CLÚSTER

Utilice el siguiente comando para mostrar la configuración completa del clúster actual.

```
pcs config
```

CAPÍTULO 4. CREACIÓN DE UN CLUSTER DE ALTA DISPONIBILIDAD DE RED HAT CON PACEMAKER

El siguiente procedimiento crea un cluster de alta disponibilidad de Red Hat de dos nodos utilizando **pcs**.

La configuración del cluster en este ejemplo requiere que su sistema incluya los siguientes componentes:

- 2 nodos, que se utilizarán para crear el cluster. En este ejemplo, los nodos utilizados son **z1.example.com** y **z2.example.com**.
- Conmutadores de red para la red privada. Recomendamos, pero no exigimos, una red privada para la comunicación entre los nodos del clúster y otro hardware del clúster, como los conmutadores de alimentación de red y los conmutadores de canal de fibra.
- Un dispositivo de cercado para cada nodo del clúster. Este ejemplo utiliza dos puertos del conmutador de potencia APC con un nombre de host de **zapc.example.com**.

4.1. INSTALACIÓN DEL SOFTWARE DEL CLÚSTER

El siguiente procedimiento instala el software del clúster y configura su sistema para la creación del clúster.

1. En cada nodo del cluster, instale los paquetes de software Red Hat High Availability Add-On junto con todos los agentes de valla disponibles en el canal de Alta Disponibilidad.

```
# yum install pcs pacemaker fence-agents-all
```

Alternativamente, puede instalar los paquetes de software de Red Hat High Availability Add-On junto con sólo el agente de valla que necesite con el siguiente comando.

```
# yum install pcs pacemaker fence-agents-model
```

El siguiente comando muestra una lista de los agentes de la valla disponibles.

```
# rpm -q -a | grep fence
fence-agents-rhevm-4.0.2-3.el7.x86_64
fence-agents-ilo-mp-4.0.2-3.el7.x86_64
fence-agents-ipmilan-4.0.2-3.el7.x86_64
...
```



AVISO

Después de instalar los paquetes del complemento de alta disponibilidad de Red Hat, debe asegurarse de que sus preferencias de actualización de software estén configuradas para que no se instale nada automáticamente. La instalación en un cluster en funcionamiento puede causar comportamientos inesperados. Para obtener más información, consulte [Prácticas recomendadas para aplicar actualizaciones de software a un cluster de alta disponibilidad o de almacenamiento resiliente de RHEL](#).

- Si está ejecutando el demonio **firewalld**, ejecute los siguientes comandos para habilitar los puertos requeridos por el complemento de alta disponibilidad de Red Hat.



NOTA

Puede determinar si el demonio **firewalld** está instalado en su sistema con el comando **rpm -q firewalld**. Si está instalado, puede determinar si se está ejecutando con el comando **firewall-cmd --state**.

```
# firewall-cmd --permanent --add-service=high-availability
# firewall-cmd --add-service=high-availability
```



NOTA

La configuración ideal del cortafuegos para los componentes del clúster depende del entorno local, en el que puede ser necesario tener en cuenta consideraciones como si los nodos tienen múltiples interfaces de red o si existe un cortafuegos fuera del host. El ejemplo que se presenta aquí, que abre los puertos que generalmente requiere un cluster Pacemaker, debe modificarse para adaptarse a las condiciones locales. [Habilitación de puertos para](#) el complemento de alta disponibilidad muestra los puertos que se deben habilitar para el complemento de alta disponibilidad de Red Hat y proporciona una explicación de para qué se utiliza cada puerto.

- Para poder utilizar **pcs** para configurar el cluster y comunicarse entre los nodos, debe establecer una contraseña en cada nodo para el usuario **hacluster**, que es la cuenta de administración **pcs**. Se recomienda que la contraseña del usuario **hacluster** sea la misma en cada nodo.

```
# passwd hacluster
Changing password for user hacluster.
New password:
Retype new password:
passwd: all authentication tokens updated successfully.
```

- Antes de poder configurar el cluster, el demonio **pcsd** debe ser iniciado y habilitado para arrancar en cada nodo. Este demonio funciona con el comando **pcs** para gestionar la configuración en todos los nodos del clúster.

En cada nodo del clúster, ejecute los siguientes comandos para iniciar el servicio **pcsd** y para habilitar **pcsd** al inicio del sistema.

```
# systemctl start pcsd.service
# systemctl enable pcsd.service
```

4.2. INSTALACIÓN DEL PAQUETE PCP-ZEROCONF (RECOMENDADO)

Cuando configure su cluster, se recomienda que instale el paquete **pcp-zeroconf** para la herramienta Performance Co-Pilot (PCP). PCP es la herramienta de monitorización de recursos recomendada por Red Hat para los sistemas RHEL. La instalación del paquete **pcp-zeroconf** le permite tener a PCP ejecutándose y recopilando datos de monitorización del rendimiento en beneficio de las investigaciones sobre cerramientos, fallos de recursos y otros eventos que perturban el cluster.



NOTA

Los despliegues de clústeres en los que se habilita PCP necesitarán suficiente espacio disponible para los datos capturados por PCP en el sistema de archivos que contiene **/var/log/pcp/**. El uso típico de espacio por parte de PCP varía según las implementaciones, pero 10Gb suelen ser suficientes cuando se utiliza la configuración predeterminada de **pcp-zeroconf**, y algunos entornos pueden requerir menos. La supervisión del uso de este directorio durante un período de 14 días de actividad típica puede proporcionar una expectativa de uso más precisa.

Para instalar el paquete **pcp-zeroconf**, ejecute el siguiente comando.

```
# yum install pcp-zeroconf
```

Este paquete permite **pmcd** y establece la captura de datos a un intervalo de 10 segundos.

Para obtener información sobre la revisión de los datos del PCP, consulte [¿Por qué se ha reiniciado un nodo del cluster de alta disponibilidad de RHEL y cómo puedo evitar que vuelva a ocurrir?](#) en el Portal del cliente de Red Hat.

4.3. CREACIÓN DE UN CLÚSTER DE ALTA DISPONIBILIDAD

Este procedimiento crea un cluster de Red Hat High Availability Add-On que consiste en los nodos **z1.example.com** y **z2.example.com**.

1. Autentifique el usuario **pcs hacluster** para cada nodo del clúster en el nodo desde el que va a ejecutar **pcs**.

El siguiente comando autentifica al usuario **hacluster** en **z1.example.com** para los dos nodos de un cluster de dos nodos que estará formado por **z1.example.com** y **z2.example.com**.

```
[root@z1 ~]# pcs host auth z1.example.com z2.example.com
Username: hacluster
Password:
z1.example.com: Authorized
z2.example.com: Authorized
```

2. Ejecute el siguiente comando desde **z1.example.com** para crear el cluster de dos nodos **my_cluster** que consta de los nodos **z1.example.com** y **z2.example.com**. Esto propagará los archivos de configuración del cluster a ambos nodos del cluster. Este comando incluye la opción

--start, que iniciará los servicios de cluster en ambos nodos del cluster.

```
[root@z1 ~]# pcs cluster setup my_cluster --start z1.example.com z2.example.com
```

- Habilite los servicios del clúster para que se ejecuten en cada nodo del clúster cuando se inicie el nodo.



NOTA

Para su entorno particular, puede optar por dejar los servicios del clúster deshabilitados omitiendo este paso. Esto le permite asegurarse de que si un nodo se cae, cualquier problema con su clúster o sus recursos se resuelve antes de que el nodo se reincorpore al clúster. Si deja los servicios de clúster deshabilitados, tendrá que iniciar manualmente los servicios cuando reinicie un nodo ejecutando el comando **pcs cluster start** en ese nodo.

```
[root@z1 ~]# pcs cluster enable --all
```

Puede mostrar el estado actual del clúster con el comando **pcs cluster status**. Debido a que puede haber un ligero retraso antes de que el clúster esté en funcionamiento cuando inicie los servicios del clúster con la opción **--start** del comando **pcs cluster setup**, debe asegurarse de que el clúster esté en funcionamiento antes de realizar cualquier acción posterior en el clúster y su configuración.

```
[root@z1 ~]# pcs cluster status
Cluster Status:
Stack: corosync
Current DC: z2.example.com (version 2.0.0-10.el8-b67d8d0de9) - partition with quorum
Last updated: Thu Oct 11 16:11:18 2018
Last change: Thu Oct 11 16:11:00 2018 by hacluster via crmd on z2.example.com
2 Nodes configured
0 Resources configured
...
```

4.4. CREACIÓN DE UN CLUSTER DE ALTA DISPONIBILIDAD CON MÚLTIPLES ENLACES

Puede utilizar el comando **pcs cluster setup** para crear un cluster de Alta Disponibilidad de Red Hat con múltiples enlaces especificando todos los enlaces para cada nodo.

El formato del comando para crear un cluster de dos nodos con dos enlaces es el siguiente.

```
pcs cluster setup cluster_name node1_name addr=node1_link0_address addr=node1_link1_address
node2_name addr=node2_link0_address addr=node2_link1_address
```

Al crear un clúster con múltiples enlaces, debe tener en cuenta lo siguiente.

- El orden de los **addr=address** es importante. La primera dirección especificada después de un nombre de nodo es para **link0**, la segunda para **link1**, y así sucesivamente.
- Es posible especificar hasta ocho enlaces utilizando el protocolo de transporte **knet**, que es el protocolo de transporte por defecto.

- Todos los nodos deben tener el mismo número de parámetros **addr=**.
- A partir de RHEL 8.1, es posible añadir, eliminar y cambiar enlaces en un clúster existente utilizando los comandos **pcs cluster link add**, **pcs cluster link remove**, **pcs cluster link delete** y **pcs cluster link update**.
- Al igual que con los clusters de un solo enlace, no mezcles direcciones IPv4 e IPv6 en un enlace, aunque puedes tener un enlace ejecutando IPv4 y el otro ejecutando IPv6.
- Al igual que con los clústeres de un solo enlace, puede especificar las direcciones como direcciones IP o como nombres siempre que los nombres se resuelvan con direcciones IPv4 o IPv6 para las que no se mezclen direcciones IPv4 e IPv6 en un enlace.

El siguiente ejemplo crea un cluster de dos nodos llamado **my_twolink_cluster** con dos nodos, **rh80-node1** y **rh80-node2**. **rh80-node1** tiene dos interfaces, la dirección IP 192.168.122.201 como **link0** y 192.168.123.201 como **link1**. **rh80-node2** tiene dos interfaces, la dirección IP 192.168.122.202 como **link0** y 192.168.123.202 como **link1**.

```
# pcs cluster setup my_twolink_cluster rh80-node1 addr=192.168.122.201
addr=192.168.123.201 rh80-node2 addr=192.168.122.202 addr=192.168.123.202
```

Para obtener información sobre cómo añadir nodos a un clúster existente con [múltiples enlaces](#), consulte [Añadir un nodo a un clúster con múltiples enlaces](#) .

Para obtener información sobre la modificación de los enlaces en un clúster existente con múltiples enlaces, consulte [Añadir y modificar enlaces en un clúster existente](#) .

4.5. CONFIGURACIÓN DE LAS VALLAS

Debe configurar un dispositivo de cercado para cada nodo en el cluster. Para obtener información sobre los comandos y opciones de configuración de fencing, consulte [Configuración de fencing en un cluster de Red Hat High Availability](#).

Para obtener información general sobre el cercado y su importancia en un [cluster](#) de alta disponibilidad de Red Hat, consulte [Cerrado en un cluster de alta disponibilidad de Red Hat](#) .



NOTA

Al configurar un dispositivo de vallado, se debe prestar atención a si ese dispositivo comparte la energía con cualquier nodo o dispositivo del clúster. Si un nodo y su dispositivo de vallado comparten la alimentación, el clúster puede correr el riesgo de no poder vallar ese nodo si se pierde la alimentación de éste y de su dispositivo de vallado. Un clúster de este tipo debería tener fuentes de alimentación redundantes para los dispositivos de vallado y los nodos, o dispositivos de vallado redundantes que no compartan la energía. Los métodos alternativos de cercado, como el SBD o el cercado de almacenamiento, también pueden aportar redundancia en caso de pérdidas aisladas de energía.

Este ejemplo utiliza el conmutador de potencia APC con un nombre de host de **zapc.example.com** para cercar los nodos, y utiliza el agente de cercado **fence_apc_snmp**. Dado que ambos nodos serán cercados por el mismo agente de cercado, puede configurar ambos dispositivos de cercado como un único recurso, utilizando la opción **pcmk_host_map**.

Se crea un dispositivo de esgrima configurando el dispositivo como un recurso **stonith** con el comando **pcs stonith create**. El siguiente comando configura un recurso **stonith** llamado **myapc** que utiliza el

agente de esgrima **fence_apc_snmp** para los nodos **z1.example.com** y **z2.example.com**. La opción **pcmk_host_map** asigna **z1.example.com** al puerto 1, y **z2.example.com** al puerto 2. El valor de inicio de sesión y la contraseña para el dispositivo APC son ambos **apc**. Por defecto, este dispositivo utilizará un intervalo de monitorización de sesenta segundos para cada nodo.

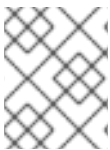
Tenga en cuenta que puede utilizar una dirección IP cuando especifique el nombre de host para los nodos.

```
[root@z1 ~]# pcs stonith create myapc fence_apc_snmp \
ipaddr="zpc.example.com" \
pcmk_host_map="z1.example.com:1;z2.example.com:2" \
login="apc" passwd="apc"
```

El siguiente comando muestra los parámetros de un dispositivo STONITH existente.

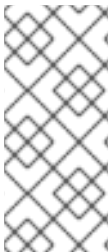
```
[root@rh7-1 ~]# pcs stonith config myapc
Resource: myapc (class=stonith type=fence_apc_snmp)
Attributes: ipaddr=zpc.example.com pcmk_host_map=z1.example.com:1;z2.example.com:2
login=apc passwd=apc
Operations: monitor interval=60s (myapc-monitor-interval-60s)
```

Después de configurar su dispositivo de valla, debe probar el dispositivo. Para obtener información sobre cómo probar un dispositivo de valla, consulte [Probar un dispositivo de valla](#).



NOTA

No pruebe su dispositivo de vallado desactivando la interfaz de red, ya que esto no probará correctamente el vallado.



NOTA

Una vez que se ha configurado el cercado y se ha iniciado un clúster, un reinicio de la red desencadenará el cercado para el nodo que reinicie la red incluso cuando no se haya superado el tiempo de espera. Por este motivo, no reinicie el servicio de red mientras el servicio de clúster esté en funcionamiento, ya que activará el cercado involuntario en el nodo.

4.6. COPIA DE SEGURIDAD Y RESTAURACIÓN DE LA CONFIGURACIÓN DE UN CLÚSTER

Puede hacer una copia de seguridad de la configuración del clúster en un archivo tar con el siguiente comando. Si no especifica un nombre de archivo, se utilizará la salida estándar.

```
pcs config backup filename
```



NOTA

El comando **pcs config backup** hace una copia de seguridad sólo de la configuración del clúster en sí, tal y como está configurada en el CIB; la configuración de los daemons de recursos está fuera del alcance de este comando. Por ejemplo, si ha configurado un recurso Apache en el clúster, se hará una copia de seguridad de la configuración del recurso (que está en el CIB), mientras que no se hará una copia de seguridad de la configuración del demonio Apache (como se establece en `/etc/httpd``) ni de los archivos que sirve. Del mismo modo, si hay un recurso de base de datos configurado en el clúster, la base de datos en sí no será respaldada, mientras que la configuración del recurso de base de datos (CIB) sí lo será.

Utilice el siguiente comando para restaurar los archivos de configuración del clúster en todos los nodos desde la copia de seguridad. Si no se especifica un nombre de archivo, se utilizará la entrada estándar. Si se especifica la opción **--local**, sólo se restaurarán los archivos del nodo actual.

```
pcs config restore [--local] [filename]
```

4.7. HABILITACIÓN DE PUERTOS PARA EL COMPLEMENTO DE ALTA DISPONIBILIDAD

La configuración ideal del cortafuegos para los componentes del clúster depende del entorno local, donde puede ser necesario tener en cuenta consideraciones tales como si los nodos tienen múltiples interfaces de red o si existe un cortafuegos fuera del host.

Si está ejecutando el demonio **firewalld**, ejecute los siguientes comandos para habilitar los puertos requeridos por el complemento de alta disponibilidad de Red Hat.

```
# firewall-cmd --permanent --add-service=high-availability
# firewall-cmd --add-service=high-availability
```

Es posible que tenga que modificar qué puertos están abiertos para adaptarse a las condiciones locales.



NOTA

Puede determinar si el demonio **firewalld** está instalado en su sistema con el comando **rpm -q firewalld**. Si el demonio **firewalld** está instalado, puede determinar si se está ejecutando con el comando **firewall-cmd --state**.

Tabla 4.1, “Puertos a habilitar para el complemento de alta disponibilidad” muestra los puertos a habilitar para el Complemento de Alta Disponibilidad de Red Hat y proporciona una explicación de para qué se utiliza el puerto.

Tabla 4.1. Puertos a habilitar para el complemento de alta disponibilidad

Puerto	Cuando sea necesario
--------	----------------------

Puerto	Cuando sea necesario
TCP 2224	<p>Puerto por defecto pcsd requerido en todos los nodos (necesario para la UI Web pcsd y requerido para la comunicación entre nodos). Puede configurar el puerto pcsd mediante el parámetro PCSD_PORT en el archivo <code>/etc/sysconfig/pcsd</code>.</p> <p>Es crucial abrir el puerto 2224 de forma que pcs desde cualquier nodo pueda hablar con todos los nodos del clúster, incluido él mismo. Cuando se utiliza el gestor de tickets del clúster de Booth o un dispositivo de quórum, se debe abrir el puerto 2224 en todos los hosts relacionados, como los árbitros de Booth o el host del dispositivo de quórum.</p>
TCP 3121	<p>Requerido en todos los nodos si el cluster tiene nodos Pacemaker Remote</p> <p>El demonio pacemaker-based de Pacemaker en los nodos del cluster completo contactará con el demonio pacemaker_remoted en los nodos Pacemaker Remote en el puerto 3121. Si se utiliza una interfaz separada para la comunicación del clúster, el puerto sólo debe estar abierto en esa interfaz. Como mínimo, el puerto debe estar abierto en los nodos Pacemaker Remote hacia los nodos del cluster completo. Dado que los usuarios pueden convertir un host entre un nodo completo y un nodo remoto, o ejecutar un nodo remoto dentro de un contenedor utilizando la red del host, puede ser útil abrir el puerto en todos los nodos. No es necesario abrir el puerto a otros hosts que no sean nodos.</p>
TCP 5403	<p>Se requiere en el host del dispositivo de quórum cuando se utiliza un dispositivo de quórum con corosync-qnetd. El valor por defecto se puede cambiar con la opción -p del comando corosync-qnetd.</p>
UDP 5404-5412	<p>Se requiere en los nodos corosync para facilitar la comunicación entre nodos. Es crucial abrir los puertos 5404-5412 de forma que corosync desde cualquier nodo pueda hablar con todos los nodos del clúster, incluido él mismo.</p>
TCP 21064	<p>Se requiere en todos los nodos si el cluster contiene algún recurso que requiera DLM (como GFS2).</p>

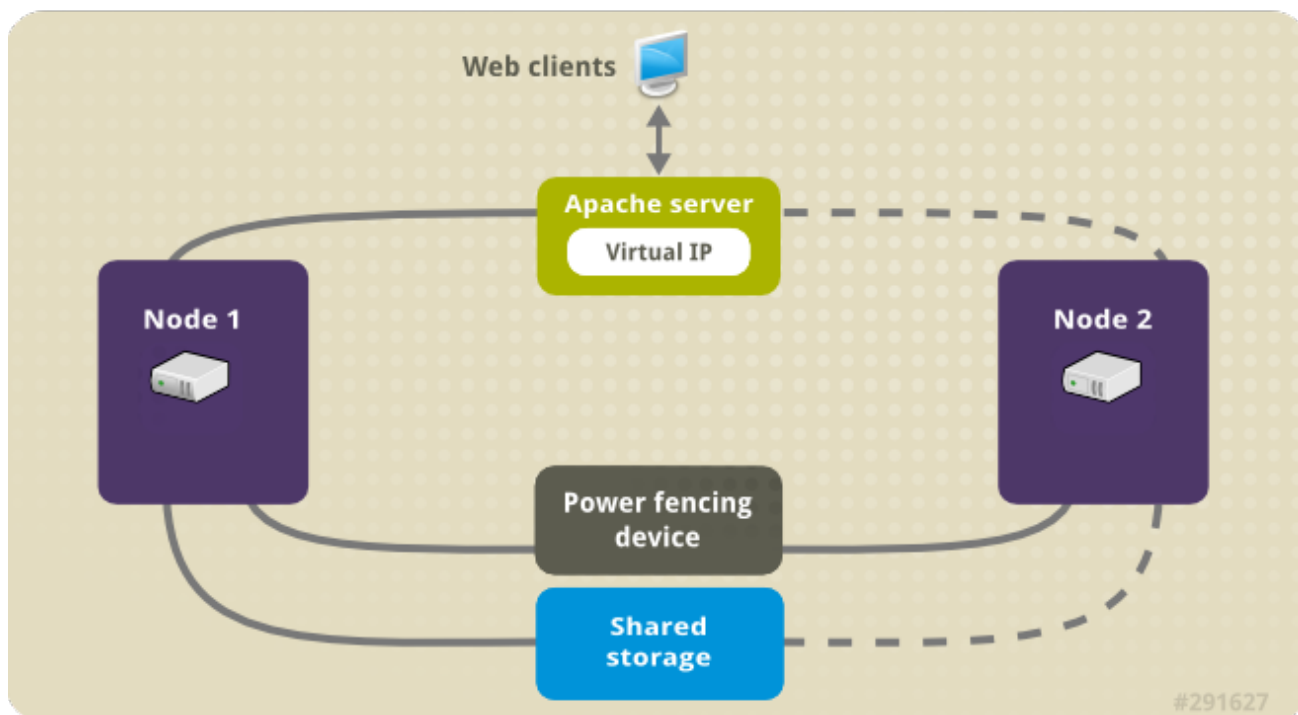
Puerto	Cuando sea necesario
TCP 9929, UDP 9929	Se requiere que esté abierto en todos los nodos del clúster y en los nodos del árbitro de la cabina para las conexiones desde cualquiera de esos mismos nodos cuando se utiliza el gestor de tickets de la cabina para establecer un clúster multisitio.

CAPÍTULO 5. CONFIGURACIÓN DE UN SERVIDOR HTTP APACHE ACTIVO/PASIVO EN UN CLUSTER DE ALTA DISPONIBILIDAD DE RED HAT

El siguiente procedimiento configura un servidor HTTP Apache activo/pasivo en un cluster de dos nodos de Red Hat Enterprise Linux High Availability Add-On utilizando **pcs** para configurar los recursos del cluster. En este caso de uso, los clientes acceden al servidor Apache HTTP a través de una dirección IP flotante. El servidor web se ejecuta en uno de los dos nodos del cluster. Si el nodo en el que se ejecuta el servidor web queda inoperativo, el servidor web vuelve a arrancar en el segundo nodo del cluster con una mínima interrupción del servicio.

Figura 5.1, “Apache en un cluster de alta disponibilidad de dos nodos de Red Hat” muestra una visión general de alto nivel del clúster en el que El clúster es un clúster de alta disponibilidad de Red Hat de dos nodos que está configurado con un interruptor de alimentación de red y con almacenamiento compartido. Los nodos del cluster están conectados a una red pública, para el acceso de los clientes al servidor Apache HTTP a través de una IP virtual. El servidor Apache se ejecuta en el Nodo 1 o en el Nodo 2, cada uno de los cuales tiene acceso al almacenamiento en el que se guardan los datos de Apache. En esta ilustración, el servidor web se ejecuta en el Nodo 1, mientras que el Nodo 2 está disponible para ejecutar el servidor si el Nodo 1 queda inoperativo.

Figura 5.1. Apache en un cluster de alta disponibilidad de dos nodos de Red Hat



Este caso de uso requiere que su sistema incluya los siguientes componentes:

- Un clúster de alta disponibilidad de Red Hat de dos nodos con un cercado de energía configurado para cada nodo. Recomendamos pero no requerimos una red privada. Este procedimiento utiliza el ejemplo de cluster proporcionado en [Creación de un cluster de alta disponibilidad de Red Hat con Pacemaker](#).
- Una dirección IP virtual pública, necesaria para Apache.
- Almacenamiento compartido para los nodos del clúster, utilizando iSCSI, Fibre Channel u otro dispositivo de bloque de red compartido.

El clúster está configurado con un grupo de recursos Apache, que contiene los componentes del clúster

que requiere el servidor web: un recurso LVM, un recurso de sistema de archivos, un recurso de dirección IP y un recurso de servidor web. Este grupo de recursos puede pasar de un nodo del clúster a otro, permitiendo que cualquiera de ellos ejecute el servidor web. Antes de crear el grupo de recursos para este clúster, usted realizará los siguientes procedimientos:

1. Configurar un sistema de archivos **ext4** en el volumen lógico **my_lv**.
2. Configurar un servidor web.

Después de realizar estos pasos, se crea el grupo de recursos y los recursos que contiene.

5.1. CONFIGURACIÓN DE UN VOLUMEN LVM CON UN SISTEMA DE ARCHIVOS EXT4 EN UN CLUSTER PACEMAKER

Este caso de uso requiere que se cree un volumen lógico LVM en el almacenamiento que se comparte entre los nodos del clúster.



NOTA

Los volúmenes LVM y las correspondientes particiones y dispositivos utilizados por los nodos del clúster deben estar conectados únicamente a los nodos del clúster.

El siguiente procedimiento crea un volumen lógico LVM y luego crea un sistema de archivos ext4 en ese volumen para utilizarlo en un cluster de Pacemaker. En este ejemplo, la partición compartida **/dev/sdb1** se utiliza para almacenar el volumen físico LVM a partir del cual se creará el volumen lógico LVM.

1. En ambos nodos del clúster, realice los siguientes pasos para establecer el valor del ID del sistema LVM al valor del identificador **uname** para el sistema. El ID del sistema LVM se utilizará para garantizar que sólo el clúster sea capaz de activar el grupo de volúmenes.
 - a. Establezca la opción de configuración **system_id_source** en el archivo de configuración **/etc/lvm/lvm.conf** como **uname**.

```
# Configuration option global/system_id_source.
system_id_source = "uname"
```

- b. Verifique que el ID del sistema LVM en el nodo coincide con el **uname** para el nodo.

```
# lvm systemid
system ID: z1.example.com
# uname -n
z1.example.com
```

2. Cree el volumen LVM y cree un sistema de archivos ext4 en ese volumen. Como la partición **/dev/sdb1** es un almacenamiento compartido, esta parte del procedimiento se realiza en un solo nodo.
 - a. Crear un volumen físico LVM en la partición **/dev/sdb1**.

```
# pvcreate /dev/sdb1
Physical volume "/dev/sdb1" successfully created
```

- b. Cree el grupo de volúmenes **my_vg** que consiste en el volumen físico **/dev/sdb1**.

```
# vgcreate my_vg /dev/sdb1
Volume group "my_vg" successfully created
```

- c. Comprueba que el nuevo grupo de volúmenes tiene el ID del sistema del nodo en el que se está ejecutando y desde el que se creó el grupo de volúmenes.

```
# vgs -o+systemid
VG   #PV #LV #SN Attr   VSize  VFree  System ID
my_vg 1   0   0 wz--n- <1.82t <1.82t z1.example.com
```

- d. Crear un volumen lógico utilizando el grupo de volumen **my_vg**.

```
# lvcreate -L450 -n my_lv my_vg
Rounding up size to full physical extent 452.00 MiB
Logical volume "my_lv" created
```

Puede utilizar el comando **lvs** para mostrar el volumen lógico.

```
# lvs
LV   VG   Attr   LSize  Pool Origin Data%  Move Log Copy%  Convert
my_lv my_vg -wi-a---- 452.00m
...
```

- e. Crear un sistema de archivos ext4 en el volumen lógico **my_lv**.

```
# mkfs.ext4 /dev/my_vg/my_lv
mke2fs 1.44.3 (10-July-2018)
Creating filesystem with 462848 1k blocks and 115824 inodes
...
```

5.2. CONFIGURACIÓN DE UN SERVIDOR HTTP APACHE

El siguiente procedimiento configura un servidor HTTP Apache.

1. Asegúrese de que el servidor HTTP Apache está instalado en cada nodo del clúster. También necesita la herramienta **wget** instalada en el clúster para poder comprobar el estado del servidor HTTP Apache.

En cada nodo, ejecute el siguiente comando.

```
# yum install -y httpd wget
```

Si está ejecutando el demonio **firewalld**, en cada nodo del cluster habilite los puertos requeridos por el complemento de alta disponibilidad de Red Hat.

```
# firewall-cmd --permanent --add-service=high-availability
# firewall-cmd --reload
```

2. Para que el agente de recursos de Apache obtenga el estado del servidor HTTP Apache, asegúrese de que el siguiente texto está presente en el archivo **/etc/httpd/conf/httpd.conf** en cada nodo del clúster, y asegúrese de que no ha sido comentado. Si este texto no está ya presente, añádalo al final del archivo.

```
<Location /server-status>
  SetHandler server-status
  Require local
</Location>
```

3. Cuando se utiliza el agente de recursos **apache** para gestionar Apache, no se utiliza **systemd**. Por ello, debe editar el script **logrotate** suministrado con Apache para que no utilice **systemctl** para recargar Apache. Elimine la siguiente línea en el archivo **/etc/logrotate.d/httpd** en cada nodo del clúster.

```
/bin/systemctl reload httpd.service > /dev/null 2>/dev/null || true
```

Sustituya la línea que ha eliminado por las tres líneas siguientes.

```
/usr/bin/test -f /run/httpd.pid >/dev/null 2>/dev/null &&
/usr/bin/ps -q $(/usr/bin/cat /run/httpd.pid) >/dev/null 2>/dev/null &&
/usr/sbin/httpd -f /etc/httpd/conf/httpd.conf \
-c "PidFile /run/httpd.pid" -k graceful > /dev/null 2>/dev/null || true
```

4. Cree una página web para que Apache la sirva. En un nodo del clúster, monte el sistema de archivos que creó en [Configurar un volumen LVM con un sistema de archivos ext4](#), cree el archivo **index.html** en ese sistema de archivos y luego desmonte el sistema de archivos.

```
# mount /dev/my_vg/my_lv /var/www/
# mkdir /var/www/html
# mkdir /var/www/cgi-bin
# mkdir /var/www/error
# restorecon -R /var/www
# cat <<-END >/var/www/html/index.html
<html>
<body>Hello</body>
</html>
END
# umount /var/www
```

5.3. CREACIÓN DE LOS RECURSOS Y GRUPOS DE RECURSOS

Este caso de uso requiere que se creen cuatro recursos de cluster. Para garantizar que todos estos recursos se ejecutan en el mismo nodo, se configuran como parte del grupo de recursos **apachegroup**. Los recursos a crear son los siguientes, listados en el orden en que se iniciarán.

1. Un recurso **LVM** llamado **my_lvm** que utiliza el grupo de volúmenes LVM que creó en [Configuración de un volumen LVM con un sistema de archivos ext4](#).
2. Un recurso **Filesystem** llamado **my_fs**, que utiliza el dispositivo del sistema de archivos **/dev/my_vg/my_lv** que creaste en [Configuración de un volumen LVM con un sistema de archivos ext4](#).
3. Un recurso **IPaddr2**, que es una dirección IP flotante para el grupo de recursos **apachegroup**. La dirección IP no debe ser una ya asociada a un nodo físico. Si no se especifica el dispositivo NIC del recurso **IPaddr2**, la IP flotante debe residir en la misma red que una de las direcciones IP asignadas estáticamente del nodo, de lo contrario, el dispositivo NIC para asignar la dirección IP flotante no puede ser detectado correctamente.

- Un recurso **apache** llamado **Website** que utiliza el archivo **index.html** y la configuración de Apache que definió en [Configuración de un servidor HTTP Apache](#) .

El siguiente procedimiento crea el grupo de recursos **apachegroup** y los recursos que contiene el grupo. Los recursos se iniciarán en el orden en que los añada al grupo y se detendrán en el orden inverso en que se añadan al grupo. Ejecute este procedimiento desde un solo nodo del clúster.

- El siguiente comando crea el recurso **LVM-activate my_lvm** . Dado que el grupo de recursos **apachegroup** aún no existe, este comando crea el grupo de recursos.



NOTA

No configure más de un recurso **LVM-activate** que utilice el mismo grupo de volúmenes LVM en una configuración de HA activa/pasiva, ya que esto podría causar la corrupción de los datos. Además, no configure un recurso **LVM-activate** como recurso clónico en una configuración de HA activa/pasiva.

```
[root@z1 ~]# pcs resource create my_lvm ocf:heartbeat:LVM-activate vgname=my_vg
vg_access_mode=system_id --group apachegroup
```

Cuando se crea un recurso, éste se inicia automáticamente. Puede utilizar el siguiente comando para confirmar que el recurso se ha creado y se ha iniciado.

```
# pcs resource status
Resource Group: apachegroup
my_lvm (ocf::heartbeat:LVM-activate): Started
```

Puede detener e iniciar manualmente un recurso individual con los comandos **pcs resource disable** y **pcs resource enable**.

- Los siguientes comandos crean los recursos restantes para la configuración, añadiéndolos al grupo de recursos existente **apachegroup**.

```
[root@z1 ~]# pcs resource create my_fs Filesystem \
device="/dev/my_vg/my_lv" directory="/var/www" fstype="ext4" \
--group apachegroup
```

```
[root@z1 ~]# pcs resource create VirtualIP IPAddr2 ip=198.51.100.3 \
cidr_netmask=24 --group apachegroup
```

```
[root@z1 ~]# pcs resource create Website apache \
configfile="/etc/httpd/conf/httpd.conf" \
statusurl="http://127.0.0.1/server-status" --group apachegroup
```

- Después de crear los recursos y el grupo de recursos que los contiene, puede comprobar el estado del clúster. Observe que los cuatro recursos se ejecutan en el mismo nodo.

```
[root@z1 ~]# pcs status
Cluster name: my_cluster
Last updated: Wed Jul 31 16:38:51 2013
Last change: Wed Jul 31 16:42:14 2013 via crm_attribute on z1.example.com
Stack: corosync
Current DC: z2.example.com (2) - partition with quorum
Version: 1.1.10-5.el7-9abe687
```

```
2 Nodes configured
6 Resources configured
```

```
Online: [ z1.example.com z2.example.com ]
```

Full list of resources:

```
myapc (stonith:fence_apc_snmp): Started z1.example.com
Resource Group: apachegroup
my_lvm (ocf::heartbeat:LVM): Started z1.example.com
my_fs (ocf::heartbeat:Filesystem): Started z1.example.com
VirtualIP (ocf::heartbeat:IPAddr2): Started z1.example.com
Website (ocf::heartbeat:apache): Started z1.example.com
```

Tenga en cuenta que si no ha configurado un dispositivo de cercado para su clúster, por defecto los recursos no se inician.

- Una vez que el clúster esté en funcionamiento, puede dirigir un navegador a la dirección IP que definió como recurso **IPAddr2** para ver la pantalla de muestra, que consiste en la simple palabra "Hola".

```
Hola
```

Si encuentra que los recursos que configuró no están funcionando, puede ejecutar el comando **pcs resource debug-start resource** para probar la configuración de los recursos.

5.4. PROBAR LA CONFIGURACIÓN DE LOS RECURSOS

En la pantalla de estado del clúster que se muestra en [Creación de los recursos y grupos de recursos](#), todos los recursos se están ejecutando en el nodo **z1.example.com**. Puede probar si el grupo de recursos falla en el nodo **z2.example.com** utilizando el siguiente procedimiento para poner el primer nodo en modo **standby**, tras lo cual el nodo ya no podrá alojar recursos.

- El siguiente comando pone el nodo **z1.example.com** en modo **standby**.

```
[root@z1 ~]# pcs node standby z1.example.com
```

- Después de poner el nodo **z1** en modo **standby**, compruebe el estado del clúster. Observe que los recursos deberían estar ahora todos funcionando en **z2**.

```
[root@z1 ~]# pcs status
Cluster name: my_cluster
Last updated: Wed Jul 31 17:16:17 2013
Last change: Wed Jul 31 17:18:34 2013 via crm_attribute on z1.example.com
Stack: corosync
Current DC: z2.example.com (2) - partition with quorum
Version: 1.1.10-5.el7-9abe687
2 Nodes configured
6 Resources configured
```

```
Node z1.example.com (1): standby
Online: [ z2.example.com ]
```

Full list of resources:

```
myapc (stonith:fence_apc_snmp): Started z1.example.com
```

```
Resource Group: apachegroup  
my_lvm (ocf::heartbeat:LVM): Started z2.example.com  
my_fs (ocf::heartbeat:Filesystem): Started z2.example.com  
VirtualIP (ocf::heartbeat:IPaddr2): Started z2.example.com  
Website (ocf::heartbeat:apache): Started z2.example.com
```

El sitio web en la dirección IP definida debería seguir mostrándose, sin interrupción.

3. Para eliminar **z1** del modo **standby**, introduzca el siguiente comando.

```
[root@z1 ~]# pcs node unstandby z1.example.com
```



NOTA

Sacar un nodo del modo **standby** no provoca por sí mismo que los recursos vuelvan a fallar en ese nodo. Esto dependerá del valor de **resource-stickiness** para los recursos. Para obtener información sobre el metaatributo **resource-stickiness**, consulte [Configurar un recurso para que prefiera su nodo actual](#) .

CAPÍTULO 6. CONFIGURACIÓN DE UN SERVIDOR NFS ACTIVO/PASIVO EN UN CLUSTER DE ALTA DISPONIBILIDAD DE RED HAT

El siguiente procedimiento configura un servidor NFS activo/pasivo de alta disponibilidad en un cluster de dos nodos de Red Hat Enterprise Linux High Availability Add-On usando almacenamiento compartido. El procedimiento utiliza **pcs** para configurar los recursos del cluster Pacemaker. En este caso de uso, los clientes acceden al sistema de archivos NFS a través de una dirección IP flotante. El servidor NFS se ejecuta en uno de los dos nodos del clúster. Si el nodo en el que se ejecuta el servidor NFS deja de funcionar, el servidor NFS vuelve a arrancar en el segundo nodo del cluster con una mínima interrupción del servicio.

6.1. REQUISITOS PREVIOS

Este caso de uso requiere que su sistema incluya los siguientes componentes:

- Un clúster de alta disponibilidad de Red Hat de dos nodos con un cercado de energía configurado para cada nodo. Recomendamos pero no requerimos una red privada. Este procedimiento utiliza el ejemplo de cluster proporcionado en [Creación de un cluster de alta disponibilidad de Red Hat con Pacemaker](#).
- Una dirección IP virtual pública, necesaria para el servidor NFS.
- Almacenamiento compartido para los nodos del clúster, utilizando iSCSI, Fibre Channel u otro dispositivo de bloque de red compartido.

6.2. RESUMEN DEL PROCEDIMIENTO

La configuración de un servidor NFS activo/pasivo de alta disponibilidad en un cluster de alta disponibilidad de dos nodos de Red Hat Enterprise Linux requiere que realice los siguientes pasos:

1. Configure un sistema de archivos **ext4** en el volumen lógico LVM **my_lv** en el almacenamiento compartido para los nodos del clúster.
2. Configure un recurso compartido NFS en el almacenamiento compartido en el volumen lógico LVM.
3. Crear los recursos del clúster.
4. Pruebe el servidor NFS que ha configurado.

6.3. CONFIGURACIÓN DE UN VOLUMEN LVM CON UN SISTEMA DE ARCHIVOS EXT4 EN UN CLUSTER PACEMAKER

Este caso de uso requiere que se cree un volumen lógico LVM en el almacenamiento que se comparte entre los nodos del clúster.



NOTA

Los volúmenes LVM y las correspondientes particiones y dispositivos utilizados por los nodos del clúster deben estar conectados únicamente a los nodos del clúster.

El siguiente procedimiento crea un volumen lógico LVM y luego crea un sistema de archivos ext4 en ese volumen para utilizarlo en un cluster de Pacemaker. En este ejemplo, la partición compartida **/dev/sdb1** se utiliza para almacenar el volumen físico LVM a partir del cual se creará el volumen lógico LVM.

1. En ambos nodos del clúster, realice los siguientes pasos para establecer el valor del ID del sistema LVM al valor del identificador **uname** para el sistema. El ID del sistema LVM se utilizará para garantizar que sólo el clúster sea capaz de activar el grupo de volúmenes.
 - a. Establezca la opción de configuración **system_id_source** en el archivo de configuración **/etc/lvm/lvm.conf** como **uname**.

```
# Configuration option global/system_id_source.
system_id_source = "uname"
```

- b. Verifique que el ID del sistema LVM en el nodo coincide con el **uname** para el nodo.

```
# lvm systemid
system ID: z1.example.com
# uname -n
z1.example.com
```

2. Cree el volumen LVM y cree un sistema de archivos ext4 en ese volumen. Como la partición **/dev/sdb1** es un almacenamiento compartido, esta parte del procedimiento se realiza en un solo nodo.

- a. Crear un volumen físico LVM en la partición **/dev/sdb1**.

```
# pvcreate /dev/sdb1
Physical volume "/dev/sdb1" successfully created
```

- b. Cree el grupo de volúmenes **my_vg** que consiste en el volumen físico **/dev/sdb1**.

```
# vgcreate my_vg /dev/sdb1
Volume group "my_vg" successfully created
```

- c. Comprueba que el nuevo grupo de volúmenes tiene el ID del sistema del nodo en el que se está ejecutando y desde el que se creó el grupo de volúmenes.

```
# vgs -o+systemid
VG #PV #LV #SN Attr VSize VFree System ID
my_vg 1 0 0 wz--n- <1.82t <1.82t z1.example.com
```

- d. Crear un volumen lógico utilizando el grupo de volumen **my_vg**.

```
# lvcreate -L450 -n my_lv my_vg
Rounding up size to full physical extent 452.00 MiB
Logical volume "my_lv" created
```

Puede utilizar el comando **lvs** para mostrar el volumen lógico.

```
# lvs
LV VG Attr LSize Pool Origin Data% Move Log Copy% Convert
my_lv my_vg -wi-a---- 452.00m
...
```


- e. Crear un sistema de archivos ext4 en el volumen lógico **my_lv**.

```
# mkfs.ext4 /dev/my_vg/my_lv
mke2fs 1.44.3 (10-July-2018)
Creating filesystem with 462848 1k blocks and 115824 inodes
...
```

6.4. CONFIGURACIÓN DE UN RECURSO COMPARTIDO NFS

El siguiente procedimiento configura el recurso compartido NFS para la conmutación por error del servicio NFS.

1. En ambos nodos del clúster, cree el directorio **/nfsshare**.

```
# mkdir /nfsshare
```

2. En un nodo del clúster, realice el siguiente procedimiento.
 - a. Monte el sistema de archivos ext4 que creó en [Configuración de un volumen LVM con un sistema de archivos ext4](#) en el directorio **/nfsshare**.

```
[root@z1 ~]# mount /dev/my_vg/my_lv /nfsshare
```

- b. Cree un árbol de directorios **exports** en el directorio **/nfsshare**.

```
[root@z1 ~]# mkdir -p /nfsshare/exports
[root@z1 ~]# mkdir -p /nfsshare/exports/export1
[root@z1 ~]# mkdir -p /nfsshare/exports/export2
```

- c. Coloque los archivos en el directorio **exports** para que los clientes NFS puedan acceder a ellos. Para este ejemplo, estamos creando archivos de prueba llamados **clientdatafile1** y **clientdatafile2**.

```
[root@z1 ~]# touch /nfsshare/exports/export1/clientdatafile1
[root@z1 ~]# touch /nfsshare/exports/export2/clientdatafile2
```

- d. Desmonte el sistema de archivos ext4 y desactive el grupo de volúmenes LVM.

```
[root@z1 ~]# umount /dev/my_vg/my_lv
[root@z1 ~]# vgchange -an my_vg
```

6.5. CONFIGURACIÓN DE LOS RECURSOS Y DEL GRUPO DE RECURSOS PARA UN SERVIDOR NFS EN UN CLÚSTER

Esta sección proporciona el procedimiento para configurar los recursos del clúster para este caso de uso.



NOTA

Si no ha configurado un dispositivo de cercado para su clúster, por defecto los recursos no se inician.

Si encuentra que los recursos que configuró no están funcionando, puede ejecutar el comando **pcs resource debug-start resource** para probar la configuración de los recursos. Esto inicia el servicio fuera del control y conocimiento del cluster. En el momento en que los recursos configurados vuelvan a funcionar, ejecute **pcs resource cleanup resource** para que el clúster conozca las actualizaciones.

El siguiente procedimiento configura los recursos del sistema. Para garantizar que todos estos recursos se ejecutan en el mismo nodo, se configuran como parte del grupo de recursos **nfsgroup**. Los recursos se iniciarán en el orden en el que los añada al grupo, y se detendrán en el orden inverso en el que se añaden al grupo. Ejecute este procedimiento desde un solo nodo del clúster.

1. Crea el recurso LVM-activate llamado **my_lvm**. Dado que el grupo de recursos **nfsgroup** aún no existe, este comando crea el grupo de recursos.



AVISO

No configure más de un recurso **LVM-activate** que utilice el mismo grupo de volúmenes LVM en una configuración de HA activa/pasiva, ya que se corre el riesgo de corromper los datos. Además, no configure un recurso **LVM-activate** como recurso clónico en una configuración de HA activa/pasiva.

```
[root@z1 ~]# pcs resource create my_lvm ocf:heartbeat:LVM-activate vgname=my_vg
vg_access_mode=system_id --group nfsgroup
```

2. Compruebe el estado del clúster para verificar que el recurso está en funcionamiento.

```
root@z1 ~]# pcs status
Cluster name: my_cluster
Last updated: Thu Jan  8 11:13:17 2015
Last change: Thu Jan  8 11:13:08 2015
Stack: corosync
Current DC: z2.example.com (2) - partition with quorum
Version: 1.1.12-a14efad
2 Nodes configured
3 Resources configured

Online: [ z1.example.com z2.example.com ]

Full list of resources:
myapc (stonith:fence_apc_snmp):   Started z1.example.com
Resource Group: nfsgroup
  my_lvm (ocf::heartbeat:LVM):   Started z1.example.com

PCSD Status:
```

```
z1.example.com: Online
z2.example.com: Online
```

Daemon Status:

```
corosync: active/enabled
pacemaker: active/enabled
pcsd: active/enabled
```

3. Configurar un recurso **Filesystem** para el cluster.

El siguiente comando configura un recurso ext4 **Filesystem** llamado **nfsshare** como parte del grupo de recursos **nfsgroup**. Este sistema de archivos utiliza el grupo de volúmenes LVM y el sistema de archivos ext4 que creó en [Configuración de un volumen LVM con un sistema de archivos ext4](#) y se montará en el directorio **/nfsshare** que creó en [Configuración de un recurso compartido NFS](#).

```
[root@z1 ~]# pcs resource create nfsshare Filesystem \
device=/dev/my_vg/my_lv directory=/nfsshare \
fstype=ext4 --group nfsgroup
```

Puede especificar las opciones de montaje como parte de la configuración de un recurso **Filesystem** con el parámetro **options=options** parámetro. Ejecute el comando **pcs resource describe Filesystem** para obtener las opciones de configuración completas.

4. Compruebe que los recursos **my_lvm** y **nfsshare** están en funcionamiento.

```
[root@z1 ~]# pcs status
...
Full list of resources:
myapc (stonith:fence_apc_snmp): Started z1.example.com
Resource Group: nfsgroup
  my_lvm (ocf::heartbeat:LVM): Started z1.example.com
  nfsshare (ocf::heartbeat:Filesystem): Started z1.example.com
...
```

5. Cree el recurso **nfserver** denominado **nfs-daemon** como parte del grupo de recursos **nfsgroup**.



NOTA

El recurso **nfserver** permite especificar un parámetro **nfs_shared_infodir**, que es un directorio que los servidores NFS utilizan para almacenar información de estado relacionada con NFS.

Se recomienda que este atributo se establezca en un subdirectorio de uno de los recursos **Filesystem** creados en esta colección de exportaciones. Esto asegura que los servidores NFS están almacenando su información de estado en un dispositivo que estará disponible para otro nodo si este grupo de recursos necesita reubicarse. En este ejemplo;

- **/nfsshare** es el directorio de almacenamiento compartido gestionado por el recurso **Filesystem**
- **/nfsshare/exports/export1** y **/nfsshare/exports/export2** son los directorios de exportación
- **/nfsshare/nfsinfo** es el directorio de información compartida del recurso **nfserver**

```
[root@z1 ~]# pcs resource create nfs-daemon nfserver \
nfs_shared_infodir=/nfsshare/nfsinfo nfs_no_notify=true \
--group nfsgroup
```

```
[root@z1 ~]# pcs status
...
```

6. Añada los recursos **exportfs** para exportar el directorio **/nfsshare/exports**. Estos recursos forman parte del grupo de recursos **nfsgroup**. Esto construye un directorio virtual para los clientes NFSv4. Los clientes NFSv3 también pueden acceder a estas exportaciones.



NOTA

La opción **fsid=0** sólo es necesaria si desea crear un directorio virtual para clientes NFSv4. Para más información, consulte [¿Cómo configuro la opción fsid en el archivo /etc/exports de un servidor NFS?](#)

```
[root@z1 ~]# pcs resource create nfs-root exportfs \
clientspec=192.168.122.0/255.255.255.0 \
options=rw,sync,no_root_squash \
directory=/nfsshare/exports \
fsid=0 --group nfsgroup
```

```
[root@z1 ~]# # pcs resource create nfs-export1 exportfs \
clientspec=192.168.122.0/255.255.255.0 \
options=rw,sync,no_root_squash directory=/nfsshare/exports/export1 \
fsid=1 --group nfsgroup
```

```
[root@z1 ~]# # pcs resource create nfs-export2 exportfs \
clientspec=192.168.122.0/255.255.255.0 \
options=rw,sync,no_root_squash directory=/nfsshare/exports/export2 \
fsid=2 --group nfsgroup
```

- Añada el recurso de dirección IP flotante que los clientes NFS utilizarán para acceder al recurso compartido NFS. Este recurso forma parte del grupo de recursos **nfsgroup**. Para esta implementación de ejemplo, estamos usando 192.168.122.200 como dirección IP flotante.

```
[root@z1 ~]# pcs resource create nfs_ip IPAddr2 \
ip=192.168.122.200 cidr_netmask=24 --group nfsgroup
```

- Añade un recurso **nfsnotify** para enviar notificaciones de reinicio de NFSv3 una vez que se haya inicializado todo el despliegue de NFS. Este recurso forma parte del grupo de recursos **nfsgroup**.



NOTA

Para que la notificación NFS se procese correctamente, la dirección IP flotante debe tener un nombre de host asociado que sea consistente tanto en los servidores NFS como en el cliente NFS.

```
[root@z1 ~]# pcs resource create nfs-notify nfsnotify \
source_host=192.168.122.200 --group nfsgroup
```

- Después de crear los recursos y las restricciones de recursos, puede comprobar el estado del clúster. Tenga en cuenta que todos los recursos se ejecutan en el mismo nodo.

```
[root@z1 ~]# pcs status
...
Full list of resources:
myapc (stonith:fence_apc_snmp): Started z1.example.com
Resource Group: nfsgroup
  my_lvm (ocf::heartbeat:LVM): Started z1.example.com
  nfsshare (ocf::heartbeat:Filesystem): Started z1.example.com
  nfs-daemon (ocf::heartbeat:nfsserver): Started z1.example.com
  nfs-root (ocf::heartbeat:exportfs): Started z1.example.com
  nfs-export1 (ocf::heartbeat:exportfs): Started z1.example.com
  nfs-export2 (ocf::heartbeat:exportfs): Started z1.example.com
  nfs_ip (ocf::heartbeat:IPAddr2): Started z1.example.com
  nfs-notify (ocf::heartbeat:nfsnotify): Started z1.example.com
...
```

6.6. PRUEBA DE LA CONFIGURACIÓN DE LOS RECURSOS NFS

Puede validar la configuración de su sistema con los siguientes procedimientos. Debería poder montar el sistema de archivos exportado con NFSv3 o NFSv4.

6.6.1. Prueba de la exportación NFS

- En un nodo fuera del clúster, que resida en la misma red que el despliegue, verifique que el recurso compartido NFS puede ser visto montando el recurso compartido NFS. Para este ejemplo, estamos utilizando la red 192.168.122.0/24.

```
# showmount -e 192.168.122.200
Export list for 192.168.122.200:
/nfsshare/exports/export1 192.168.122.0/255.255.255.0
```

```
/nfsshare/exports      192.168.122.0/255.255.255.0
/nfsshare/exports/export2 192.168.122.0/255.255.255.0
```

- Para verificar que puede montar el recurso compartido NFS con NFSv4, monte el recurso compartido NFS en un directorio del nodo cliente. Tras el montaje, compruebe que el contenido de los directorios de exportación es visible. Desmonte el recurso compartido después de la prueba.

```
# mkdir nfsshare
# mount -o "vers=4" 192.168.122.200:export1 nfsshare
# ls nfsshare
clientdatafile1
# umount nfsshare
```

- Compruebe que puede montar el recurso compartido NFS con NFSv3. Tras el montaje, compruebe que el archivo de prueba **clientdatafile1** es visible. A diferencia de NFSv4, como NFSv3 no utiliza el sistema de archivos virtual, debe montar una exportación específica. Desmonte el recurso compartido después de la prueba.

```
# mkdir nfsshare
# mount -o "vers=3" 192.168.122.200:/nfsshare/exports/export2 nfsshare
# ls nfsshare
clientdatafile2
# umount nfsshare
```

6.6.2. Pruebas de conmutación por error

- En un nodo fuera del clúster, monte el recurso compartido NFS y verifique el acceso a la dirección **clientdatafile1** que creamos en [Configuración de un recurso compartido NFS](#)

```
# mkdir nfsshare
# mount -o "vers=4" 192.168.122.200:export1 nfsshare
# ls nfsshare
clientdatafile1
```

- Desde un nodo del clúster, determine qué nodo del clúster está ejecutando **nfsgroup**. En este ejemplo, **nfsgroup** se está ejecutando en **z1.example.com**.

```
[root@z1 ~]# pcs status
...
Full list of resources:
myapc (stonith:fence_apc_snmp): Started z1.example.com
Resource Group: nfsgroup
  my_lvm (ocf::heartbeat:LVM): Started z1.example.com
  nfsshare (ocf::heartbeat:Filesystem): Started z1.example.com
  nfs-daemon (ocf::heartbeat:nfsserver): Started z1.example.com
  nfs-root (ocf::heartbeat:exportfs): Started z1.example.com
  nfs-export1 (ocf::heartbeat:exportfs): Started z1.example.com
  nfs-export2 (ocf::heartbeat:exportfs): Started z1.example.com
  nfs_ip (ocf::heartbeat:IPaddr2): Started z1.example.com
  nfs-notify (ocf::heartbeat:nfsnotify): Started z1.example.com
...
```

- Desde un nodo del clúster, ponga el nodo que está ejecutando **nfsgroup** en modo de espera.

```
[root@z1 ~]# pcs node standby z1.example.com
```

- Compruebe que **nfsgroup** se inicia con éxito en el otro nodo del clúster.

```
[root@z1 ~]# pcs status
```

```
...
```

```
Full list of resources:
```

```
Resource Group: nfsgroup
```

```
my_lvm (ocf::heartbeat:LVM): Started z2.example.com
```

```
nfsshare (ocf::heartbeat:Filesystem): Started z2.example.com
```

```
nfs-daemon (ocf::heartbeat:nfsserver): Started z2.example.com
```

```
nfs-root (ocf::heartbeat:exportfs): Started z2.example.com
```

```
nfs-export1 (ocf::heartbeat:exportfs): Started z2.example.com
```

```
nfs-export2 (ocf::heartbeat:exportfs): Started z2.example.com
```

```
nfs_ip (ocf::heartbeat:IPaddr2): Started z2.example.com
```

```
nfs-notify (ocf::heartbeat:nfsnotify): Started z2.example.com
```

```
...
```

- Desde el nodo externo al clúster en el que ha montado el recurso compartido NFS, verifique que este nodo externo sigue teniendo acceso al archivo de prueba dentro del montaje NFS.

```
# ls nfsshare
```

```
clientdatafile1
```

El servicio se perderá brevemente para el cliente durante la conmutación por error, pero el cliente debería recuperarlo sin intervención del usuario. Por defecto, los clientes que utilizan NFSv4 pueden tardar hasta 90 segundos en recuperar el montaje; estos 90 segundos representan el periodo de gracia de arrendamiento de archivos NFSv4 que observa el servidor al iniciarse. Los clientes NFSv3 deberían recuperar el acceso al montaje en cuestión de pocos segundos.

- Desde un nodo del clúster, retire del modo de espera el nodo que inicialmente estaba ejecutando **nfsgroup**.



NOTA

Sacar un nodo del modo **standby** no provoca por sí mismo que los recursos vuelvan a fallar en ese nodo. Esto dependerá del valor de **resource-stickiness** para los recursos. Para obtener información sobre el metaatributo **resource-stickiness**, consulte [Configurar un recurso para que prefiera su nodo actual](#) .

```
[root@z1 ~]# pcs node unstandby z1.example.com
```

CAPÍTULO 7. SISTEMAS DE ARCHIVOS GFS2 EN UN CLÚSTER

Esta sección establece:

- Un procedimiento para configurar un clúster Pacemaker que incluya sistemas de archivos GFS2
- Un procedimiento para migrar volúmenes lógicos de RHEL 7 que contienen sistemas de archivos GFS2 a un clúster de RHEL 8

7.1. CONFIGURACIÓN DE UN SISTEMA DE ARCHIVOS GFS2 EN UN CLÚSTER

Este procedimiento es un resumen de los pasos necesarios para configurar un cluster Pacemaker que incluya sistemas de archivos GFS2. Este ejemplo crea tres sistemas de archivos GFS2 en tres volúmenes lógicos.

Como requisito previo a este procedimiento, debe instalar e iniciar el software de clúster en todos los nodos y crear un clúster básico de dos nodos. También debe configurar el cercado para el cluster. Para obtener información sobre la creación de un cluster Pacemaker y la configuración del cercado para el cluster, consulte [Creación de un cluster de alta disponibilidad de Red Hat con Pacemaker](#).

Procedimiento

1. En ambos nodos del clúster, instale los paquetes **lvm2-lockd**, **gfs2-utils** y **dlm**. El paquete **lvm2-lockd** forma parte del canal AppStream y los paquetes **gfs2-utils** y **dlm** forman parte del canal Resilient Storage.

```
# yum install lvm2-lockd gfs2-utils dlm
```

2. Ajuste el parámetro global de Marcapasos **no_quorum_policy** a **freeze**.



NOTA

Por defecto, el valor de **no-quorum-policy** se establece en **stop**, lo que indica que una vez que se pierde el quórum, todos los recursos de la partición restante se detendrán inmediatamente. Normalmente este valor por defecto es la opción más segura y óptima, pero a diferencia de la mayoría de los recursos, GFS2 requiere quórum para funcionar. Cuando se pierde el quórum, tanto las aplicaciones que utilizan los montajes de GFS2 como el propio montaje de GFS2 no pueden detenerse correctamente. Cualquier intento de detener estos recursos sin quórum fallará, lo que en última instancia provocará que todo el clúster quede vallado cada vez que se pierda el quórum.

Para solucionar esta situación, configure **no-quorum-policy** a **freeze** cuando GFS2 esté en uso. Esto significa que cuando se pierde el quórum, la partición restante no hará nada hasta que se recupere el quórum.

```
# pcs property set no-quorum-policy=freeze
```

3. Configure un recurso **dml**. Esta es una dependencia necesaria para configurar un sistema de archivos GFS2 en un clúster. Este ejemplo crea el recurso **dml** como parte de un grupo de recursos llamado **locking**.


```
[root@z1 ~]# pcs resource create dlm --group locking ocf:pacemaker:controld op
monitor interval=30s on-fail=fence
```

4. Clone el grupo de recursos **locking** para que el grupo de recursos pueda estar activo en ambos nodos del clúster.

```
[root@z1 ~]# pcs resource clone locking interleave=true
```

5. Configure un recurso **lvmlockd** como parte del grupo **locking**.

```
[root@z1 ~]# pcs resource create lvmlockd --group locking ocf:heartbeat:lvmlockd op
monitor interval=30s on-fail=fence
```

6. Compruebe el estado del clúster para asegurarse de que el grupo de recursos **locking** se ha iniciado en ambos nodos del clúster.

```
[root@z1 ~]# pcs status --full
Cluster name: my_cluster
[...]

Online: [ z1.example.com (1) z2.example.com (2) ]

Full list of resources:

smoke-apc (stonith:fence_apc): Started z1.example.com
Clone Set: locking-clone [locking]
  Resource Group: locking:0
    dlm (ocf::pacemaker:controld): Started z1.example.com
    lvmlockd (ocf::heartbeat:lvmlockd): Started z1.example.com
  Resource Group: locking:1
    dlm (ocf::pacemaker:controld): Started z2.example.com
    lvmlockd (ocf::heartbeat:lvmlockd): Started z2.example.com
Started: [ z1.example.com z2.example.com ]
```

7. Compruebe que el demonio **lvmlockd** se está ejecutando en ambos nodos del clúster.

```
[root@z1 ~]# ps -ef | grep lvmlockd
root 12257 1 0 17:45 ? 00:00:00 lvmlockd -p /run/lvmlockd.pid -A 1 -g dlm
[root@z2 ~]# ps -ef | grep lvmlockd
root 12270 1 0 17:45 ? 00:00:00 lvmlockd -p /run/lvmlockd.pid -A 1 -g dlm
```

8. En un nodo del clúster, cree dos grupos de volúmenes compartidos. Un grupo de volumen contendrá dos sistemas de archivos GFS2 y el otro grupo de volumen contendrá un sistema de archivos GFS2.

El siguiente comando crea el grupo de volúmenes compartidos **shared_vg1** en **/dev/vdb**.

```
[root@z1 ~]# vgcreate --shared shared_vg1 /dev/vdb
Physical volume "/dev/vdb" successfully created.
Volume group "shared_vg1" successfully created
VG shared_vg1 starting dlm lockspace
Starting locking. Waiting until locks are ready...
```

El siguiente comando crea el grupo de volúmenes compartidos **shared_vg2** en **/dev/vdc**.

```
[root@z1 ~]# vgcreate --shared shared_vg2 /dev/vdc
Physical volume "/dev/vdc" successfully created.
Volume group "shared_vg2" successfully created
VG shared_vg2 starting dlm lockspace
Starting locking. Waiting until locks are ready...
```

9. En el segundo nodo del clúster, inicie el gestor de bloqueos para cada uno de los grupos de volúmenes compartidos.

```
[root@z2 ~]# vgchange --lock-start shared_vg1
VG shared_vg1 starting dlm lockspace
Starting locking. Waiting until locks are ready...
[root@z2 ~]# vgchange --lock-start shared_vg2
VG shared_vg2 starting dlm lockspace
Starting locking. Waiting until locks are ready...
```

10. En un nodo del clúster, cree los volúmenes lógicos compartidos y formatee los volúmenes con un sistema de archivos GFS2. Se necesita un diario para cada nodo que monte el sistema de archivos. Asegúrese de crear suficientes diarios para cada uno de los nodos del clúster.

```
[root@z1 ~]# lvcreate --activate sy -L5G -n shared_lv1 shared_vg1
Logical volume "shared_lv1" created.
[root@z1 ~]# lvcreate --activate sy -L5G -n shared_lv2 shared_vg1
Logical volume "shared_lv2" created.
[root@z1 ~]# lvcreate --activate sy -L5G -n shared_lv1 shared_vg2
Logical volume "shared_lv1" created.

[root@z1 ~]# mkfs.gfs2 -j2 -p lock_dlm -t my_cluster:gfs2-demo1
/dev/shared_vg1/shared_lv1
[root@z1 ~]# mkfs.gfs2 -j2 -p lock_dlm -t my_cluster:gfs2-demo2
/dev/shared_vg1/shared_lv2
[root@z1 ~]# mkfs.gfs2 -j2 -p lock_dlm -t my_cluster:gfs2-demo3
/dev/shared_vg2/shared_lv1
```

11. Cree un recurso **LVM-activate** para cada volumen lógico para activar automáticamente ese volumen lógico en todos los nodos.
 - a. Crea un recurso **LVM-activate** llamado **sharedlv1** para el volumen lógico **shared_lv1** en el grupo de volúmenes **shared_vg1**. Este comando también crea el grupo de recursos **shared_vg1** que incluye el recurso. En este ejemplo, el grupo de recursos tiene el mismo nombre que el grupo de volumen compartido que incluye el volumen lógico.

```
[root@z1 ~]# pcs resource create sharedlv1 --group shared_vg1 ocf:heartbeat:LVM-
activate lvname=shared_lv1 vgroup=shared_vg1 activation_mode=shared
vg_access_mode=lvmlockd
```

- b. Crea un recurso **LVM-activate** llamado **sharedlv2** para el volumen lógico **shared_lv2** en el grupo de volúmenes **shared_vg1**. Este recurso también formará parte del grupo de recursos **shared_vg1**.

```
[root@z1 ~]# pcs resource create sharedlv2 --group shared_vg1 ocf:heartbeat:LVM-
activate lvname=shared_lv2 vgroup=shared_vg1 activation_mode=shared
vg_access_mode=lvmlockd
```

- c. Crea un recurso **LVM-activate** llamado **sharedlv3** para el volumen lógico **shared_lv1** en el grupo de volúmenes **shared_vg2**. Este comando también crea el grupo de recursos **shared_vg2** que incluye el recurso.

```
[root@z1 ~]# pcs resource create sharedlv3 --group shared_vg2 ocf:heartbeat:LVM-activate lvname=shared_lv1 vgname=shared_vg2 activation_mode=shared_vg_access_mode=lvmlockd
```

12. Clona los dos nuevos grupos de recursos.

```
[root@z1 ~]# pcs resource clone shared_vg1 interleave=true
[root@z1 ~]# pcs resource clone shared_vg2 interleave=true
```

13. Configure las restricciones de ordenación para garantizar que el grupo de recursos **locking** que incluye los recursos **dlm** y **lvmlockd** se inicie primero.

```
[root@z1 ~]# pcs constraint order start locking-clone then shared_vg1-clone
Adding locking-clone shared_vg1-clone (kind: Mandatory) (Options: first-action=start then-action=start)
[root@z1 ~]# pcs constraint order start locking-clone then shared_vg2-clone
Adding locking-clone shared_vg2-clone (kind: Mandatory) (Options: first-action=start then-action=start)
```

14. Configure las restricciones de colocación para garantizar que los grupos de recursos **vg1** y **vg2** se inicien en el mismo nodo que el grupo de recursos **locking**.

```
[root@z1 ~]# pcs constraint colocation add shared_vg1-clone with locking-clone
[root@z1 ~]# pcs constraint colocation add shared_vg2-clone with locking-clone
```

15. En ambos nodos del clúster, verifique que los volúmenes lógicos estén activos. Puede haber un retraso de unos segundos.

```
[root@z1 ~]# lvs
LV      VG      Attr    LSize
shared_lv1 shared_vg1 -wi-a----- 5.00g
shared_lv2 shared_vg1 -wi-a----- 5.00g
shared_lv1 shared_vg2 -wi-a----- 5.00g

[root@z2 ~]# lvs
LV      VG      Attr    LSize
shared_lv1 shared_vg1 -wi-a----- 5.00g
shared_lv2 shared_vg1 -wi-a----- 5.00g
shared_lv1 shared_vg2 -wi-a----- 5.00g
```

16. Cree un recurso de sistema de archivos para montar automáticamente cada sistema de archivos GFS2 en todos los nodos.

No debe añadir el sistema de archivos al archivo **/etc/fstab** porque se gestionará como un recurso de clúster de Pacemaker. Las opciones de montaje se pueden especificar como parte de la configuración del recurso con **options=options**. Ejecute el comando **pcs resource describe Filesystem** para obtener las opciones de configuración completas.

Los siguientes comandos crean los recursos del sistema de archivos. Estos comandos añaden cada recurso al grupo de recursos que incluye el recurso de volumen lógico para ese sistema de archivos.

```
[root@z1 ~]# pcs resource create sharedfs1 --group shared_vg1
ocf:heartbeat:Filesystem device="/dev/shared_vg1/shared_lv1" directory="/mnt/gfs1"
fstype="gfs2" options=noatime op monitor interval=10s on-fail=fence
[root@z1 ~]# pcs resource create sharedfs2 --group shared_vg1
ocf:heartbeat:Filesystem device="/dev/shared_vg1/shared_lv2" directory="/mnt/gfs2"
fstype="gfs2" options=noatime op monitor interval=10s on-fail=fence
[root@z1 ~]# pcs resource create sharedfs3 --group shared_vg2
ocf:heartbeat:Filesystem device="/dev/shared_vg2/shared_lv1" directory="/mnt/gfs3"
fstype="gfs2" options=noatime op monitor interval=10s on-fail=fence
```

17. Compruebe que los sistemas de archivos GFS2 están montados en ambos nodos del clúster.

```
[root@z1 ~]# mount | grep gfs2
/dev/mapper/shared_vg1-shared_lv1 on /mnt/gfs1 type gfs2 (rw,noatime,seclabel)
/dev/mapper/shared_vg1-shared_lv2 on /mnt/gfs2 type gfs2 (rw,noatime,seclabel)
/dev/mapper/shared_vg2-shared_lv1 on /mnt/gfs3 type gfs2 (rw,noatime,seclabel)

[root@z2 ~]# mount | grep gfs2
/dev/mapper/shared_vg1-shared_lv1 on /mnt/gfs1 type gfs2 (rw,noatime,seclabel)
/dev/mapper/shared_vg1-shared_lv2 on /mnt/gfs2 type gfs2 (rw,noatime,seclabel)
/dev/mapper/shared_vg2-shared_lv1 on /mnt/gfs3 type gfs2 (rw,noatime,seclabel)
```

18. Comprueba el estado del clúster.

```
[root@z1 ~]# pcs status --full
Cluster name: my_cluster
[...]

Full list of resources:

smoke-apc (stonith:fence_apc): Started z1.example.com
Clone Set: locking-clone [locking]
Resource Group: locking:0
    dlm (ocf::pacemaker:controld): Started z2.example.com
    lvmlockd (ocf::heartbeat:lvmlockd): Started z2.example.com
Resource Group: locking:1
    dlm (ocf::pacemaker:controld): Started z1.example.com
    lvmlockd (ocf::heartbeat:lvmlockd): Started z1.example.com
Started: [ z1.example.com z2.example.com ]
Clone Set: shared_vg1-clone [shared_vg1]
Resource Group: shared_vg1:0
    sharedlv1 (ocf::heartbeat:LVM-activate): Started z2.example.com
    sharedlv2 (ocf::heartbeat:LVM-activate): Started z2.example.com
    sharedfs1 (ocf::heartbeat:Filesystem): Started z2.example.com
    sharedfs2 (ocf::heartbeat:Filesystem): Started z2.example.com
Resource Group: shared_vg1:1
    sharedlv1 (ocf::heartbeat:LVM-activate): Started z1.example.com
    sharedlv2 (ocf::heartbeat:LVM-activate): Started z1.example.com
    sharedfs1 (ocf::heartbeat:Filesystem): Started z1.example.com
    sharedfs2 (ocf::heartbeat:Filesystem): Started example.co
Started: [ z1.example.com z2.example.com ]
Clone Set: shared_vg2-clone [shared_vg2]
Resource Group: shared_vg2:0
    sharedlv3 (ocf::heartbeat:LVM-activate): Started z2.example.com
    sharedfs3 (ocf::heartbeat:Filesystem): Started z2.example.com
```

```
Resource Group: shared_vg2:1
  sharedlv3 (ocf::heartbeat:LVM-activate): Started z1.example.com
  sharedfs3 (ocf::heartbeat:Filesystem): Started z1.example.com
Started: [ z1.example.com z2.example.com ]
```

...

Recursos adicionales

- Para obtener información sobre la configuración del [almacenamiento](#) en bloque compartido para un clúster de Red Hat High Availability con Microsoft Azure Shared Disks, consulte [Configuración del almacenamiento en bloque compartido](#) .
- Para obtener información sobre la configuración del [almacenamiento](#) en bloque compartido para un clúster de Red Hat High Availability con volúmenes de Amazon EBS Multi-Attach, consulte [Configuración del almacenamiento en bloque compartido](#) .
- Para obtener información sobre la configuración del almacenamiento en bloque compartido para un [clúster](#) de alta disponibilidad de Red Hat en Alibaba [Cloud](#), consulte [Configuración del almacenamiento en bloque compartido para un clúster de alta disponibilidad de Red Hat en Alibaba Cloud](#).

7.2. MIGRACIÓN DE UN SISTEMA DE ARCHIVOS GFS2 DE RHEL7 A RHEL8

En Red Hat Enterprise Linux 8, LVM utiliza el demonio de bloqueo de LVM **lvmlockd** en lugar de **clvmd** para gestionar los dispositivos de almacenamiento compartido en un cluster activo/activo. Esto requiere que configure los volúmenes lógicos que su cluster activo/activo requerirá como volúmenes lógicos compartidos. Además, esto requiere que utilice el recurso **LVM-activate** para gestionar un volumen LVM y que utilice el agente de recursos **lvmlockd** para gestionar el demonio **lvmlockd**. Consulte [Configuración de un sistema de archivos GFS2 en un clúster](#) para conocer el procedimiento completo para configurar un clúster Pacemaker que incluya sistemas de archivos GFS2 mediante volúmenes lógicos compartidos.

Para utilizar sus volúmenes lógicos existentes de Red Hat Enterprise Linux 7 cuando configure un cluster RHEL8 que incluya sistemas de archivos GFS2, realice el siguiente procedimiento desde el cluster RHEL8. En este ejemplo, el volumen lógico RHEL 7 en cluster es parte del grupo de volúmenes **upgrade_gfs_vg**.



NOTA

El clúster RHEL8 debe tener el mismo nombre que el clúster RHEL7 que incluye el sistema de archivos GFS2 para que el sistema de archivos existente sea válido.

1. Asegúrese de que los volúmenes lógicos que contienen los sistemas de archivos GFS2 están actualmente inactivos. Este procedimiento sólo es seguro si todos los nodos han dejado de utilizar el grupo de volúmenes.
2. Desde un nodo del clúster, cambie a la fuerza el grupo de volúmenes para que sea local.

```
[root@rhel8-01 ~]# vgchange --lock-type none --lock-opt force upgrade_gfs_vg
Forcibly change VG lock type to none? [y/n]: y
Volume group "upgrade_gfs_vg" successfully changed
```

3. Desde un nodo del clúster, cambie el grupo de volumen local a un grupo de volumen compartido

```
[root@rhel8-01 ~]# vgchange --lock-type dlm upgrade_gfs_vg  
Volume group "upgrade_gfs_vg" successfully changed
```

4. En cada nodo del clúster, inicie el bloqueo del grupo de volúmenes.

```
[root@rhel8-01 ~]# vgchange --lock-start upgrade_gfs_vg  
VG upgrade_gfs_vg starting dlm lockspace  
Starting locking. Waiting until locks are ready...  
[root@rhel8-02 ~]# vgchange --lock-start upgrade_gfs_vg  
VG upgrade_gfs_vg starting dlm lockspace  
Starting locking. Waiting until locks are ready...
```

Después de realizar este procedimiento, puede crear un recurso **LVM-activate** para cada volumen lógico.

CAPÍTULO 8. INTRODUCCIÓN A LA INTERFAZ WEB DE PCSD

La **pcsd** Web UI es una interfaz gráfica de usuario para crear y configurar clusters Pacemaker/Corosync.

8.1. INSTALACIÓN DEL SOFTWARE DEL CLÚSTER

El siguiente procedimiento instala el software del clúster y configura su sistema para la creación del clúster.

1. En cada nodo del cluster, instale los paquetes de software Red Hat High Availability Add-On junto con todos los agentes de valla disponibles en el canal de Alta Disponibilidad.

```
# yum install pcs pacemaker fence-agents-all
```

Alternativamente, puede instalar los paquetes de software de Red Hat High Availability Add-On junto con sólo el agente de valla que necesite con el siguiente comando.

```
# yum install pcs pacemaker fence-agents-model
```

El siguiente comando muestra una lista de los agentes de la valla disponibles.

```
# rpm -q -a | grep fence
fence-agents-rhevm-4.0.2-3.el7.x86_64
fence-agents-ilo-mp-4.0.2-3.el7.x86_64
fence-agents-ipmilan-4.0.2-3.el7.x86_64
...
```



AVISO

Después de instalar los paquetes del complemento de alta disponibilidad de Red Hat, debe asegurarse de que sus preferencias de actualización de software estén configuradas para que no se instale nada automáticamente. La instalación en un cluster en funcionamiento puede causar comportamientos inesperados. Para obtener más información, consulte [Prácticas recomendadas para aplicar actualizaciones de software a un cluster de alta disponibilidad o de almacenamiento resiliente de RHEL](#).

2. Si está ejecutando el demonio **firewalld**, ejecute los siguientes comandos para habilitar los puertos requeridos por el complemento de alta disponibilidad de Red Hat.



NOTA

Puede determinar si el demonio **firewalld** está instalado en su sistema con el comando **rpm -q firewalld**. Si está instalado, puede determinar si se está ejecutando con el comando **firewall-cmd --state**.

```
# firewall-cmd --permanent --add-service=high-availability
# firewall-cmd --add-service=high-availability
```



NOTA

La configuración ideal del cortafuegos para los componentes del clúster depende del entorno local, en el que puede ser necesario tener en cuenta consideraciones como si los nodos tienen múltiples interfaces de red o si existe un cortafuegos fuera del host. El ejemplo que se presenta aquí, que abre los puertos que generalmente requiere un cluster Pacemaker, debe modificarse para adaptarse a las condiciones locales. [Habilitación de puertos para el complemento de alta disponibilidad](#) muestra los puertos que se deben habilitar para el complemento de alta disponibilidad de Red Hat y proporciona una explicación de para qué se utiliza cada puerto.

- Para poder utilizar **pcs** para configurar el cluster y comunicarse entre los nodos, debe establecer una contraseña en cada nodo para el usuario **hacluster**, que es la cuenta de administración **pcs**. Se recomienda que la contraseña del usuario **hacluster** sea la misma en cada nodo.

```
# passwd hacluster
Changing password for user hacluster.
New password:
Retype new password:
passwd: all authentication tokens updated successfully.
```

- Antes de poder configurar el cluster, el demonio **pcsd** debe ser iniciado y habilitado para arrancar en cada nodo. Este demonio funciona con el comando **pcs** para gestionar la configuración en todos los nodos del clúster. En cada nodo del clúster, ejecute los siguientes comandos para iniciar el servicio **pcsd** y para habilitar **pcsd** al inicio del sistema.

```
# systemctl start pcsd.service
# systemctl enable pcsd.service
```

8.2. CONFIGURACIÓN DE LA INTERFAZ WEB DE PCSD

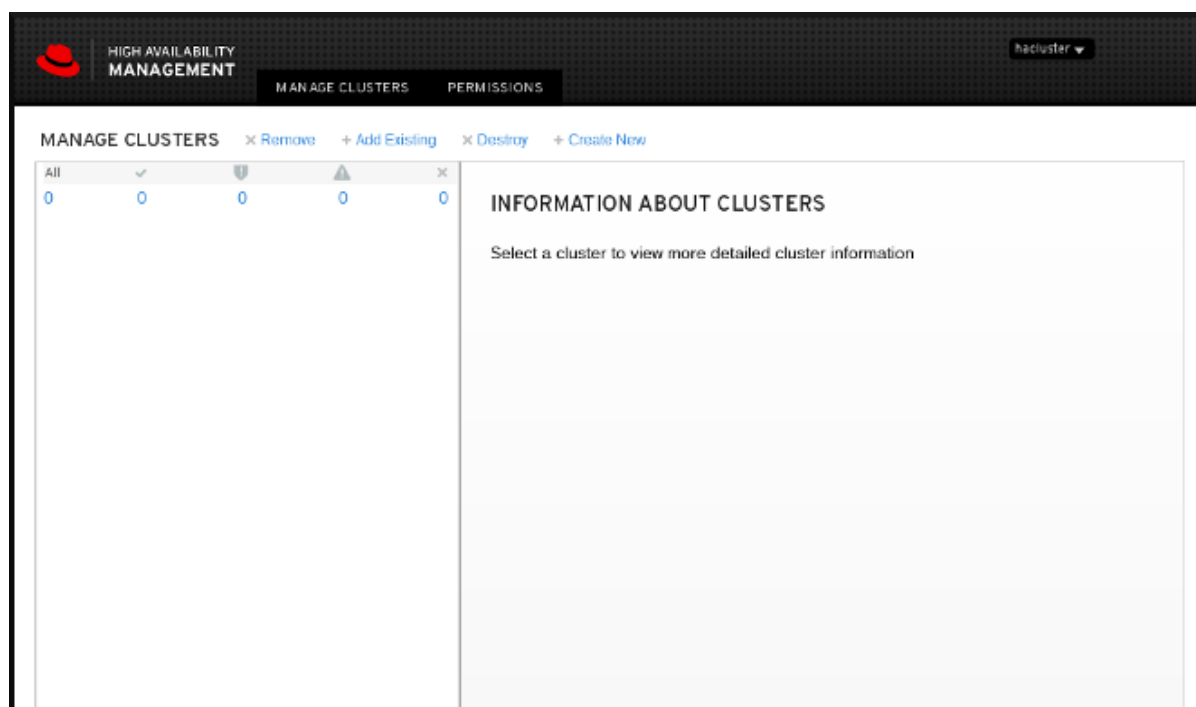
Una vez que haya instalado las herramientas de configuración de Pacemaker y haya configurado su sistema para la configuración de clústeres, utilice el siguiente procedimiento para configurar su sistema y utilizar la interfaz web **pcsd** para configurar un clúster.

- En cualquier sistema, abra un navegador a la siguiente URL, especificando uno de los nodos del clúster (tenga en cuenta que esto utiliza el protocolo **https**). Aparecerá la pantalla de inicio de sesión de **pcsd** Web UI.

```
https://nodename:2224
```

- Inicie sesión como usuario **hacluster**. Esto hace que aparezca la página **Manage Clusters** como se muestra en [Figura 8.1, "Página de gestión de clusters"](#).

Figura 8.1. Página de gestión de clusters



8.3. CREACIÓN DE UN CLÚSTER CON LA INTERFAZ WEB DE PCSD

Desde la página de Gestión de Clusters, puede crear un nuevo cluster, añadir un cluster existente a la Web UI, o eliminar un cluster de la Web UI.

- Para crear un clúster, haga clic en **Create New**. Introduzca el nombre del cluster a crear y los nodos que lo constituyen. Si no ha autenticado previamente el usuario **hacluster** para cada nodo del clúster, se le pedirá que autentique los nodos del clúster.
- Al crear el clúster, puede configurar las opciones avanzadas del clúster haciendo clic en **Go to advanced settings** en esta pantalla. Las configuraciones avanzadas del clúster que puede configurar se describen en [Configuración de las opciones avanzadas de configuración del clúster con la UI web de pcsd](#).
- Para añadir un clúster existente a la interfaz web, haga clic en **Add Existing** e introduzca el nombre de host o la dirección IP de un nodo del clúster que desee gestionar con la interfaz web.

Una vez que haya creado o añadido un clúster, el nombre del clúster se muestra en la página Gestionar clúster. Al seleccionar el clúster se muestra información sobre el mismo.



NOTA

Cuando se utiliza la interfaz web **pcsd** para configurar un clúster, se puede pasar el ratón por encima del texto que describe muchas de las opciones para ver descripciones más largas de esas opciones en forma de pantalla **tooltip**.

8.3.1. Configuración de las opciones avanzadas de configuración del clúster con la interfaz web de pcsd

Al crear un clúster, puede configurar opciones adicionales del clúster haciendo clic en **Ir a la configuración avanzada** en la pantalla Crear clúster. Esto le permite modificar los ajustes configurables de los siguientes componentes del clúster:

- Configuración del transporte: Valores para el mecanismo de transporte utilizado para la comunicación del clúster
- Opciones de quórum: Valores para las opciones de quórum del servicio **votequorum**
- Configuración de Totem: Valores para el protocolo Totem utilizado por Corosync

Al seleccionar estas opciones se muestran los ajustes que se pueden configurar. Para obtener información sobre cada uno de los ajustes, sitúe el puntero del ratón sobre la opción concreta.

8.3.2. Establecer los permisos de gestión del clúster

Hay dos conjuntos de permisos de clúster que puede conceder a los usuarios:

- Permisos para gestionar el clúster con la interfaz web, que también concede permisos para ejecutar comandos de **pcs** que se conectan a los nodos a través de una red. Esta sección describe cómo configurar esos permisos con la Web UI.
- Permisos para los usuarios locales para permitir el acceso de sólo lectura o de lectura-escritura a la configuración del clúster, utilizando ACLs. La configuración de las ACL con la interfaz web se describe en [Configuración de los componentes del clúster con la interfaz web de pcsd](#) .

Puede conceder permiso a usuarios específicos que no sean el usuario **hacluster** para gestionar el clúster a través de la interfaz web y para ejecutar comandos **pcs** que se conectan a los nodos a través de una red, añadiéndolos al grupo **haclient**. A continuación, puede configurar los permisos establecidos para un miembro individual del grupo **haclient** haciendo clic en la pestaña Permisos de la página Gestionar clústeres y estableciendo los permisos en la pantalla resultante. Desde esta pantalla, también puedes configurar los permisos para los grupos.

Puede conceder los siguientes permisos:

- Permisos de lectura, para ver la configuración del clúster
- Permisos de escritura, para modificar la configuración del clúster (excepto los permisos y las ACL)
- Conceder permisos, para modificar los permisos del clúster y las ACL
- Permisos completos, para un acceso sin restricciones a un cluster, incluyendo la adición y eliminación de nodos, con acceso a claves y certificados

8.4. CONFIGURACIÓN DE LOS COMPONENTES DEL CLÚSTER CON LA INTERFAZ WEB DE PCSD

Para configurar los componentes y atributos de un clúster, haga clic en el nombre del clúster que aparece en la pantalla **Manage Clusters**. Esto hace que aparezca la página **Nodes**, como se describe en [Sección 8.4.1, "Configuración de los nodos del clúster con la interfaz web de pcsd"](#) . Esta página muestra un menú en la parte superior de la página con las siguientes entradas:

- Los nodos, como se describe en [Sección 8.4.1, "Configuración de los nodos del clúster con la interfaz web de pcsd"](#)
- Recursos, como se describe en [Sección 8.4.2, "Configuración de los recursos del clúster con la interfaz web de pcsd"](#)

- Dispositivos de vallas, como se describe en [Sección 8.4.3, "Configuración de los dispositivos de la valla con la interfaz web de pcsd"](#)
- ACL, como se describe en [Sección 8.4.4, "Configuración de ACLs con la Web UI de pcsd"](#)
- Propiedades del clúster, como se describe en [Sección 8.4.5, "Configuración de las propiedades del clúster con la interfaz web de pcsd"](#)

8.4.1. Configuración de los nodos del clúster con la interfaz web de pcsd

Al seleccionar la opción **Nodes** en el menú de la parte superior de la página de gestión de clústeres, se muestran los nodos actualmente configurados y el estado del nodo actualmente seleccionado, incluyendo los recursos que se están ejecutando en el nodo y las preferencias de ubicación de los recursos. Esta es la página por defecto que se muestra cuando se selecciona un clúster en la pantalla **Manage Clusters**.

Desde esta página puede añadir o eliminar nodos. También puede iniciar, detener, reiniciar o poner un nodo en modo de espera o de mantenimiento. Para obtener información sobre el modo de espera, consulte Poner un [nodo](#) en modo de espera. Para obtener información sobre el modo de mantenimiento, consulte Poner un [clúster](#) en modo de mantenimiento.

También puede configurar los dispositivos de valla directamente desde esta página, como se describe en seleccionando **Configure Fencing**. La configuración de los dispositivos de vallas se describe en [Sección 8.4.3, "Configuración de los dispositivos de la valla con la interfaz web de pcsd"](#) .

8.4.2. Configuración de los recursos del clúster con la interfaz web de pcsd

Al seleccionar la opción **Resources** en el menú de la parte superior de la página de gestión del clúster, se muestran los recursos actualmente configurados para el clúster, organizados según los grupos de recursos. Al seleccionar un grupo o un recurso se muestran los atributos de ese grupo o recurso.

Desde esta pantalla, puede añadir o eliminar recursos, puede editar la configuración de los recursos existentes y puede crear un grupo de recursos.

Para añadir un nuevo recurso al cluster:

- Haga clic en **Add**. Aparecerá la pantalla **Add Resource**.
- Cuando selecciona un tipo de recurso en el menú desplegable **Type**, los argumentos que debe especificar para ese recurso aparecen en el menú.
- Puede hacer clic en **Optional Arguments** para mostrar los argumentos adicionales que puede especificar para el recurso que está definiendo.
- Tras introducir los parámetros del recurso que está creando, haga clic en **Create Resource**.

Al configurar los argumentos de un recurso, aparece una breve descripción del argumento en el menú. Si se desplaza el cursor al campo, se muestra una descripción de ayuda más larga de ese argumento.

Puede definir un recurso como un recurso clonado o como un recurso clonado promocionable. Para obtener información sobre estos tipos de recursos, consulte [Creación de recursos de clúster activos en varios nodos \(recursos clonados\)](#).

Una vez que haya creado al menos un recurso, puede crear un grupo de recursos. Para obtener información general sobre los grupos de recursos, consulte [Configuración de grupos de recursos](#) .

Para crear un grupo de recursos:

- Seleccione los recursos que formarán parte del grupo en la pantalla **Resources** y, a continuación, haga clic en **Create Group**. Esto muestra la pantalla **Create Group**.
- Desde la pantalla **Create Group**, puede reorganizar el orden de los recursos en un grupo de recursos utilizando la función de arrastrar y soltar para desplazar la lista de los recursos.
- Introduzca un nombre de grupo y haga clic en **Create Group**. Esto le devuelve a la pantalla **Resources**, que ahora muestra el nombre del grupo y los recursos dentro de ese grupo.

Después de crear un grupo de recursos, puede indicar el nombre de ese grupo como parámetro de recurso cuando cree o modifique recursos adicionales.

8.4.3. Configuración de los dispositivos de la valla con la interfaz web de pcsd

Seleccionando la opción **Fence Devices** en el menú de la parte superior de la página de gestión del clúster se muestra la pantalla **Fence Devices**, que muestra los dispositivos de valla actualmente configurados.

Para añadir un nuevo dispositivo de valla al clúster:

- Haga clic en **Add**. Aparecerá la pantalla **Add Fence Device**.
- Cuando selecciona un tipo de dispositivo de valla en el menú desplegable **Type**, los argumentos que debe especificar para ese dispositivo de valla aparecen en el menú.
- Puede hacer clic en **Optional Arguments** para mostrar los argumentos adicionales que puede especificar para el dispositivo de valla que está definiendo.
- Tras introducir los parámetros del nuevo dispositivo de vallado, haga clic en **Create Fence Instance**.

Para configurar un dispositivo de cercado SBD, haga clic en **SBD** en la pantalla **Fence Devices**. Esto llama a una pantalla que permite activar o desactivar el SBD en el clúster.

Para más información sobre los dispositivos de cercado, consulte [Configuración de cercado en un cluster de Alta Disponibilidad de Red Hat](#).

8.4.4. Configuración de ACLs con la Web UI de pcsd

Al seleccionar la opción **ACLs** en el menú de la parte superior de la página de gestión del clúster, aparece una pantalla desde la que se pueden establecer permisos para los usuarios locales, permitiendo el acceso de sólo lectura o de lectura y escritura a la configuración del clúster mediante el uso de listas de control de acceso (ACL).

Para asignar permisos ACL, se crea un rol y se definen los permisos de acceso para ese rol. Cada rol puede tener un número ilimitado de permisos (lectura/escritura/denegación) aplicados a una consulta XPath o al ID de un elemento específico. Después de definir el rol, puede asignarlo a un usuario o grupo existente.

Para obtener más información sobre la asignación de permisos mediante ACL, consulte [Configuración de permisos locales mediante ACL](#).

8.4.5. Configuración de las propiedades del clúster con la interfaz web de pcsd

Al seleccionar la opción **Cluster Properties** en el menú de la parte superior de la página de gestión del clúster, se muestran las propiedades del clúster y se pueden modificar sus valores predeterminados.

Para obtener información sobre las propiedades del clúster de Pacemaker, consulte [Propiedades del clúster de Pacemaker](#).

8.5. CONFIGURACIÓN DE UNA INTERFAZ WEB DE ALTA DISPONIBILIDAD PCSD

Cuando se utiliza la interfaz web **pcsd**, se conecta a uno de los nodos del clúster para mostrar las páginas de gestión del mismo. Si el nodo al que se conecta se cae o deja de estar disponible, puede volver a conectarse al clúster abriendo el navegador a una URL que especifique un nodo diferente del clúster.

Sin embargo, es posible configurar la propia interfaz web **pcsd** para la alta disponibilidad, en cuyo caso se puede seguir gestionando el clúster sin necesidad de introducir una nueva URL.

Para configurar la interfaz web **pcsd** para una alta disponibilidad, realice los siguientes pasos.

1. Asegúrese de que los certificados de **pcsd** están sincronizados entre los nodos del clúster estableciendo **PCSD_SSL_CERT_SYNC_ENABLED** a **true** en el archivo de configuración **/etc/sysconfig/pcsd**. La activación de la sincronización de certificados hace que **pcsd** sincronice los certificados para los comandos de configuración del clúster y de adición de nodos. En RHEL 8, **PCSD_SSL_CERT_SYNC_ENABLED** está configurado como **false** por defecto.
2. Cree un recurso de clúster **IPaddr2**, que es una dirección IP flotante que utilizará para conectarse a la interfaz web **pcsd**. La dirección IP no debe ser una ya asociada a un nodo físico. Si no se especifica el dispositivo NIC del recurso **IPaddr2**, la IP flotante debe residir en la misma red que una de las direcciones IP asignadas estáticamente del nodo, de lo contrario no se podrá detectar correctamente el dispositivo NIC que asignará la dirección IP flotante.
3. Cree certificados SSL personalizados para utilizarlos con **pcsd** y asegúrese de que son válidos para las direcciones de los nodos utilizados para conectarse a la interfaz web **pcsd**.
 - a. Para crear certificados SSL personalizados, puede utilizar certificados comodín o puede utilizar la extensión de certificado Subject Alternative Name. Para obtener información sobre Red Hat Certificate System, consulte el [Manual de administración de Red Hat Certificate System](#).
 - b. Instale los certificados personalizados para **pcsd** con el comando **pcs pcsd certkey**.
 - c. Sincronice los certificados de **pcsd** en todos los nodos del clúster con el comando **pcs pcsd sync-certificates**.
4. Conéctese a la interfaz web **pcsd** utilizando la dirección IP flotante que configuró como recurso del clúster.



NOTA

Aunque configure la interfaz web **pcsd** para una alta disponibilidad, se le pedirá que vuelva a conectarse cuando el nodo al que se está conectando se caiga.

CAPÍTULO 9. CONFIGURACIÓN DEL CERCADO EN UN CLUSTER DE ALTA DISPONIBILIDAD DE RED HAT

Un nodo que no responde puede seguir accediendo a los datos. La única manera de estar seguro de que sus datos están a salvo es cercar el nodo utilizando STONITH. STONITH es un acrónimo de "Shoot The Other Node In The Head" (dispara al otro nodo en la cabeza) y protege tus datos de ser corrompidos por nodos deshonestos o accesos concurrentes. Usando STONITH, puedes estar seguro de que un nodo está realmente desconectado antes de permitir que se acceda a los datos desde otro nodo.

STONITH también tiene un papel que desempeñar en el caso de que un servicio en clúster no pueda ser detenido. En este caso, el clúster utiliza STONITH para forzar la desconexión de todo el nodo, lo que hace que sea seguro iniciar el servicio en otro lugar.

Para una información general más completa sobre el cercado y su importancia en un [cluster](#) de alta disponibilidad de Red Hat, consulte [Cerrado en un cluster de alta disponibilidad de Red Hat](#).

La implementación de STONITH en un clúster Pacemaker se realiza mediante la configuración de dispositivos de valla para los nodos del clúster.

9.1. VISUALIZACIÓN DE LOS AGENTES DE LA VALLA DISPONIBLES Y SUS OPCIONES

Utilice el siguiente comando para ver la lista de todos los agentes STONITH disponibles. Cuando se especifica un filtro, este comando muestra sólo los agentes STONITH que coinciden con el filtro.

```
pcs stonith [filter]
```

Utilice el siguiente comando para ver las opciones del agente STONITH especificado.

```
pcs stonith describir stonith_agent
```

Por ejemplo, el siguiente comando muestra las opciones del agente de la valla para APC sobre telnet/SSH.

```
# pcs stonith describe fence_apc
Stonith options for: fence_apc
ipaddr (required): IP Address or Hostname
login (required): Login Name
passwd: Login password or passphrase
passwd_script: Script to retrieve password
cmd_prompt: Force command prompt
secure: SSH connection
port (required): Physical plug number or name of virtual machine
identity_file: Identity file for ssh
switch: Physical switch number on device
inet4_only: Forces agent to use IPv4 addresses only
inet6_only: Forces agent to use IPv6 addresses only
ipport: TCP port to use for connection with device
action (required): Fencing Action
verbose: Verbose mode
debug: Write debug information to given file
version: Display version information and exit
help: Display help and exit
```

separator: Separator for CSV created by operation list
 power_timeout: Test X seconds for status change after ON/OFF
 shell_timeout: Wait X seconds for cmd prompt after issuing command
 login_timeout: Wait X seconds for cmd prompt after login
 power_wait: Wait X seconds after issuing ON/OFF
 delay: Wait X seconds before fencing is started
 retry_on: Count of attempts to retry power on



AVISO

Para los agentes de la valla que proporcionan una opción **method**, un valor de **cycle** no es compatible y no debe ser especificado, ya que puede causar la corrupción de datos.

9.2. CREACIÓN DE UN DISPOSITIVO DE VALLAS

El formato del comando para crear un dispositivo stonith es el siguiente. Para ver un listado de las opciones de creación de stoniths disponibles, consulte la pantalla **pcs stonith -h**.

```
pcs stonith create stonith_id stonith_device_type [stonith_device_options] [op operation_action operation_options]
```

El siguiente comando crea un único dispositivo de esgrima para un solo nodo.

```
# pcs stonith create MyStonith fence_virt pcmk_host_list=f1 op monitor interval=30s
```

Algunos dispositivos de cercado sólo pueden cercar un único nodo, mientras que otros dispositivos pueden cercar varios nodos. Los parámetros que se especifican al crear un dispositivo de vallado dependen de lo que el dispositivo de vallado admita y requiera.

- Algunos dispositivos de vallado pueden determinar automáticamente los nodos que pueden vallar.
- Puede utilizar el parámetro **pcmk_host_list** al crear un dispositivo de cercado para especificar todas las máquinas que están controladas por ese dispositivo de cercado.
- Algunos dispositivos de vallado requieren una asignación de nombres de host a las especificaciones que el dispositivo de vallado entiende. Puede asignar nombres de host con el parámetro **pcmk_host_map** al crear un dispositivo de vallado.

Para obtener información sobre los parámetros **pcmk_host_list** y **pcmk_host_map**, consulte [Propiedades generales de los dispositivos de cercado](#) .

Después de configurar un dispositivo de valla, es imprescindible que pruebe el dispositivo para asegurarse de que funciona correctamente. Para obtener información sobre cómo probar un dispositivo de valla, consulte [Probar un dispositivo de valla](#).

9.3. PROPIEDADES GENERALES DE LOS DISPOSITIVOS DE CERCADO

Cualquier nodo del clúster puede cercar cualquier otro nodo del clúster con cualquier dispositivo de cercado, independientemente de si el recurso de cercado está iniciado o detenido. El hecho de que el recurso esté iniciado sólo controla el monitor recurrente para el dispositivo, no si se puede utilizar, con las siguientes excepciones:

- Puede desactivar un dispositivo de esgrima ejecutando el comando **pcs stonith disable *stonith_id*** comando. Esto evitará que cualquier nodo utilice ese dispositivo.
- Para evitar que un nodo específico utilice un dispositivo de esgrima, puede configurar las restricciones de ubicación para el recurso de esgrima con el comando **pcs constraint location ... avoids**.
- La configuración de **stonith-enabled=false** deshabilitará el cercado por completo. Tenga en cuenta, sin embargo, que Red Hat no soporta clusters cuando el fencing está deshabilitado, ya que no es adecuado para un entorno de producción.

en [Tabla 9.1, "Propiedades generales de los dispositivos de esgrima"](#) se describen las propiedades generales que se pueden establecer para los dispositivos de cercado.

Tabla 9.1. Propiedades generales de los dispositivos de esgrima

Campo	Tipo	Por defecto	Descripción
pcmk_host_map	cadena		Un mapeo de nombres de host a números de puerto para dispositivos que no soportan nombres de host. Por ejemplo: node1:1;node2:2,3 indica al clúster que utilice el puerto 1 para el nodo1 y los puertos 2 y 3 para el nodo2
pcmk_host_list	cadena		Una lista de máquinas controladas por este dispositivo (Opcional a menos que pcmk_host_check=static-list).

Campo	Tipo	Por defecto	Descripción
pcmk_host_check	cadena	<p>* static-list si se establece pcmk_host_list o pcmk_host_map</p> <p>* En caso contrario, dynamic-list si el dispositivo de la valla admite la acción list</p> <p>* En caso contrario, status si el dispositivo de la valla admite la acción status</p> <p>*En caso contrario, none.</p>	Cómo determinar qué máquinas controla el dispositivo. Valores permitidos: dynamic-list (consultar el dispositivo), static-list (comprobar el atributo pcmk_host_list), ninguno (asumir que cada dispositivo puede cercar cada máquina)

9.4. OPCIONES AVANZADAS DE CONFIGURACIÓN DEL CERCADO

Tabla 9.2, “Propiedades avanzadas de los dispositivos de esgrima” resume las propiedades adicionales que puede configurar para los dispositivos de cercado. Tenga en cuenta que estas propiedades son sólo para uso avanzado.

Tabla 9.2. Propiedades avanzadas de los dispositivos de esgrima

Campo	Tipo	Por defecto	Descripción
pcmk_host_argument	cadena	puerto	Un parámetro alternativo para suministrar en lugar del puerto. Algunos dispositivos no admiten el parámetro de puerto estándar o pueden proporcionar otros adicionales. Utilícelo para especificar un parámetro alternativo, específico del dispositivo, que debe indicar la máquina que se va a cercar. Un valor de none puede utilizarse para indicar al clúster que no suministre ningún parámetro adicional.

Campo	Tipo	Por defecto	Descripción
pcmk_reboot_action	cadena	reiniciar	Un comando alternativo para ejecutar en lugar de reboot . Algunos dispositivos no soportan los comandos estándar o pueden proporcionar otros adicionales. Usa esto para especificar un comando alternativo, específico del dispositivo, que implemente la acción de reinicio.
pcmk_reboot_timeout	tiempo	60s	Especifique un tiempo de espera alternativo para utilizar en las acciones de reinicio en lugar de stonith-timeout . Algunos dispositivos necesitan mucho más/menos tiempo de lo normal para completarse. Utilice esta opción para especificar un tiempo de espera alternativo, específico del dispositivo, para las acciones de reinicio.
pcmk_reboot_retries	entero	2	El número máximo de veces para reintentar el comando reboot dentro del periodo de tiempo de espera. Algunos dispositivos no admiten conexiones múltiples. Las operaciones pueden fallar si el dispositivo está ocupado con otra tarea, por lo que Pacemaker reintentará automáticamente la operación, si queda tiempo. Utilice esta opción para modificar el número de veces que Pacemaker reintentará las acciones de reinicio antes de rendirse.

Campo	Tipo	Por defecto	Descripción
pcmk_off_action	cadena	fuera de	Un comando alternativo para ejecutar en lugar de off . Algunos dispositivos no soportan los comandos estándar o pueden proporcionar otros adicionales. Utilízalo para especificar un comando alternativo, específico del dispositivo, que implemente la acción de apagado.
pcmk_off_timeout	tiempo	60s	Especifique un tiempo de espera alternativo que se utilizará para las acciones de apagado en lugar de stonith-timeout . Algunos dispositivos necesitan mucho más o mucho menos tiempo para completarse de lo normal. Utilice esta opción para especificar un tiempo de espera alternativo, específico del dispositivo, para las acciones de desconexión.

Campo	Tipo	Por defecto	Descripción
pcmk_off_retries	entero	2	El número máximo de veces que se debe reintentar el comando de desconexión dentro del periodo de tiempo de espera. Algunos dispositivos no admiten conexiones múltiples. Las operaciones pueden fallar si el dispositivo está ocupado con otra tarea, por lo que Pacemaker reintentará automáticamente la operación, si queda tiempo. Utilice esta opción para modificar el número de veces que Pacemaker reintentará las acciones de desconexión antes de rendirse.
pcmk_list_action	cadena	lista	Un comando alternativo para ejecutar en lugar de list . Algunos dispositivos no soportan los comandos estándar o pueden proporcionar otros adicionales. Usa esto para especificar un comando alternativo, específico del dispositivo, que implemente la acción de la lista.
pcmk_list_timeout	tiempo	60s	Especifique un tiempo de espera alternativo para usar en las acciones de la lista. Algunos dispositivos necesitan mucho más o mucho menos tiempo de lo normal para completarse. Utilice esta opción para especificar un tiempo de espera alternativo, específico del dispositivo, para las acciones de la lista.

Campo	Tipo	Por defecto	Descripción
pcmk_list_retries	entero	2	El número máximo de veces para reintentar el comando list dentro del periodo de tiempo de espera. Algunos dispositivos no admiten conexiones múltiples. Las operaciones pueden fallar si el dispositivo está ocupado con otra tarea, por lo que Pacemaker reintentará automáticamente la operación, si queda tiempo. Utilice esta opción para modificar el número de veces que Pacemaker reintentará las acciones de la lista antes de rendirse.
pcmk_monitor_action	cadena	monitor	Un comando alternativo para ejecutar en lugar de monitor . Algunos dispositivos no soportan los comandos estándar o pueden proporcionar otros adicionales. Utilícelo para especificar un comando alternativo, específico del dispositivo, que implemente la acción del monitor.
pcmk_monitor_timeout	tiempo	60s	Especifique un tiempo de espera alternativo que se utilizará para las acciones del monitor en lugar de stonith-timeout . Algunos dispositivos necesitan mucho más o mucho menos tiempo para completarse de lo normal. Utilice esta opción para especificar un tiempo de espera alternativo, específico del dispositivo, para las acciones de monitorización.

Campo	Tipo	Por defecto	Descripción
pcmk_monitor_retries	entero	2	El número máximo de veces para reintentar el comando monitor dentro del periodo de tiempo de espera. Algunos dispositivos no admiten conexiones múltiples. Las operaciones pueden fallar si el dispositivo está ocupado con otra tarea, por lo que Pacemaker reintentará automáticamente la operación, si queda tiempo. Utilice esta opción para modificar el número de veces que Pacemaker reintentará las acciones de monitorización antes de rendirse.
pcmk_status_action	cadena	estado	Un comando alternativo para ejecutar en lugar de status . Algunos dispositivos no soportan los comandos estándar o pueden proporcionar otros adicionales. Usa esto para especificar un comando alternativo, específico del dispositivo, que implemente la acción de estado.

Campo	Tipo	Por defecto	Descripción
pcmk_status_timeout	tiempo	60s	Especifique un tiempo de espera alternativo para utilizar en las acciones de estado en lugar de stonith-timeout . Algunos dispositivos necesitan mucho más o mucho menos tiempo para completarse de lo normal. Utilice esta opción para especificar un tiempo de espera alternativo, específico del dispositivo, para las acciones de estado.
pcmk_status_retries	entero	2	El número máximo de veces para reintentar el comando de estado dentro del periodo de tiempo de espera. Algunos dispositivos no admiten conexiones múltiples. Las operaciones pueden fallar si el dispositivo está ocupado con otra tarea, por lo que Pacemaker reintentará automáticamente la operación, si queda tiempo. Utilice esta opción para modificar el número de veces que Pacemaker reintenta las acciones de estado antes de rendirse.

Campo	Tipo	Por defecto	Descripción
pcmk_delay_base	tiempo	0s	<p>Habilite un retardo base para las acciones de stonith y especifique un valor de retardo base. En un clúster con un número par de nodos, configurar un retardo puede ayudar a evitar que los nodos se cerquen entre sí al mismo tiempo en una división uniforme. Un retardo aleatorio puede ser útil cuando se utiliza el mismo dispositivo de vallado para todos los nodos, y diferentes retardos estáticos pueden ser útiles en cada dispositivo de vallado cuando se utiliza un dispositivo independiente para cada nodo. El retardo global se deriva de un valor de retardo aleatorio añadiendo este retardo estático para que la suma se mantenga por debajo del retardo máximo. Si se establece pcmk_delay_base pero no se establece pcmk_delay_max, no hay ningún componente aleatorio en el retardo y éste será el valor de pcmk_delay_base.</p> <p>Algunos agentes de vallas individuales implementan un parámetro de "retraso", que es independiente de los retrasos configurados con una propiedad pcmk_delay_*. Si se configuran ambos retardos, se suman y por lo tanto generalmente no se usarían en conjunto.</p>

Campo <code>delay_max</code>	Tipo	Por defecto	Descripción
			<p>retardo aleatorio para las acciones de stonith y especifique el máximo de retardo aleatorio. En un clúster con un número par de nodos, configurar un retardo puede ayudar a evitar que los nodos se cerquen entre sí al mismo tiempo en una división uniforme. Un retardo aleatorio puede ser útil cuando se utiliza el mismo dispositivo de vallado para todos los nodos, y diferentes retardos estáticos pueden ser útiles en cada dispositivo de vallado cuando se utiliza un dispositivo separado para cada nodo. El retardo global se deriva de este valor de retardo aleatorio añadiendo un retardo estático para que la suma se mantenga por debajo del retardo máximo. Si se establece <code>pcmk_delay_max</code> pero no se establece <code>pcmk_delay_base</code> no hay ningún componente estático en el retardo.</p> <p>Algunos agentes de vallas individuales implementan un parámetro de "retraso", que es independiente de los retrasos configurados con una propiedad <code>pcmk_delay_*</code>. Si se configuran ambos retardos, se suman y por lo tanto generalmente no se usarían en conjunto.</p>

Campo	Tipo	Por defecto	Descripción
pcmk_action_limit	entero	1	El número máximo de acciones que se pueden realizar en paralelo en este dispositivo. Es necesario configurar primero la propiedad de clúster concurrent-fencing=true (este es el valor por defecto para RHEL 8.1 y posteriores). Un valor de -1 es ilimitado.
pcmk_on_action	cadena	en	Sólo para uso avanzado: Un comando alternativo para ejecutar en lugar de on . Algunos dispositivos no admiten los comandos estándar o pueden proporcionar otros adicionales. Utilícelo para especificar un comando alternativo, específico del dispositivo, que implemente la acción on .
pcmk_on_timeout	tiempo	60s	Sólo para uso avanzado: Especifique un tiempo de espera alternativo que se utilizará para las acciones de on en lugar de stonith-timeout . Algunos dispositivos necesitan mucho más o mucho menos tiempo de lo normal para completarse. Utilice esta opción para especificar un tiempo de espera alternativo, específico del dispositivo, para las acciones de on .

Campo	Tipo	Por defecto	Descripción
pcmk_on_retries	entero	2	Sólo para uso avanzado: El número máximo de veces para reintentar el comando on dentro del periodo de tiempo de espera. Algunos dispositivos no admiten conexiones múltiples. Las operaciones pueden fail si el dispositivo está ocupado con otra tarea, por lo que Pacemaker reintentará automáticamente la operación, si queda tiempo. Utilice esta opción para modificar el número de veces que Pacemaker reintentará las acciones de on antes de darse por vencido.

Además de las propiedades que puede establecer para los dispositivos de vallado individuales, también hay propiedades de cluster que puede establecer que determinan el comportamiento del vallado, como se describe en [Tabla 9.3, "Propiedades de los racimos que determinan el comportamiento del cercado"](#) .

Tabla 9.3. Propiedades de los racimos que determinan el comportamiento del cercado

Opción	Por defecto	Descripción
stonith-enabled	verdadero	Indica que los nodos fallidos y los nodos con recursos que no pueden ser detenidos deben ser cercados. La protección de los datos requiere que se configure esta opción true . Si true , o no está configurado, el cluster se negará a iniciar recursos a menos que uno o más recursos STONITH hayan sido configurados también. Red Hat sólo admite clusters con este valor establecido en true .

Opción	Por defecto	Descripción
stonith-action	reiniciar	Acción a enviar al dispositivo STONITH. Valores permitidos: reboot , off . El valor poweroff también está permitido, pero sólo se utiliza para los dispositivos heredados.
stonith-timeout	60s	Cuánto tiempo hay que esperar para que se complete una acción de STONITH.
stonith-max-attempts	10	Cuántas veces puede fallar el esgrima para un objetivo antes de que el clúster deje de reintentarlo inmediatamente.
stonith-watchdog-timeout		El tiempo máximo de espera hasta que se pueda asumir que un nodo ha sido eliminado por el perro guardián del hardware. Se recomienda que este valor sea el doble del valor del tiempo de espera del perro guardián del hardware. Esta opción es necesaria sólo si se utiliza el SBD basado en watchdog para el fencing.
concurrent-fencing	true (RHEL 8.1 y posteriores)	Permitir que las operaciones de cercado se realicen en paralelo.

Opción	Por defecto	Descripción
fence-reaction	detener	<p>(Red Hat Enterprise Linux 8.2 y posterior) Determina cómo debe reaccionar un nodo del cluster si se le notifica su propio cercado. Un nodo de cluster puede recibir una notificación de su propio fencing si el fencing está mal configurado, o si el fabric fencing está en uso y no corta la comunicación del cluster. Los valores permitidos son stop para intentar detener inmediatamente Pacemaker y permanecer detenido, o panic para intentar reiniciar inmediatamente el nodo local, volviendo a detenerse en caso de fallo.</p> <p>Aunque el valor por defecto para esta propiedad es stop, la opción más segura para este valor es panic, que intenta reiniciar inmediatamente el nodo local. Si se prefiere el comportamiento de parada, como es más probable que ocurra en conjunción con el fabric fencing, se recomienda establecerlo explícitamente.</p>

Para obtener información sobre la configuración de las propiedades del [clúster](#), consulte [Configuración y eliminación de las propiedades del clúster](#).

9.5. PRUEBA DE UN DISPOSITIVO DE VALLA

El cercado es una parte fundamental de la infraestructura de Red Hat Cluster y por lo tanto es importante validar o probar que el cercado funciona correctamente.

Utilice el siguiente procedimiento para probar un dispositivo de valla.

1. Utilice ssh, telnet, HTTP o cualquier protocolo remoto que se utilice para conectarse al dispositivo para iniciar sesión manualmente y probar el dispositivo de vallado o ver qué salida se da. Por ejemplo, si va a configurar el cercado para un dispositivo habilitado para IPMI, entonces intente iniciar la sesión de forma remota con **ipmitool**. Tome nota de las opciones utilizadas al iniciar la sesión manualmente porque esas opciones podrían ser necesarias al utilizar el agente de cercado.

Si no puede iniciar sesión en el dispositivo de vallado, verifique que el dispositivo es pingable, que no hay nada, como una configuración de cortafuegos, que esté impidiendo el acceso al dispositivo de vallado, que el acceso remoto está activado en el dispositivo de vallado y que las credenciales son correctas.

2. Ejecute el agente de vallas manualmente, utilizando el script del agente de vallas. Esto no requiere que los servicios del clúster se estén ejecutando, por lo que puede realizar este paso

antes de que el dispositivo se configure en el clúster. Esto puede asegurar que el dispositivo de valla está respondiendo correctamente antes de proceder.



NOTA

Los ejemplos de esta sección utilizan el script **fence_ipmilan** fence agent para un dispositivo iLO. El agente de valla real que utilizará y el comando que llama a ese agente dependerá del hardware de su servidor. Deberá consultar la página de manual del agente de vallas que esté utilizando para determinar qué opciones debe especificar. Normalmente necesitará conocer el nombre de usuario y la contraseña del dispositivo de valla y otra información relacionada con el dispositivo de valla.

El siguiente ejemplo muestra el formato que se utilizaría para ejecutar el script del agente de vallado **fence_ipmilan** con el parámetro **-o status** para comprobar el estado de la interfaz del dispositivo de vallado en otro nodo sin llegar a vallarlo. Esto le permite probar el dispositivo y hacerlo funcionar antes de intentar reiniciar el nodo. Cuando se ejecuta este comando, se especifica el nombre y la contraseña de un usuario iLO que tiene permisos de encendido y apagado para el dispositivo iLO.

```
# fence_ipmilan -a ipaddress -l username -p password -o status
```

El siguiente ejemplo muestra el formato que se utilizaría para ejecutar el script del agente de la valla **fence_ipmilan** con el parámetro **-o reboot**. La ejecución de este comando en un nodo reinicia el nodo gestionado por este dispositivo iLO.

```
# fence_ipmilan -a ipaddress -l username -p password -o reboot
```

Si el agente de la valla no pudo realizar correctamente una acción de estado, apagado, encendido o reinicio, debe comprobar el hardware, la configuración del dispositivo de la valla y la sintaxis de sus comandos. Además, puede ejecutar el script del agente de valla con la salida de depuración activada. La salida de depuración es útil para algunos agentes de cercado para ver en qué parte de la secuencia de eventos el script del agente de cercado está fallando cuando se registra en el dispositivo de cercado.

```
# fence_ipmilan -a ipaddress -l username -p password -o status -D /tmp/${hostname}-fence_agent.debug
```

Al diagnosticar un fallo que se ha producido, debe asegurarse de que las opciones que especificó al iniciar sesión manualmente en el dispositivo de valla son idénticas a las que pasó al agente de valla con el script del agente de valla.

En el caso de los agentes de valla que admiten una conexión encriptada, es posible que aparezca un error debido a que falla la validación del certificado, lo que requiere que confíe en el host o que utilice el parámetro **ssl-insecure** del agente de valla. Del mismo modo, si SSL/TLS está deshabilitado en el dispositivo de destino, es posible que deba tenerlo en cuenta al configurar los parámetros SSL para el agente de valla.



NOTA

Si el agente de cercado que se está probando es un **fence_drac**, **fence_ilo**, o algún otro agente de cercado para un dispositivo de gestión de sistemas que sigue fallando, entonces vuelva a intentar **fence_ipmilan**. La mayoría de las tarjetas de gestión de sistemas soportan el inicio de sesión remoto IPMI y el único agente de cercado soportado es **fence_ipmilan**.

- Una vez que se ha configurado el dispositivo de vallado en el cluster con las mismas opciones que funcionaron manualmente y se ha iniciado el cluster, pruebe el vallado con el comando **pcs stonith fence** desde cualquier nodo (o incluso varias veces desde diferentes nodos), como en el siguiente ejemplo. El comando **pcs stonith fence** lee la configuración del cluster desde el CIB y llama al agente de vallado tal y como está configurado para ejecutar la acción de vallado. Esto verifica que la configuración del cluster es correcta.

pcs stonith fence node_name

Si el comando **pcs stonith fence** funciona correctamente, significa que la configuración de vallado del clúster debería funcionar cuando se produce un evento de vallado. Si el comando falla, significa que la administración del clúster no puede invocar el dispositivo de vallado a través de la configuración que ha recuperado. Compruebe los siguientes problemas y actualice la configuración del clúster según sea necesario.

- Comprueba la configuración de tu valla. Por ejemplo, si ha utilizado un mapa de host, debe asegurarse de que el sistema puede encontrar el nodo utilizando el nombre de host que ha proporcionado.
 - Comprueba si la contraseña y el nombre de usuario del dispositivo incluyen algún carácter especial que pueda ser malinterpretado por el shell bash. Asegurarse de introducir las contraseñas y los nombres de usuario entre comillas podría solucionar este problema.
 - Comprueba si puedes conectarte al dispositivo utilizando la dirección IP o el nombre de host exactos que has especificado en el comando **pcs stonith**. Por ejemplo, si das el nombre de host en el comando stonith pero pruebas utilizando la dirección IP, esa no es una prueba válida.
 - Si el protocolo que utiliza el dispositivo de la valla es accesible para usted, utilice ese protocolo para intentar conectarse al dispositivo. Por ejemplo, muchos agentes utilizan ssh o telnet. Debería intentar conectarse al dispositivo con las credenciales que proporcionó al configurar el dispositivo, para ver si obtiene un aviso válido y puede iniciar sesión en el dispositivo.
Si usted determina que todos sus parámetros son apropiados pero todavía tiene problemas para conectarse a su dispositivo de valla, puede comprobar el registro en el propio dispositivo de valla, si el dispositivo lo proporciona, que mostrará si el usuario se ha conectado y qué comando emitió el usuario. También puede buscar en el archivo **/var/log/messages** las instancias de stonith y error, que podrían dar alguna idea de lo que está ocurriendo, pero algunos agentes pueden proporcionar información adicional.
- Una vez que las pruebas del dispositivo de la valla están funcionando y el clúster está en marcha, pruebe un fallo real. Para ello, realiza una acción en el clúster que debería iniciar una pérdida de tokens.
 - Desmontar una red. La forma de desmontar una red depende de su configuración específica. En muchos casos, puede sacar físicamente los cables de red o de alimentación del host. Para obtener información sobre la simulación de un fallo de red, consulte [¿Cuál es la forma adecuada de simular un fallo de red en un clúster RHEL?](#)



NOTA

Desactivar la interfaz de red en el host local en lugar de desconectar físicamente los cables de red o de alimentación no se recomienda como prueba de cercado porque no simula con precisión un fallo típico del mundo real.

- Bloquea el tráfico de corosync tanto de entrada como de salida utilizando el firewall local. El siguiente ejemplo bloquea corosync, asumiendo que se utiliza el puerto por defecto de corosync, **firewalld** se utiliza como firewall local, y la interfaz de red utilizada por corosync está en la zona de firewall por defecto:

```
# firewall-cmd --direct --add-rule ipv4 filter OUTPUT 2 -p udp --dport=5405 -j DROP
# firewall-cmd --add-rich-rule='rule family="ipv4" port port="5405" protocol="udp" drop'
```

- Simule un fallo y haga entrar en pánico a su máquina con **sysrq-trigger**. Tenga en cuenta, sin embargo, que desencadenar un pánico del kernel puede causar la pérdida de datos; se recomienda que primero desactive los recursos de su clúster.

```
# echo c > /proc/sysrq-trigger
```

9.6. CONFIGURACIÓN DE LOS NIVELES DE VALLADO

Pacemaker admite nodos de esgrima con múltiples dispositivos a través de una función denominada topologías de esgrima. Para implementar topologías, cree los dispositivos individuales como lo haría normalmente y luego defina uno o más niveles de esgrima en la sección de topología de esgrima en la configuración.

- Cada nivel se intenta en orden numérico ascendente, empezando por el 1.
- Si un dispositivo falla, el proceso termina para el nivel actual. No se ejercitan más dispositivos en ese nivel y se intenta el siguiente nivel en su lugar.
- Si todos los dispositivos son cercados con éxito, entonces ese nivel ha tenido éxito y no se intentan otros niveles.
- La operación finaliza cuando se ha superado un nivel (éxito), o se han intentado todos los niveles (fracaso).

Utilice el siguiente comando para añadir un nivel de cercado a un nodo. Los dispositivos se dan como una lista separada por comas de ids de stoniths, que se intentan para el nodo en ese nivel.

```
pcs stonith level add level node devices
```

El siguiente comando muestra todos los niveles de esgrima que están configurados actualmente.

```
pcs stonith level
```

En el siguiente ejemplo, hay dos dispositivos de valla configurados para el nodo **rh7-2**: un dispositivo de valla ilo llamado **my_ilo** y un dispositivo de valla apc llamado **my_apc**. Estos comandos configuran los niveles de vallado para que, si el dispositivo **my_ilo** falla y no puede vallar el nodo, Pacemaker intente utilizar el dispositivo **my_apc**. Este ejemplo también muestra la salida del comando **pcs stonith level** después de configurar los niveles.


```
# pcs stonith level add 1 rh7-2 my_ilo
# pcs stonith level add 2 rh7-2 my_apc
# pcs stonith level
Node: rh7-2
Level 1 - my_ilo
Level 2 - my_apc
```

El siguiente comando elimina el nivel de valla para el nodo y los dispositivos especificados. Si no se especifica ningún nodo o dispositivo, el nivel de valla que se especifique se eliminará de todos los nodos.

```
pcs stonith level remove level [node_id] [stonith_id] ... [stonith_id]
```

El siguiente comando borra los niveles de valla en el nodo o id de piedra especificado. Si no se especifica un nodo o un id de stonith, se borran todos los niveles de valla.

```
pcs stonith level clear [node|stonith_id(s)]
```

Si especifica más de un id de stonith, deben estar separados por una coma y sin espacios, como en el siguiente ejemplo.

```
# pcs stonith level clear dev_a,dev_b
```

El siguiente comando verifica que todos los dispositivos y nodos especificados en los niveles de la valla existen.

```
pcs stonith level verify
```

Puede especificar nodos en la topología de vallado mediante una expresión regular aplicada a un nombre de nodo y mediante un atributo de nodo y su valor. Por ejemplo, los siguientes comandos configuran los nodos **node1**, **node2**, y **node3** para utilizar los dispositivos de vallado **apc1** y **apc2**, y los nodos **node4**, **node5**, y **node6** para utilizar los dispositivos de vallado **apc3** y **apc4**.

```
pcs stonith level add 1 "regexp%node[1-3]" apc1,apc2
pcs stonith level add 1 "regexp%node[4-6]" apc3,apc4
```

Los siguientes comandos producen los mismos resultados utilizando la coincidencia de atributos de nodos.

```
pcs node attribute node1 rack=1
pcs node attribute node2 rack=1
pcs node attribute node3 rack=1
pcs node attribute node4 rack=2
pcs node attribute node5 rack=2
pcs node attribute node6 rack=2
pcs stonith level add 1 attrib%rack=1 apc1,apc2
pcs stonith level add 1 attrib%rack=2 apc3,apc4
```

9.7. CONFIGURAR EL CERCADO PARA LAS FUENTES DE ALIMENTACIÓN REDUNDANTES

Cuando se configura el cercado para las fuentes de alimentación redundantes, el clúster debe asegurarse de que cuando se intenta reiniciar un host, ambas fuentes de alimentación se apagan antes de que se vuelva a encender cualquiera de ellas.

Si el nodo nunca pierde completamente la energía, es posible que no libere sus recursos. Esto abre la posibilidad de que los nodos accedan a estos recursos simultáneamente y los corrompan.

Es necesario definir cada dispositivo sólo una vez y especificar que ambos son necesarios para cercar el nodo, como en el siguiente ejemplo.

```
# pcs stonith create apc1 fence_apc_snmp ipaddr=apc1.example.com login=user
passwd='7a4D#1j!pz864' pcmk_host_map="node1.example.com:1;node2.example.com:2"

# pcs stonith create apc2 fence_apc_snmp ipaddr=apc2.example.com login=user
passwd='7a4D#1j!pz864' pcmk_host_map="node1.example.com:1;node2.example.com:2"

# pcs stonith level add 1 node1.example.com apc1,apc2
# pcs stonith level add 1 node2.example.com apc1,apc2
```

9.8. VISUALIZACIÓN DE LOS DISPOSITIVOS DE VALLAS CONFIGURADOS

El siguiente comando muestra todos los dispositivos de valla actualmente configurados. Si se especifica un *stonith_id*, el comando muestra las opciones de ese dispositivo stonith configurado solamente. Si se especifica la opción **--full**, se muestran todas las opciones de stonith configuradas.

```
pcs stonith config [stonith_id] [--full]
```

9.9. MODIFICACIÓN Y SUPRESIÓN DE LOS DISPOSITIVOS DE LA VALLA

Utilice el siguiente comando para modificar o añadir opciones a un dispositivo de cercado actualmente configurado.

```
pcs stonith update stonith_id [stonith_device_options]
```

Utilice el siguiente comando para eliminar un dispositivo de cercado de la configuración actual.

```
pcs stonith delete stonith_id
```

9.10. CERCAR MANUALMENTE UN NODO DEL CLÚSTER

Puedes cercar un nodo manualmente con el siguiente comando. Si se especifica **--off** se utilizará la llamada de la API **off** a stonith que apagará el nodo en lugar de reiniciarlo.

```
pcs stonith fence node [--off]
```

En una situación en la que ningún dispositivo stonith es capaz de cercar un nodo aunque ya no esté activo, es posible que el clúster no pueda recuperar los recursos del nodo. Si esto ocurre, después de asegurarse manualmente de que el nodo está apagado, puede introducir el siguiente comando para confirmar al clúster que el nodo está apagado y liberar sus recursos para su recuperación.



AVISO

Si el nodo que especifica no está realmente apagado, sino que ejecuta el software del clúster o los servicios normalmente controlados por el clúster, se producirá una corrupción de datos/un fallo del clúster.

```
pcs stonith confirmar node
```

9.11. DESACTIVACIÓN DE UN DISPOSITIVO DE VALLA

Para desactivar un dispositivo/recurso de esgrima, se ejecuta el comando **pcs stonith disable**.

El siguiente comando desactiva el dispositivo de la valla **myapc**.

```
# pcs stonith disable myapc
```

9.12. IMPEDIR QUE UN NODO UTILICE UN DISPOSITIVO DE VALLA

Para evitar que un nodo específico utilice un dispositivo de esgrima, puede configurar restricciones de ubicación para el recurso de esgrima.

El siguiente ejemplo evita que el dispositivo de valla **node1-ipmi** se ejecute en **node1**.

```
# pcs constraint location node1-ipmi avoids node1
```

9.13. CONFIGURACIÓN DE ACPI PARA SU USO CON DISPOSITIVOS DE VALLA INTEGRADOS

Si su clúster utiliza dispositivos de cercado integrados, debe configurar ACPI (Advanced Configuration and Power Interface) para garantizar un cercado inmediato y completo.

Si un nodo del clúster está configurado para ser cercado por un dispositivo de cercado integrado, desactive el ACPI Soft-Off para ese nodo. La desactivación de ACPI Soft-Off permite que un dispositivo de valla integrado apague un nodo de forma inmediata y completa en lugar de intentar un apagado limpio (por ejemplo, **shutdown -h now**). De lo contrario, si el ACPI Soft-Off está habilitado, un dispositivo de barrera integrado puede tardar cuatro o más segundos en apagar un nodo (vea la nota que sigue). Además, si el ACPI Soft-Off está habilitado y un nodo entra en pánico o se congela durante el apagado, un dispositivo de cercado integrado puede no ser capaz de apagar el nodo. En esas circunstancias, el cercado se retrasa o no tiene éxito. En consecuencia, cuando un nodo está cercado con un dispositivo de cercado integrado y el ACPI Soft-Off está habilitado, un clúster se recupera lentamente o requiere la intervención administrativa para recuperarse.



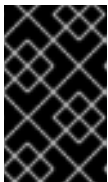
NOTA

El tiempo necesario para cercar un nodo depende del dispositivo de cercado integrado que se utilice. Algunos dispositivos de vallado integrados realizan el equivalente a pulsar y mantener el botón de encendido; por lo tanto, el dispositivo de vallado apaga el nodo en cuatro o cinco segundos. Otros dispositivos de valla integrados realizan el equivalente a pulsar el botón de encendido momentáneamente, confiando en el sistema operativo para apagar el nodo; por lo tanto, el dispositivo de valla apaga el nodo en un lapso de tiempo mucho mayor que cuatro o cinco segundos.

- La forma preferida de desactivar el ACPI Soft-Off es cambiar la configuración de la BIOS a "instant-off" o una configuración equivalente que apague el nodo sin demora, como se describe en [Sección 9.13.1, "Desactivación de ACPI Soft-Off con la BIOS"](#).

La desactivación de ACPI Soft-Off con la BIOS puede no ser posible en algunos sistemas. Si la desactivación de ACPI con la BIOS no es satisfactoria para su clúster, puede desactivar ACPI Soft-Off con uno de los siguientes métodos alternativos:

- Configurando **HandlePowerKey=ignore** en el archivo `/etc/systemd/logind.conf` y verificando que el nodo se apague inmediatamente cuando sea cercado, como se describe en [Sección 9.13.2, "Desactivación de ACPI Soft-Off en el archivo logind.conf"](#). Este es el primer método alternativo para desactivar el ACPI Soft-Off.
- Añadiendo **acpi=off** a la línea de comandos de arranque del kernel, como se describe en [Sección 9.13.3, "Desactivar completamente ACPI en el archivo GRUB 2"](#). Este es el segundo método alternativo para desactivar el ACPI Soft-Off, si el método preferido o el primer método alternativo no está disponible.



IMPORTANTE

Este método desactiva completamente ACPI; algunos ordenadores no arrancan correctamente si ACPI está completamente desactivado. Utilice este método *only* si los otros métodos no son efectivos para su clúster.

9.13.1. Desactivación de ACPI Soft-Off con la BIOS

Puede desactivar el ACPI Soft-Off configurando la BIOS de cada nodo del cluster con el siguiente procedimiento.



NOTA

El procedimiento para desactivar el ACPI Soft-Off con la BIOS puede variar entre los sistemas de servidor. Debe verificar este procedimiento con la documentación de su hardware.

1. Reinicie el nodo e inicie el programa **BIOS CMOS Setup Utility**.
2. Navegue hasta el menú Power (o el menú de gestión de energía equivalente).
3. En el menú Power, ajuste la función **Soft-Off by PWR-BTTN** (o su equivalente) a **Instant-Off** (o el ajuste equivalente que apague el nodo mediante el botón de encendido sin demora). **BIOS CMOS Setup Utility**: muestra un menú Power con **ACPI Function** ajustado a **Enabled** y **Soft-Off by PWR-BTTN** ajustado a **Instant-Off**.

**NOTA**

Los equivalentes a **ACPI Function**, **Soft-Off by PWR-BTTN**, y **Instant-Off** pueden variar entre ordenadores. Sin embargo, el objetivo de este procedimiento es configurar la BIOS para que el ordenador se apague mediante el botón de encendido sin demora.

4. Salga del programa **BIOS CMOS Setup Utility**, guardando la configuración de la BIOS.
5. Compruebe que el nodo se apaga inmediatamente cuando se le pone una valla. Para obtener información sobre la comprobación de un dispositivo de vallado, consulte [Probar un dispositivo de vallado](#).

BIOS CMOS Setup Utility:

```
`Soft-Off by PWR-BTTN` set to
`Instant-Off`
```

```
+-----+-----+
| ACPI Function      [Enabled] | Item Help | |
| ACPI Suspend Type [S1(POS)] |-----|
| x Run VGABIOS if S3 Resume Auto | Menu Level * |
| Suspend Mode      [Disabled] | |
| HDD Power Down    [Disabled] | |
| Soft-Off by PWR-BTTN [Instant-Off | |
| CPU THRM-Throttling [50.0%] | |
| Wake-Up by PCI card [Enabled] | |
| Power On by Ring   [Enabled] | |
| Wake Up On LAN     [Enabled] | |
| x USB KB Wake-Up From S3 Disabled | |
| Resume by Alarm    [Disabled] | |
| x Date(of Month) Alarm 0 | |
| x Time(hh:mm:ss) Alarm 0 : 0 : | |
| POWER ON Function [BUTTON ONLY | |
| x KB Power ON Password Enter | |
| x Hot Key Power ON Ctrl-F1 | |
| | | |
| | | |
+-----+-----+
```

Este ejemplo muestra **ACPI Function** ajustado a **Enabled**, y **Soft-Off by PWR-BTTN** ajustado a **Instant-Off**.

9.13.2. Desactivación de ACPI Soft-Off en el archivo logind.conf

Para desactivar el manejo de la tecla de encendido en el archivo `/etc/systemd/logind.conf`, utilice el siguiente procedimiento.

1. Defina la siguiente configuración en el archivo `/etc/systemd/logind.conf`:

```
HandlePowerKey=ignorar
```

2. Reinicie el servicio **systemd-logind**:

```
# systemctl restart systemd-logind.service
```

3. Compruebe que el nodo se apaga inmediatamente cuando se le pone una valla. Para obtener información sobre la comprobación de un dispositivo [de vallado](#), consulte [Probar un dispositivo de vallado](#).

9.13.3. Desactivar completamente ACPI en el archivo GRUB 2

Puede desactivar el ACPI Soft-Off añadiendo **acpi=off** a la entrada del menú GRUB para un kernel.



IMPORTANTE

Este método desactiva completamente ACPI; algunos ordenadores no arrancan correctamente si ACPI está completamente desactivado. Utilice este método *only* si los otros métodos no son efectivos para su clúster.

Utilice el siguiente procedimiento para desactivar ACPI en el archivo GRUB 2:

1. Utilice la opción **--args** en combinación con la opción **--update-kernel** de la herramienta **grubby** para cambiar el archivo **grub.cfg** de cada nodo del clúster de la siguiente manera:

```
# grubby --args=acpi=off --update-kernel=ALL
```

2. Reinicia el nodo.
3. Compruebe que el nodo se apaga inmediatamente cuando se le pone una valla. Para obtener información sobre la comprobación de un dispositivo [de vallado](#), consulte [Probar un dispositivo de vallado](#).

CAPÍTULO 10. CONFIGURACIÓN DE LOS RECURSOS DEL CLÚSTER

El formato del comando para crear un recurso de cluster es el siguiente:

```
pcs resource create resource_id [standard:[provider:]]type [resource_options] [op operation_action
operation_options [operation_action operation_options ]...] [meta meta_options...] [clone
[clone_options] | master [master_options] | --group group_name [--before resource_id | --after
resource_id] | [bundle bundle_id] [--disabled] [--wait[=n]]
```

Las principales opciones de creación de recursos de clúster son las siguientes:

- Cuando se especifica la opción **--group**, el recurso se añade al grupo de recursos nombrado. Si el grupo no existe, se crea el grupo y se añade este recurso al grupo.
- Las opciones **--before** y **--after** especifican la posición del recurso añadido en relación con un recurso que ya existe en un grupo de recursos.
- Especificar la opción **--disabled** indica que el recurso no se inicia automáticamente.

Puede determinar el comportamiento de un recurso en un clúster configurando restricciones para ese recurso.

Ejemplos de creación de recursos

El siguiente comando crea un recurso con el nombre **VirtualIP** de estándar **ocf**, proveedor **heartbeat**, y tipo **IPAddr2**. La dirección flotante de este recurso es 192.168.0.120, y el sistema comprobará si el recurso está funcionando cada 30 segundos.

```
# pcs resource create VirtualIP ocf:heartbeat:IPAddr2 ip=192.168.0.120 cidr_netmask=24 op
monitor interval=30s
```

Alternativamente, puede omitir los campos *standard* y *provider* y utilizar el siguiente comando. De este modo, el estándar será **ocf** y el proveedor **heartbeat**.

```
# pcs resource create VirtualIP IPAddr2 ip=192.168.0.120 cidr_netmask=24 op monitor
interval=30s
```

Borrar un recurso configurado

Utilice el siguiente comando para eliminar un recurso configurado.

```
supresión de recursos pcs resource_id
```

Por ejemplo, el siguiente comando elimina un recurso existente con un ID de recurso de **VirtualIP**.

```
# pcs resource delete VirtualIP
```

10.1. IDENTIFICADORES DE AGENTES DE RECURSOS

Los identificadores que se definen para un recurso indican al clúster qué agente debe utilizar para el recurso, dónde encontrar ese agente y a qué estándares se ajusta. [Tabla 10.1, "Identificadores de agentes de recursos"](#), describe estas propiedades.

Tabla 10.1. Identificadores de agentes de recursos

Campo	Descripción
estándar	<p>La norma a la que se ajusta el agente. Valores permitidos y su significado:</p> <ul style="list-style-type: none"> * ocf - El <i>type</i> especificado es el nombre de un archivo ejecutable que se ajusta a la API del agente de recursos de Open Cluster Framework y que se encuentra bajo /usr/lib/ocf/resource.d/provider * lsb - El <i>type</i> especificado es el nombre de un archivo ejecutable que se ajusta a las acciones de script de inicio de base de Linux. Si el tipo no especifica una ruta completa, el sistema lo buscará en el directorio /etc/init.d. * systemd - El <i>type</i> especificado es el nombre de una unidad instalada systemd * service - Pacemaker buscará el <i>type</i> especificado, primero como agente lsb y luego como agente systemd * nagios - El <i>type</i> especificado es el nombre de un archivo ejecutable conforme a la API de plugins de Nagios y ubicado en el directorio /usr/libexec/nagios/plugins, con metadatos de estilo OCF almacenados por separado en el directorio /usr/share/nagios/plugins-metadata (disponible en el paquete nagios-agents-metadata para ciertos plugins comunes).
tipo	El nombre del agente de recursos que desea utilizar, por ejemplo IPaddr o Filesystem
proveedor	La especificación OCF permite que varios proveedores suministren el mismo agente de recursos. La mayoría de los agentes suministrados por Red Hat utilizan heartbeat como proveedor.

Tabla 10.2, “Comandos para mostrar las propiedades de los recursos” resume los comandos que muestran las propiedades de los recursos disponibles.

Tabla 10.2. Comandos para mostrar las propiedades de los recursos

comando de visualización de pcs	Salida
pcs resource list	Muestra una lista de todos los recursos disponibles.
pcs resource standards	Muestra una lista de estándares de agentes de recursos disponibles.

comando de visualización de pcs	Salida
pcs resource providers	Muestra una lista de proveedores de agentes de recursos disponibles.
pcs resource list <i>string</i>	Muestra una lista de recursos disponibles filtrados por la cadena especificada. Puede utilizar este comando para mostrar los recursos filtrados por el nombre de un estándar, un proveedor o un tipo.

10.2. VISUALIZACIÓN DE LOS PARÁMETROS ESPECÍFICOS DE LOS RECURSOS

Para cualquier recurso individual, puede utilizar el siguiente comando para mostrar una descripción del recurso, los parámetros que puede establecer para ese recurso y los valores por defecto que se establecen para el recurso.

```
pcs resource describe [standard:[provider:]]type
```

Por ejemplo, el siguiente comando muestra información de un recurso de tipo **apache**.

```
# pcs resource describe ocf:heartbeat:apache
This is the resource agent for the Apache Web server.
This resource agent operates both version 1.x and version 2.x Apache
servers.

...
```

10.3. CONFIGURACIÓN DE LAS OPCIONES META DE LOS RECURSOS

Además de los parámetros específicos del recurso, puede configurar opciones adicionales para cualquier recurso. Estas opciones son utilizadas por el clúster para decidir cómo debe comportarse su recurso.

[Tabla 10.3, “Meta opciones de recursos”](#) describe las opciones meta de los recursos.

Tabla 10.3. Meta opciones de recursos

C a m p o	Por defecto	D e s c r i p c i ó n
p r i o r i t y	0	S i n o t o d o s l o s r e c u r s o s p u e d e n e s t a r a c t i v o s , e l c l

C a m p o	Por defecto	Ú D e s c r i p c i ó n
		a l o s r e c u r s o s d e m e n o r p r i o r i d a d p a r a m a n t e n e r a c t i

C a m p o	Por defecto	D e s c r i p c i ó n
		r p r i o r i d a d .
t a r g e t - r o l e	Started	¿ E n q u é e s t a d o d e b e r í a e l c l ú s t e r i n t

C a m p o	Por defecto	D e s c r i p c i ó n
		r e s t e r e c u r s o ? V a l o r e s p e r m i t i d o s : * S t o p p e d - F o r

C a m p o	Por defecto	D e s c r i p c i ó n
		o n d e l r e c u r s o * S t a r t e d - P e r m i t i r q u e e l r e c u r s o s e

C a m p o	Por defecto	D e s c r i p c i o n
		C a s o d e l o s c l o n e s p r o m o c i o n a b l e s , q u e s e p r o m u e v a a

C a m p o	Por defecto	D e s c r i p c i o n
		o s i e s a p r o p i a d o) * <i>M</i> <i>a</i> <i>s</i> <i>t</i> <i>e</i> <i>r</i> - <i>P</i> <i>e</i> <i>r</i> <i>m</i> <i>i</i> <i>t</i> <i>i</i> <i>r</i> <i>q</i> <i>u</i> <i>e</i> <i>e</i> <i>l</i> <i>r</i> <i>e</i> <i>c</i> <i>u</i> <i>r</i> <i>s</i> <i>o</i>

C a m p o	Por defecto	D e s c r i p c i ó n
		S u c c a s o , s e p r o m u e v a * S / a v e - P e r m i t e q u e e l r e c u r s o

C a m p o	Por defecto	D e s c r i p c i o n
		o s ó l o e n m o d o e s c l a v o s i e l r e c u r s o e s p r o m o c i o n a b l e

C a m p o n a g e d	Por defecto	D e s c r i p c i ó n s
		t e r p e r m i s o p a r a i n i c i a r y d e t e n e r e l r e c u r s o ? V a l o r

C a m p o	Por defecto	D e s c r i p c i ó n
		: t r u e , f a l s e
r e s o u r c e - s t i c k i n e s s	0	V a l o r q u e i n d i c a c u á n t o p r e f i e r e l r

C a m p o	Por defecto	D e s c r i p c i ó n
		S e d o n d e e s t á .
r e q u i r e s	Calculado	I n d i c a e n q u é c o n d i c i o n e s s e p u e d e i n

C a m p o	Por defecto	D e s c r i p c i ó n
		S o · E l v a l o r p o r d e f e c t o e s f e n c i n g , e x c e p t o e n l a s c

C a m p o	Por defecto	D e s c r i p c i ó n
		d i c a d a s a c o n t i n u a c i ó n · V a l o r e s p o s i b l e s : * n o t h i

C a m p o	Por defecto	D e s c r i p c i o n
		S i e m p r e p u e d e i n i c i a r e l r e c u r s o . * q u o r u m - E l c l u s

C a m p o	Por defecto	D e s c r i p c i ó n
		I n i c i a r e s t e r e c u r s o s i l a m a y o r í a d e l o s n o d o s c o n f i

C a m p o	Por defecto	D e s c r i p c i ó n
		a c t i v o s . E s t e e s e l v a l o r p o r d e f e c t o s i n i t i a l

C a m p o	Por defecto	b D e s c r i p c i o n
		e l s t a n d a r d d e l r e c u r s o e s s t o n i t h . * f e n c i n g - E l c l

C a m p o	Por defecto	D e s c r i p c i ó n
		d e i n i c i a r e s t e r e c u r s o s i l a m a y o r í a d e l o s n o d o s c o n

C a m p o	Por defecto	D e s c r i p c i o n
		a n a c t i v o s a n d c u a l q u i e r n o d o f a l l i d o o d e s c o n o c i d o

C a m p o	Por defecto	D e s c r i p c i ó n
		o . * u n f e n c i n g - E l c l ú s t e r s ó l o p u e d e i n i c i a r e s t e r

C a m p o	Por defecto	D e s c r i p c i o n
		Y o r í a d e l o s n o d o s c o n f i g u r a d o s e s t á n a c t i v o s a n d l o

C a m p o	Por defecto	D e s c r i p c i ó n
		O s o d e s c o n o c i d o s h a n s i d o c e r c a d o s a n d s ó l o e n l o s n o

C a m p o	Por defecto	D e s c r i p c i ó n
		o u n f e n c e d . E s t e e s e l v a l o r p o r d e f e c t o s i s e h a c o n f i

C a m p o	Por defecto	D e s c r i p c i o n
		o n p r o v i d e s = u n f e n c i n g s t o n i t h m e t a p a r a u n d i s p o s i

C a m p o	Por defecto	D e s c r i p c i ó n
		o
m i g r a t i o n - t h r e s h o l d	INFINITY	C u á n t o s f a l l o s p u e d e n o c u r r i r p a r a e s t e r e c u r s o

C a m p o	Por defecto	D e s c r i p c i ó n
		e s d e q u e e s t e n o d o s e a m a r c a d o c o m o i n e l e g i b l e p a r a a

C a m p o	Por defecto	D e s c r i p c i ó n
		e c u r s o · U n v a l o r d e O i n d i c a q u e e s t a c a r a c t e r í s t i c a

C a m p o	Por defecto	D e s c r i p c i ó n
		I t a d a (e l n o d o n u n c a s e r á m a r c a d o c o m o n o e l e g i b l e) ;

C a m p o	Por defecto	D e s c r i p c i ó n
		l o , e l c l u s t e r t r a t a I N F I N I T Y (e l v a l o r p o r d e f e c t o)

C a m p o	Por defecto	D e s c r i p c i ó n
		m u y g r a n d e p e r o f i n i t o . E s t a o p c i ó n s ó l o t i e n e e f e c t

C a m p o	Por defecto	D e s c r i p c i ó n
		o n f a l l i d a t i e n e o n - f a i l = r e s t a r t (e l v a l o r p o r d e f e

C a m p o	Por defecto	D e s c r i p c i ó n
		n a l m e n t e p a r a l a s o p e r a c i o n e s d e i n i c i o f a l l i d a s s i

C a m p o	Por defecto	D e s c r i p c i ó n
		e l c l u s t e r s t a r t - f a i l u r e - i s - f a t a l e s f a l s e .
f a i l	0 (desactivado)	U t i l

U C a m p o m e o u t	Por defecto	D e s c r i p c i ó n
		n l a o p c i ó n m i g r a t i o n - t h r e s h o l d , i n d i

C a m p o	Por defecto	D e s c r i p c i o n
		e s p e r

C a m p o	Por defecto	D e s c r i p c i ó n
		e r m i t i r p o t e n c i a l m e n t e q u e e l r e c u r s o v u e l v a a l n o d

C a m p o	Por defecto	D e s c r i p c i ó n
		o · C o m o c o n c u a l q u i e r a c c i ó n b a s a d a e n e l t i e m p o , n o s

C a m p o	Por defecto	D e s c r i p c i ó n
		e s e c o m p r u e b e c o n m á s f r e c u e n c i a q u e e l v a l o r d e l p a r

C a m p o	Por defecto	D e s c r i p c i ó n
		t e r c l u s t e r - r e c h e c k - i n t e r v a l .
m u l t i p l e - a c t	stop_start	Q u é d e b e h a c e r e l

i C a m p o	Por defecto	D e s c r i p c i ó n
		u n a v e z e n c u e n t r a e l r e c u r s o a c t i v o e n m á s d e u n n o d o .

C a m p o	Por defecto	V D e s c r i p c i ó n
		t i d o s : * b l o c k - m a r c a r e l r e c u r s o c o m o n o g e s t i o n

C a m p o	Por defecto	D e s c r i p c i ó n
		y - d e t e n e r t o d a s l a s i n s t a n c i a s a c t i v a s y d e j a r l a s

C a m p o	Por defecto	D e s c r i p c i ó n
		a r t - d e t e n e r t o d a s l a s i n s t a n c i a s a c t i v a s e i n i c i a r

C a m p o	Por defecto	D e s c r i p c i ó n
		n a s o l a u b i c a c i ó n

10.3.1. Cambiar el valor por defecto de una opción de recurso

A partir de Red Hat Enterprise Linux 8.3, puede cambiar el valor por defecto de una opción de recurso para todos los recursos con el comando **pcs resource defaults update**. El siguiente comando restablece el valor por defecto de **resource-stickiness** a 100.

```
# pcs resource defaults update resource-stickiness=100
```

El comando original **pcs resource defaults name=value** que establecía los valores predeterminados para todos los recursos en las versiones anteriores de RHEL 8, sigue siendo compatible a menos que haya más de un conjunto de valores predeterminados configurados. Sin embargo, **pcs resource defaults update** es ahora la versión preferida del comando.

10.3.2. Cambio del valor por defecto de una opción de recurso para conjuntos de recursos (RHEL 8.3 y posteriores)

A partir de Red Hat Enterprise Linux 8.3, puede crear múltiples conjuntos de recursos por defecto con el comando **pcs resource defaults set create**, que le permite especificar una regla que contenga expresiones **resource**. Sólo las expresiones **resource**, incluyendo **and**, **or** y paréntesis, están permitidas en las reglas que usted especifica con este comando.

Con el comando **pcs resource defaults set create**, puede configurar un valor de recurso por defecto para todos los recursos de un tipo determinado. Si, por ejemplo, está ejecutando bases de datos que tardan mucho en detenerse, puede aumentar el valor por defecto de **resource-stickiness** para todos los recursos del tipo de base de datos para evitar que esos recursos se trasladen a otros nodos con más frecuencia de la deseada.

El siguiente comando establece el valor por defecto de **resource-stickiness** a 100 para todos los recursos de tipo **pqsq**.

- La opción **id**, que nombra el conjunto de recursos por defecto, no es obligatoria. Si no se establece esta opción, **pcs** generará un ID automáticamente. Si establece este valor, podrá proporcionar un nombre más descriptivo.
- En este ejemplo, **::pqsq** significa un recurso de cualquier clase, cualquier proveedor, del tipo **pqsq**.
 - Especificar **ocf:heartbeat:pqsq** indicaría la clase **ocf**, el proveedor **heartbeat**, el tipo **pqsq**,
 - Especificando **ocf:pacemaker:** se indicarían todos los recursos de la clase **ocf**, proveedor **pacemaker**, de cualquier tipo.

```
# pcs resource defaults set create id=pqsq-stickiness meta resource-stickiness=100 rule
resource ::pqsq
```

Para cambiar los valores por defecto de un conjunto existente, utilice el comando **pcs resource defaults set update**.

10.3.3. Visualización de los valores predeterminados de los recursos configurados actualmente

El comando **pcs resource defaults** muestra una lista de los valores por defecto actualmente configurados para las opciones de recursos, incluyendo cualquier regla que haya especificado.

El siguiente ejemplo muestra la salida de este comando después de haber restablecido el valor por defecto de **resource-stickiness** a 100.

```
# pcs resource defaults
Meta Attrs: rsc_defaults-meta_attributes
resource-stickiness=100
```

El siguiente ejemplo muestra la salida de este comando después de haber restablecido el valor por defecto de **resource-stickiness** a 100 para todos los recursos de tipo **pqsq** y de haber establecido la opción **id** a **id=pqsq-stickiness**.

```
# pcs resource defaults
Meta Attrs: pqsq-stickiness
resource-stickiness=100
Rule: boolean-op=and score=INFINITY
Expression: resource ::pqsq
```

10.3.4. Configuración de las opciones meta en la creación de recursos

Tanto si se ha restablecido el valor por defecto de una metaopción de recurso como si no, se puede establecer una opción de recurso para un determinado recurso con un valor distinto al predeterminado cuando se crea el recurso. A continuación se muestra el formato del comando **pcs resource create** que se utiliza al especificar un valor para una metaopción de recurso.

```
pcs resource create resource_id [standard:[provider:]]type [resource options] [meta meta_options...]
```

Por ejemplo, el siguiente comando crea un recurso con un valor **resource-stickiness** de 50.

```
# pcs resource create VirtualIP ocf:heartbeat:IPaddr2 ip=192.168.0.120 meta resource-
stickiness=50
```

También se puede establecer el valor de una metaopción de recurso para un recurso existente, un grupo o un recurso clonado con el siguiente comando.

```
pcs resource meta resource_id | group_id | clone_id meta_options
```

En el siguiente ejemplo, hay un recurso existente llamado **dummy_resource**. Este comando establece la opción meta **failure-timeout** a 20 segundos, para que el recurso pueda intentar reiniciarse en el mismo nodo en 20 segundos.

```
# pcs resource meta dummy_resource failure-timeout=20s
```

Después de ejecutar este comando, puede visualizar los valores del recurso para verificar que **failure-timeout=20s** está configurado.

```
# pcs resource config dummy_resource
Resource: dummy_resource (class=ocf provider=heartbeat type=Dummy)
Meta Attrs: failure-timeout=20s
...
```

10.4. CONFIGURACIÓN DE LOS GRUPOS DE RECURSOS

Uno de los elementos más comunes de un clúster es un conjunto de recursos que deben ubicarse juntos, iniciarse secuencialmente y detenerse en el orden inverso. Para simplificar esta configuración, Pacemaker admite el concepto de grupos de recursos.

10.4.1. Creación de un grupo de recursos

Se crea un grupo de recursos con el siguiente comando, especificando los recursos que se incluirán en el grupo. Si el grupo no existe, este comando crea el grupo. Si el grupo existe, este comando añade recursos adicionales al grupo. Los recursos se iniciarán en el orden que se especifique con este comando, y se detendrán en el orden inverso al de inicio.

```
pcs resource group add group_name resource_id [resource_id] ... [resource_id] [--before resource_id
| --after resource_id]
```

Puede utilizar las opciones **--before** y **--after** de este comando para especificar la posición de los recursos añadidos en relación con un recurso que ya existe en el grupo.

También puede añadir un nuevo recurso a un grupo existente cuando crea el recurso, utilizando el siguiente comando. El recurso que se crea se añade al grupo denominado *group_name*. Si el grupo *group_name* no existe, se creará.

```
pcs resource create resource_id [standard:[provider:]]type [resource_options] [op operation_action
operation_options] --group group_name
```

No hay límite en el número de recursos que puede contener un grupo. Las propiedades fundamentales de un grupo son las siguientes.

- Los recursos se colocan dentro de un grupo.

- Los recursos se inician en el orden en que se especifican. Si un recurso del grupo no puede ejecutarse en ningún sitio, ningún recurso especificado después de ese recurso podrá ejecutarse.
- Los recursos se detienen en el orden inverso al especificado.

El siguiente ejemplo crea un grupo de recursos denominado **shortcut** que contiene los recursos existentes **IPAddr** y **Email**.

```
# pcs resource group add shortcut IPAddr Email
```

En este ejemplo:

- Primero se pone en marcha el **IPAddr** y luego el **Email**.
- Primero se detiene el recurso **Email** y luego **IPAddr**.
- Si **IPAddr** no puede correr en ningún sitio, tampoco **Email**.
- Sin embargo, si **Email** no puede ejecutarse en ningún sitio, esto no afecta en absoluto a **IPAddr**.

10.4.2. Eliminar un grupo de recursos

Se elimina un recurso de un grupo con el siguiente comando. Si no quedan recursos en el grupo, este comando elimina el propio grupo.

```
pcs resource group remove group_name resource_id ...
```

10.4.3. Visualización de los grupos de recursos

El siguiente comando muestra todos los grupos de recursos configurados actualmente.

```
lista de grupos de recursos pcs
```

10.4.4. Opciones de grupo

Puede establecer las siguientes opciones para un grupo de recursos, y mantienen el mismo significado que cuando se establecen para un solo recurso: **priority**, **target-role**, **is-managed**. Para obtener información sobre las opciones meta de los recursos, consulte [Configuración de las opciones meta de los recursos](#).

10.4.5. Adherencia al grupo

La pegajosidad, la medida de cuánto quiere un recurso quedarse donde está, es aditiva en los grupos. Cada recurso activo del grupo contribuirá con su valor de pegajosidad al total del grupo. Así, si el valor por defecto de **resource-stickiness** es 100, y un grupo tiene siete miembros, cinco de los cuales son activos, entonces el grupo en su conjunto preferirá su ubicación actual con una puntuación de 500.

10.5. DETERMINACIÓN DEL COMPORTAMIENTO DE LOS RECURSOS

Puede determinar el comportamiento de un recurso en un cluster configurando restricciones para ese recurso. Puede configurar las siguientes categorías de restricciones:

- **location** limitaciones
- **order** limitaciones
- **colocation** limitaciones

Pacemaker admite el concepto de grupos de recursos como una forma de abreviar la configuración de un conjunto de restricciones que ubicarán un conjunto de recursos juntos y garantizarán que los recursos se inicien de forma secuencial y se detengan en orden inverso. Una vez que haya creado un grupo de recursos, puede configurar las restricciones del propio grupo del mismo modo que configura las restricciones de los recursos individuales. Para obtener información sobre los grupos de recursos, consulte [Configuración de grupos de recursos](#).

CAPÍTULO 11. DETERMINACIÓN DE LOS NODOS EN LOS QUE PUEDE EJECUTARSE UN RECURSO

Las restricciones de ubicación determinan en qué nodos puede ejecutarse un recurso. Puede configurar las restricciones de ubicación para determinar si un recurso preferirá o evitará un nodo específico.

Además de las restricciones de ubicación, el nodo en el que se ejecuta un recurso está influenciado por el valor **resource-stickiness** para ese recurso, que determina hasta qué punto un recurso prefiere permanecer en el nodo donde se está ejecutando actualmente. Para obtener información sobre la configuración del valor **resource-stickiness**, consulte [Configuración de un recurso para que prefiera su nodo actual](#).

11.1. CONFIGURAR LAS RESTRICCIONES DE UBICACIÓN

Puede configurar una restricción de ubicación básica para especificar si un recurso prefiere o evita un nodo, con un valor opcional **score** para indicar el grado relativo de preferencia de la restricción.

El siguiente comando crea una restricción de ubicación para que un recurso prefiera el nodo o nodos especificados. Tenga en cuenta que es posible crear restricciones en un recurso concreto para más de un nodo con un solo comando.

```
pcs restricción ubicación rsc prefiere node[=score] [node[=score]] ...
```

El siguiente comando crea una restricción de ubicación para que un recurso evite el nodo o nodos especificados.

```
pcs restricción ubicación rsc evita node[=score] [node[=score]] ...
```

Tabla 11.1, “Opciones de restricción de ubicación” resume el significado de las opciones básicas para configurar las restricciones de ubicación.

Tabla 11.1. Opciones de restricción de ubicación

Campo	Descripción
rsc	Un nombre de recurso
node	El nombre de un nodo

Campo	Descripción
score	<p>Valor entero positivo para indicar el grado de preferencia para que el recurso dado prefiera o evite el nodo dado. INFINITY es el valor por defecto de score para una restricción de localización de recursos.</p> <p>Un valor de INFINITY para score en un comando pcs constraint location rsc prefers indica que el recurso preferirá ese nodo si el nodo está disponible, pero no impide que el recurso se ejecute en otro nodo si el nodo especificado no está disponible.</p> <p>Un valor de INFINITY para score en un comando pcs constraint location rsc avoids indica que el recurso nunca se ejecutará en ese nodo, incluso si no hay otro nodo disponible. Esto es el equivalente a establecer un comando pcs constraint location add con un valor de -INFINITY.</p> <p>Una puntuación numérica (es decir, no INFINITY) significa que la restricción es opcional, y se respetará a menos que algún otro factor la supere. Por ejemplo, si el recurso ya está colocado en un nodo diferente, y su puntuación resource-stickiness es mayor que la de una restricción de ubicación prefers, entonces el recurso se dejará donde está.</p>

El siguiente comando crea una restricción de ubicación para especificar que el recurso **Webserver** prefiere el nodo **node1**.

```
pcs ubicación de la restricción El servidor web prefiere el nodo1
```

pcs admite expresiones regulares en las restricciones de ubicación en la línea de comandos. Estas restricciones se aplican a múltiples recursos basados en la expresión regular que coincide con el nombre del recurso. Esto le permite configurar múltiples restricciones de ubicación con una sola línea de comandos.

El siguiente comando crea una restricción de ubicación para especificar que los recursos **dummy0** a **dummy9** prefieren **node1**.

```
pcs ubicación de la restricción 'regexp mmy[0-9]' prefiere el nodo1
```

Dado que Pacemaker utiliza expresiones regulares extendidas de POSIX, como se documenta en http://pubs.opengroup.org/onlinepubs/9699919799/basedefs/V1_chap09.html#tag_09_04 puede especificar la misma restricción con el siguiente comando.

```
pcs constraint location 'regexp mmy[[:digit:]]' prefiere el nodo1
```

11.2. LIMITACIÓN DEL DESCUBRIMIENTO DE RECURSOS A UN SUBCONJUNTO DE NODOS

Antes de que Pacemaker inicie un recurso en cualquier lugar, primero ejecuta una operación de monitorización única (a menudo denominada "sondeo") en cada nodo, para saber si el recurso ya se está ejecutando. Este proceso de descubrimiento de recursos puede dar lugar a errores en los nodos que no pueden ejecutar el monitor.

Al configurar una restricción de ubicación en un nodo, puede utilizar la opción **resource-discovery** del comando **pcs constraint location** para indicar una preferencia sobre si Pacemaker debe realizar el descubrimiento de recursos en este nodo para el recurso especificado. Limitar el descubrimiento de recursos a un subconjunto de nodos en los que el recurso puede ejecutarse físicamente puede aumentar significativamente el rendimiento cuando hay un gran conjunto de nodos. Cuando **pacemaker_remote** está en uso para ampliar el número de nodos en el rango de cientos de nodos, esta opción debe ser considerada.

El siguiente comando muestra el formato para especificar la opción **resource-discovery** del comando **pcs constraint location**. En este comando, un valor positivo para `score` corresponde a una restricción de ubicación básica que configura un recurso para preferir un nodo, mientras que un valor negativo para `score` corresponde a una restricción de ubicación básica que configura un recurso para evitar un nodo. Al igual que con las restricciones de ubicación básicas, también puedes utilizar expresiones regulares para los recursos con estas restricciones.

```
pcs constraint location add id rsc node score [resource-discovery=option]
```

Tabla 11.2, "Parámetros de restricción de descubrimiento de recursos" resume los significados de los parámetros básicos para configurar las restricciones para el descubrimiento de recursos.

Tabla 11.2. Parámetros de restricción de descubrimiento de recursos

Campo	Descripción
id	Un nombre elegido por el usuario para la propia restricción.
rsc	Un nombre de recurso
node	El nombre de un nodo

<p>score</p>	<p>Valor entero para indicar el grado de preferencia para que el recurso dado prefiera o evite el nodo dado. Un valor positivo para la puntuación corresponde a una restricción de ubicación básica que configura un recurso para preferir un nodo, mientras que un valor negativo para la puntuación corresponde a una restricción de ubicación básica que configura un recurso para evitar un nodo.</p> <p>Un valor de INFINITY para score indica que el recurso preferirá ese nodo si el nodo está disponible, pero no impide que el recurso se ejecute en otro nodo si el nodo especificado no está disponible. Un valor de -INFINITY para score indica que el recurso nunca se ejecutará en ese nodo, incluso si no hay otro nodo disponible.</p> <p>Una puntuación numérica (es decir, no INFINITY o -INFINITY) significa que la restricción es opcional, y se respetará a menos que algún otro factor la supere. Por ejemplo, si el recurso ya está colocado en un nodo diferente y su puntuación en resource-stickiness es mayor que la de una restricción de ubicación en prefers, el recurso se dejará donde está.</p>
<p>resource-discovery opciones</p>	<p>* always - Realiza siempre el descubrimiento de recursos para el recurso especificado en este nodo. Este es el valor predeterminado de resource-discovery para una restricción de ubicación de recursos.</p> <p>* never - No realizar nunca el descubrimiento de recursos para el recurso especificado en este nodo.</p> <p>* exclusive - Realiza el descubrimiento de recursos para el recurso especificado sólo en este nodo (y otros nodos marcados de forma similar como exclusive). Las restricciones de ubicación múltiples que utilizan el descubrimiento de exclusive para el mismo recurso a través de diferentes nodos crean un subconjunto de nodos en los que resource-discovery es exclusivo. Si un recurso está marcado para el descubrimiento de exclusive en uno o más nodos, ese recurso sólo se puede colocar dentro de ese subconjunto de nodos.</p>



AVISO

Configurar **resource-discovery** como **never** o **exclusive** elimina la capacidad de Pacemaker de detectar y detener las instancias no deseadas de un servicio que se ejecuta donde no debe estar. Depende del administrador del sistema asegurarse de que el servicio nunca pueda estar activo en los nodos sin descubrimiento de recursos (por ejemplo, dejando desinstalado el software correspondiente).

11.3. CONFIGURAR UNA ESTRATEGIA DE RESTRICCIÓN DE UBICACIÓN

Cuando se utilizan restricciones de ubicación, se puede configurar una estrategia general para especificar en qué nodos puede ejecutarse un recurso:

- Agrupaciones Opt-In
- Agrupaciones Opt-Out

La elección de configurar su clúster como un clúster opt-in o opt-out depende tanto de sus preferencias personales como de la composición de su clúster. Si la mayoría de sus recursos pueden ejecutarse en la mayoría de los nodos, es probable que una disposición opt-out resulte en una configuración más sencilla. Por otro lado, si la mayoría de los recursos sólo pueden ejecutarse en un pequeño subconjunto de nodos, una configuración opt-in podría ser más sencilla.

11.3.1. Configuración de un clúster "Opt-In"

Para crear un clúster opcional, establezca la propiedad de clúster **symmetric-cluster** en **false** para evitar que los recursos se ejecuten en cualquier lugar por defecto.

```
# pcs property set symmetric-cluster=false
```

Habilitar nodos para recursos individuales. Los siguientes comandos configuran las restricciones de ubicación para que el recurso **Webserver** prefiera el nodo **example-1**, el recurso **Database** prefiera el nodo **example-2**, y ambos recursos puedan fallar en el nodo **example-3** si su nodo preferido falla. Cuando se configuran las restricciones de ubicación para un cluster opt-in, establecer una puntuación de cero permite que un recurso se ejecute en un nodo sin indicar ninguna preferencia para preferir o evitar el nodo.

```
# pcs constraint location Webserver prefers example-1=200
# pcs constraint location Webserver prefers example-3=0
# pcs constraint location Database prefers example-2=200
# pcs constraint location Database prefers example-3=0
```

11.3.2. Configuración de un clúster "Opt-Out"

Para crear un clúster de exclusión, establezca la propiedad de clúster **symmetric-cluster** en **true** para permitir que los recursos se ejecuten en todas partes por defecto. Esta es la configuración por defecto si **symmetric-cluster** no se establece explícitamente.

```
# pcs property set symmetric-cluster=true
```

Los siguientes comandos darán lugar a una configuración equivalente al ejemplo de [Sección 11.3.1, "Configuración de un clúster "Opt-In"'](#). Ambos recursos pueden fallar al nodo **example-3** si su nodo preferido falla, ya que cada nodo tiene una puntuación implícita de 0.

```
# pcs constraint location Webserver prefers example-1=200
# pcs constraint location Webserver avoids example-2=INFINITY
# pcs constraint location Database avoids example-1=INFINITY
# pcs constraint location Database prefers example-2=200
```

Tenga en cuenta que no es necesario especificar una puntuación de INFINITO en estos comandos, ya que ese es el valor por defecto para la puntuación.

11.4. CONFIGURAR UN RECURSO PARA QUE PREFIERA SU NODO ACTUAL

Los recursos tienen un valor **resource-stickiness** que puedes establecer como un meta atributo cuando creas el recurso, como se describe en [Configuración de las opciones meta de los recursos](#) . El valor **resource-stickiness** determina cuánto quiere permanecer un recurso en el nodo donde se está ejecutando actualmente. Pacemaker tiene en cuenta el valor de **resource-stickiness** junto con otros ajustes (por ejemplo, los valores de puntuación de las restricciones de ubicación) para determinar si se debe mover un recurso a otro nodo o dejarlo en su lugar.

Por defecto, un recurso se crea con un valor **resource-stickiness** de 0. El comportamiento predeterminado de Pacemaker cuando **resource-stickiness** se establece en 0 y no hay restricciones de ubicación es mover los recursos para que se distribuyan uniformemente entre los nodos del clúster. Esto puede dar lugar a que los recursos sanos se muevan con más frecuencia de la deseada. Para evitar este comportamiento, puede establecer el valor predeterminado de **resource-stickiness** en 1. Este valor predeterminado se aplicará a todos los recursos del clúster. Este pequeño valor puede ser fácilmente anulado por otras restricciones que usted cree, pero es suficiente para evitar que Pacemaker mueva innecesariamente los recursos sanos por el cluster.

El siguiente comando establece el valor por defecto de **resource-stickiness** en 1.

```
# pcs resource defaults resource-stickiness=1
```

Si se establece el valor **resource-stickiness**, no se moverá ningún recurso a un nodo recién añadido. Si se desea equilibrar los recursos en ese momento, se puede volver a establecer temporalmente el valor **resource-stickiness** a 0.

Tenga en cuenta que si la puntuación de una restricción de ubicación es mayor que el valor de **resource-stickiness**, el clúster puede seguir moviendo un recurso sano al nodo donde apunta la restricción de ubicación.

Para obtener más información sobre cómo Pacemaker determina dónde colocar un recurso, consulte [Configuración de una estrategia de colocación de nodos](#) .

CAPÍTULO 12. DETERMINACIÓN DEL ORDEN DE EJECUCIÓN DE LOS RECURSOS DEL CLÚSTER

Para determinar el orden de ejecución de los recursos, se configura una restricción de orden.

A continuación se muestra el formato del comando para configurar una restricción de ordenación.

```
pcs constraint order [action] resource_id then [action] resource_id [options]
```

Tabla 12.1, "Propiedades de una restricción de orden", resume las propiedades y opciones para configurar las restricciones de ordenación.

Tabla 12.1. Propiedades de una restricción de orden

Campo	Descripción
resource_id	El nombre de un recurso sobre el que se realiza una acción.
acción	<p>La acción a realizar sobre un recurso. Los posibles valores de la propiedad <i>action</i> son los siguientes:</p> <ul style="list-style-type: none"> * start - Inicie el recurso. * stop - Detener el recurso. * promote - Promover el recurso de un recurso esclavo a un recurso maestro. * demote - Descolocar el recurso de un recurso maestro a un recurso esclavo. <p>Si no se especifica ninguna acción, la acción por defecto es start.</p>

Campo	Descripción
kind opción	<p>Cómo aplicar la restricción. Los posibles valores de la opción kind son los siguientes:</p> <ul style="list-style-type: none"> * Optional - Sólo se aplica si ambos recursos están ejecutando la acción especificada. Para obtener información sobre el ordenamiento opcional, consulte Configurar el ordenamiento de los avisos. * Mandatory - Siempre (valor por defecto). Si el primer recurso especificado se detiene o no puede iniciarse, el segundo recurso especificado debe detenerse. Para obtener información sobre el ordenamiento obligatorio, consulte Configuración del ordenamiento obligatorio. * Serialize - Asegúrese de que no se produzcan dos acciones de parada/arranque simultáneamente para los recursos que especifique. El primer y el segundo recurso que especifique pueden iniciarse en cualquier orden, pero uno debe terminar de iniciarse antes de que el otro pueda iniciarse. Un caso de uso típico es cuando el inicio de recursos pone una alta carga en el host.
symmetrical opción	<p>Si es verdadera, se aplica la inversa de la restricción para la acción opuesta (por ejemplo, si B comienza después de que A comience, entonces B se detiene antes. Las restricciones de ordenación para las que kind es Serialize no pueden ser simétricas. El valor por defecto es true para las clases Mandatory y Ordering, false para Serialize.</p>

Utilice el siguiente comando para eliminar los recursos de cualquier restricción de ordenación.

```
pcs constraint order remove resource1 [resourceN]...
```

12.1. CONFIGURACIÓN DE LA ORDENACIÓN OBLIGATORIA

Una restricción de orden obligatoria indica que la segunda acción no debe iniciarse para el segundo recurso a menos que y hasta que la primera acción se complete con éxito para el primer recurso. Las acciones que pueden ordenarse son **stop**, **start**, y adicionalmente para los clones promocionables, **demote** y **promote**. Por ejemplo, \ "A entonces B" (que equivale a "iniciar A y luego iniciar B") significa que B no se iniciará a menos que A se inicie con éxito. Una restricción de ordenación es obligatoria si la opción **kind** de la restricción se establece en **Mandatory** o se deja por defecto.

Si la opción **symmetrical** se establece en **true** o se deja por defecto, las acciones opuestas se ordenarán de forma inversa. Las acciones **start** y **stop** son opuestas, y **demote** y **promote** son opuestas. Por ejemplo, una ordenación simétrica \ "promocionar A y luego iniciar B" implica \ "detener B y luego degradar A", lo que significa que A no puede ser degradado hasta que y a menos que B se detenga con éxito. Un ordenamiento simétrico significa que los cambios en el estado de A pueden causar acciones que se programen para B. Por ejemplo, dado "A entonces B", si A se reinicia debido a una falla, B se detendrá primero, luego A se detendrá, luego A se iniciará, luego B se iniciará.

Tenga en cuenta que el clúster reacciona a cada cambio de estado. Si el primer recurso se reinicia y vuelve a estar en estado de inicio antes de que el segundo recurso inicie una operación de parada, no será necesario reiniciar el segundo recurso.

12.2. CONFIGURAR LA ORDENACIÓN DE LOS AVISOS

Cuando se especifica la opción **kind=Optional** para una restricción de ordenación, la restricción se considera opcional y sólo se aplica si ambos recursos están ejecutando las acciones especificadas. Cualquier cambio de estado del primer recurso especificado no tendrá efecto en el segundo recurso especificado.

El siguiente comando configura una restricción de ordenación de asesoramiento para los recursos denominados **VirtuallP** y **dummy_resource**.

```
# pcs constraint order VirtuallP then dummy_resource kind=Optional
```

12.3. CONFIGURACIÓN DE CONJUNTOS DE RECURSOS ORDENADOS

Una situación común es que un administrador cree una cadena de recursos ordenados, donde, por ejemplo, el recurso A se inicia antes que el recurso B, que se inicia antes que el recurso C. Si su configuración requiere que cree un conjunto de recursos que se coloquen y se inicien en orden, puede configurar un grupo de recursos que contenga esos recursos, como se describe en [Configuración de grupos de recursos](#).

Sin embargo, hay algunas situaciones en las que no es adecuado configurar los recursos que deben iniciarse en un orden determinado como un grupo de recursos:

- Es posible que tenga que configurar los recursos para que se inicien en orden y que los recursos no estén necesariamente colocados.
- Puede tener un recurso C que debe iniciarse después de que se haya iniciado el recurso A o el B, pero no hay ninguna relación entre A y B.
- Puede tener recursos C y D que deben iniciarse después de que se hayan iniciado los recursos A y B, pero no hay ninguna relación entre A y B o entre C y D.

En estas situaciones, puede crear una restricción de ordenación en un conjunto o conjuntos de recursos con el comando **pcs constraint order set**.

Puede establecer las siguientes opciones para un conjunto de recursos con el comando **pcs constraint order set**.

- **sequential**, que puede establecerse como **true** o **false** para indicar si el conjunto de recursos debe estar ordenado de forma relativa. El valor por defecto es **true**.
Si se establece **sequential** en **false**, se puede ordenar un conjunto en relación con otros conjuntos en la restricción de ordenación, sin que sus miembros se ordenen entre sí. Por lo tanto, esta opción sólo tiene sentido si hay varios conjuntos en la restricción; en caso contrario, la restricción no tiene ningún efecto.
- **require-all**, que puede establecerse en **true** o **false** para indicar si todos los recursos del conjunto deben estar activos antes de continuar. Establecer **require-all** a **false** significa que sólo un recurso del conjunto debe iniciarse antes de continuar con el siguiente conjunto. Establecer **require-all** a **false** no tiene ningún efecto a menos que se utilice junto con conjuntos desordenados, que son conjuntos para los que **sequential** se establece en **false**. El valor por defecto es **true**.

- **action**, que puede establecerse en **start**, **promote**, **demote** o **stop**, como se describe en [Propiedades de una restricción de orden](#) .
- **role**, que puede ajustarse a **Stopped**, **Started**, **Master** o **Slave**.

Puede establecer las siguientes opciones de restricción para un conjunto de recursos siguiendo el parámetro **setoptions** del comando **pcs constraint order set**.

- **id**, para dar un nombre a la restricción que está definiendo.
- **kind**, que indica cómo hacer cumplir la restricción, como se describe en [Propiedades de una restricción de orden](#).
- **symmetrical**, para establecer si la inversa de la restricción se aplica para la acción opuesta, como se describe en [Propiedades de una restricción de orden](#) .

```
pcs constraint order set resource1 resource2 [resourceN]... [options] [set resourceX resourceY ...
[options]] [setoptions [constraint_options]]
```

Si tiene tres recursos llamados **D1**, **D2**, y **D3**, el siguiente comando los configura como un conjunto de recursos ordenados.

```
# pcs constraint order set D1 D2 D3
```

Si tiene seis recursos denominados **A**, **B**, **C**, **D**, **E**, y **F**, este ejemplo configura una restricción de ordenación para el conjunto de recursos que comenzará de la siguiente manera:

- **A** y **B** se inician de forma independiente
- **C** se inicia una vez que se ha iniciado **A** o **B**
- **D** se pone en marcha una vez que se ha iniciado **C**
- **E** y **F** se inician de forma independiente una vez que se ha iniciado **D**

La detención de los recursos no se ve influida por esta restricción, ya que se establece **symmetrical=false**.

```
# pcs constraint order set A B sequential=false require-all=false set C D set E F
sequential=false setoptions symmetrical=false
```

12.4. CONFIGURACIÓN DEL ORDEN DE INICIO PARA LAS DEPENDENCIAS DE RECURSOS NO GESTIONADAS POR PACEMAKER

Es posible que un clúster incluya recursos con dependencias que no son gestionadas por el clúster. En este caso, debe asegurarse de que esas dependencias se inicien antes de iniciar Pacemaker y se detengan después de detenerlo.

Puede configurar su orden de inicio para tener en cuenta esta situación mediante el objetivo **systemd resource-agents-deps** . Puede crear una unidad de arranque **systemd** para este objetivo y Pacemaker se ordenará adecuadamente en relación con este objetivo.

Por ejemplo, si un clúster incluye un recurso que depende del servicio externo **foo** que no está gestionado por el clúster, realice el siguiente procedimiento.

1. Cree la unidad de entrega `/etc/systemd/system/resource-agents-deps.target.d/foo.conf` que contiene lo siguiente:

```
[Unit]
Requires=foo.service
After=foo.service
```

2. Ejecute el comando `systemctl daemon-reload`.

Una dependencia del clúster especificada de esta manera puede ser algo distinto a un servicio. Por ejemplo, puede tener una dependencia para montar un sistema de archivos en `/srv`, en cuyo caso realizaría el siguiente procedimiento:

1. Asegúrese de que `/srv` aparece en el archivo `/etc/fstab`. Esto se convertirá automáticamente en el archivo `systemd srv.mount` en el arranque cuando se recargue la configuración del administrador del sistema. Para más información, consulte las páginas de manual `systemd.mount`(5) y `systemd-fstab-generator`(8).
2. Para asegurarse de que Pacemaker se inicie después de montar el disco, cree la unidad drop-in `/etc/systemd/system/resource-agents-deps.target.d/srv.conf` que contiene lo siguiente.

```
[Unit]
Requires=srv.mount
After=srv.mount
```

3. Ejecute el comando `systemctl daemon-reload`.

CAPÍTULO 13. COLOCACIÓN DE LOS RECURSOS DEL CLÚSTER

Para especificar que la ubicación de un recurso depende de la ubicación de otro recurso, se configura una restricción de colocación.

La creación de una restricción de colocación entre dos recursos tiene un efecto secundario importante: afecta al orden en que se asignan los recursos a un nodo. Esto se debe a que no se puede colocar el recurso A en relación con el recurso B a menos que se sepa dónde está el recurso B. Por lo tanto, cuando se crean restricciones de colocación, es importante considerar si se debe colocar el recurso A con el recurso B o el recurso B con el recurso A.

Otra cosa que hay que tener en cuenta al crear las restricciones de colocación es que, suponiendo que el recurso A esté colocado con el recurso B, el clúster también tendrá en cuenta las preferencias del recurso A a la hora de decidir qué nodo elegir para el recurso B.

El siguiente comando crea una restricción de colocación.

```
pcs constraint colocation add [master|slave] source_resource with [master|slave] target_resource
[score] [options]
```

Tabla 13.1, “Propiedades de una restricción de colocación”, resume las propiedades y opciones para configurar las restricciones de colocación.

Tabla 13.1. Propiedades de una restricción de colocación

Campo	Descripción
recurso_fuente	La fuente de colocación. Si la restricción no puede ser satisfecha, el clúster puede decidir no permitir que el recurso se ejecute en absoluto.
recurso_objetivo	El objetivo de colocación. El clúster decidirá dónde colocar este recurso primero y luego decidirá dónde colocar el recurso de origen.
puntuación	Los valores positivos indican que el recurso debe ejecutarse en el mismo nodo. Los valores negativos indican que los recursos no deben ejecutarse en el mismo nodo. Un valor de INFINITY , el valor por defecto, indica que el <i>source_resource</i> debe ejecutarse en el mismo nodo que el <i>target_resource</i> . Un valor de -INFINITY indica que el <i>source_resource</i> no debe ejecutarse en el mismo nodo que el <i>target_resource</i> .

13.1. ESPECIFICACIÓN DE LA UBICACIÓN OBLIGATORIA DE LOS RECURSOS

La colocación obligatoria se produce cada vez que la puntuación de la restricción es **INFINITY** o **-INFINITY**. En estos casos, si la restricción no puede satisfacerse, no se permite la ejecución de *source_resource*. En el caso de **score=INFINITY**, esto incluye los casos en los que el *target_resource* no está activo.

Si necesita que **myresource1** se ejecute siempre en la misma máquina que **myresource2**, deberá añadir la siguiente restricción:

```
# pcs constraint colocation add myresource1 with myresource2 score=INFINITY
```

Como se ha utilizado **INFINITY**, si **myresource2** no puede ejecutarse en ninguno de los nodos del clúster (por la razón que sea), entonces **myresource1** no podrá ejecutarse.

Alternativamente, puede querer configurar lo contrario, un cluster en el que **myresource1** no pueda ejecutarse en la misma máquina que **myresource2**. En este caso, utilice **score=-INFINITY**

```
# pcs constraint colocation add myresource1 with myresource2 score=-INFINITY
```

De nuevo, al especificar **-INFINITY**, la restricción es vinculante. Así que si el único lugar que queda por ejecutar es donde ya está **myresource2**, entonces **myresource1** no puede ejecutarse en ningún sitio.

13.2. ESPECIFICACIÓN DE LA COLOCACIÓN DE LOS RECURSOS CON FINES DE ASESORAMIENTO

Si la colocación obligatoria se trata de \N "debe" y \N "no debe", entonces la colocación consultiva es la alternativa \N "preferiría si". Para las restricciones con puntuaciones superiores a **-INFINITY** e inferiores a **INFINITY**, el clúster intentará acomodar sus deseos pero puede ignorarlos si la alternativa es detener algunos de los recursos del clúster.

13.3. COLOCACIÓN DE CONJUNTOS DE RECURSOS

Si su configuración requiere que cree un conjunto de recursos que se coloquen y se inicien en orden, puede configurar un grupo de recursos que contenga esos recursos, como se describe en [Configuración de grupos de recursos](#). Sin embargo, hay algunas situaciones en las que no es apropiado configurar los recursos que deben colocarse como un grupo de recursos:

- Es posible que tenga que colocar un conjunto de recursos, pero los recursos no tienen que empezar necesariamente en orden.
- Puede tener un recurso C que debe estar colocado con el recurso A o B, pero no hay relación entre A y B.
- Es posible que haya recursos C y D que deban colocarse con los recursos A y B, pero no hay relación entre A y B ni entre C y D.

En estas situaciones, puede crear una restricción de colocación en un conjunto o conjuntos de recursos con el comando **pcs constraint colocation set**.

Puede establecer las siguientes opciones para un conjunto de recursos con el comando **pcs constraint colocation set**.

- **sequential**, que puede establecerse como **true** o **false** para indicar si los miembros del conjunto deben estar colocados entre sí.
Si se establece **sequential** en **false**, los miembros de este conjunto se colocan con otro conjunto que aparece más adelante en la restricción, independientemente de los miembros de este conjunto que estén activos. Por lo tanto, esta opción sólo tiene sentido si otro conjunto aparece después de éste en la restricción; de lo contrario, la restricción no tiene efecto.
- **role**, que puede ajustarse a **Stopped**, **Started**, **Master** o **Slave**.

Puede establecer la siguiente opción de restricción para un conjunto de recursos siguiendo el parámetro **setoptions** del comando **pcs constraint colocation set**.

- **id**, para dar un nombre a la restricción que está definiendo.
- **score**, para indicar el grado de preferencia de esta restricción. Para obtener información sobre esta opción, consulte [Opciones de restricción de ubicación](#).

Cuando se enumeran los miembros de un conjunto, cada miembro se coloca con el anterior. Por ejemplo, "conjunto A B" significa que "B está colocado con A". Sin embargo, cuando se enumeran varios conjuntos, cada conjunto se coloca con el que le sigue. Por ejemplo, "conjunto C D secuencial=falso conjunto A B" significa que "el conjunto C D (donde C y D no tienen relación entre sí) está colocado con el conjunto A B (donde B está colocado con A)".

El siguiente comando crea una restricción de colocación en un conjunto o conjuntos de recursos.

```
pcs constraint colocation set resource1 resource2 [resourceN]... [options] [set resourceX resourceY ... [options]] [setoptions [constraint_options]]
```

13.4. ELIMINACIÓN DE LAS RESTRICCIONES DE COLOCACIÓN

Utilice el siguiente comando para eliminar las restricciones de colocación con *source_resource*.

```
pcs constraint colocation remove source_resource target_resource
```


CAPÍTULO 14. VISUALIZACIÓN DE LAS LIMITACIONES DE RECURSOS

Hay varios comandos que se pueden utilizar para mostrar las restricciones que se han configurado.

14.1. VISUALIZACIÓN DE TODAS LAS RESTRICCIONES CONFIGURADAS

El siguiente comando muestra todas las restricciones de ubicación, orden y colocación actuales. Si se especifica la opción **--full**, muestra los ID de las restricciones internas.

```
pcs constraint [list|show] [--full]
```

A partir de RHEL 8.2, el listado de restricciones de recursos ya no muestra por defecto las restricciones caducadas. Para incluir las restricciones caducadas, utilice la opción **--all** del comando **pcs constraint**. Esto listará las restricciones expiradas, anotando las restricciones y sus reglas asociadas como **(expired)** en la pantalla.

14.2. VISUALIZACIÓN DE LAS RESTRICCIONES DE UBICACIÓN

El siguiente comando muestra todas las restricciones de ubicación actuales.

- Si se especifica **resources**, las restricciones de ubicación se muestran por recurso. Este es el comportamiento por defecto.
- Si se especifica **nodes**, las restricciones de ubicación se muestran por nodo.
- Si se especifican recursos o nodos concretos, sólo se mostrará información sobre esos recursos o nodos.

```
pcs constraint location [show [resources [resource...]] | [nodos [node...]]] [--full]
```

14.3. VISUALIZACIÓN DE LAS RESTRICCIONES DE ORDENACIÓN

La siguiente orden enumera todas las restricciones de ordenación actuales.

```
pcs orden de restricción [mostrar]
```

14.4. VISUALIZACIÓN DE LAS RESTRICCIONES DE COLOCACIÓN

El siguiente comando muestra todas las restricciones de colocación actuales.

```
pcs constraint colocation [show]
```

14.5. VISUALIZACIÓN DE LAS RESTRICCIONES ESPECÍFICAS DE LOS RECURSOS

El siguiente comando enumera las restricciones que hacen referencia a recursos específicos.

```
pcs constraint ref resource...
```

14.6. VISUALIZACIÓN DE LAS DEPENDENCIAS DE RECURSOS (RED HAT ENTERPRISE LINUX 8.2 Y POSTERIORES)

El siguiente comando muestra las relaciones entre los recursos del cluster en una estructura de árbol.

```
pcs resource relations resource [--full]
```

Si se utiliza la opción **--full**, el comando muestra información adicional, incluyendo los ID de las restricciones y los tipos de recursos.

En el siguiente ejemplo, hay 3 recursos configurados: C, D y E.

```
# pcs constraint order start C then start D
Adding C D (kind: Mandatory) (Options: first-action=start then-action=start)
# pcs constraint order start D then start E
Adding D E (kind: Mandatory) (Options: first-action=start then-action=start)

# pcs resource relations C
C
`- order
  | start C then start D
  `- D
    `- order
      | start D then start E
      `- E

# pcs resource relations D
D
|- order
| | start C then start D
| `- C
`- order
  | start D then start E
  `- E

# pcs resource relations E
E
`- order
  | start D then start E
  `- D
    `- order
      | start C then start D
      `- C
```

En el siguiente ejemplo, hay 2 recursos configurados: A y B. Los recursos A y B forman parte del grupo de recursos G.

```
# pcs resource relations A
A
`- outer resource
  `- G
    `- inner resource(s)
      | members: A B
```

```
`- B
# pcs resource relations B
B
`- outer resource
  `- G
    `- inner resource(s)
      | members: A B
      `- A
# pcs resource relations G
G
`- inner resource(s)
  | members: A B
  |- A
  `- B
```

CAPÍTULO 15. DETERMINACIÓN DE LA UBICACIÓN DE LOS RECURSOS CON REGLAS

Para restricciones de ubicación más complicadas, puede utilizar las reglas de Pacemaker para determinar la ubicación de un recurso.

15.1. REGLAS DEL MARCAPASOS

Las reglas pueden ser utilizadas para hacer su configuración más dinámica. Un uso de las reglas podría ser asignar máquinas a diferentes grupos de procesamiento (utilizando un atributo de nodo) en función del tiempo y luego utilizar ese atributo al crear restricciones de ubicación.

Cada regla puede contener varias expresiones, expresiones de fecha e incluso otras reglas. Los resultados de las expresiones se combinan basándose en el campo **boolean-op** de la regla para determinar si la regla se evalúa finalmente como **true** o **false**. Lo que ocurre a continuación depende del contexto en el que se utiliza la regla.

Tabla 15.1. Propiedades de una regla

Campo	Descripción
role	Limita la regla para que se aplique sólo cuando el recurso está en ese rol. Valores permitidos: Started , Slave , y Master . NOTA: Una regla con role="Master" no puede determinar la ubicación inicial de una instancia clonada. Sólo afectará a cuál de las instancias activas será promovida.
score	La puntuación a aplicar si la regla se evalúa como true . Limitado a su uso en reglas que forman parte de restricciones de ubicación.
score-attribute	El atributo del nodo que se busca y se utiliza como puntuación si la regla se evalúa como true . Limitado a su uso en reglas que forman parte de restricciones de ubicación.
boolean-op	Cómo combinar el resultado de varios objetos de expresión. Valores permitidos: and y or . El valor por defecto es and .

15.1.1. Expresiones de atributos de los nodos

Las expresiones de atributos de nodo se utilizan para controlar un recurso en función de los atributos definidos por un nodo o nodos.

Tabla 15.2. Propiedades de una expresión

Campo	Descripción
attribute	El atributo del nodo a probar

Campo	Descripción
type	Determina cómo se deben comprobar los valores. Valores permitidos: string , integer , version . El valor por defecto es string .
operation	La comparación a realizar. Valores permitidos: <ul style="list-style-type: none"> * lt - Verdadero si el valor del atributo del nodo es menor que value * gt - Verdadero si el valor del atributo del nodo es mayor que value * lte - Verdadero si el valor del atributo del nodo es menor o igual a value * gte - Verdadero si el valor del atributo del nodo es mayor o igual a value * eq - Verdadero si el valor del atributo del nodo es igual a value * ne - Verdadero si el valor del atributo del nodo no es igual a value * defined - Verdadero si el nodo tiene el atributo nombrado * not_defined - Verdadero si el nodo no tiene el atributo nombrado
value	Valor suministrado por el usuario para la comparación (obligatorio a menos que operation sea defined o not_defined)

Además de los atributos añadidos por el administrador, el clúster define atributos de nodo especiales e incorporados para cada nodo que también pueden utilizarse, como se describe en [Tabla 15.3, "Atributos de los nodos incorporados"](#).

Tabla 15.3. Atributos de los nodos incorporados

Nombre	Descripción
#uname	Nombre del nodo
#id	ID de nodo

Nombre	Descripción
#kind	Tipo de nodo. Los valores posibles son cluster , remote y container . El valor de kind es remote para los nodos Pacemaker Remote creados con el recurso ocf:pacemaker:remote , y container para los nodos invitados Pacemaker Remote y los nodos bundle.
#is_dc	true si este nodo es un controlador designado (DC), false en caso contrario
#cluster_name	El valor de la propiedad cluster-name cluster, si se ha establecido
#site_name	El valor del atributo del nodo site-name , si está establecido, de lo contrario es idéntico a #cluster-name
#role	La función que el clon promocionable correspondiente tiene en este nodo. Válido sólo dentro de una regla para una restricción de ubicación para un clon promocionable.

15.1.2. Expresiones basadas en la fecha y la hora

Las expresiones de fecha se utilizan para controlar un recurso o una opción del clúster en función de la fecha/hora actual. Pueden contener una especificación de fecha opcional.

Tabla 15.4. Propiedades de una expresión de fecha

Campo	Descripción
start	Una fecha/hora conforme a la especificación ISO8601.
end	Una fecha/hora conforme a la especificación ISO8601.

Campo	Descripción
operation	<p>Compara la fecha/hora actual con la fecha de inicio o de finalización o con ambas, dependiendo del contexto. Valores permitidos:</p> <ul style="list-style-type: none"> * gt - Verdadero si la fecha/hora actual es posterior a start * lt - Verdadero si la fecha/hora actual es anterior a end * in_range - Verdadero si la fecha/hora actual es posterior a start y anterior a end * date-spec - realiza una comparación tipo cron con la fecha/hora actual

15.1.3. Especificaciones de fecha

Las especificaciones de fecha se utilizan para crear expresiones tipo cron relacionadas con el tiempo. Cada campo puede contener un solo número o un solo rango. En lugar de ponerlo a cero por defecto, cualquier campo no suministrado se ignora.

Por ejemplo, **monthdays="1"** coincide con el primer día de cada mes y **hours="09-17"** con las horas comprendidas entre las 9 y las 17 horas (inclusive). Sin embargo, no se puede especificar **weekdays="1,2"** o **weekdays="1-2,5-6"** ya que contienen múltiples rangos.

Tabla 15.5. Propiedades de una especificación de fecha

Campo	Descripción
id	Un nombre único para la fecha
hours	Valores permitidos: 0-23
monthdays	Valores permitidos: 0-31 (según el mes y el año)
weekdays	Valores permitidos: 1-7 (1=lunes, 7=domingos)
yeardays	Valores permitidos: 1-366 (dependiendo del año)
months	Valores permitidos: 1-12
weeks	Valores permitidos: 1-53 (dependiendo de weekyear)
years	Año según el calendario gregoriano

Campo	Descripción
weekyears	Puede diferir de los años gregorianos; por ejemplo, 2005-001 Ordinal es también 2005-01-01 Gregorian es también 2004-W53-6 Weekly
moon	Valores permitidos: 0-7 (0 es luna nueva, 4 es luna llena).

15.2. CONFIGURACIÓN DE UNA RESTRICCIÓN DE UBICACIÓN DE MARCAPASOS MEDIANTE REGLAS

Utilice el siguiente comando para configurar una restricción de Pacemaker que utilice reglas. Si se omite **score**, el valor por defecto es INFINITO. Si se omite **resource-discovery**, el valor predeterminado es **always**.

Para obtener información sobre la opción **resource-discovery**, consulte [Limitar el descubrimiento de recursos a un subconjunto de nodos](#).

Al igual que con las restricciones de ubicación básicas, también puede utilizar expresiones regulares para los recursos con estas restricciones.

Cuando se utilizan reglas para configurar las restricciones de ubicación, el valor de **score** puede ser positivo o negativo, con un valor positivo que indica "prefiere" y un valor negativo que indica "evita".

```
pcs constraint location rsc rule [resource-discovery=option] [role=master|slave] [score=score | score-attribute=attribute] expression
```

La opción *expression* puede ser una de las siguientes, donde *duration_options* y *date_spec_options* son: horas, días del mes, días de la semana, días del año, meses, semanas, años, años de la semana, luna, como se describe en [Propiedades de una especificación de fecha](#) .

- **defined|not_defined *attribute***
- ***attribute* lt|gt|lte|gte|eq|ne [*string|integer|version*] *value***
- **date gt|lt *date***
- **date in_range *date* to *date***
- **date in_range *date* to duration *duration_options* ...**
- **date-spec *date_spec_options***
- ***expression* and|or *expression***
- **(*expression*)**

Tenga en cuenta que las duraciones son una forma alternativa de especificar un final para las operaciones de **in_range** mediante cálculos. Por ejemplo, puede especificar una duración de 19 meses.

La siguiente restricción de ubicación configura una expresión que es verdadera si ahora es cualquier momento del año 2018.

```
# pcs constraint location Webservice rule score=INFINITY date-spec years=2018
```

El siguiente comando configura una expresión que es verdadera de 9 am a 5 pm, de lunes a viernes. Tenga en cuenta que el valor de horas de 16 coincide hasta las 16:59:59, ya que el valor numérico (hora) sigue coincidiendo.

```
# pcs constraint location Webservice rule score=INFINITY date-spec hours="9-16"  
weekdays="1-5"
```

El siguiente comando configura una expresión que es verdadera cuando hay luna llena el viernes trece.

```
# pcs constraint location Webservice rule date-spec weekdays=5 monthdays=13 moon=4
```

Para eliminar una regla, utilice el siguiente comando. Si la regla que está eliminando es la última regla de su restricción, la restricción será eliminada.

```
regla de restricción pcs eliminar rule_id
```

CAPÍTULO 16. GESTIÓN DE LOS RECURSOS DEL CLÚSTER

Esta sección describe varios comandos que puede utilizar para gestionar los recursos del clúster.

16.1. VISUALIZACIÓN DE LOS RECURSOS CONFIGURADOS

Para mostrar una lista de todos los recursos configurados, utilice el siguiente comando.

```
pcs estado de los recursos pcs
```

Por ejemplo, si su sistema está configurado con un recurso llamado **VirtualIP** y un recurso llamado **WebSite**, el comando **pcs resource show** produce la siguiente salida.

```
# pcs resource status
VirtualIP (ocf::heartbeat:IPaddr2): Started
WebSite (ocf::heartbeat:apache): Started
```

Para mostrar una lista de todos los recursos configurados y los parámetros configurados para esos recursos, utilice la opción **--full** del comando **pcs resource config**, como en el siguiente ejemplo.

```
# pcs resource config
Resource: VirtualIP (type=IPaddr2 class=ocf provider=heartbeat)
Attributes: ip=192.168.0.120 cidr_netmask=24
Operations: monitor interval=30s
Resource: WebSite (type=apache class=ocf provider=heartbeat)
Attributes: statusurl=http://localhost/server-status configfile=/etc/httpd/conf/httpd.conf
Operations: monitor interval=1min
```

Para mostrar los parámetros configurados para un recurso, utilice el siguiente comando.

```
pcs resource config resource_id
```

Por ejemplo, el siguiente comando muestra los parámetros actualmente configurados para el recurso **VirtualIP**.

```
# pcs resource config VirtualIP
Resource: VirtualIP (type=IPaddr2 class=ocf provider=heartbeat)
Attributes: ip=192.168.0.120 cidr_netmask=24
Operations: monitor interval=30s
```

16.2. MODIFICACIÓN DE LOS PARÁMETROS DE LOS RECURSOS

Para modificar los parámetros de un recurso configurado, utilice el siguiente comando.

```
pcs resource update resource_id [resource_options]
```

La siguiente secuencia de comandos muestra los valores iniciales de los parámetros configurados para el recurso **VirtualIP**, el comando para cambiar el valor del parámetro **ip** y los valores posteriores al comando de actualización.

```
# pcs resource config VirtualIP
Resource: VirtualIP (type=IPaddr2 class=ocf provider=heartbeat)
```

```

Attributes: ip=192.168.0.120 cidr_netmask=24
Operations: monitor interval=30s
# pcs resource update VirtualIP ip=192.169.0.120
# pcs resource config VirtualIP
Resource: VirtualIP (type=IPAddr2 class=ocf provider=heartbeat)
Attributes: ip=192.169.0.120 cidr_netmask=24
Operations: monitor interval=30s

```



NOTA

Cuando se actualiza el funcionamiento de un recurso con el comando **pcs resource update**, cualquier opción que no se indique específicamente se restablece a sus valores por defecto.

16.3. BORRAR EL ESTADO DE FALLO DE LOS RECURSOS DEL CLÚSTER

Si un recurso ha fallado, aparece un mensaje de fallo cuando se muestra el estado del cluster. Si resuelve ese recurso, puede borrar ese estado de fallo con el comando **pcs resource cleanup**. Este comando restablece el estado del recurso y **failcount**, indicando al clúster que olvide el historial de operaciones de un recurso y vuelva a detectar su estado actual.

El siguiente comando limpia el recurso especificado por *resource_id*.

```
limpieza de recursos pcs resource_id
```

Si no se especifica un *resource_id*, este comando restablece el estado del recurso y **failcount** para todos los recursos.

El comando **pcs resource cleanup** sondea sólo los recursos que se muestran como una acción fallida. Para sondear todos los recursos en todos los nodos puede introducir el siguiente comando:

```
actualización de recursos pcs
```

Por defecto, el comando **pcs resource refresh** sondea sólo los nodos en los que se conoce el estado de un recurso. Para sondear todos los recursos aunque no se conozca su estado, introduzca el siguiente comando:

```
pcs resource refresh --full
```

16.4. DESPLAZAMIENTO DE RECURSOS EN UN CLÚSTER

Pacemaker proporciona una variedad de mecanismos para configurar un recurso para que se mueva de un nodo a otro y para mover manualmente un recurso cuando sea necesario.

Puede mover manualmente los recursos de un clúster con los comandos **pcs resource move** y **pcs resource relocate**, como se describe en [Mover manualmente los recursos del clúster](#).

Además de estos comandos, también puede controlar el comportamiento de los recursos del clúster habilitando, deshabilitando y prohibiendo recursos, como se describe en [Habilitación, deshabilitación y prohibición de recursos del clúster](#).

Puedes configurar un recurso para que se mueva a un nuevo nodo después de un número definido de fallos, y puedes configurar un clúster para que mueva los recursos cuando se pierda la conectividad externa.

16.4.1. Desplazamiento de recursos por avería

Cuando creas un recurso, puedes configurar el recurso para que se mueva a un nuevo nodo después de un número definido de fallos, estableciendo la opción **migration-threshold** para ese recurso. Una vez alcanzado el umbral, este nodo ya no podrá ejecutar el recurso fallido hasta:

- El administrador restablece manualmente el recurso **failcount** mediante el comando **pcs resource cleanup**.
- Se alcanza el valor del recurso **failure-timeout**.

El valor de **migration-threshold** se establece por defecto en **INFINITY**. **INFINITY** se define internamente como un número muy grande pero finito. Un valor de 0 desactiva la función **migration-threshold**.



NOTA

Configurar un **migration-threshold** para un recurso no es lo mismo que configurar un recurso para la migración, en la que el recurso se mueve a otra ubicación sin pérdida de estado.

El siguiente ejemplo añade un umbral de migración de 10 al recurso llamado **dummy_resource**, que indica que el recurso se moverá a un nuevo nodo después de 10 fallos.

```
# pcs resource meta dummy_resource migration-threshold=10
```

Puede añadir un umbral de migración a los valores predeterminados para todo el clúster con el siguiente comando.

```
# pcs resource defaults migration-threshold=10
```

Para determinar el estado actual de los fallos y los límites del recurso, utilice el comando **pcs resource failcount show**.

Hay dos excepciones al concepto de umbral de migración; ocurren cuando un recurso falla al arrancar o falla al parar. Si la propiedad del clúster **start-failure-is-fatal** se establece en **true** (que es el valor predeterminado), los fallos de arranque hacen que **failcount** se establezca en **INFINITY** y, por lo tanto, siempre hacen que el recurso se mueva inmediatamente.

Los fallos de parada son ligeramente diferentes y cruciales. Si un recurso falla al detenerse y STONITH está habilitado, entonces el cluster cercará el nodo para poder iniciar el recurso en otro lugar. Si STONITH no está habilitado, entonces el cluster no tiene forma de continuar y no intentará iniciar el recurso en otro lugar, sino que intentará detenerlo de nuevo después del tiempo de espera del fallo.

16.4.2. Desplazamiento de recursos por cambios de conectividad

Configurar el clúster para mover los recursos cuando se pierde la conectividad externa es un proceso de dos pasos.

1. Añade un recurso **ping** al cluster. El recurso **ping** utiliza la utilidad del sistema del mismo

nombre para comprobar si una lista de máquinas (especificadas por el nombre de host DNS o la dirección IPv4/IPv6) son alcanzables y utiliza los resultados para mantener un atributo de nodo llamado **pingd**.

2. Configure una restricción de ubicación para el recurso que lo trasladará a un nodo diferente cuando se pierda la conectividad.

Tabla 10.1, “Identificadores de agentes de recursos” describe las propiedades que se pueden establecer para un recurso **ping**.

Tabla 16.1. Propiedades de un recurso de ping

Campo	Descripción
dampen	El tiempo de espera (dampening) para que se produzcan más cambios. Esto evita que un recurso rebote por el clúster cuando los nodos del clúster notan la pérdida de conectividad en momentos ligeramente diferentes.
multiplier	El número de nodos de ping conectados se multiplica por este valor para obtener una puntuación. Es útil cuando hay varios nodos de ping configurados.
host_list	Las máquinas con las que hay que contactar para determinar el estado actual de la conectividad. Los valores permitidos incluyen nombres de host DNS resolubles y direcciones IPv4 e IPv6. Las entradas de la lista de hosts están separadas por espacios.

El siguiente comando de ejemplo crea un recurso **ping** que verifica la conectividad con **gateway.example.com**. En la práctica, usted verificaría la conectividad con su puerta de enlace/enrutador de red. Configure el recurso **ping** como un clon para que el recurso se ejecute en todos los nodos del clúster.

```
# pcs resource create ping ocf:pacemaker:ping dampen=5s multiplier=1000
host_list=gateway.example.com clone
```

El siguiente ejemplo configura una regla de restricción de ubicación para el recurso existente llamado **Webserver**. Esto hará que el recurso **Webserver** se mueva a un host que sea capaz de hacer ping a **gateway.example.com** si el host en el que se está ejecutando actualmente no puede hacer ping a **gateway.example.com**.

```
# pcs constraint location Webserver rule score=-INFINITY pingd lt 1 or not_defined pingd
```

```
Module included in the following assemblies:
```

```
//
```

```
// <List assemblies here, each on a new line>
```

```
// rhel-8-docs/enterprise/assemblies/assembly_managing_cluster_resources.adoc
```

16.5. DESACTIVACIÓN DE UNA OPERACIÓN DE MONITORIZACIÓN

La forma más fácil de detener un monitor recurrente es eliminarlo. Sin embargo, puede haber ocasiones en las que sólo quiera desactivarlo temporalmente. En estos casos, añada **enabled="false"** a la definición de la operación. Cuando quiera restablecer la operación de monitorización, ponga **enabled="true"** en la definición de la operación.

Cuando se actualiza la operación de un recurso con el comando **pcs resource update**, cualquier opción que no se indique específicamente se restablece a sus valores por defecto. Por ejemplo, si ha configurado una operación de monitorización con un valor de tiempo de espera personalizado de 600, la ejecución de los siguientes comandos restablecerá el valor de tiempo de espera al valor predeterminado de 20 (o al que haya establecido el valor predeterminado con el comando **pcs resource ops default**).

```
# pcs resource update resourceXZY op monitor enabled=false
# pcs resource update resourceXZY op monitor enabled=true
```

Para mantener el valor original de 600 para esta opción, al restablecer la operación de supervisión debe especificar ese valor, como en el siguiente ejemplo.

```
# pcs resource update resourceXZY op monitor timeout=600 enabled=true
```

16.6. CONFIGURACIÓN Y GESTIÓN DE LAS ETIQUETAS DE RECURSOS DEL CLÚSTER (RHEL 8.3 Y POSTERIOR)

A partir de Red Hat Enterprise Linux 8.3, puede utilizar el comando **pcs** para etiquetar los recursos del cluster. Esto le permite habilitar, deshabilitar, gestionar o desgestionar un conjunto específico de recursos con un solo comando.

16.6.1. Etiquetado de los recursos del clúster para su administración por categorías

El siguiente procedimiento etiqueta dos recursos con una etiqueta de recurso y desactiva los recursos etiquetados. En este ejemplo, los recursos existentes que se van a etiquetar se llaman **d-01** y **d-02**.

1. Cree una etiqueta denominada **special-resources** para los recursos **d-01** y **d-02**.

```
[root@node-01]# pcs tag create special-resources d-01 d-02
```

2. Muestra la configuración de la etiqueta de recursos.

```
[root@node-01]# pcs tag config
special-resources
  d-01
  d-02
```

3. Desactivar todos los recursos etiquetados con la etiqueta **special-resources**.

```
[root@node-01]# pcs resource disable special-resources
```

4. Muestra el estado de los recursos para confirmar que los recursos **d-01** y **d-02** están desactivados.

```
[root@node-01]# pcs resource
* d-01      (ocf::pacemaker:Dummy): Stopped (disabled)
* d-02      (ocf::pacemaker:Dummy): Stopped (disabled)
```

■

Además del comando **pcs resource disable**, los comandos **pcs resource enable**, **pcs resource manage**, y **pcs resource unmanage** apoyan la administración de recursos etiquetados.

Después de haber creado una etiqueta de recurso:

- Puede eliminar una etiqueta de recurso con el comando **pcs tag delete**.
- Puede modificar la configuración de una etiqueta de recurso existente con el comando **pcs tag update**.

16.6.2. Eliminar un recurso de clúster etiquetado

No se puede eliminar un recurso de cluster etiquetado con el comando **pcs**. Para eliminar un recurso etiquetado, utilice el siguiente procedimiento.

1. Eliminar la etiqueta de recurso.

- a. El siguiente comando elimina la etiqueta de recurso **special-resources** de todos los recursos con esa etiqueta,

```
[root@node-01]# pcs tag remove special-resources
[root@node-01]# pcs tag
No tags defined
```

- b. El siguiente comando elimina la etiqueta de recurso **special-resources** sólo del recurso **d-01**.

```
[root@node-01]# pcs tag update special-resources remove d-01
```

2. Eliminar el recurso.

```
[root@node-01]# pcs resource delete d-01
Attempting to stop: d-01... Stopped
```

CAPÍTULO 17. CREACIÓN DE RECURSOS DE CLUSTER ACTIVOS EN VARIOS NODOS (RECURSOS CLONADOS)

Puede clonar un recurso de clúster para que el recurso pueda estar activo en varios nodos. Por ejemplo, puede utilizar los recursos clonados para configurar varias instancias de un recurso IP para distribuirlo en un clúster para el equilibrio de nodos. Puede clonar cualquier recurso siempre que el agente de recursos lo soporte. Un clon consiste en un recurso o un grupo de recursos.



NOTA

Sólo los recursos que pueden estar activos en varios nodos al mismo tiempo son adecuados para la clonación. Por ejemplo, un recurso **Filesystem** que monte un sistema de archivos no agrupado como **ext4** desde un dispositivo de memoria compartida no debe ser clonado. Dado que la partición **ext4** no es consciente del clúster, este sistema de archivos no es adecuado para las operaciones de lectura/escritura que se producen desde varios nodos al mismo tiempo.

17.1. CREACIÓN Y ELIMINACIÓN DE UN RECURSO CLONADO

Puedes crear un recurso y un clon de ese recurso al mismo tiempo con el siguiente comando.

```
pcs resource create resource_id [standard:[provider:]]type [resource options] [meta resource meta options] clone [clone options]
```

El nombre del clon será ***resource_id-clone***.

No se puede crear un grupo de recursos y un clon de ese grupo de recursos en un solo comando.

Alternativamente, puede crear un clon de un recurso o grupo de recursos previamente creado con el siguiente comando.

```
pcs resource clone resource_id | group_name [clone options]...
```

El nombre del clon será ***resource_id-clone*** o ***group_name-clone***.



NOTA

Es necesario configurar los cambios de configuración de los recursos en un solo nodo.



NOTA

Al configurar las restricciones, utilice siempre el nombre del grupo o del clon.

Cuando se crea un clon de un recurso, el clon toma el nombre del recurso con **-clone** añadido al nombre. Los siguientes comandos crean un recurso de tipo **apache** llamado **webfarm** y un clon de ese recurso llamado **webfarm-clone**.

```
# pcs resource create webfarm apache clone
```




NOTA

Cuando se crea un clon de un recurso o grupo de recursos que se ordenará después de otro clon, casi siempre se debe establecer la opción **interleave=true**. Esto asegura que las copias del clon dependiente puedan detenerse o iniciarse cuando el clon del que depende se haya detenido o iniciado en el mismo nodo. Si no establece esta opción, si un recurso clonado B depende de un recurso clonado A y un nodo abandona el clúster, cuando el nodo regresa al clúster y el recurso A se inicia en ese nodo, entonces todas las copias del recurso B en todos los nodos se reiniciarán. Esto se debe a que cuando un recurso clonado dependiente no tiene la opción **interleave** establecida, todas las instancias de ese recurso dependen de cualquier instancia en ejecución del recurso del que depende.

Utilice el siguiente comando para eliminar un clon de un recurso o grupo de recursos. Esto no elimina el recurso o el grupo de recursos en sí.

```
pcs resource unclone resource_id | group_name
```

Tabla 17.1, “Opciones de clonación de recursos” describe las opciones que puede especificar para un recurso clonado.

Tabla 17.1. Opciones de clonación de recursos

Campo	Descripción
priority, target-role, is-managed	Opciones heredadas del recurso que se está clonando, como se describe en Tabla 10.3, “Meta opciones de recursos” .
clone-max	Cuántas copias del recurso se van a iniciar. Por defecto, el número de nodos del clúster.
clone-node-max	Cuántas copias del recurso pueden iniciarse en un solo nodo; el valor por defecto es 1 .
notify	Cuando se detiene o se inicia una copia del clon, se informa a todas las demás copias de antemano y cuando la acción ha tenido éxito. Valores permitidos: false, true . El valor por defecto es false .

Campo	Descripción
globally-unique	<p>¿Cada copia del clon realiza una función diferente? Valores permitidos: false, true</p> <p>Si el valor de esta opción es false, estos recursos se comportan de forma idéntica en todos los lugares donde se ejecutan y, por tanto, sólo puede haber una copia del clon activa por máquina.</p> <p>Si el valor de esta opción es true, una copia del clon que se ejecuta en una máquina no es equivalente a otra instancia, ya sea que esa instancia se ejecute en otro nodo o en el mismo nodo. El valor por defecto es true si el valor de clone-node-max es mayor que uno; en caso contrario, el valor por defecto es false.</p>
ordered	<p>Si las copias se inician en serie (en lugar de en paralelo). Valores permitidos: false, true. El valor por defecto es false.</p>
interleave	<p>Cambia el comportamiento de las restricciones de ordenación (entre clones) para que las copias del primer clon puedan iniciarse o detenerse tan pronto como la copia en el mismo nodo del segundo clon se haya iniciado o detenido (en lugar de esperar hasta que cada instancia del segundo clon se haya iniciado o detenido). Valores permitidos: false, true. El valor por defecto es false.</p>
clone-min	<p>Si se especifica un valor, cualquier clon que se ordene después de este clon no podrá iniciarse hasta que se ejecute el número especificado de instancias del clon original, incluso si la opción interleave se establece en true.</p>

Para lograr un patrón de asignación estable, los clones son ligeramente pegajosos por defecto, lo que indica que tienen una ligera preferencia por permanecer en el nodo donde se están ejecutando. Si no se proporciona ningún valor para **resource-stickiness**, el clon utilizará un valor de 1. Al ser un valor pequeño, causa una perturbación mínima en los cálculos de puntuación de otros recursos, pero es suficiente para evitar que Pacemaker mueva innecesariamente las copias por el clúster. Para obtener información sobre la configuración de la meta-opción de recursos **resource-stickiness**, consulte [Configuración de meta-opciones de recursos](#).

17.2. CONFIGURAR LAS LIMITACIONES DE RECURSOS DE LOS CLONES

En la mayoría de los casos, un clon tendrá una sola copia en cada nodo activo del clúster. Sin embargo, puede establecer **clone-max** para el clon de recursos a un valor que sea menor que el número total de nodos en el clúster. Si este es el caso, puedes indicar a qué nodos debe asignar preferentemente las copias el clúster con restricciones de ubicación de recursos. Estas restricciones no se escriben de forma diferente a las de los recursos normales, salvo que se debe utilizar el id del clon.

El siguiente comando crea una restricción de ubicación para que el clúster asigne preferentemente el clon de recursos **webfarm-clone** a **node1**.

```
# pcs constraint location webfarm-clone prefers node1
```

Las restricciones de orden se comportan de forma ligeramente diferente para los clones. En el ejemplo siguiente, como la opción de clon de **interleave** se deja por defecto como **false**, ninguna instancia de **webfarm-stats** se iniciará hasta que todas las instancias de **webfarm-clone** que necesiten iniciarse lo hayan hecho. Sólo si no se puede iniciar ninguna copia de **webfarm-clone**, se impedirá que **webfarm-stats** se active. Además, **webfarm-clone** esperará a que **webfarm-stats** se detenga antes de detenerse.

```
# pcs constraint order start webfarm-clone then webfarm-stats
```

La colocación de un recurso regular (o de grupo) con un clon significa que el recurso puede ejecutarse en cualquier máquina con una copia activa del clon. El clúster elegirá una copia en función de dónde se ejecute el clon y de las preferencias de ubicación del propio recurso.

También es posible la colocación entre clones. En estos casos, el conjunto de ubicaciones permitidas para el clon se limita a los nodos en los que el clon está (o estará) activo. La asignación se realiza entonces de forma normal.

El siguiente comando crea una restricción de colocación para garantizar que el recurso **webfarm-stats** se ejecute en el mismo nodo que una copia activa de **webfarm-clone**.

```
# pcs constraint colocation add webfarm-stats with webfarm-clone
```

17.3. CREACIÓN DE RECURSOS CLONADOS PROMOCIONABLES

Los recursos clónicos promocionables son recursos clónicos con el meta atributo **promotable** establecido en **true**. Permiten que las instancias estén en uno de los dos modos de funcionamiento; éstos se denominan **Master** y **Slave**. Los nombres de los modos no tienen significados específicos, excepto por la limitación de que cuando una instancia se inicia, debe aparecer en el estado **Slave**.

17.3.1. Crear un recurso promocionable

Puede crear un recurso como clon promocionable con el siguiente comando único.

```
pcs resource create resource_id [standard:[provider:]]type [resource options] promocionable [clone options]
```

El nombre del clon promocionable será **resource_id-clone**.

Alternativamente, puede crear un recurso promocionable a partir de un recurso o grupo de recursos creado previamente con el siguiente comando. El nombre del clon promocionable será **resource_id-clone** o **group_name-clone**.

```
recurso pcs promocionable resource_id [clone options]
```

Tabla 17.2, “Opciones adicionales de clonación disponibles para los clones promocionables” describe las opciones adicionales de clonación que se pueden especificar para un recurso promocionable.

Tabla 17.2. Opciones adicionales de clonación disponibles para los clones promocionables

Campo	Descripción
promoted-max	Cuántas copias del recurso se pueden promover; por defecto 1.
promoted-node-max	Cuántas copias del recurso pueden promoverse en un solo nodo; por defecto 1.

17.3.2. Configuración de las limitaciones de los recursos promocionables

En la mayoría de los casos, un recurso promocionable tendrá una única copia en cada nodo activo del clúster. Si este no es el caso, puede indicar a qué nodos debe asignar preferentemente las copias el clúster con restricciones de ubicación de los recursos. Estas restricciones no se escriben de forma diferente a las de los recursos normales.

Puede crear una restricción de colocación que especifique si los recursos están operando en un rol de maestro o de esclavo. El siguiente comando crea una restricción de colocación de recursos.

```
pcs constraint colocation add [master|slave] source_resource with [master|slave] target_resource [score] [options]
```

Para obtener información sobre las restricciones de colocación, consulte [Colocación de los recursos del clúster](#).

Al configurar una restricción de ordenación que incluya recursos promocionables, una de las acciones que puede especificar para los recursos es **promote**, que indica que el recurso sea promocionado del rol esclavo al rol maestro. Además, puede especificar una acción de **demote**, que indica que el recurso sea degradado de rol maestro a rol esclavo.

El comando para configurar una restricción de orden es el siguiente.

```
pcs constraint order [action] resource_id then [action] resource_id [options]
```

Para obtener información sobre las restricciones de orden de los recursos, consulte [ifdef:: Determinación del orden de ejecución de los recursos del clúster](#) .

17.4. DESPLAZAMIENTO DE UN RECURSO PROMOCIONADO EN CASO DE FALLO (RHEL 8.3 Y POSTERIORES)

Puede configurar un recurso promocionable para que cuando una acción de **promote** o **monitor** falle para ese recurso, o la partición en la que el recurso se está ejecutando pierda el quórum, el recurso se degradará pero no se detendrá completamente. Esto puede evitar la necesidad de una intervención manual en situaciones en las que sería necesario detener completamente el recurso.

- Para configurar un recurso promocionable para que sea degradado cuando falle una acción de **promote**, establezca la meta-opción de operación **on-fail** a **demote**, como en el siguiente ejemplo.

```
# pcs resource op add my-rsc promote on-fail="demote"
```

- Para configurar un recurso promocionable para que sea degradado cuando una acción de **monitor** falle, establezca **interval** a un valor no nulo, establezca la meta-opción de operación **on-fail** a **demote**, y establezca **role** a **Master**, como en el siguiente ejemplo.

```
# pcs resource op add my-rsc monitor interval="10s" on-fail="demote" role="Master"
```

- Para configurar un clúster de manera que cuando una partición del clúster pierda el quórum cualquier recurso promovido sea degradado pero se deje en funcionamiento y todos los demás recursos se detengan, establezca la propiedad del clúster **no-quorum-policy** en **demote**

La especificación de un metaatributo **demote** para una operación no afecta al modo en que se determina la promoción de un recurso. Si el nodo afectado sigue teniendo la mayor puntuación de promoción, será seleccionado para ser promovido de nuevo.

CAPÍTULO 18. GESTIÓN DE LOS NODOS DEL CLÚSTER

Las siguientes secciones describen los comandos que se utilizan para gestionar los nodos del clúster, incluidos los comandos para iniciar y detener los servicios del clúster y para añadir y eliminar nodos del clúster.

18.1. DETENCIÓN DE LOS SERVICIOS DEL CLÚSTER

El siguiente comando detiene los servicios de cluster en el nodo o nodos especificados. Al igual que con **pcs cluster start**, la opción **--all** detiene los servicios de clúster en todos los nodos y, si no se especifica ningún nodo, los servicios de clúster se detienen únicamente en el nodo local.

```
pcs cluster stop [--all | node] [...]
```

Puede forzar una parada de los servicios del clúster en el nodo local con el siguiente comando, que realiza un comando **kill -9**.

```
pcs cluster kill
```

18.2. ACTIVACIÓN Y DESACTIVACIÓN DE LOS SERVICIOS DEL CLÚSTER

Utilice el siguiente comando para habilitar los servicios del clúster, que configura los servicios del clúster para que se ejecuten al inicio en el nodo o nodos especificados. La habilitación permite que los nodos se reincorporen automáticamente al clúster después de que se hayan cercado, lo que minimiza el tiempo que el clúster está a menos de su capacidad. Si los servicios de clúster no están habilitados, un administrador puede investigar manualmente qué ha fallado antes de iniciar los servicios de clúster manualmente, de modo que, por ejemplo, no se permita que un nodo con problemas de hardware vuelva a entrar en el clúster cuando es probable que vuelva a fallar.

- Si se especifica la opción **--all**, el comando habilita los servicios de cluster en todos los nodos.
- Si no se especifica ningún nodo, los servicios del clúster se activan sólo en el nodo local.

```
pcs cluster enable [--all | node] [...]
```

Utilice el siguiente comando para configurar los servicios del clúster para que no se ejecuten al inicio en el nodo o nodos especificados.

- Si se especifica la opción **--all**, el comando desactiva los servicios de cluster en todos los nodos.
- Si no se especifica ningún nodo, los servicios de cluster se desactivan sólo en el nodo local.

```
pcs cluster disable [--all | node] [...]
```

18.3. AÑADIR NODOS DEL CLÚSTER



NOTA

Se recomienda encarecidamente añadir nodos a los clusters existentes sólo durante una ventana de mantenimiento de producción. Esto le permite realizar las pruebas de recursos y de despliegue adecuadas para el nuevo nodo y su configuración de cercado.

Utilice el siguiente procedimiento para añadir un nuevo nodo a un cluster existente. Este procedimiento añade nodos de clusters estándar que ejecutan **corosync**. Para obtener información sobre la integración de nodos **no corosync** en un cluster, consulte [Integración de nodos no corosync en un cluster: el servicio pacemaker_remote](#).

En este ejemplo, los nodos de cluster existentes son **clusternode-01.example.com**, **clusternode-02.example.com**, y **clusternode-03.example.com**. El nuevo nodo es **newnode.example.com**.

En el nuevo nodo a añadir al cluster, realice las siguientes tareas.

1. Instale los paquetes del cluster. Si el cluster utiliza SBD, el gestor de tickets de Booth o un dispositivo de quórum, debe instalar manualmente también los paquetes respectivos (**sbd**, **booth-site**, **corosync-qdevice**) en el nuevo nodo.

```
[root@newnode ~]# yum install -y pcs fence-agents-all
```

Además de los paquetes del cluster, también necesitará instalar y configurar todos los servicios que esté ejecutando en el cluster, que haya instalado en los nodos existentes del cluster. Por ejemplo, si está ejecutando un servidor Apache HTTP en un cluster de alta disponibilidad de Red Hat, necesitará instalar el servidor en el nodo que está añadiendo, así como la herramienta **wget** que comprueba el estado del servidor.

2. Si está ejecutando el demonio **firewalld**, ejecute los siguientes comandos para habilitar los puertos requeridos por el complemento de alta disponibilidad de Red Hat.

```
# firewall-cmd --permanent --add-service=high-availability
# firewall-cmd --add-service=high-availability
```

3. Establezca una contraseña para el ID de usuario **hacluster**. Se recomienda utilizar la misma contraseña para cada nodo del cluster.

```
[root@newnode ~]# passwd hacluster
Changing password for user hacluster.
New password:
Retype new password:
passwd: all authentication tokens updated successfully.
```

4. Ejecute los siguientes comandos para iniciar el servicio **pcsd** y para habilitar **pcsd** al inicio del sistema.

```
# systemctl start pcsd.service
# systemctl enable pcsd.service
```

En un nodo del cluster existente, realice las siguientes tareas.

1. Autenticar al usuario **hacluster** en el nuevo nodo del cluster.

```
[root@clusternode-01 ~]# pcs host auth newnode.example.com
```

```
Username: hacluster
Password:
newnode.example.com: Authorized
```

2. Añade el nuevo nodo al clúster existente. Este comando también sincroniza el archivo de configuración del clúster **corosync.conf** con todos los nodos del clúster, incluido el nuevo nodo que está añadiendo.

```
[root@clusternode-01 ~]# pcs cluster node add newnode.example.com
```

En el nuevo nodo a añadir al clúster, realice las siguientes tareas.

1. Inicie y habilite los servicios del clúster en el nuevo nodo.

```
[root@newnode ~]# pcs cluster start
Starting Cluster...
[root@newnode ~]# pcs cluster enable
```

2. Asegúrese de configurar y probar un dispositivo de cercado para el nuevo nodo del clúster.

18.4. ELIMINACIÓN DE LOS NODOS DEL CLÚSTER

El siguiente comando apaga el nodo especificado y lo elimina del archivo de configuración del clúster, **corosync.conf**, en todos los demás nodos del clúster.

```
pcs cluster node remove node
```

18.5. AÑADIR UN NODO A UN CLUSTER CON MÚLTIPLES ENLACES

Al añadir un nodo a un cluster con múltiples enlaces, debe especificar las direcciones para todos los enlaces. El siguiente ejemplo añade el nodo **rh80-node3** a un cluster, especificando la dirección IP 192.168.122.203 para el primer enlace y 192.168.123.203 como segundo enlace.

```
# pcs cluster node add rh80-node3 addr=192.168.122.203 addr=192.168.123.203
```

18.6. AÑADIR Y MODIFICAR ENLACES EN UN CLÚSTER EXISTENTE (RHEL 8.1 Y POSTERIOR)

En la mayoría de los casos, puede añadir o modificar los enlaces de un clúster existente sin reiniciar el clúster.

18.6.1. Añadir y eliminar enlaces en un clúster existente

Para añadir un nuevo enlace a un clúster en funcionamiento, utilice el comando **pcs cluster link add**.

- Al añadir un enlace, debe especificar una dirección para cada nodo.
- Añadir y eliminar un enlace sólo es posible cuando se utiliza el protocolo de transporte **knet**.
- Al menos un enlace del clúster debe estar definido en todo momento.

- El número máximo de enlaces en un cluster es de 8, numerados del 0 al 7. No importa qué enlaces se definan, así que, por ejemplo, puedes definir sólo los enlaces 3, 6 y 7.
- Cuando se añade un enlace sin especificar su número de enlace, **pcs** utiliza el enlace más bajo disponible.
- Los números de los enlaces actualmente configurados se encuentran en el archivo **corosync.conf**. Para mostrar el archivo **corosync.conf**, ejecute el comando **pcs cluster corosync**.

El siguiente comando añade el enlace número 5 a un cluster de tres nodos.

```
[root@node1 ~] # pcs cluster link add node1=10.0.5.11 node2=10.0.5.12 node3=10.0.5.31
options linknumber=5
```

Para eliminar un enlace existente, utilice el comando **pcs cluster link delete** o **pcs cluster link remove**. Cualquiera de los siguientes comandos eliminará el enlace número 5 del clúster.

```
[root@node1 ~] # pcs cluster link delete 5
[root@node1 ~] # pcs cluster link remove 5
```

18.6.2. Modificación de un enlace en un cluster con múltiples enlaces

Si hay varios enlaces en el cluster y quiere cambiar uno de ellos, realice el siguiente procedimiento.

1. Elimina el enlace que quieres cambiar.

```
[root@node1 ~] # pcs cluster link remove 2
```

2. Añade el enlace al clúster con las direcciones y opciones actualizadas.

```
[root@node1 ~] # pcs cluster link add node1=10.0.5.11 node2=10.0.5.12
node3=10.0.5.31 options linknumber=2
```

18.6.3. Modificación de las direcciones de enlace en un clúster con un solo enlace

Si su clúster utiliza un solo enlace y quiere modificar ese enlace para utilizar diferentes direcciones, realice el siguiente procedimiento. En este ejemplo, el enlace original es el enlace 1.

1. Añade un nuevo enlace con las nuevas direcciones y opciones.

```
[root@node1 ~] # pcs cluster link add node1=10.0.5.11 node2=10.0.5.12
node3=10.0.5.31 options linknumber=2
```

2. Quite el enlace original.

```
[root@node1 ~] # pcs cluster link remove 1
```

Tenga en cuenta que no puede especificar direcciones que estén actualmente en uso al añadir enlaces a un clúster. Esto significa, por ejemplo, que si tienes un cluster de dos nodos con un enlace y quieres cambiar la dirección de un solo nodo, no puedes usar el procedimiento anterior para añadir un nuevo

enlace que especifique una nueva dirección y una dirección existente. En su lugar, puede añadir un enlace temporal antes de eliminar el enlace existente y volver a añadirlo con la dirección actualizada, como en el siguiente ejemplo.

En este ejemplo:

- El enlace para el cluster existente es el enlace 1, que utiliza la dirección 10.0.5.11 para el nodo 1 y la dirección 10.0.5.12 para el nodo 2.
- Desea cambiar la dirección del nodo 2 a 10.0.5.31.

Para actualizar sólo una de las direcciones de un cluster de dos nodos con un único enlace, utilice el siguiente procedimiento.

1. Añade un nuevo enlace temporal al clúster existente, utilizando direcciones que no están actualmente en uso.

```
[root@node1 ~] # pcs cluster link add node1=10.0.5.13 node2=10.0.5.14 options linknumber=2
```

2. Quite el enlace original.

```
[root@node1 ~] # pcs cluster link remove 1
```

3. Añade el nuevo enlace modificado.

```
[root@node1 ~] # pcs cluster link add node1=10.0.5.11 node2=10.0.5.31 options linknumber=1
```

4. Elimina el enlace temporal que has creado

```
[root@node1 ~] # pcs cluster link remove 2
```

18.6.4. Modificación de las opciones de enlace para un enlace en un cluster con un solo enlace

Si su clúster utiliza sólo un enlace y quiere modificar las opciones de ese enlace pero no quiere cambiar la dirección a utilizar, puede añadir un enlace temporal antes de eliminar y actualizar el enlace a modificar.

En este ejemplo:

- El enlace para el cluster existente es el enlace 1, que utiliza la dirección 10.0.5.11 para el nodo 1 y la dirección 10.0.5.12 para el nodo 2.
- Desea cambiar la opción de enlace **link_priority** a 11.

Utilice el siguiente procedimiento para modificar la opción de enlace en un cluster con un solo enlace.

1. Añade un nuevo enlace temporal al clúster existente, utilizando direcciones que no están actualmente en uso.

```
[root@node1 ~] # pcs cluster link add node1=10.0.5.13 node2=10.0.5.14 options linknumber=2
```

- Quite el enlace original.

```
[root@node1 ~] # pcs cluster link remove 1
```

- Vuelve a añadir el enlace original con las opciones actualizadas.

```
[root@node1 ~] # pcs cluster link add node1=10.0.5.11 node2=10.0.5.12 options
linknumber=1 link_priority=11
```

- Elimina el enlace temporal.

```
[root@node1 ~] # pcs cluster link remove 2
```

18.6.5. No es posible modificar un enlace cuando se añade uno nuevo

Si por alguna razón no es posible añadir un nuevo enlace en su configuración y su única opción es modificar un enlace existente, puede utilizar el siguiente procedimiento, que requiere que apague su cluster.

El siguiente procedimiento de ejemplo actualiza el enlace número 1 en el cluster y establece la opción **link_priority** para el enlace a 11.

- Detenga los servicios de cluster para el cluster.

```
[root@node1 ~] # pcs cluster stop --all
```

- Actualice las direcciones y opciones de los enlaces.

El comando **pcs cluster link update** no requiere que se especifiquen todas las direcciones y opciones de los nodos. En su lugar, puede especificar sólo las direcciones a modificar. Este ejemplo modifica las direcciones de **node1** y **node3** y la opción **link_priority** solamente.

```
[root@node1 ~] # pcs cluster link update 1 node1=10.0.5.11 node3=10.0.5.31 options
link_priority=11
```

Para eliminar una opción, puede establecer un valor nulo con el formato **option=** formato.

- Reiniciar el clúster

```
[root@node1 ~] # pcs cluster start --all
```

18.7. CONFIGURACIÓN DE UN GRAN CLÚSTER CON MUCHOS RECURSOS

Si el clúster que va a desplegar está formado por un gran número de nodos y muchos recursos, es posible que tenga que modificar los valores por defecto de los siguientes parámetros para su clúster.

La propiedad del clúster **cluster-ipc-limit**

La propiedad de clúster **cluster-ipc-limit** es la máxima acumulación de mensajes IPC antes de que un demonio de clúster desconecte a otro. Cuando un gran número de recursos se limpian o modifican simultáneamente en un cluster grande, un gran número de actualizaciones CIB llegan a la

vez. Esto podría hacer que los clientes más lentos sean desalojados si el servicio Pacemaker no tiene tiempo de procesar todas las actualizaciones de configuración antes de que se alcance el umbral de la cola de eventos CIB.

El valor recomendado de **cluster-ipc-limit** para su uso en clusters grandes es el número de recursos en el cluster multiplicado por el número de nodos. Este valor puede aumentarse si se ven mensajes de "desalojo de clientes" para PIDs de demonio de clúster en los registros.

Puede aumentar el valor de **cluster-ipc-limit** desde su valor por defecto de 500 con el comando **pcs property set**. Por ejemplo, para un clúster de diez nodos con 200 recursos, puede establecer el valor de **cluster-ipc-limit** en 2000 con el siguiente comando.

```
# pcs property set cluster-ipc-limit=2000
```

El parámetro **PCMK_ipc_buffer** Marcapasos

En implementaciones muy grandes, los mensajes internos de Pacemaker pueden exceder el tamaño del búfer de mensajes. Cuando esto ocurra, verá un mensaje en los registros del sistema con el siguiente formato:

```
Compressed message exceeds X% of configured IPC limit (X bytes); consider setting PCMK_ipc_buffer to X or higher
```

Cuando vea este mensaje, puede aumentar el valor de **PCMK_ipc_buffer** en el archivo de configuración **/etc/sysconfig/pacemaker** en cada nodo. Por ejemplo, para aumentar el valor de **PCMK_ipc_buffer** de su valor predeterminado a 13396332 bytes, cambie el campo no comentado **PCMK_ipc_buffer** en el archivo **/etc/sysconfig/pacemaker** en cada nodo del clúster de la siguiente manera.

```
PCMK_ipc_buffer=13396332
```

Para aplicar este cambio, ejecute el siguiente comando.

```
# systemctl restart pacemaker
```

CAPÍTULO 19. CONFIGURACIÓN DE LOS PERMISOS DE LOS USUARIOS PARA UN CLÚSTER PACEMAKER

Puede conceder permiso a usuarios específicos distintos del usuario **hacluster** para gestionar un cluster Pacemaker. Hay dos conjuntos de permisos que puede conceder a usuarios individuales:

- Permisos que permiten a los usuarios individuales gestionar el clúster a través de la interfaz web y ejecutar los comandos de **pcs** que se conectan a los nodos a través de una red. Los comandos que se conectan a los nodos a través de una red incluyen comandos para configurar un clúster, o para añadir o eliminar nodos de un clúster.
- Permisos para los usuarios locales para permitir el acceso de sólo lectura o de lectura-escritura a la configuración del clúster. Los comandos que no requieren conectarse a través de una red incluyen comandos que editan la configuración del clúster, como los que crean recursos y configuran restricciones.

En las situaciones en las que se han asignado ambos conjuntos de permisos, se aplican primero los permisos para los comandos que se conectan a través de una red y luego se aplican los permisos para editar la configuración del clúster en el nodo local. La mayoría de los comandos de **pcs** no requieren acceso a la red y en esos casos no se aplicarán los permisos de red.

19.1. ESTABLECER PERMISOS PARA EL ACCESO A LOS NODOS A TRAVÉS DE UNA RED

Para conceder permiso a determinados usuarios para gestionar el clúster a través de la interfaz web y para ejecutar comandos de **pcs** que se conectan a los nodos a través de una red, añada esos usuarios al grupo **haclient**. Esto debe hacerse en cada nodo del clúster.

19.2. ESTABLECIMIENTO DE PERMISOS LOCALES MEDIANTE ACLS

Puede utilizar el comando **pcs acl** para establecer los permisos de los usuarios locales para permitir el acceso de sólo lectura o de lectura-escritura a la configuración del clúster mediante el uso de listas de control de acceso (ACL).

Por defecto, las ACLs no están habilitadas. Cuando las ACLs no están habilitadas, el usuario root y cualquier usuario que sea miembro del grupo **haclient** en todos los nodos tiene acceso local completo de lectura/escritura a la configuración del cluster mientras que los usuarios que no son miembros de **haclient** no tienen acceso. Sin embargo, cuando las ACLs están activadas, incluso los usuarios que son miembros del grupo **haclient** tienen acceso sólo a lo que se le ha concedido a ese usuario por las ACLs.

El establecimiento de permisos para los usuarios locales es un proceso de dos pasos:

1. Ejecute el comando **pcs acl role create...** para crear un *role* que defina los permisos para ese rol.
2. Asigne el rol que creó a un usuario con el comando **pcs acl user create**. Si asigna varios roles al mismo usuario, cualquier permiso de **deny** tiene prioridad, luego **write**, luego **read**.

El siguiente procedimiento de ejemplo proporciona acceso de sólo lectura para una configuración de clúster a un usuario local llamado **rouser**. Tenga en cuenta que también es posible restringir el acceso sólo a determinadas partes de la configuración.



AVISO

Es importante realizar este procedimiento como root o guardar todas las actualizaciones de la configuración en un archivo de trabajo que luego puede empujar a la CIB activa cuando haya terminado. De lo contrario, puede bloquearse a sí mismo para realizar más cambios. Para obtener información sobre cómo guardar las actualizaciones de configuración en un archivo de trabajo, consulte [Guardar un cambio de configuración en un archivo de trabajo](#).

1. Este procedimiento requiere que el usuario **rouser** exista en el sistema local y que el usuario **rouser** sea miembro del grupo **haclient**.

```
# adduser rouser
# usermod -a -G haclient rouser
```

2. Habilite las ACL de Pacemaker con el comando **pcs acl enable**.

```
# pcs acl enable
```

3. Cree un rol llamado **read-only** con permisos de sólo lectura para el cib.

```
# pcs acl role create read-only description="Read access to cluster" read xpath /cib
```

4. Cree el usuario **rouser** en el sistema pcs ACL y asigne a ese usuario el rol **read-only**.

```
# pcs acl user create rouser read-only
```

5. Ver las ACLs actuales.

```
# pcs acl
User: rouser
Roles: read-only
Role: read-only
Description: Read access to cluster
Permission: read xpath /cib (read-only-read)
```

CAPÍTULO 20. OPERACIONES DE CONTROL DE RECURSOS

Para asegurarse de que los recursos se mantienen sanos, puede añadir una operación de monitorización a la definición de un recurso. Si no se especifica una operación de monitorización para un recurso, por defecto el comando **pcs** creará una operación de monitorización, con un intervalo determinado por el agente de recursos. Si el agente de recursos no proporciona un intervalo de monitorización por defecto, el comando **pcs** creará una operación de monitorización con un intervalo de 60 segundos.

Tabla 20.1, “Propiedades de una operación” resume las propiedades de una operación de supervisión de recursos.

Tabla 20.1. Propiedades de una operación

Campo	Descripción
id	Nombre único para la acción. El sistema lo asigna cuando se configura una operación.
name	La acción a realizar. Valores comunes: monitor , start , stop
interval	<p>Si se establece un valor distinto de cero, se crea una operación recurrente que se repite con esta frecuencia, en segundos. Un valor distinto de cero sólo tiene sentido cuando la acción name se establece en monitor. Una acción de monitorización recurrente se ejecutará inmediatamente después de que se complete el inicio de un recurso, y las acciones de monitorización subsiguientes se programan a partir del momento en que se completó la acción de monitorización anterior. Por ejemplo, si una acción de monitorización con interval=20s se ejecuta a las 01:00:00, la siguiente acción de monitorización no se produce a las 01:00:20, sino a los 20 segundos de completarse la primera acción de monitorización.</p> <p>Si se establece en cero, que es el valor por defecto, este parámetro permite proporcionar los valores que se utilizarán para las operaciones creadas por el clúster. Por ejemplo, si el interval se establece en cero, el name de la operación se establece en start, y el valor timeout se establece en 40, entonces Pacemaker utilizará un tiempo de espera de 40 segundos al iniciar este recurso. Una operación monitor con un intervalo de cero permite establecer los valores timeout/on-fail/enabled para las sondas que Pacemaker realiza al iniciar para obtener el estado actual de todos los recursos cuando los valores predeterminados no son deseables.</p>
timeout	<p>Si la operación no se completa en la cantidad de tiempo establecida por este parámetro, aborta la operación y la considera fallida. El valor por defecto es el valor de timeout si se establece con el comando pcs resource op defaults, o 20 segundos si no se establece. Si encuentra que su sistema incluye un recurso que requiere más tiempo del que el sistema permite para realizar una operación (como start, stop, o monitor), investigue la causa y si se espera que el tiempo de ejecución sea largo puede aumentar este valor.</p> <p>El valor de timeout no es un retraso de ningún tipo, ni el clúster espera todo el periodo de tiempo de espera si la operación vuelve antes de que el periodo de tiempo de espera se haya completado.</p>

Campo	Descripción
on-fail	<p>La acción a realizar si esta acción falla. Valores permitidos:</p> <ul style="list-style-type: none"> * ignore - Finge que el recurso no ha fallado * block - No realizar más operaciones en el recurso * stop - Detener el recurso y no iniciarlo en otro lugar * restart - Detener el recurso y volver a iniciarlo (posiblemente en un nodo diferente) * fence - STONITH el nodo en el que falló el recurso * standby - Alejar los recursos de <i>all</i> del nodo en el que ha fallado el recurso * demote - Cuando una acción promote falla para el recurso, el recurso será degradado pero no será detenido completamente. Cuando una acción de monitor falla para un recurso, si interval se establece en un valor distinto de cero y role se establece en Master, el recurso se degradará pero no se detendrá por completo. <p>El valor por defecto de la operación stop es fence cuando STONITH está activado y block en caso contrario. Todas las demás operaciones son por defecto restart.</p>
enabled	Si false , la operación se trata como si no existiera. Valores permitidos true, false

20.1. CONFIGURACIÓN DE LAS OPERACIONES DE SUPERVISIÓN DE RECURSOS

Puede configurar las operaciones de supervisión cuando cree un recurso, utilizando el siguiente comando.

```
pcs resource create resource_id standard:provider:type|type [resource_options] [op operation_action operation_options [operation_type operation_options]...]
```

Por ejemplo, el siguiente comando crea un recurso **IPaddr2** con una operación de monitorización. El nuevo recurso se llama **VirtualIP** con una dirección IP de 192.168.0.99 y una máscara de red de 24 en **eth2**. Se realizará una operación de monitorización cada 30 segundos.

```
# pcs resource create VirtualIP ocf:heartbeat:IPaddr2 ip=192.168.0.99 cidr_netmask=24 nic=eth2 op monitor interval=30s
```

Alternativamente, puede añadir una operación de supervisión a un recurso existente con el siguiente comando.

```
pcs resource op add resource_id operation_action [operation_properties]
```

Utilice el siguiente comando para eliminar una operación de recurso configurada.

```
pcs resource op remove resource_id operation_name operation_properties
```




NOTA

Debe especificar las propiedades exactas de la operación para eliminar correctamente una operación existente.

Para cambiar los valores de una opción de monitorización, puede actualizar el recurso. Por ejemplo, puede crear un **VirtualIP** con el siguiente comando.

```
# pcs resource create VirtualIP ocf:heartbeat:IPaddr2 ip=192.168.0.99 cidr_netmask=24
nic=eth2
```

Por defecto, este comando crea estas operaciones.

```
Operations: start interval=0s timeout=20s (VirtualIP-start-timeout-20s)
            stop interval=0s timeout=20s (VirtualIP-stop-timeout-20s)
            monitor interval=10s timeout=20s (VirtualIP-monitor-interval-10s)
```

Para cambiar la operación de parada del tiempo de espera, ejecute el siguiente comando.

```
# pcs resource update VirtualIP op stop interval=0s timeout=40s

# pcs resource show VirtualIP
Resource: VirtualIP (class=ocf provider=heartbeat type=IPaddr2)
Attributes: ip=192.168.0.99 cidr_netmask=24 nic=eth2
Operations: start interval=0s timeout=20s (VirtualIP-start-timeout-20s)
            monitor interval=10s timeout=20s (VirtualIP-monitor-interval-10s)
            stop interval=0s timeout=40s (VirtualIP-name-stop-interval-0s-timeout-40s)
```

20.2. CONFIGURACIÓN DE LOS VALORES PREDETERMINADOS DE LAS OPERACIONES DE RECURSOS GLOBALES

A partir de Red Hat Enterprise Linux 8.3, puede cambiar el valor por defecto de una operación de recursos para todos los recursos con el comando **pcs resource op defaults update**. El siguiente comando establece un valor global por defecto de **timeout** de 240 segundos para todas las operaciones de monitoreo.

```
# pcs resource op defaults update timeout=240s
```

El comando original **pcs resource op defaults name=value** que establecía los valores predeterminados de funcionamiento de los recursos en las versiones anteriores de RHEL 8, sigue siendo compatible a menos que haya más de un conjunto de valores predeterminados configurados. Sin embargo, **pcs resource op defaults update** es ahora la versión preferida del comando.

20.2.1. Anulación de los valores de las operaciones específicas de los recursos

Tenga en cuenta que un recurso de clúster utilizará el valor global por defecto sólo cuando la opción no se especifique en la definición del recurso de clúster. Por defecto, los agentes de recursos definen la opción **timeout** para todas las operaciones. Para que se respete el valor global de tiempo de espera de la operación, debe crear el recurso de clúster sin la opción **timeout** explícitamente o debe eliminar la opción **timeout** actualizando el recurso de clúster, como en el siguiente comando.

```
# pcs resource update VirtualIP op monitor interval=10s
```

Por ejemplo, después de establecer un valor global por defecto de **timeout** de 240 segundos para todas las operaciones de monitorización y de actualizar el recurso de cluster **VirtualIP** para eliminar el valor de tiempo de espera para la operación **monitor**, el recurso **VirtualIP** tendrá entonces valores de tiempo de espera para las operaciones **start**, **stop** y **monitor** de 20s, 40s y 240s, respectivamente. El valor global por defecto para las operaciones de tiempo de espera se aplica aquí sólo en la operación **monitor**, donde la opción por defecto **timeout** fue eliminada por el comando anterior.

```
# pcs resource show VirtualIP
Resource: VirtualIP (class=ocf provider=heartbeat type=IPAddr2)
Attributes: ip=192.168.0.99 cidr_netmask=24 nic=eth2
Operations: start interval=0s timeout=20s (VirtualIP-start-timeout-20s)
            monitor interval=10s (VirtualIP-monitor-interval-10s)
            stop interval=0s timeout=40s (VirtualIP-name-stop-interval-0s-timeout-40s)
```

20.2.2. Cambio del valor por defecto de una operación de recursos para conjuntos de recursos (RHEL 8.3 y posteriores)

A partir de Red Hat Enterprise Linux 8.3, puede crear múltiples conjuntos de valores predeterminados de operaciones de recursos con el comando **pcs resource op defaults set create**, que le permite especificar una regla que contenga **resource** y expresiones de operación. Sólo **resource** y las expresiones de operación, incluyendo **and**, **or** y paréntesis, están permitidas en las reglas que usted especifica con este comando.

Con este comando, se puede configurar un valor de operación de recursos por defecto para todos los recursos de un tipo determinado. Por ejemplo, ahora es posible configurar los recursos implícitos de **podman** creados por Pacemaker cuando los paquetes están en uso.

El siguiente comando establece un valor de tiempo de espera por defecto de 90 para todas las operaciones de todos los recursos de **podman**. En este ejemplo, **::podman** significa un recurso de cualquier clase, cualquier proveedor, de tipo **podman**.

La opción **id**, que nombra el conjunto de operaciones de recursos por defecto, no es obligatoria. Si no se establece esta opción, **pcs** generará un ID automáticamente. La configuración de este valor le permite proporcionar un nombre más descriptivo.

```
# pcs resource op defaults set create id=podman-timeout meta timeout=90s rule resource
::podman
```

El siguiente comando establece un valor de tiempo de espera por defecto de 120s para la operación **stop** para todos los recursos.

```
# pcs resource op defaults set create id=stop-timeout meta timeout=120s rule op stop
```

Es posible establecer el valor de tiempo de espera por defecto para una operación específica para todos los recursos de un tipo particular. El siguiente ejemplo establece un valor de tiempo de espera por defecto de 120s para la operación **stop** para todos los recursos **podman**.

```
# pcs resource op defaults set create id=podman-stop-timeout meta timeout=120s rule
resource ::podman and op stop
```

20.2.3. Visualización de los valores por defecto de las operaciones de los recursos actualmente configurados

El comando **pcs resource op defaults** muestra una lista de los valores por defecto actualmente configurados para las operaciones de recursos, incluyendo cualquier regla que haya especificado.

El siguiente comando muestra los valores de operación por defecto para un cluster que ha sido configurado con un valor de tiempo de espera por defecto de 90s para todas las operaciones de todos los recursos de **podman**, y para el cual se ha establecido un ID para el conjunto de valores por defecto de las operaciones de los recursos como **podman-timeout**.

```
# pcs resource op defaults
Meta Attrs: podman-timeout
timeout=90s
Rule: boolean-op=and score=INFINITY
Expression: resource ::podman
```

El siguiente comando muestra los valores de operación por defecto para un cluster que ha sido configurado con un valor de tiempo de espera por defecto de 120s para la operación **stop** para todos los recursos **podman**, y para el cual se ha establecido un ID para el conjunto de valores por defecto de operación de recursos como **podman-stop-timeout**.

```
# pcs resource op defaults
Meta Attrs: podman-stop-timeout
timeout=120s
Rule: boolean-op=and score=INFINITY
Expression: resource ::podman
Expression: op stop
```

20.3. CONFIGURACIÓN DE MÚLTIPLES OPERACIONES DE SUPERVISIÓN

Puede configurar un único recurso con tantas operaciones de monitorización como soporte tenga el agente de recursos. De este modo, puede hacer una comprobación de salud superficial cada minuto y otras progresivamente más intensas a intervalos mayores.



NOTA

Cuando se configuran operaciones de monitorización múltiple, hay que asegurarse de que no se realizan dos operaciones en el mismo intervalo.

Para configurar operaciones de supervisión adicionales para un recurso que admita comprobaciones más exhaustivas a diferentes niveles, se añade una opción **OCF_CHECK_LEVEL=*n*** opción.

Por ejemplo, si se configura el siguiente recurso **IPAddr2**, por defecto se crea una operación de supervisión con un intervalo de 10 segundos y un valor de tiempo de espera de 20 segundos.

```
# pcs resource create VirtualIP ocf:heartbeat:IPAddr2 ip=192.168.0.99 cidr_netmask=24
nic=eth2
```

Si la IP virtual admite una comprobación diferente con una profundidad de 10, el siguiente comando hace que Pacemaker realice la comprobación de supervisión más avanzada cada 60 segundos, además de la comprobación normal de la IP virtual cada 10 segundos. (Como se indica, no debe configurar la operación de supervisión adicional con un intervalo de 10 segundos también)

```
# pcs resource op add VirtualIP monitor interval=60s OCF_CHECK_LEVEL=10
```

CAPÍTULO 21. PROPIEDADES DE LA AGRUPACIÓN DE MARCAPASOS

Las propiedades del clúster controlan cómo se comporta el clúster ante situaciones que pueden darse durante su funcionamiento.

21.1. RESUMEN DE LAS PROPIEDADES Y OPCIONES DEL CLÚSTER

Tabla 21.1, “Propiedades del clúster” resume las propiedades del cluster Pacemaker, mostrando los valores por defecto de las propiedades y los posibles valores que puede establecer para esas propiedades.

Hay propiedades adicionales del clúster que determinan el comportamiento del cercado. Para obtener información sobre estas propiedades, consulte [Opciones avanzadas de configuración de cercado](#).



NOTA

Además de las propiedades descritas en esta tabla, existen otras propiedades del cluster que son expuestas por el software del cluster. Para estas propiedades, se recomienda no cambiar sus valores de los predeterminados.

Tabla 21.1. Propiedades del clúster

Opción	Por defecto	Descripción
batch-limit	0	El número de acciones de recursos que el clúster puede ejecutar en paralelo. El valor "correcto" dependerá de la velocidad y la carga de su red y de los nodos del clúster. El valor predeterminado de 0 significa que el clúster impondrá dinámicamente un límite cuando algún nodo tenga una carga elevada de CPU.
migration-limit	-1 (ilimitado)	El número de trabajos de migración que el clúster puede ejecutar en paralelo en un nodo.

Opción	Por defecto	Descripción
no-quorum-policy	detener	<p>Qué hacer cuando el cluster no tiene quórum. Valores permitidos:</p> <ul style="list-style-type: none"> * ignorar - continuar con la gestión de todos los recursos * congelar - continuar la gestión de los recursos, pero no recuperar los recursos de los nodos que no están en la partición afectada * detener - detener todos los recursos de la partición del clúster afectada * suicidio - valla todos los nodos de la partición del clúster afectado * degradar - si una partición del clúster pierde el quórum, degradar cualquier recurso promovido y detener todos los demás recursos
symmetric-cluster	verdadero	Indica si los recursos pueden ejecutarse en cualquier nodo por defecto.
cluster-delay	60s	Retraso de ida y vuelta en la red (excluyendo la ejecución de la acción). El valor "correcto" dependerá de la velocidad y la carga de su red y de los nodos del clúster.
stop-orphan-resources	verdadero	Indica si los recursos eliminados deben ser detenidos.
stop-orphan-actions	verdadero	Indica si las acciones borradas deben ser canceladas.

Opción	Por defecto	Descripción
start-failure-is-fatal	verdadero	<p>Indica si un fallo en el arranque de un recurso en un nodo concreto impide nuevos intentos de arranque en ese nodo. Cuando se establece en false, el clúster decidirá si se intenta iniciar de nuevo en el mismo nodo basándose en el recuento de fallos actual del recurso y en el umbral de migración. Para obtener información sobre la configuración de la opción migration-threshold para un recurso, consulte Configuración de las opciones meta de los recursos.</p> <p>Si se establece start-failure-is-fatal en false se corre el riesgo de que un nodo defectuoso que no pueda iniciar un recurso retenga todas las acciones dependientes. Por este motivo, start-failure-is-fatal tiene por defecto el valor verdadero. El riesgo de establecer start-failure-is-fatal=false se puede mitigar estableciendo un umbral de migración bajo para que otras acciones puedan continuar después de ese número de fallos.</p>
pe-error-series-max	-1 (todos)	El número de entradas del programador que resultan en ERRORES para guardar. Se utiliza cuando se informa de los problemas.
pe-warn-series-max	-1 (todos)	El número de entradas del programador que resultan en ADVERTENCIAS para guardar. Se utiliza cuando se informa de los problemas.
pe-input-series-max	-1 (todos)	El número de entradas "normales" del programador que hay que guardar. Se utiliza cuando se informa de los problemas.

Opción	Por defecto	Descripción
cluster-infrastructure		La pila de mensajería en la que se está ejecutando actualmente Pacemaker. Se utiliza con fines informativos y de diagnóstico; no es configurable por el usuario.
dc-version		Versión de Pacemaker en el controlador designado (DC) del cluster. Se utiliza con fines de diagnóstico; no es configurable por el usuario.
cluster-recheck-interval	15 minutos	Intervalo de sondeo para los cambios basados en el tiempo de las opciones, los parámetros de los recursos y las restricciones. Valores permitidos: Cero desactiva el sondeo, los valores positivos son un intervalo en segundos (a menos que se especifiquen otras unidades SI, como 5min). Tenga en cuenta que este valor es el tiempo máximo entre comprobaciones; si un evento del cluster ocurre antes del tiempo especificado por este valor, la comprobación se hará antes.
maintenance-mode	falso	El modo de mantenimiento indica al clúster que pase a un modo "sin manos" y que no inicie ni detenga ningún servicio hasta que se le indique lo contrario. Cuando el modo de mantenimiento se ha completado, el clúster realiza una comprobación del estado actual de los servicios y detiene o inicia los que lo necesiten.
shutdown-escalation	20min	El tiempo después del cual hay que renunciar a tratar de cerrar elegantemente y simplemente salir. Sólo para uso avanzado.
stop-all-resources	falso	En caso de que el clúster detenga todos los recursos.

Opción	Por defecto	Descripción
enable-acl	falso	Indica si el clúster puede utilizar listas de control de acceso, tal y como se establece con el comando pcs acl .
placement-strategy	default	Indica si el clúster tendrá en cuenta los atributos de utilización a la hora de determinar la ubicación de los recursos en los nodos del clúster y cómo lo hará.
priority-fencing-delay	0 (desactivado)	<p>(RHEL 8.3 y posteriores) Permite configurar un clúster de dos nodos para que, en una situación de división del cerebro, el nodo con menos recursos en ejecución sea el que se cerque.</p> <p>La propiedad priority-fencing-delay puede establecerse con una duración de tiempo. El valor por defecto de esta propiedad es 0 (desactivado). Si esta propiedad se establece con un valor distinto de cero, y el meta-atributo priority está configurado para al menos un recurso, entonces en una situación de división del cerebro el nodo con la mayor prioridad combinada de todos los recursos que se ejecutan en él tendrá más probabilidades de sobrevivir.</p> <p>Por ejemplo, si se establece pcs resource defaults priority=1 y pcs property set priority-fencing-delay=15s y no se establece ninguna otra prioridad, el nodo que ejecute más recursos tendrá más probabilidades de sobrevivir porque el otro nodo esperará 15 segundos antes de iniciar el cercado. Si un recurso en particular es más importante que el resto, puedes darle una mayor prioridad.</p> <p>El nodo que ejecuta el rol de maestro de un clon promocionable obtiene 1 punto extra si se ha configurado una prioridad para ese clon.</p>

Opción	Por defecto	Descripción
		<p>Cualquier retardo establecido con la propiedad priority-fencing-delay se añadirá a cualquier retardo de las propiedades del dispositivo de vallado pcmk_delay_base y pcmk_delay_max. Este comportamiento permite un cierto retraso cuando ambos nodos tienen la misma prioridad, o ambos nodos necesitan ser cercados por alguna razón que no sea la pérdida del nodo (por ejemplo, si on-fail=fencing se establece para una operación de monitorización de recursos). Si se utiliza en combinación, se recomienda establecer la propiedad priority-fencing-delay a un valor que sea significativamente mayor que el retardo máximo de pcmk_delay_base y pcmk_delay_max, para estar seguro de que el nodo priorizado es el preferido (el doble del valor sería completamente seguro).</p> <p>Sólo el cercado programado por el propio Marcapasos observará priority-fencing-delay. El cercado programado por un código externo como dlm_controld no proporcionará la información necesaria al dispositivo de cercado.</p>

21.2. ESTABLECER Y ELIMINAR LAS PROPIEDADES DEL CLÚSTER

Para establecer el valor de una propiedad del clúster, utilice el siguiente **pcs** comando.

```
pcs property set property=value
```

Por ejemplo, para establecer el valor de **symmetric-cluster** en **false**, utilice el siguiente comando.

```
# pcs property set symmetric-cluster=false
```

Puede eliminar una propiedad del clúster de la configuración con el siguiente comando.

```
propiedad pcs unset property
```

Alternativamente, puede eliminar una propiedad del clúster de una configuración dejando en blanco el

campo de valor del comando **pcs property set**. Esto restablece esa propiedad a su valor por defecto. Por ejemplo, si ha establecido previamente la propiedad **symmetric-cluster** a **false**, el siguiente comando elimina el valor que ha establecido de la configuración y restaura el valor de **symmetric-cluster** a **true**, que es su valor por defecto.

```
# pcs property set symmetric-cluster=
```

21.3. CONSULTA DE LA CONFIGURACIÓN DE LAS PROPIEDADES DEL CLÚSTER

En la mayoría de los casos, cuando se utiliza el comando **pcs** para mostrar los valores de los distintos componentes del cluster, se puede utilizar **pcs list** o **pcs show** indistintamente. En los siguientes ejemplos, **pcs list** es el formato utilizado para mostrar una lista completa de todos los ajustes de más de una propiedad, mientras que **pcs show** es el formato utilizado para mostrar los valores de una propiedad específica.

Para mostrar los valores de la configuración de las propiedades que se han establecido para el clúster, utilice el siguiente **pcs** comando.

```
lista de propiedades de pcs
```

Para mostrar todos los valores de las configuraciones de las propiedades del cluster, incluyendo los valores por defecto de las configuraciones de las propiedades que no se han establecido explícitamente, utilice el siguiente comando.

```
pcs property list --all
```

Para mostrar el valor actual de una propiedad específica del clúster, utilice el siguiente comando.

```
pcs property show property
```

Por ejemplo, para mostrar el valor actual de la propiedad **cluster-infrastructure**, ejecute el siguiente comando:

```
# pcs property show cluster-infrastructure
Cluster Properties:
cluster-infrastructure: cman
```

Con fines informativos, puede mostrar una lista de todos los valores por defecto de las propiedades, tanto si se han establecido como si no, utilizando el siguiente comando.

```
propiedad pcs [list|show] --defaults
```

CAPÍTULO 22. CONFIGURAR LOS RECURSOS PARA QUE PERMANEZCAN DETENIDOS AL APAGAR UN NODO LIMPIO (RHEL 8.2 Y POSTERIORES)

Cuando un nodo del clúster se apaga, la respuesta predeterminada de Pacemaker es detener todos los recursos que se ejecutan en ese nodo y recuperarlos en otro lugar, incluso si el apagado es limpio. A partir de RHEL 8.2, puede configurar Pacemaker para que, cuando un nodo se apague limpiamente, los recursos conectados al nodo queden bloqueados en él y no puedan arrancar en otro lugar hasta que vuelvan a arrancar cuando el nodo que se ha apagado se reincorpore al clúster. Esto permite apagar los nodos durante las ventanas de mantenimiento, cuando las interrupciones del servicio son aceptables, sin provocar que los recursos de ese nodo fallen en otros nodos del clúster.

22.1. PROPIEDADES DEL CLÚSTER PARA CONFIGURAR QUE LOS RECURSOS PERMANEZCAN PARADOS AL APAGAR UN NODO LIMPIO

La capacidad de evitar que los recursos fallen en un cierre de nodo limpio se implementa por medio de las siguientes propiedades del clúster.

shutdown-lock

Cuando esta propiedad del clúster se establece en el valor predeterminado de **false**, el clúster recuperará los recursos que estén activos en los nodos que se apaguen limpiamente. Cuando esta propiedad se establece en **true**, los recursos que están activos en los nodos que se apagan limpiamente no pueden iniciarse en otro lugar hasta que se inicien en el nodo de nuevo después de que se reincorpore al clúster.

La propiedad **shutdown-lock** funcionará tanto para los nodos del cluster como para los nodos remotos, pero no para los nodos invitados.

Si **shutdown-lock** está configurado como **true**, puede eliminar el bloqueo de un recurso del clúster cuando un nodo está inactivo para que el recurso pueda iniciarse en otro lugar, realizando un refresco manual en el nodo con el siguiente comando.

pcs resource refresh resource node=nodename

Ten en cuenta que, una vez desbloqueados los recursos, el clúster es libre de moverlos a otro lugar. Puedes controlar la probabilidad de que esto ocurra utilizando valores de pegajosidad o preferencias de ubicación para el recurso.



NOTA

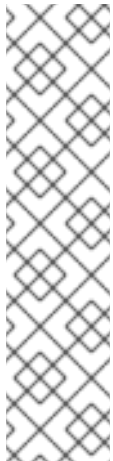
Una actualización manual sólo funcionará con nodos remotos si primero ejecuta los siguientes comandos:

1. Ejecute el comando **systemctl stop pacemaker_remote** en el nodo remoto para detener el nodo.
2. Ejecute el **pcs resource disable remote-connection-resource** comando.

A continuación, puede realizar una actualización manual en el nodo remoto.

shutdown-lock-limit

Cuando esta propiedad del clúster se establece en un tiempo diferente al valor predeterminado de 0, los recursos estarán disponibles para la recuperación en otros nodos si el nodo no se reincorpora dentro del tiempo especificado desde que se inició el apagado.



NOTA

La propiedad **shutdown-lock-limit** funcionará con nodos remotos sólo si primero ejecuta los siguientes comandos:

1. Ejecute el comando **systemctl stop pacemaker_remote** en el nodo remoto para detener el nodo.
2. Ejecute el **pcs resource disable remote-connection-resource** comando.

Después de ejecutar estos comandos, los recursos que se habían estado ejecutando en el nodo remoto estarán disponibles para su recuperación en otros nodos cuando haya transcurrido la cantidad de tiempo especificada en **shutdown-lock-limit**.

22.2. ESTABLECIMIENTO DE LA PROPIEDAD DE CLUSTER SHUTDOWN-LOCK

El siguiente ejemplo establece la propiedad de clúster **shutdown-lock** a **true** en un clúster de ejemplo y muestra el efecto que esto tiene cuando el nodo se apaga y se inicia de nuevo. Este clúster de ejemplo consta de tres nodos: **z1.example.com**, **z2.example.com**, y **z3.example.com**.

1. Establezca la propiedad **shutdown-lock** en **true** y verifique su valor. En este ejemplo la propiedad **shutdown-lock-limit** mantiene su valor por defecto de 0.

```
[root@z3.example.com ~]# pcs property set shutdown-lock=true
[root@z3.example.com ~]# pcs property list --all | grep shutdown-lock
shutdown-lock: true
shutdown-lock-limit: 0
```

2. Compruebe el estado del clúster. En este ejemplo, los recursos **third** y **fifth** se están ejecutando en **z1.example.com**.

```
[root@z3.example.com ~]# pcs status
...
Full List of Resources:
...
* first (ocf::pacemaker:Dummy): Started z3.example.com
* second (ocf::pacemaker:Dummy): Started z2.example.com
* third (ocf::pacemaker:Dummy): Started z1.example.com
* fourth (ocf::pacemaker:Dummy): Started z2.example.com
* fifth (ocf::pacemaker:Dummy): Started z1.example.com
...
```

3. Apagar **z1.example.com**, lo que detendrá los recursos que se están ejecutando en ese nodo.

```
[root@z3.example.com ~] # pcs cluster stop z1.example.com
Stopping Cluster (pacemaker)...
Stopping Cluster (corosync)...
```

4. La ejecución del comando **pcs status** muestra que el nodo **z1.example.com** está fuera de línea y que los recursos que se habían estado ejecutando en **z1.example.com** son **LOCKED** mientras el nodo está fuera de línea.

```
[root@z3.example.com ~]# pcs status
...

Node List:
* Online: [ z2.example.com z3.example.com ]
* OFFLINE: [ z1.example.com ]

Full List of Resources:
...
* first (ocf::pacemaker:Dummy): Started z3.example.com
* second (ocf::pacemaker:Dummy): Started z2.example.com
* third (ocf::pacemaker:Dummy): Stopped z1.example.com (LOCKED)
* fourth (ocf::pacemaker:Dummy): Started z3.example.com
* fifth (ocf::pacemaker:Dummy): Stopped z1.example.com (LOCKED)
...
```

5. Inicie de nuevo los servicios del clúster en **z1.example.com** para que se reincorpore al clúster. Los recursos bloqueados deberían iniciarse en ese nodo, aunque una vez iniciados no necesariamente permanecerán en el mismo nodo.

```
[root@z3.example.com ~]# pcs cluster start z1.example.com
Starting Cluster...
```

6. En este ejemplo, los recursos **third** y **fifth** se recuperan en el nodo **z1.example.com**.

```
[root@z3.example.com ~]# pcs status
...

Node List:
* Online: [ z1.example.com z2.example.com z3.example.com ]

Full List of Resources:
..
* first (ocf::pacemaker:Dummy): Started z3.example.com
* second (ocf::pacemaker:Dummy): Started z2.example.com
* third (ocf::pacemaker:Dummy): Started z1.example.com
* fourth (ocf::pacemaker:Dummy): Started z3.example.com
* fifth (ocf::pacemaker:Dummy): Started z1.example.com
...
```

CAPÍTULO 23. CONFIGURAR UNA ESTRATEGIA DE COLOCACIÓN DE NODOS

Pacemaker decide dónde colocar un recurso según las puntuaciones de asignación de recursos en cada nodo. El recurso se asignará al nodo en el que tenga la mayor puntuación. Esta puntuación de asignación se deriva de una combinación de factores, entre los que se incluyen las restricciones de recursos, la configuración de **resource-stickiness**, el historial de fallos anteriores de un recurso en cada nodo y la utilización de cada nodo.

Si las puntuaciones de asignación de recursos en todos los nodos son iguales, por la estrategia de colocación por defecto Pacemaker elegirá un nodo con el menor número de recursos asignados para equilibrar la carga. Si el número de recursos en cada nodo es igual, se elegirá el primer nodo elegible listado en el CIB para ejecutar el recurso.

Sin embargo, a menudo los diferentes recursos utilizan proporciones significativamente diferentes de las capacidades de un nodo (como la memoria o la E/S). No siempre se puede equilibrar la carga de forma ideal teniendo en cuenta sólo el número de recursos asignados a un nodo. Además, si los recursos se colocan de forma que sus requisitos combinados superen la capacidad proporcionada, es posible que no se inicien completamente o que funcionen con un rendimiento degradado. Para tener en cuenta estos factores, Pacemaker permite configurar los siguientes componentes:

- la capacidad que proporciona un nodo concreto
- la capacidad que requiere un determinado recurso
- una estrategia global de colocación de recursos

23.1. ATRIBUTOS DE UTILIZACIÓN Y ESTRATEGIA DE COLOCACIÓN

Para configurar la capacidad que proporciona un nodo o que requiere un recurso, puedes utilizar *utilization attributes* para nodos y recursos. Esto se hace estableciendo una variable de utilización para un recurso y asignando un valor a esa variable para indicar lo que el recurso requiere, y luego estableciendo esa misma variable de utilización para un nodo y asignando un valor a esa variable para indicar lo que ese nodo proporciona.

Puede nombrar los atributos de utilización según sus preferencias y definir tantos pares de nombre y valor como necesite su configuración. Los valores de los atributos de utilización deben ser enteros.

23.1.1. Configuración de la capacidad de nodos y recursos

El siguiente ejemplo configura un atributo de utilización de la capacidad de la CPU para dos nodos, estableciendo este atributo como la variable **cpu**. También configura un atributo de utilización de la capacidad de RAM, estableciendo este atributo como la variable **memory**. En este ejemplo:

- El nodo 1 se define por tener una capacidad de CPU de dos y una capacidad de RAM de 2048
- El nodo 2 se define por ofrecer una capacidad de CPU de cuatro y una capacidad de RAM de 2048

```
# pcs node utilization node1 cpu=2 memory=2048
# pcs node utilization node2 cpu=4 memory=2048
```

El siguiente ejemplo especifica los mismos atributos de utilización que requieren tres recursos diferentes. En este ejemplo:

- recurso **dummy-small** requiere una capacidad de CPU de 1 y una capacidad de RAM de 1024
- recurso **dummy-medium** requiere una capacidad de CPU de 2 y una capacidad de RAM de 2048
- recurso **dummy-large** requiere una capacidad de CPU de 1 y una capacidad de RAM de 3072

```
# pcs resource utilization dummy-small cpu=1 memory=1024
# pcs resource utilization dummy-medium cpu=2 memory=2048
# pcs resource utilization dummy-large cpu=3 memory=3072
```

Se considera que un nodo es elegible para un recurso si tiene suficiente capacidad libre para satisfacer los requisitos del recurso, definidos por los atributos de utilización.

23.1.2. Configuración de la estrategia de colocación

Después de haber configurado las capacidades que proporcionan sus nodos y las capacidades que requieren sus recursos, es necesario establecer la propiedad del clúster **placement-strategy**, de lo contrario las configuraciones de capacidad no tienen ningún efecto.

Hay cuatro valores disponibles para la propiedad **placement-strategy** cluster:

- **default**
- **utilization**
- **balanced**
- **minimal**

El siguiente comando de ejemplo establece el valor de **placement-strategy** en **balanced**. Después de ejecutar este comando, Pacemaker se asegurará de que la carga de sus recursos se distribuya uniformemente por todo el clúster, sin necesidad de complicados conjuntos de restricciones de colocación.

```
# pcs property set placement-strategy=balanced
```

23.2. ASIGNACIÓN DE RECURSOS DEL MARCAPASOS

Las siguientes subsecciones resumen cómo Pacemaker asigna los recursos.

23.2.1. Preferencia de nodo

Pacemaker determina qué nodo tiene preferencia a la hora de asignar recursos según la siguiente estrategia.

- El nodo con el mayor peso de nodo se consume primero. El peso del nodo es una puntuación mantenida por el clúster para representar la salud del nodo.
- Si varios nodos tienen el mismo peso de nodo:
 - Si la propiedad del cluster **placement-strategy** es **default** o **utilization**:
 - El nodo que tiene el menor número de recursos asignados se consume primero.

- Si el número de recursos asignados es igual, el primer nodo elegible que aparece en la CIB se consume primero.
- Si la propiedad del cluster **placement-strategy** es **balanced**:
 - El nodo que tiene más capacidad libre se consume primero.
 - Si las capacidades libres de los nodos son iguales, el nodo que tiene el menor número de recursos asignados se consume primero.
 - Si las capacidades libres de los nodos son iguales y el número de recursos asignados es igual, el primer nodo elegible que aparece en el CIB se consume primero.
- Si la propiedad del cluster **placement-strategy** es **minimal**, el primer nodo elegible listado en el CIB se consume primero.

23.2.2. Capacidad del nodo

Pacemaker determina qué nodo tiene más capacidad libre según la siguiente estrategia.

- Si sólo se ha definido un tipo de atributo de utilización, la capacidad libre es una simple comparación numérica.
- Si se han definido varios tipos de atributos de utilización, entonces el nodo que es numéricamente más alto en la mayoría de los tipos de atributos tiene la mayor capacidad libre. Por ejemplo:
 - Si el NodoA tiene más CPUs libres, y el NodoB tiene más memoria libre, entonces sus capacidades libres son iguales.
 - Si el NodoA tiene más CPUs libres, mientras que el NodoB tiene más memoria y almacenamiento libres, entonces el NodoB tiene más capacidad libre.

23.2.3. Preferencia de asignación de recursos

Pacemaker determina qué recurso se asigna primero según la siguiente estrategia.

- El recurso que tiene la mayor prioridad se asigna primero. Puede establecer la prioridad de un recurso cuando lo crea.
- Si las prioridades de los recursos son iguales, el recurso que tiene la puntuación más alta en el nodo en el que se está ejecutando se asigna primero, para evitar que se barajen los recursos.
- Si las puntuaciones de los recursos en los nodos donde se ejecutan los recursos son iguales o los recursos no se ejecutan, el recurso que tiene la puntuación más alta en el nodo preferido se asigna primero. Si las puntuaciones de los recursos en el nodo preferido son iguales en este caso, se asigna primero el primer recurso ejecutable listado en el CIB.

23.3. DIRECTRICES DE LA ESTRATEGIA DE COLOCACIÓN DE RECURSOS

Para garantizar que la estrategia de colocación de recursos de Pacemaker funcione de la manera más eficaz, debe tener en cuenta las siguientes consideraciones al configurar su sistema.

- Asegúrate de que tienes suficiente capacidad física.

Si la capacidad física de sus nodos se está utilizando casi al máximo en condiciones normales, podrían producirse problemas durante la conmutación por error. Incluso sin la función de utilización, puede empezar a experimentar tiempos de espera y fallos secundarios.

- Incorpore un poco de amortiguación en las capacidades que configure para los nodos. Anuncie ligeramente más recursos de nodo de los que tiene físicamente, asumiendo que un recurso de Pacemaker no utilizará el 100% de la cantidad configurada de CPU, memoria, etc. todo el tiempo. Esta práctica se denomina a veces overcommit.
- Especificar las prioridades de los recursos.
Si el clúster va a sacrificar servicios, deberían ser los que menos te importan. Asegúrate de que las prioridades de los recursos están bien configuradas para que tus recursos más importantes se programen primero.

23.4. EL AGENTE DE RECURSOS NODEUTILIZATION

El agente NodeUtilization puede detectar los parámetros del sistema de la CPU disponible, la disponibilidad de la memoria del host y la disponibilidad de la memoria del hipervisor y añadir estos parámetros en el CIB. Puede ejecutar el agente como un recurso clonado para que rellene automáticamente estos parámetros en cada nodo.

Para obtener información sobre el agente de recursos **NodeUtilization** y las opciones de recursos para este agente, ejecute el comando **pcs resource describe NodeUtilization**.

CAPÍTULO 24. CONFIGURACIÓN DE UN DOMINIO VIRTUAL COMO RECURSO

Puede configurar un dominio virtual gestionado por el marco de virtualización **libvirt** como recurso de clúster con el comando **pcs resource create**, especificando **VirtualDomain** como tipo de recurso.

Al configurar un dominio virtual como recurso, tenga en cuenta las siguientes consideraciones:

- Un dominio virtual debe ser detenido antes de configurarlo como recurso de cluster.
- Una vez que un dominio virtual es un recurso del clúster, no debe iniciarse, detenerse o migrar, excepto a través de las herramientas del clúster.
- No configure un dominio virtual que haya configurado como recurso de cluster para que se inicie cuando su host arranque.
- Todos los nodos autorizados a ejecutar un dominio virtual deben tener acceso a los archivos de configuración y dispositivos de almacenamiento necesarios para ese dominio virtual.

Si desea que el clúster gestione servicios dentro del propio dominio virtual, puede configurar el dominio virtual como nodo invitado.

24.1. OPCIONES DE RECURSOS DEL DOMINIO VIRTUAL

Tabla 24.1, “Opciones de recursos para los recursos del dominio virtual” describe las opciones de recursos que puede configurar para un recurso **VirtualDomain**.

Tabla 24.1. Opciones de recursos para los recursos del dominio virtual

C a m p o	Por defecto	D e s c r i p c i ó n
c o n f i g		(o b l i g a t o r i o) R

C a m p o	Por defecto	D e s c r i p c i ó n
		I a r c h i v o d e c o n f i g u r a c i ó n l i b v i r t p a r a e s t e d o m i n i

C a m p o	Por defecto	D e s c r i p c i ó n
h y p e r v i s o r	Depende del sistema	R l d e l h i p e r v i s o r a l q u e c o n e c t a r s e . P u e d e d e t e r m i n

C a m p o	Por defecto	D e s c r i p c i ó n
		f e c t o d e l s i s t e m a e j e c u t a n d o e l c o m a n d o v i r s h - - q u i

C a m p o	Por defecto	D e s c r i
f o r c e	0	p c i o n p
- s t o p		r e s e a p a g a l a f u e r z a (\ "d e s t r u i r") e l d o m i n i o a l d e

C a m p o	Por defecto	D e s c r i p c i o n
		m p o r t a m i e n t o p o r d e f e c t o e s r e c u r r i r a u n a p a g a d o f o

C a m p o	Por defecto	D e s c r i p c i ó n
		P u é s d e q u e u n i n t e n t o d e a p a g a d o g r a c i o s o h a y a f a l l a

C a m p o	Por defecto	D e s c r i p c i ó n
		l e c e r e s t a o p c i ó n e n t r u e s ó l o s i s u d o m i n i o v i r t u a l (

C a m p o	Por defecto	D e s c r i p c i ó n
		e v i r t u a l i z a c i ó n) n o s o p o r t a e l a p a g a d o g r a c e f u l .

m C a m p o i o n - t r a n s p o r t	Depende del sistema Por defecto	D e s c r i p c i o n
		I l i z a d o p a r a c o n e c t a r s e a l h i p e r v i s o r r e m o t o d u r a n

C a m p o	Por defecto	D e s c r i p c i ó n
		n . S i s e o m i t e e s t e p a r á m e t r o , e l r e c u r s o u t i l i z a r á e

C a m p o	Por defecto	D e s c r i p c i ó n
		r e d e t e r m i n a d o d e l i b v i r t p a r a c o n e c t a r s e a l h i p e r v

C a m p o	Por defecto	D e s c r i p c i ó n
m i g r a t i o n _ n e t w o r k _ s u f f i x		U t i l i z a r u n a r e d d e m i g r a c i ó n d e d i c a d a · L a U R I d e m i g

C a m p o	Por defecto	D e s c r i p c i ó n
		o n e a ñ a d i e n d o e l v a l o r d e e s t e p a r á m e t r o a l f i n a l d e l

C a m p o	Por defecto	D e s c r i p c i ó n
		o . S i e l n o m b r e d e l n o d o e s u n n o m b r e d e d o m i n i o c o m p l e

C a m p o	Por defecto	D e s c r i p c i ó n
		e r t e e l s u f i j o i n m e d i a t a m e n t e a n t e s d e l p r i m e r p u n t

C a m p o	Por defecto	D e s c r i p c i ó n
		· A s e g ú r e s e d e q u e e s t e n o m b r e d e h o s t c o m p u e s t o s e p u

C a m p o	Por defecto	D e s c r i p c i ó n
		o c a l m e n t e y q u e l a d i r e c c i ó n l p a s o c i a d a e s a c c e s i b l

C a m p o	Por defecto	D e s c r i p c i ó n
		r e d f a v o r e c i d a .
m o n i t o r - s c r i p t s		P a r a s u p e r v i s a r a d i c i o n a l m e n t e l

C a m p o	Por defecto	D e s c r i p c i ó n
		e n t r o d e l d o m i n i o v i r t u a l , a ñ a d a e s t e p a r á m e t r o c o n

C a m p o	Por defecto	D e s c r i p c i o n
		r i p t s a s u p e r v i s a r . <i>Note</i> : C u a n d o s e u t i l i z a n s c r i p t

C a m p o	Por defecto	D e s c r i p c i o n
		o n / l a s o p e r a c i o n e s s t a r t y m i g r a t e - f r o m s ó l o s e c o m

C a m p o	Por defecto	D e s c r i p c i ó n
		d o t o d o s l o s s c r i p t s d e s u p e r v i s i ó n h a y a n f i n a l i z a d

C a m p o	Por defecto	D e s c r i p c i ó n
		e g ú r e s e d e e s t a b l e c e r e l t i e m p o d e e s p e r a d e e s t a s o p

C a m p o	Por defecto	D e s c r i p c i o n
		a a c o m o d a r e s t e r e t r a s o
a u t o s e t _ u t i l i z a t i o n _	true	S i s e e s t a b l e c e n t r u e , e l

Cam po	Por defecto	De s c r i p c i ó n
		a r á e l n ú m e r o d e d o m a i n U , s v C P U s d e v i r s h , y l o p o n d r á

C a m p o	Por defecto	D e s c r i p c i ó n
		l ó n d e l a C P U d e l r e c u r s o c u a n d o s e e j e c u t e e l m o n i t

C a m p o s e t _ u t i	Por defecto true	D e s c r i p c i ó n
l i z a t i o n _ h v _ m e m o r y		c e c o m o v e r d a d e r o , e l a g e n t e d e t e c t a r á e l n ú m e r o d e M

C a m p o	Por defecto	a D e s c r i p c i ó n
		r s h , y l o p o n d r á e

C a m p o	Por defecto	D e s c r i p c i ó n
		J e c u t e e l m o n i t o r .
m i g r a t e p o r t	puerto alto aleatorio	E s t e p u e r t o s e u t i l i z a r á e n e l U

C a m p o	Por defecto	D e s c r i p c i o n
		n d e q e m u · S i n o s e e s t a b l e c e , e l p u e r t o s e r á u n p u e r t o

C a m p o	Por defecto	D e s c r i p c i ó n
		o
s n a p s h o t		R u t a a l d i r e c t o r i o d e i n s t a n t á n e a s d o n d e s e a l m a

C a m p o	Por defecto	D e s c r i p c i ó n
		e n d e l a m á q u i n a v i r t u a l . C u a n d o s e e s t a b l e c e s t e p a

C a m p o	Por defecto	D e s c r i p c i ó n
		t a d o R A M d e l a m á q u i n a v i r t u a l s e g u a r d a r á e n u n a r c h i

C a m p o	Por defecto	D e s c r i p c i o n
		O r d e i n s t a n t á n e a s c u a n d o s e d e t e n g a . S i a l i n i c i a

C a m p o	Por defecto	D e s c r i p c i o n
		I v o d e e s t a d o p a r a e l d o m i n i o , é s t e s e r e s t a u r a r á a l m

C a m p o	Por defecto	D e s c r i p c i o n
		e l q u e e s t a b a j u s t o a n t e s d e d e t e n e r s e p o r ú l t i m a v e z

C a m p o	Por defecto	D e s c r i p c i ó n
		S i n c o m p a t i b l e c o n l a o p c i ó n f o r c e _ s t o p .

Además de las opciones del recurso **VirtualDomain**, puede configurar la opción de metadatos **allow-migrate** para permitir la migración en vivo del recurso a otro nodo. Cuando esta opción se establece en **true**, el recurso puede ser migrado sin pérdida de estado. Cuando esta opción se establece en **false**, que es el estado por defecto, el dominio virtual se apagará en el primer nodo y luego se reiniciará en el segundo nodo cuando se mueva de un nodo a otro.

24.2. CREACIÓN DEL RECURSO DE DOMINIO VIRTUAL

Utilice el siguiente procedimiento para crear un recurso **VirtualDomain** en un clúster para una máquina virtual que haya creado previamente:

1. Para crear el agente de recursos **VirtualDomain** para la gestión de la máquina virtual, Pacemaker requiere que el archivo de configuración **xml** de la máquina virtual se vuelque a un archivo en el disco. Por ejemplo, si creó una máquina virtual llamada **guest1**, vuelque el archivo **xml** a un archivo en algún lugar de uno de los nodos del clúster que podrá ejecutar el huésped. Puede utilizar un nombre de archivo de su elección; este ejemplo utiliza **/etc/pacemaker/guest1.xml**.

```
# virsh dumpxml guest1 > /etc/pacemaker/guest1.xml
```

2. Copie el archivo de configuración **xml** de la máquina virtual a todos los demás nodos del clúster que podrán ejecutar el invitado, en la misma ubicación de cada nodo.
3. Asegúrese de que todos los nodos autorizados a ejecutar el dominio virtual tienen acceso a los dispositivos de almacenamiento necesarios para ese dominio virtual.
4. Compruebe por separado que el dominio virtual puede iniciarse y detenerse en cada nodo que ejecutará el dominio virtual.
5. Si se está ejecutando, apague el nodo invitado. Pacemaker iniciará el nodo cuando esté configurado en el cluster. La máquina virtual no debe configurarse para que se inicie automáticamente al arrancar el host.
6. Configure el recurso **VirtualDomain** con el comando **pcs resource create**. Por ejemplo, el siguiente comando configura un recurso **VirtualDomain** llamado **VM**. Dado que la opción **allow-migrate** está configurada como **true** un **pcs move VM nodeX** comando se haría como una migración en vivo.

En este ejemplo **migration_transport** está configurado como **ssh**. Tenga en cuenta que para que la migración SSH funcione correctamente, el registro sin clave debe funcionar entre nodos.

```
# pcs resource create VM VirtualDomain config=/etc/pacemaker/guest1.xml  
migration_transport=ssh meta allow-migrate=true
```


CAPÍTULO 25. QUÓRUM DEL CLÚSTER

Un cluster de Red Hat Enterprise Linux High Availability Add-On utiliza el servicio **votequorum**, en conjunto con fencing, para evitar situaciones de split brain. Se asigna un número de votos a cada sistema en el cluster y las operaciones del cluster se permiten sólo cuando hay una mayoría de votos. El servicio debe cargarse en todos los nodos o en ninguno; si se carga en un subconjunto de nodos del clúster, los resultados serán imprevisibles. Para obtener información sobre la configuración y el funcionamiento del servicio **votequorum**, consulte la página de manual **votequorum(5)**.

25.1. CONFIGURACIÓN DE LAS OPCIONES DE QUÓRUM

Hay algunas características especiales de la configuración del quórum que puede establecer cuando crea un clúster con el comando **pcs cluster setup**. [Tabla 25.1, "Opciones de quórum"](#) resume estas opciones.

Tabla 25.1. Opciones de quórum

Opción	Descripción
auto_tie_breaker	<p>Cuando se activa, el clúster puede sufrir hasta el 50% de los nodos que fallan al mismo tiempo, de forma determinista. La partición del clúster, o el conjunto de nodos que aún están en contacto con el nodeid configurado en auto_tie_breaker_node (o el nodeid más bajo si no se ha configurado), permanecerá en quórum. Los demás nodos estarán sin quórum.</p> <p>La opción auto_tie_breaker se utiliza principalmente para clústeres con un número par de nodos, ya que permite que el clúster siga funcionando con una división uniforme. Para fallos más complejos, como divisiones múltiples y desiguales, se recomienda utilizar un dispositivo de quórum, como se describe en Dispositivos de quórum.</p> <p>La opción auto_tie_breaker es incompatible con los dispositivos de quórum.</p>

Opción	Descripción
wait_for_all	<p>Cuando se habilita, el clúster será quórum por primera vez sólo después de que todos los nodos hayan sido visibles al menos una vez al mismo tiempo.</p> <p>La opción wait_for_all se utiliza principalmente para clústeres de dos nodos y para clústeres de nodos pares que utilizan el dispositivo de quórum lms (último hombre en pie).</p> <p>La opción wait_for_all se activa automáticamente cuando un clúster tiene dos nodos, no utiliza un dispositivo de quórum y auto_tie_breaker está desactivado. Puede anular esto estableciendo explícitamente wait_for_all a 0.</p>
last_man_standing	<p>Cuando se activa, el clúster puede recalcular dinámicamente expected_votes y el quórum en circunstancias específicas. Debe habilitar wait_for_all cuando habilite esta opción. La opción last_man_standing es incompatible con los dispositivos de quórum.</p>
last_man_standing_window	<p>El tiempo, en milisegundos, que se debe esperar antes de recalcular expected_votes y el quórum después de que un clúster pierda nodos.</p>

Para más información sobre la configuración y el uso de estas opciones, consulte la página de manual **votequorum**(5).

25.2. MODIFICACIÓN DE LAS OPCIONES DE QUÓRUM

Puede modificar las opciones generales de quórum de su clúster con el comando **pcs quorum update**. La ejecución de este comando requiere que el clúster esté detenido. Para obtener información sobre las opciones de quórum, consulte la página de manual **votequorum**(5).

El formato del comando **pcs quorum update** es el siguiente.

```
pcs actualización del quórum [auto_tie_breaker=[0|1]] [last_man_standing=[0|1]]
[last_man_standing_window=time-in-ms] [wait_for_all=[0|1]]
```

La siguiente serie de comandos modifica la opción de quórum **wait_for_all** y muestra el estado actualizado de la opción. Tenga en cuenta que el sistema no permite ejecutar este comando mientras el clúster está en funcionamiento.

```
[root@node1:~]# pcs quorum update wait_for_all=1
Checking corosync is not running on nodes...
Error: node1: corosync is running
Error: node2: corosync is running
```

```
[root@node1:~]# pcs cluster stop --all
```

```

node2: Stopping Cluster (pacemaker)...
node1: Stopping Cluster (pacemaker)...
node1: Stopping Cluster (corosync)...
node2: Stopping Cluster (corosync)...

[root@node1:~]# pcs quorum update wait_for_all=1
Checking corosync is not running on nodes...
node2: corosync is not running
node1: corosync is not running
Sending updated corosync.conf to nodes...
node1: Succeeded
node2: Succeeded

[root@node1:~]# pcs quorum config
Options:
  wait_for_all: 1

```

25.3. VISUALIZACIÓN DE LA CONFIGURACIÓN Y EL ESTADO DEL QUÓRUM

Una vez que el clúster está funcionando, puede introducir los siguientes comandos de quórum del clúster.

El siguiente comando muestra la configuración del quórum.

```
pcs quorum [config]
```

El siguiente comando muestra el estado de ejecución del quórum.

```
estado del quórum del pcs
```

25.4. EJECUCIÓN DE CLÚSTERES INQUEMADOS

Si retira nodos de un clúster durante un largo período de tiempo y la pérdida de esos nodos provocaría la pérdida de quórum, puede cambiar el valor del parámetro **expected_votes** para el clúster en vivo con el comando **pcs quorum expected-votes**. Esto permite que el clúster siga funcionando cuando no tiene quórum.



AVISO

El cambio de los votos esperados en un clúster en vivo debe hacerse con extrema precaución. Si menos del 50% del clúster se está ejecutando porque has cambiado manualmente los votos esperados, entonces los otros nodos del clúster podrían iniciarse por separado y ejecutar los servicios del clúster, causando corrupción de datos y otros resultados inesperados. Si cambia este valor, debe asegurarse de que el parámetro **wait_for_all** está activado.

El siguiente comando establece los votos esperados en el cluster en vivo al valor especificado. Esto afecta sólo al clúster en vivo y no cambia el archivo de configuración; el valor de **expected_votes** se restablece al valor del archivo de configuración en caso de recarga.

```
pcs quorum esperado-votos votes
```

En una situación en la que usted sabe que el clúster está inerte pero quiere que el clúster proceda a la gestión de recursos, puede utilizar el comando **pcs quorum unblock** para evitar que el clúster espere a todos los nodos al establecer el quórum.



NOTA

Este comando debe utilizarse con extrema precaución. Antes de emitir este comando, es imprescindible asegurarse de que los nodos que no están actualmente en el clúster están apagados y no tienen acceso a los recursos compartidos.

```
# pcs quorum unblock
```

25.5. DISPOSITIVOS DE QUÓRUM

Puede permitir que un clúster soporte más fallos de nodo de los que permiten las reglas de quórum estándar configurando un dispositivo de quórum independiente que actúe como dispositivo de arbitraje de terceros para el clúster. Se recomienda un dispositivo de quórum para clusters con un número par de nodos. Con clusters de dos nodos, el uso de un dispositivo de quórum puede determinar mejor qué nodo sobrevive en una situación de cerebro dividido.

Al configurar un dispositivo de quórum hay que tener en cuenta lo siguiente.

- Se recomienda que un dispositivo de quórum se ejecute en una red física diferente en el mismo sitio que el clúster que utiliza el dispositivo de quórum. Lo ideal es que el host del dispositivo de quórum esté en un rack distinto al del clúster principal, o al menos en una fuente de alimentación distinta y no en el mismo segmento de red que el anillo o anillos de corosync.
- No se puede utilizar más de un dispositivo de quórum en un clúster al mismo tiempo.
- Aunque no se puede utilizar más de un dispositivo de quórum en un clúster al mismo tiempo, un único dispositivo de quórum puede ser utilizado por varios clústeres al mismo tiempo. Cada clúster que utilice ese dispositivo de quórum puede utilizar diferentes algoritmos y opciones de quórum, ya que éstos se almacenan en los propios nodos del clúster. Por ejemplo, un único dispositivo de quórum puede ser utilizado por un cluster con un algoritmo **ffsplit** (fifty/fifty split) y por un segundo cluster con un algoritmo **lms** (last man standing).
- Un dispositivo de quórum no debe ejecutarse en un nodo de clúster existente.

25.5.1. Instalación de paquetes de dispositivos de quórum

La configuración de un dispositivo de quórum para un clúster requiere la instalación de los siguientes paquetes:

- Instale **corosync-qdevice** en los nodos de un clúster existente.

```
[root@node1:~]# yum install corosync-qdevice
[root@node2:~]# yum install corosync-qdevice
```

- Instale **pcs** y **corosync-qnetd** en el host del dispositivo de quórum.

```
[root@qdevice:~]# yum install pcs corosync-qnetd
```

- Inicie el servicio **pcsd** y habilite **pcsd** al iniciar el sistema en el host del dispositivo de quórum.

```
[root@qdevice:~]# systemctl start pcsd.service
[root@qdevice:~]# systemctl enable pcsd.service
```

25.5.2. Configuración de un dispositivo de quórum

El siguiente procedimiento configura un dispositivo de quórum y lo añade al clúster. En este ejemplo:

- El nodo utilizado para un dispositivo de quórum es **qdevice**.
- El modelo de dispositivo de quórum es **net**, que es actualmente el único modelo soportado. El modelo **net** admite los siguientes algoritmos:
 - **ffsplit**: división al cincuenta por ciento. Esto proporciona exactamente un voto a la partición con el mayor número de nodos activos.
 - **lms**: último hombre en pie. Si el nodo es el único que queda en el clúster que puede ver el servidor **qnetd**, entonces devuelve un voto.



AVISO

El algoritmo LMS permite que el clúster siga siendo quórum incluso con un solo nodo restante, pero también significa que el poder de voto del dispositivo de quórum es grande, ya que es igual al número_de_nodos - 1. Perder la conexión con el dispositivo de quórum significa perder el número_de_nodos - 1 de votos, lo que significa que sólo un clúster con todos los nodos activos puede seguir siendo quórum (sobrevotando al dispositivo de quórum); cualquier otro clúster se convierte en no quórum.

Para obtener información más detallada sobre la implementación de estos algoritmos, consulte la página de manual **corosync-qdevice(8)**.

- Los nodos del clúster son **node1** y **node2**.

El siguiente procedimiento configura un dispositivo de quórum y añade ese dispositivo de quórum a un clúster.

1. En el nodo que utilizará para alojar su dispositivo de quórum, configure el dispositivo de quórum con el siguiente comando. Este comando configura e inicia el modelo de dispositivo de quórum **net** y configura el dispositivo para que se inicie al arrancar.

```
[root@qdevice:~]# pcs qdevice setup model net --enable --start
Quorum device 'net' initialized
quorum device enabled
```

```
Starting quorum device...
quorum device started
```

Después de configurar el dispositivo de quórum, puedes comprobar su estado. Esto debería mostrar que el demonio **corosync-qnetd** se está ejecutando y, en este momento, no hay clientes conectados a él. La opción del comando **--full** proporciona una salida detallada.

```
[root@qdevice:~]# pcs qdevice status net --full
QNetd address:      *:5403
TLS:                Supported (client certificate required)
Connected clients: 0
Connected clusters: 0
Maximum send/receive size: 32768/32768 bytes
```

- Habilite los puertos del cortafuegos necesarios para el demonio **pcsd** y el dispositivo de quórum **net** habilitando el servicio **high-availability** en **firewalld** con los siguientes comandos.

```
[root@qdevice:~]# firewall-cmd --permanent --add-service=high-availability
[root@qdevice:~]# firewall-cmd --add-service=high-availability
```

- Desde uno de los nodos del clúster existente, autentique al usuario **hacluster** en el nodo que aloja el dispositivo de quórum. Esto permite que **pcs** en el clúster se conecte a **pcs** en el host **qdevice**, pero no permite que **pcs** en el host **qdevice** se conecte a **pcs** en el clúster.

```
[root@node1:~] # pcs host auth qdevice
Username: hacluster
Password:
qdevice: Authorized
```

- Añade el dispositivo de quórum al clúster.

Antes de añadir el dispositivo de quórum, puede comprobar la configuración actual y el estado del dispositivo de quórum para su posterior comparación. La salida de estos comandos indica que el clúster aún no está utilizando un dispositivo de quórum, y el estado de pertenencia a **Qdevice** para cada nodo es **NR** (No registrado).

```
[root@node1:~]# pcs quorum config
Options:
```

```
[root@node1:~]# pcs quorum status
Quorum information
-----
Date:          Wed Jun 29 13:15:36 2016
Quorum provider: corosync_votequorum
Nodes:        2
Node ID:      1
Ring ID:      1/8272
Quorate:     Yes
```

```
Votequorum information
-----
Expected votes: 2
Highest expected: 2
Total votes:    2
Quorum:        1
```

```
Flags:          2Node Quorate
```

Membership information

```
-----
```

Nodeid	Votes	Qdevice Name
1	1	NR node1 (local)
2	1	NR node2

El siguiente comando añade al clúster el dispositivo de quórum que ha creado previamente. No se puede utilizar más de un dispositivo de quórum en un clúster al mismo tiempo. Sin embargo, un dispositivo de quórum puede ser utilizado por varios clústeres al mismo tiempo. Este comando de ejemplo configura el dispositivo de quórum para utilizar el algoritmo **ffsplit**. Para obtener información sobre las opciones de configuración del dispositivo de quórum, consulte la página de manual **corosync-qdevice(8)**.

```
[root@node1:~]# pcs quorum device add model net host=qdevice \
algorithm=ffsplit
Setting up qdevice certificates on nodes...
node2: Succeeded
node1: Succeeded
Enabling corosync-qdevice...
node1: corosync-qdevice enabled
node2: corosync-qdevice enabled
Sending updated corosync.conf to nodes...
node1: Succeeded
node2: Succeeded
Corosync configuration reloaded
Starting corosync-qdevice...
node1: corosync-qdevice started
node2: corosync-qdevice started
```

5. Compruebe el estado de configuración del dispositivo de quórum. Desde el lado del clúster, puede ejecutar los siguientes comandos para ver cómo ha cambiado la configuración.

En **pcs quorum config** se muestra el dispositivo de quórum que se ha configurado.

```
[root@node1:~]# pcs quorum config
Options:
Device:
  Model: net
  algorithm: ffsplit
  host: qdevice
```

El comando **pcs quorum status** muestra el estado de ejecución del quórum, indicando que el dispositivo de quórum está en uso. Los significados de los valores de estado de la información de membresía de **Qdevice** para cada nodo del clúster son los siguientes:

- **A/NA**
- **V/NV**
- **MW/NMW**

```
[root@node1:~]# pcs quorum status
```

Quorum information

```

-----
Date:      Wed Jun 29 13:17:02 2016
Quorum provider: corosync_votequorum
Nodes:     2
Node ID:   1
Ring ID:   1/8272
Quorate:   Yes

```

Votequorum information

```

-----
Expected votes: 3
Highest expected: 3
Total votes: 3
Quorum: 2
Flags: Quorate Qdevice

```

Membership information

```

-----
Nodeid  Votes  Qdevice Name
  1      1  A,V,NMW node1 (local)
  2      1  A,V,NMW node2
  0      1      Qdevice

```

La página **pcs quorum device status** muestra el estado de ejecución del dispositivo de quórum.

```
[root@node1:~]# pcs quorum device status
```

Qdevice information

```

-----
Model:      Net
Node ID:    1
Configured node list:
  0 Node ID = 1
  1 Node ID = 2
Membership node list: 1, 2

```

Qdevice-net information

```

-----
Cluster name:  mycluster
QNetd host:   qdevice:5403
Algorithm:    ffsplit
Tie-breaker:  Node with lowest node ID
State:        Connected

```

Desde el lado del dispositivo de quórum, puede ejecutar el siguiente comando de estado, que muestra el estado del demonio **corosync-qnetd**.

```
[root@qdevice:~]# pcs qdevice status net --full
```

```

QNetd address:  *:5403
TLS:           Supported (client certificate required)
Connected clients: 2
Connected clusters: 1
Maximum send/receive size: 32768/32768 bytes
Cluster "mycluster":

```



```

Algorithm:      ffsplit
Tie-breaker:   Node with lowest node ID
Node ID 2:
  Client address:  ::ffff:192.168.122.122:50028
  HB interval:    8000ms
  Configured node list:  1, 2
  Ring ID:       1.2050
  Membership node list:  1, 2
  TLS active:    Yes (client certificate verified)
  Vote:         ACK (ACK)
Node ID 1:
  Client address:  ::ffff:192.168.122.121:48786
  HB interval:    8000ms
  Configured node list:  1, 2
  Ring ID:       1.2050
  Membership node list:  1, 2
  TLS active:    Yes (client certificate verified)
  Vote:         ACK (ACK)

```

25.5.3. Gestión del servicio de dispositivos de quórum

PCS proporciona la capacidad de gestionar el servicio de dispositivo de quórum en el host local (**corosync-qnetd**), como se muestra en los siguientes comandos de ejemplo. Tenga en cuenta que estos comandos sólo afectan al servicio **corosync-qnetd**.

```

[root@qdevice:~]# pcs qdevice start net
[root@qdevice:~]# pcs qdevice stop net
[root@qdevice:~]# pcs qdevice enable net
[root@qdevice:~]# pcs qdevice disable net
[root@qdevice:~]# pcs qdevice kill net

```

25.5.4. Gestión de la configuración del dispositivo de quórum en un clúster

Las siguientes secciones describen los comandos PCS que puede utilizar para gestionar la configuración de los dispositivos de quórum en un clúster.

25.5.4.1. Cambiar la configuración del dispositivo de quórum

Puede cambiar la configuración de un dispositivo de quórum con el comando **pcs quorum device update**.



AVISO

Para cambiar la opción **host** del modelo de dispositivo de quórum **net**, utilice los comandos **pcs quorum device remove** y **pcs quorum device add** para establecer la configuración correctamente, a menos que el host antiguo y el nuevo sean la misma máquina.

El siguiente comando cambia el algoritmo del dispositivo de quórum a **lms**.

```
[root@node1:~]# pcs quorum device update model algorithm=lms
Sending updated corosync.conf to nodes...
node1: Succeeded
node2: Succeeded
Corosync configuration reloaded
Reloading qdevice configuration on nodes...
node1: corosync-qdevice stopped
node2: corosync-qdevice stopped
node1: corosync-qdevice started
node2: corosync-qdevice started
```

25.5.4.2. Eliminación de un dispositivo de quórum

Utilice el siguiente comando para eliminar un dispositivo de quórum configurado en un nodo del clúster.

```
[root@node1:~]# pcs quorum device remove
Sending updated corosync.conf to nodes...
node1: Succeeded
node2: Succeeded
Corosync configuration reloaded
Disabling corosync-qdevice...
node1: corosync-qdevice disabled
node2: corosync-qdevice disabled
Stopping corosync-qdevice...
node1: corosync-qdevice stopped
node2: corosync-qdevice stopped
Removing qdevice certificates from nodes...
node1: Succeeded
node2: Succeeded
```

Después de haber eliminado un dispositivo de quórum, debería ver el siguiente mensaje de error al mostrar el estado del dispositivo de quórum.

```
[root@node1:~]# pcs quorum device status
Error: Unable to get quorum status: corosync-qdevice-tool: Can't connect to QDevice socket (is QDevice running?): No such file or directory
```

25.5.4.3. Destrucción de un dispositivo de quórum

Para desactivar y detener un dispositivo de quórum en el host del dispositivo de quórum y eliminar todos sus archivos de configuración, utilice el siguiente comando.

```
[root@qdevice:~]# pcs qdevice destroy net
Stopping quorum device...
quorum device stopped
quorum device disabled
Quorum device 'net' configuration files removed
```

CAPÍTULO 26. SCRIPTS DE ACTIVACIÓN DE EVENTOS DE CLUSTER

Un clúster Pacemaker es un sistema impulsado por eventos, en el que un evento puede ser un fallo de un recurso o nodo, un cambio de configuración o el inicio o la detención de un recurso. Puede configurar las alertas de clúster de Pacemaker para que realicen alguna acción externa cuando se produzca un evento de clúster mediante agentes de alerta, que son programas externos a los que el clúster llama de la misma manera que el clúster llama a los agentes de recursos para gestionar la configuración y el funcionamiento de los recursos.

El clúster pasa información sobre el evento al agente mediante variables de entorno. Los agentes pueden hacer cualquier cosa con esta información, como enviar un mensaje de correo electrónico o registrar en un archivo o actualizar un sistema de monitorización.

- Pacemaker proporciona varios agentes de alerta de muestra, que se instalan en **/usr/share/pacemaker/alerts** por defecto. Estos scripts de muestra pueden copiarse y utilizarse tal cual, o bien pueden utilizarse como plantillas que pueden editarse para adaptarse a sus propósitos. Consulte el código fuente de los agentes de muestra para conocer el conjunto completo de atributos que admiten.
- Si los agentes de alerta de muestra no satisfacen sus necesidades, puede escribir sus propios agentes de alerta para que llame una alerta de Marcapasos.

26.1. INSTALACIÓN Y CONFIGURACIÓN DE AGENTES DE ALERTA DE MUESTRA

Cuando utilice uno de los agentes de alerta de muestra, deberá revisar el script para asegurarse de que se adapta a sus necesidades. Estos agentes de muestra se proporcionan como punto de partida para scripts personalizados para entornos de cluster específicos. Tenga en cuenta que mientras Red Hat soporta las interfaces que los scripts de agentes de alerta utilizan para comunicarse con Pacemaker, Red Hat no proporciona soporte para los agentes personalizados en sí mismos.

Para utilizar uno de los agentes de alerta de ejemplo, debe instalar el agente en cada nodo del clúster. Por ejemplo, el siguiente comando instala el script **alert_file.sh.sample** como **alert_file.sh**.

```
# install --mode=0755 /usr/share/pacemaker/alerts/alert_file.sh.sample
/var/lib/pacemaker/alert_file.sh
```

Una vez instalado el script, puede crear una alerta que utilice el script.

El siguiente ejemplo configura una alerta que utiliza el agente de alerta instalado **alert_file.sh** para registrar eventos en un archivo. Los agentes de alerta se ejecutan como el usuario **hacluster**, que tiene un conjunto mínimo de permisos.

Este ejemplo crea el archivo de registro **pcmk_alert_file.log** que se utilizará para registrar los eventos. A continuación, crea el agente de alerta y añade la ruta del archivo de registro como su destinatario.

```
# touch /var/log/pcmk_alert_file.log
# chown hacluster:haclient /var/log/pcmk_alert_file.log
# chmod 600 /var/log/pcmk_alert_file.log
# pcs alert create id=alert_file description="Log events to a file."
path=/var/lib/pacemaker/alert_file.sh
# pcs alert recipient add alert_file id=my-alert_logfile value=/var/log/pcmk_alert_file.log
```

El siguiente ejemplo instala el script **alert_snmp.sh.sample** como **alert_snmp.sh** y configura una alerta que utiliza el agente de alerta instalado **alert_snmp.sh** para enviar los eventos del cluster como traps SNMP. Por defecto, el script enviará todos los eventos excepto las llamadas de monitorización exitosas al servidor SNMP. Este ejemplo configura el formato de marca de tiempo como una opción meta. Después de configurar la alerta, este ejemplo configura un destinatario para la alerta y muestra la configuración de la alerta.

```
# install --mode=0755 /usr/share/pacemaker/alerts/alert_snmp.sh.sample
/var/lib/pacemaker/alert_snmp.sh
# pcs alert create id=snmp_alert path=/var/lib/pacemaker/alert_snmp.sh meta timestamp-
format="%Y-%m-%d,%H:%M:%S.%01N"
# pcs alert recipient add snmp_alert value=192.168.1.2
# pcs alert
Alerts:
Alert: snmp_alert (path=/var/lib/pacemaker/alert_snmp.sh)
Meta options: timestamp-format=%Y-%m-%d,%H:%M:%S.%01N.
Recipients:
Recipient: snmp_alert-recipient (value=192.168.1.2)
```

El siguiente ejemplo instala el agente **alert_smtp.sh** y luego configura una alerta que utiliza el agente de alertas instalado para enviar los eventos del clúster como mensajes de correo electrónico. Después de configurar la alerta, este ejemplo configura un destinatario y muestra la configuración de la alerta.

```
# install --mode=0755 /usr/share/pacemaker/alerts/alert_smtp.sh.sample
/var/lib/pacemaker/alert_smtp.sh
# pcs alert create id=smtp_alert path=/var/lib/pacemaker/alert_smtp.sh options
email_sender=donotreply@example.com
# pcs alert recipient add smtp_alert value=admin@example.com
# pcs alert
Alerts:
Alert: smtp_alert (path=/var/lib/pacemaker/alert_smtp.sh)
Options: email_sender=donotreply@example.com
Recipients:
Recipient: smtp_alert-recipient (value=admin@example.com)
```

26.2. CREACIÓN DE UNA ALERTA DE CLÚSTER

El siguiente comando crea una alerta de cluster. Las opciones que se configuran son valores de configuración específicos del agente que se pasan al script del agente de alertas en la ruta que se especifica como variables de entorno adicionales. Si no especifica un valor para **id**, se generará uno.

```
pcs alert create path=path [id=alert-id] [description=description] [options [option=value]...] [meta
[meta-option=value]...]
```

Se pueden configurar varios agentes de alerta; el clúster los llamará a todos para cada evento. Los agentes de alerta serán llamados sólo en los nodos del cluster. Serán llamados para eventos que involucren nodos Pacemaker Remote, pero nunca serán llamados en esos nodos.

El siguiente ejemplo crea una simple alerta que llamará a **myscript.sh** para cada evento.

```
# pcs alert create id=my_alert path=/path/to/myscript.sh
```

26.3. VISUALIZACIÓN, MODIFICACIÓN Y ELIMINACIÓN DE ALERTAS DE CLUSTER

El siguiente comando muestra todas las alertas configuradas junto con los valores de las opciones configuradas.

```
pcs alert [config|show]
```

El siguiente comando actualiza una alerta existente con el valor especificado en *alert-id*.

```
pcs alert update alert-id [path=path] [description=description] [options [option=value]...] [meta [meta-option=value]...]
```

El siguiente comando elimina una alerta con el valor especificado *alert-id*.

```
pcs alerta eliminar alert-id
```

Alternativamente, puede ejecutar el comando **pcs alert delete**, que es idéntico al comando **pcs alert remove**. Tanto el comando **pcs alert delete** como el **pcs alert remove** permiten especificar más de una alerta para ser eliminada.

26.4. CONFIGURAR LOS DESTINATARIOS DE LAS ALERTAS

Normalmente las alertas se dirigen a un destinatario. Así, cada alerta puede configurarse adicionalmente con uno o varios destinatarios. El clúster llamará al agente por separado para cada destinatario.

El destinatario puede ser cualquier cosa que el agente de alertas pueda reconocer: una dirección IP, una dirección de correo electrónico, un nombre de archivo o cualquier cosa que el agente en particular admita.

El siguiente comando añade un nuevo destinatario a la alerta especificada.

```
pcs alert recipient add alert-id value=recipient-value [id=recipient-id] [description=description] [options [option=value]...] [meta [meta-option=value]...]
```

El siguiente comando actualiza un destinatario de alerta existente.

```
pcs alert recipient update recipient-id [value=recipient-value] [description=description] [options [option=value]...] [meta [meta-option=value]...]
```

El siguiente comando elimina el destinatario de la alerta especificado.

```
pcs alerta receptor eliminar recipient-id
```

Alternativamente, puede ejecutar el comando **pcs alert recipient delete**, que es idéntico al comando **pcs alert recipient remove**. Tanto el comando **pcs alert recipient remove** como el **pcs alert recipient delete** le permiten eliminar más de un destinatario de alerta.

El siguiente comando de ejemplo añade el destinatario de la alerta **my-alert-recipient** con un ID de destinatario de **my-recipient-id** a la alerta **my-alert**. Esto configurará el cluster para llamar al script de alerta que ha sido configurado para **my-alert** para cada evento, pasando el destinatario **some-address** como una variable de entorno.

```
# pcs alert recipient add my-alert value=my-alert-recipient id=my-recipient-id options
value=some-address
```

26.5. OPCIONES DE META-ALERTA

Al igual que en el caso de los agentes de recursos, las opciones meta pueden configurarse para los agentes de alerta para afectar a la forma en que Pacemaker los llama. [Tabla 26.1, “Meta Opciones de Alerta”](#) describe las opciones meta de las alertas. Las opciones meta pueden configurarse por agente de alerta así como por destinatario.

Tabla 26.1. Meta Opciones de Alerta

Meta-atributo	Por defecto	Descripción
timestamp-format	%H:%M:%S.N	Formato que utilizará el clúster al enviar la marca de tiempo del evento al agente. Se trata de una cadena como la que se utiliza con el comando date (1).
timeout	30s	Si el agente de alerta no se completa dentro de este tiempo, se terminará.

El siguiente ejemplo configura una alerta que llama al script **myscript.sh** y luego añade dos destinatarios a la alerta. El primer destinatario tiene un ID de **my-alert-recipient1** y el segundo tiene un ID de **my-alert-recipient2**. El script será llamado dos veces para cada evento, con cada llamada usando un tiempo de espera de 15 segundos. Una llamada se pasará al destinatario **someuser@example.com** con una marca de tiempo en el formato %H:%M, mientras que la otra llamada se pasará al destinatario **otheruser@example.com** con una marca de tiempo en el formato

```
# pcs alert create id=my-alert path=/path/to/myscript.sh meta timeout=15s
# pcs alert recipient add my-alert value=someuser@example.com id=my-alert-recipient1 meta
timestamp-format="%D %H:%M"
# pcs alert recipient add my-alert value=otheruser@example.com id=my-alert-recipient2 meta
timestamp-format="%c"
```

26.6. EJEMPLOS DE COMANDOS DE CONFIGURACIÓN DE ALERTAS

Los siguientes ejemplos secuenciales muestran algunos comandos básicos de configuración de alertas para mostrar el formato a utilizar para crear alertas, añadir destinatarios y mostrar las alertas configuradas.

Tenga en cuenta que, si bien debe instalar los agentes de alerta en sí en cada nodo de un clúster, sólo debe ejecutar los comandos de **pcs** una vez.

Los siguientes comandos crean una alerta simple, añaden dos destinatarios a la alerta y muestran los valores configurados.

- Como no se especifica ningún valor de ID de alerta, el sistema crea un valor de ID de alerta de **alert**.

- El primer comando de creación de destinatarios especifica un destinatario de **rec_value**. Como este comando no especifica un ID de destinatario, se utiliza el valor de **alert-recipient** como ID de destinatario.
- El segundo comando de creación de destinatario especifica un destinatario de **rec_value2**. Este comando especifica un ID de destinatario de **my-recipient** para el destinatario.

```
# pcs alert create path=/my/path
# pcs alert recipient add alert value=rec_value
# pcs alert recipient add alert value=rec_value2 id=my-recipient
# pcs alert config
Alerts:
Alert: alert (path=/my/path)
Recipients:
Recipient: alert-recipient (value=rec_value)
Recipient: my-recipient (value=rec_value2)
```

Los siguientes comandos añaden una segunda alerta y un destinatario para esa alerta. El ID de la segunda alerta es **my-alert** y el valor del destinatario es **my-other-recipient**. Como no se especifica ningún ID de destinatario, el sistema proporciona un ID de destinatario de **my-alert-recipient**.

```
# pcs alert create id=my-alert path=/path/to/script description=alert_description options
option1=value1 opt=val meta timeout=50s timestamp-format="%H%B%S"
# pcs alert recipient add my-alert value=my-other-recipient
# pcs alert
Alerts:
Alert: alert (path=/my/path)
Recipients:
Recipient: alert-recipient (value=rec_value)
Recipient: my-recipient (value=rec_value2)
Alert: my-alert (path=/path/to/script)
Description: alert_description
Options: opt=val option1=value1
Meta options: timestamp-format=%H%B%S timeout=50s
Recipients:
Recipient: my-alert-recipient (value=my-other-recipient)
```

Los siguientes comandos modifican los valores de la alerta **my-alert** y del destinatario **my-alert-recipient**.

```
# pcs alert update my-alert options option1=newvalue1 meta timestamp-format="%H%M%S"
# pcs alert recipient update my-alert-recipient options option1=new meta timeout=60s
# pcs alert
Alerts:
Alert: alert (path=/my/path)
Recipients:
Recipient: alert-recipient (value=rec_value)
Recipient: my-recipient (value=rec_value2)
Alert: my-alert (path=/path/to/script)
Description: alert_description
Options: opt=val option1=newvalue1
Meta options: timestamp-format=%H%M%S timeout=50s
Recipients:
```

```

Recipient: my-alert-recipient (value=my-other-recipient)
Options: option1=new
Meta options: timeout=60s

```

El siguiente comando elimina el destinatario **my-alert-recipient** de **alert**.

```

# pcs alert recipient remove my-recipient
# pcs alert
Alerts:
Alert: alert (path=/my/path)
Recipients:
  Recipient: alert-recipient (value=rec_value)
Alert: my-alert (path=/path/to/script)
Description: alert_description
Options: opt=val option1=newvalue1
Meta options: timestamp-format="%M%B%S" timeout=50s
Recipients:
  Recipient: my-alert-recipient (value=my-other-recipient)
  Options: option1=new
  Meta options: timeout=60s

```

El siguiente comando elimina **myalert** de la configuración.

```

# pcs alert remove myalert
# pcs alert
Alerts:
Alert: alert (path=/my/path)
Recipients:
  Recipient: alert-recipient (value=rec_value)

```

26.7. ESCRIBIR UN AGENTE DE ALERTA

Hay tres tipos de alertas de Pacemaker: alertas de nodo, alertas de cercado y alertas de recursos. Las variables de entorno que se pasan a los agentes de alerta pueden variar, dependiendo del tipo de alerta. [Tabla 26.2, "Variables de entorno transmitidas a los agentes de alerta"](#) describe las variables de entorno que se pasan a los agentes de alerta y especifica cuándo la variable de entorno está asociada a un tipo de alerta específico.

Tabla 26.2. Variables de entorno transmitidas a los agentes de alerta

Variable de entorno	Descripción
CRM_alert_kind	El tipo de alerta (nodo, cercado o recurso)
CRM_alert_version	La versión de Pacemaker que envía la alerta
CRM_alert_recipient	El destinatario configurado

Variable de entorno	Descripción
CRM_alert_node_sequence	Un número de secuencia que se incrementa cada vez que se emite una alerta en el nodo local, que puede utilizarse para referenciar el orden en el que Pacemaker ha emitido las alertas. Una alerta para un evento que ocurrió más tarde en el tiempo tiene confiablemente un número de secuencia más alto que las alertas para eventos anteriores. Tenga en cuenta que este número no tiene ningún significado en todo el clúster.
CRM_alert_timestamp	Una marca de tiempo creada antes de ejecutar el agente, en el formato especificado por la opción meta timestamp-format . Esto permite que el agente tenga una hora fiable y de alta precisión de cuándo ocurrió el evento, independientemente de cuándo se invocó el propio agente (que podría retrasarse debido a la carga del sistema u otras circunstancias).
CRM_alert_node	Nombre del nodo afectado
CRM_alert_desc	Detalle del evento. Para las alertas de nodo, es el estado actual del nodo (miembro o perdido). Para las alertas de esgrima, es un resumen de la operación de esgrima solicitada, incluyendo el origen, el objetivo y el código de error de la operación de esgrima, si lo hay. Para las alertas de recursos, se trata de una cadena legible equivalente a CRM_alert_status .
CRM_alert_nodeid	ID del nodo cuyo estado ha cambiado (sólo se proporciona con las alertas de nodo)
CRM_alert_task	La operación de cercado o de recursos solicitada (sólo se proporciona con las alertas de cercado y de recursos)
CRM_alert_rc	El código numérico de retorno de la operación de esgrima o de recursos (sólo se proporciona con las alertas de esgrima y de recursos)
CRM_alert_rsc	El nombre del recurso afectado (sólo alertas de recursos)
CRM_alert_interval	El intervalo de la operación de recursos (sólo alertas de recursos)
CRM_alert_target_rc	El código de retorno numérico esperado de la operación (sólo alertas de recursos)

Variable de entorno	Descripción
CRM_alert_status	Un código numérico utilizado por Pacemaker para representar el resultado de la operación (sólo alertas de recursos)

A la hora de redactar un agente de alerta, hay que tener en cuenta los siguientes aspectos.

- Los agentes de alerta pueden ser llamados sin destinatario (si no está configurado ninguno), por lo que el agente debe ser capaz de manejar esta situación, aunque sólo salga en ese caso. Los usuarios pueden modificar la configuración por etapas, y añadir un destinatario más tarde.
- Si se configura más de un destinatario para una alerta, el agente de alerta será llamado una vez por cada destinatario. Si un agente no puede ejecutarse simultáneamente, debe configurarse con un solo destinatario. Sin embargo, el agente es libre de interpretar el destinatario como una lista.
- Cuando se produce un evento de cluster, todas las alertas se disparan al mismo tiempo como procesos separados. Dependiendo de cuántas alertas y destinatarios se configuren y de lo que se haga dentro de los agentes de alertas, puede producirse una explosión de carga significativa. El agente podría escribirse para tener esto en cuenta, por ejemplo, poniendo en cola las acciones que consumen muchos recursos en alguna otra instancia, en lugar de ejecutarlas directamente.
- Los agentes de alerta se ejecutan como el usuario **hacluster**, que tiene un conjunto mínimo de permisos. Si un agente requiere privilegios adicionales, se recomienda configurar **sudo** para permitir que el agente ejecute los comandos necesarios como otro usuario con los privilegios adecuados.
- Tenga cuidado de validar y sanear los parámetros configurados por el usuario, como **CRM_alert_timestamp** (cuyo contenido está especificado por el usuario **timestamp-format**), **CRM_alert_recipient**, y todas las opciones de alerta. Esto es necesario para protegerse de los errores de configuración. Además, si algún usuario puede modificar el CIB sin tener **hacluster**-nivel de acceso a los nodos del clúster, esto es una preocupación potencial de seguridad también, y usted debe evitar la posibilidad de inyección de código.
- Si un clúster contiene recursos con operaciones para las que el parámetro **on-fail** está configurado como **fence**, habrá múltiples notificaciones de valla en caso de fallo, una por cada recurso para el que esté configurado este parámetro más una notificación adicional. Tanto **pacemaker-fenced** como **pacemaker-controld** enviarán notificaciones. Sin embargo, Pacemaker sólo realiza una operación de valla real en este caso, sin importar cuántas notificaciones se envíen.



NOTA

La interfaz de alertas está diseñada para ser compatible con la interfaz de scripts externos utilizada por el recurso **ocf:pacemaker:ClusterMon**. Para preservar esta compatibilidad, las variables de entorno que se pasan a los agentes de alertas están disponibles precedidas de **CRM_notify_** así como de **CRM_alert_**. Una ruptura en la compatibilidad es que el recurso **ClusterMon** ejecutaba los scripts externos como el usuario **root**, mientras que los agentes de alerta se ejecutan como el usuario **hacluster**.

CAPÍTULO 27. CONFIGURACIÓN DE CLUSTERS MULTISITIO CON PACEMAKER

Cuando un clúster abarca más de un sitio, los problemas de conectividad de red entre los sitios pueden llevar a situaciones de cerebro dividido. Cuando la conectividad se cae, no hay manera de que un nodo en un sitio determine si un nodo en otro sitio ha fallado o sigue funcionando con un interlink de sitio fallado. Además, puede ser problemático proporcionar servicios de alta disponibilidad en dos sitios que están demasiado alejados para mantener la sincronización.

Para solucionar estos problemas, Pacemaker ofrece una compatibilidad total con la capacidad de configurar clústeres de alta disponibilidad que abarcan varios sitios mediante el uso de un gestor de tickets de clústeres de Booth.

27.1. VISIÓN GENERAL DEL GESTOR DE TICKETS DEL CLÚSTER BOOTH

El Booth *ticket manager* es un servicio distribuido que está pensado para ejecutarse en una red física diferente a las redes que conectan los nodos del clúster en sitios concretos. Produce otro clúster suelto, un *Booth formation*, que se sitúa encima de los clústeres regulares de los sitios. Esta capa de comunicación agregada facilita los procesos de decisión basados en el consenso para las entradas individuales de Booth.

Una cabina *ticket* es un singleton en la formación de la cabina y representa una unidad de autorización temporal y móvil. Los recursos pueden configurarse para que requieran un determinado ticket para ejecutarse. Esto puede garantizar que los recursos se ejecuten sólo en un sitio a la vez, para el que se ha concedido un ticket o tickets.

Se puede pensar en una formación Booth como un clúster superpuesto que consiste en clústeres que se ejecutan en diferentes sitios, donde todos los clústeres originales son independientes entre sí. Es el servicio Booth el que comunica a los clusters si se les ha concedido un ticket, y es Pacemaker el que determina si se ejecutan recursos en un cluster basándose en una restricción de ticket de Pacemaker. Esto significa que cuando se utiliza el gestor de tickets, cada uno de los clusters puede ejecutar sus propios recursos, así como los recursos compartidos. Por ejemplo, puede haber recursos A, B y C que se ejecuten sólo en un clúster, recursos D, E y F que se ejecuten sólo en el otro clúster, y recursos G y H que se ejecuten en cualquiera de los dos clústeres según lo determine un ticket. También es posible tener un recurso adicional J que podría ejecutarse en cualquiera de los dos clústeres según lo determinado por un ticket separado.

27.2. CONFIGURACIÓN DE CLUSTERS MULTISITIO CON PACEMAKER

El siguiente procedimiento proporciona un resumen de los pasos que se siguen para configurar una configuración multisitio que utiliza el gestor de tickets de Booth.

Estos comandos de ejemplo utilizan la siguiente disposición:

- El clúster 1 está formado por los nodos **cluster1-node1** y **cluster1-node2**
- El cluster 1 tiene asignada una dirección IP flotante de 192.168.11.100
- El grupo 2 está formado por **cluster2-node1** y **cluster2-node2**
- El clúster 2 tiene asignada una dirección IP flotante de 192.168.22.100
- El nodo árbitro es **arbitrator-node** con una dirección ip de 192.168.99.100

- El nombre del ticket de Booth que utiliza esta configuración es **apacheticket**

Estos comandos de ejemplo asumen que los recursos de cluster para un servicio Apache han sido configurados como parte del grupo de recursos **apachegroup** para cada cluster. No es necesario que los recursos y los grupos de recursos sean los mismos en cada clúster para configurar una restricción de tickets para esos recursos, ya que la instancia de Pacemaker para cada clúster es independiente, pero ese es un escenario común de conmutación por error.

Tenga en cuenta que en cualquier momento del procedimiento de configuración puede introducir el comando **pcs booth config** para mostrar la configuración de booth para el nodo o clúster actual o el comando **pcs booth status** para mostrar el estado actual de booth en el nodo local.

1. Instale el paquete **booth-site** Booth ticket manager en cada nodo de ambos clusters.

```
[root@cluster1-node1 ~]# yum install -y booth-site
[root@cluster1-node2 ~]# yum install -y booth-site
[root@cluster2-node1 ~]# yum install -y booth-site
[root@cluster2-node2 ~]# yum install -y booth-site
```

2. Instale los paquetes **pcs**, **booth-core**, y **booth-arbitrator** en el nodo árbitro.

```
[root@arbitrator-node ~]# yum install -y pcs booth-core booth-arbitrator
```

3. Si está ejecutando el demonio **firewalld**, ejecute los siguientes comandos en todos los nodos de ambos clústeres así como en el nodo árbitro para habilitar los puertos requeridos por el complemento de alta disponibilidad de Red Hat.

```
# firewall-cmd --permanent --add-service=high-availability`
# firewall-cmd --add-service=high-availability`
```

Es posible que tenga que modificar los puertos que están abiertos para adaptarse a las condiciones locales. Para más información sobre los puertos requeridos por el complemento de [alta](#) disponibilidad de Red Hat, consulte [Habilitación de puertos para el complemento de alta disponibilidad](#).

4. Cree una configuración de Booth en un nodo de un cluster. Las direcciones que especifique para cada clúster y para el árbitro deben ser direcciones IP. Para cada clúster, se especifica una dirección IP flotante.

```
[cluster1-node1 ~] # pcs booth setup sites 192.168.11.100 192.168.22.100 arbitrators
192.168.99.100
```

Este comando crea los archivos de configuración **/etc/booth/booth.conf** y **/etc/booth/booth.key** en el nodo desde el que se ejecuta.

5. Cree un ticket para la configuración de Booth. Este es el ticket que utilizará para definir la restricción de recursos que permitirá que los recursos se ejecuten solo cuando este ticket se haya concedido al clúster.

Este procedimiento básico de configuración de la conmutación por error utiliza sólo un ticket, pero puede crear tickets adicionales para escenarios más complicados en los que cada ticket esté asociado a un recurso o recursos diferentes.

```
[cluster1-node1 ~] # pcs booth ticket add apacheticket
```

- Sincronizar la configuración de Booth en todos los nodos del clúster actual.

```
[cluster1-node1 ~] # pcs booth sync
```

- Desde el nodo del árbitro, extraiga la configuración de Booth al árbitro. Si no lo ha hecho previamente, debe autenticar primero **pcs** en el nodo desde el que está extrayendo la configuración.

```
[arbitrator-node ~] # pcs host auth cluster1-node1
[arbitrator-node ~] # pcs booth pull cluster1-node1
```

- Extraiga la configuración de Booth al otro clúster y sincronice a todos los nodos de ese clúster. Al igual que con el nodo árbitro, si no lo ha hecho previamente, debe autenticar primero **pcs** en el nodo del que está tirando la configuración.

```
[cluster2-node1 ~] # pcs host auth cluster1-node1
[cluster2-node1 ~] # pcs booth pull cluster1-node1
[cluster2-node1 ~] # pcs booth sync
```

- Iniciar y habilitar Booth en el árbitro.



NOTA

No debe iniciar o habilitar manualmente Booth en ninguno de los nodos de los clusters, ya que Booth se ejecuta como un recurso de Pacemaker en esos clusters.

```
[arbitrator-node ~] # pcs booth start
[arbitrator-node ~] # pcs booth enable
```

- Configure Booth para que se ejecute como un recurso de clúster en ambos sitios de clúster. Esto crea un grupo de recursos con **booth-ip** y **booth-service** como miembros de ese grupo.

```
[cluster1-node1 ~] # pcs booth create ip 192.168.11.100
[cluster2-node1 ~] # pcs booth create ip 192.168.22.100
```

- Añade una restricción de ticket al grupo de recursos que has definido para cada clúster.

```
[cluster1-node1 ~] # pcs constraint ticket add apacheticket apachegroup
[cluster2-node1 ~] # pcs constraint ticket add apacheticket apachegroup
```

Puede introducir el siguiente comando para mostrar las restricciones de billetes actualmente configuradas.

```
pcs constraint ticket [show]
```

- Conceda el ticket que creó para esta configuración al primer clúster. Tenga en cuenta que no es necesario haber definido las restricciones de los tickets antes de concederlos. Una vez que haya concedido inicialmente un ticket a un clúster, Booth se encarga de la gestión de los tickets a menos que lo anule manualmente con el comando **pcs booth ticket revoke**. Para obtener información sobre los comandos de administración de **pcs booth**, consulte la pantalla de ayuda de PCS para el comando **pcs booth**.

```
[cluster1-node1 ~] # pcs booth ticket grant apacheticket
```

Es posible añadir o eliminar tickets en cualquier momento, incluso después de completar este procedimiento. Sin embargo, después de añadir o eliminar un ticket, debe sincronizar los archivos de configuración con los otros nodos y clústeres, así como con el árbitro, y conceder el ticket como se muestra en este procedimiento.

Para obtener información sobre otros comandos de administración de Booth que puede utilizar para limpiar y eliminar archivos de configuración, tickets y recursos de Booth, consulte la pantalla de ayuda de PCS para el comando **pcs booth**.

CAPÍTULO 28. INTEGRACIÓN DE NODOS NO SINCRONIZADOS EN UN CLÚSTER: EL SERVICIO PACEMAKER_REMOTE

El servicio **pacemaker_remote** permite que los nodos que no ejecutan **corosync** se integren en el clúster y que éste gestione sus recursos como si fueran nodos reales del clúster.

Entre las capacidades que ofrece el servicio **pacemaker_remote** se encuentran las siguientes:

- El servicio **pacemaker_remote** le permite escalar más allá del límite de soporte de Red Hat de 32 nodos para RHEL 8.1.
- El servicio **pacemaker_remote** permite gestionar un entorno virtual como recurso de cluster y también gestionar servicios individuales dentro del entorno virtual como recursos de cluster.

Los siguientes términos se utilizan para describir el servicio **pacemaker_remote**.

- *cluster node*
- *remote node*
- *guest node*
- *pacemaker_remote*

Un cluster Pacemaker que ejecuta el servicio **pacemaker_remote** tiene las siguientes características.

- Los nodos remotos y los nodos invitados ejecutan el servicio **pacemaker_remote** (con muy poca configuración necesaria en el lado de la máquina virtual).
- La pila del clúster (**pacemaker** y **corosync**), que se ejecuta en los nodos del clúster, se conecta al servicio **pacemaker_remote** en los nodos remotos, lo que les permite integrarse en el clúster.
- La pila del clúster (**pacemaker** y **corosync**), que se ejecuta en los nodos del clúster, lanza los nodos invitados y se conecta inmediatamente al servicio **pacemaker_remote** en los nodos invitados, permitiéndoles integrarse en el clúster.

La diferencia clave entre los nodos del clúster y los nodos remotos e invitados que gestionan los nodos del clúster es que los nodos remotos e invitados no están ejecutando la pila del clúster. Esto significa que los nodos remotos e invitados tienen las siguientes limitaciones:

- no tienen lugar en el quórum
- no ejecutan las acciones del dispositivo de esgrima
- no son elegibles para ser el controlador designado (DC) de la agrupación
- no ejecutan por sí mismos toda la gama de comandos de **pcs**

Por otro lado, los nodos remotos y los nodos invitados no están sujetos a los límites de escalabilidad asociados a la pila del clúster.

Aparte de estas limitaciones, los nodos remotos e invitados se comportan igual que los nodos del clúster en lo que respecta a la gestión de recursos, y los propios nodos remotos e invitados pueden ser cercados. El clúster es totalmente capaz de gestionar y supervisar los recursos de cada nodo remoto e

invitado: Puede crear restricciones contra ellos, ponerlos en espera o realizar cualquier otra acción que realice en los nodos del clúster con los comandos **pcs**. Los nodos remotos e invitados aparecen en la salida del estado del clúster igual que los nodos del clúster.

28.1. AUTENTICACIÓN DE HOST Y GUEST DE LOS NODOS PACEMAKER_REMOTE

La conexión entre los nodos del clúster y `pacemaker_remote` está asegurada mediante Transport Layer Security (TLS) con cifrado de clave precompartida (PSK) y autenticación a través de TCP (utilizando el puerto 3121 por defecto). Esto significa que tanto el nodo del clúster como el nodo que ejecuta **pacemaker_remote** deben compartir la misma clave privada. Por defecto, esta clave debe colocarse en `/etc/pacemaker/authkey` tanto en los nodos del clúster como en los nodos remotos.

El comando **pcs cluster node add-guest** configura el **authkey** para los nodos invitados y el comando **pcs cluster node add-remote** configura el **authkey** para los nodos remotos.

28.2. CONFIGURACIÓN DE LOS NODOS INVITADOS KVM

Un nodo invitado de Pacemaker es un nodo invitado virtual que ejecuta el servicio **pacemaker_remote**. El nodo invitado virtual es gestionado por el clúster.

28.2.1. Opciones de recursos del nodo invitado

Al configurar una máquina virtual para que actúe como nodo invitado, se crea un recurso **VirtualDomain**, que gestiona la máquina virtual. Para ver las descripciones de las opciones que puede establecer para un recurso **VirtualDomain**, consulte [Tabla 24.1, "Opciones de recursos para los recursos del dominio virtual"](#).

Además de las opciones del recurso **VirtualDomain**, las opciones de metadatos definen el recurso como nodo invitado y definen los parámetros de conexión. Estas opciones de recursos se establecen con el comando **pcs cluster node add-guest**. [Tabla 28.1, "Opciones de metadatos para configurar recursos KVM como nodos remotos"](#) describe estas opciones de metadatos.

Tabla 28.1. Opciones de metadatos para configurar recursos KVM como nodos remotos

Campo	Por defecto	Descripción
remote-node	<one>	El nombre del nodo invitado que define este recurso. Esto habilita el recurso como nodo invitado y define el nombre único utilizado para identificar el nodo invitado. <i>WARNING:</i> Este valor no puede solaparse con ningún recurso o ID de nodo.
remote-port	3121	Configura un puerto personalizado para utilizar la conexión de invitados a pacemaker_remote
remote-addr	La dirección proporcionada en el comando pcs host auth	La dirección IP o el nombre de host al que conectarse

Campo	Por defecto	Descripción
remote-connect-timeout	60s	Cantidad de tiempo antes de que se agote el tiempo de una conexión pendiente de invitados

28.2.2. Integración de una máquina virtual como nodo invitado

El siguiente procedimiento es un resumen de alto nivel de los pasos a realizar para que Pacemaker lance una máquina virtual e integre esa máquina como nodo invitado, utilizando **libvirt** y los invitados virtuales KVM.

1. Configure los recursos de **VirtualDomain**.
2. Introduzca los siguientes comandos en cada máquina virtual para instalar los paquetes de **pacemaker_remote**, iniciar el servicio **pcsd** y permitir que se ejecute al inicio, y permitir el puerto TCP 3121 a través del cortafuegos.

```
# yum install pacemaker-remote resource-agents pcs
# systemctl start pcsd.service
# systemctl enable pcsd.service
# firewall-cmd --add-port 3121/tcp --permanent
# firewall-cmd --add-port 2224/tcp --permanent
# firewall-cmd --reload
```

3. Dé a cada máquina virtual una dirección de red estática y un nombre de host único, que debe ser conocido por todos los nodos. Para obtener información sobre la configuración de una dirección IP estática para la máquina virtual invitada, consulte la sección *Virtualization Deployment and Administration Guide*.
4. Si aún no lo has hecho, autentifica **pcs** en el nodo que vas a integrar como nodo de búsqueda.

```
# pcs host auth nodename
```

5. Utilice el siguiente comando para convertir un recurso existente de **VirtualDomain** en un nodo invitado. Este comando debe ejecutarse en un nodo del clúster y no en el nodo invitado que se está añadiendo. Además de convertir el recurso, este comando copia el **/etc/pacemaker/authkey** al nodo invitado e inicia y habilita el demonio **pacemaker_remote** en el nodo invitado. El nombre del nodo invitado, que se puede definir de forma arbitraria, puede diferir del nombre del host del nodo.

```
# pcs cluster node add-guest nodename resource_id [options]
```

6. Después de crear el recurso **VirtualDomain**, puede tratar el nodo invitado como trataría cualquier otro nodo del clúster. Por ejemplo, puede crear un recurso y colocar una restricción de recursos para que se ejecute en el nodo invitado, como en los siguientes comandos, que se ejecutan desde un nodo del clúster. Puede incluir nodos invitados en grupos, lo que le permite agrupar un dispositivo de almacenamiento, un sistema de archivos y una máquina virtual.

```
# pcs resource create webserver apache configfile=/etc/httpd/conf/httpd.conf op
monitor interval=30s
# pcs constraint location webserver prefers nodename
```

28.3. CONFIGURACIÓN DE LOS NODOS REMOTOS DE PACEMAKER

Un nodo remoto se define como un recurso de cluster con **ocf:pacemaker:remote** como agente de recursos. Este recurso se crea con el comando **pcs cluster node add-remote**.

28.3.1. Opciones de recursos del nodo remoto

Tabla 28.2, "Opciones de recursos para nodos remotos" describe las opciones de recursos que puede configurar para un recurso **remote**.

Tabla 28.2. Opciones de recursos para nodos remotos

C a m p o	Por defecto	D e s c r i p c i ó n
r e c o n n e c t _ i n t e r v a l	0	T i e m p o e n s e g u n d o s p a r a e s p e r a r a n t e

C a m p o	Por defecto	D e s c r i p c i ó n
		e c o n e c t a r s e a u n n o d o r e m o t o d e s p u é s d e q u e s e h a y a c o

C a m p o	Por defecto	D e s c r i p c i ó n
		X i ó n a c t i v a c o n e l n o d o r e m o t o . E s t a e s p e r a e s r e c u r r

C a m p o	Por defecto	D e s c r i p c i ó n
		o n e x i ó n f a l l a d e s p u é s d e l p e r í o d e e s p e r a , s e r e a l

C a m p o	Por defecto	D e s c r i p c i ó n
		I n t e n t o d e r e c o n e x i ó n d e s p u é s d e o b s e r v a r e l t i e m p o

C a m p o	Por defecto	D e s c r i p c i ó n
		n d o s e u t i l i z a e s t a o p c i ó n , P a c e m a k e r s e g u i r á i n t e n

C a m p o	Por defecto	D e s c r i p c i ó n
		r y c o n e c t a r c o n e l n o d o r e m o t o i n d e f i n i d a m e n t e d e s p

C a m p o	Por defecto	D e s c r i p c i ó n
		e r v a l o d e e s p e r a .
s e r v e r	Dirección especificada con el comando pcs host auth	S e r v i d o r a l q u e c o n e c t a r s e . P u e d

C a m p o	Por defecto	D e s c r i p c i ó n
		C i ó n l P o u n n o m b r e d e h o s t .

C a m p o	Por defecto	D e s c r i p c i ó n
p o r t		P u e r t o T C P a l q u e c o n e c t a r s e .

28.3.2. Resumen de la configuración del nodo remoto

Esta sección proporciona un resumen de alto nivel de los pasos a realizar para configurar un nodo Pacemaker Remote e integrar ese nodo en un entorno de cluster Pacemaker existente.

1. En el nodo que va a configurar como nodo remoto, permita los servicios relacionados con el cluster a través del firewall local.

```
# firewall-cmd --permanent --add-service=high-availability
success
# firewall-cmd --reload
success
```



NOTA

Si está usando **iptables** directamente, o alguna otra solución de firewall además de **firewalld**, simplemente abra los siguientes puertos: Puertos TCP 2224 y 3121.

2. Instale el demonio **pacemaker_remote** en el nodo remoto.

```
# yum install -y pacemaker-remote resource-agents pcs
```

3. Inicie y habilite **pcsd** en el nodo remoto.

```
# systemctl start pcsd.service
# systemctl enable pcsd.service
```

4. Si aún no lo ha hecho, autentique **pcs** en el nodo que va a añadir como nodo remoto.

```
# pcs host auth remote1
```

5. Añade el recurso del nodo remoto al clúster con el siguiente comando. Este comando también sincroniza todos los archivos de configuración relevantes con el nuevo nodo, arranca el nodo y lo configura para iniciar **pacemaker_remote** en el arranque. Este comando debe ejecutarse en un nodo del clúster y no en el nodo remoto que se está añadiendo.

```
# pcs cluster node add-remote remote1
```

6. Después de añadir el recurso **remote** al clúster, puede tratar el nodo remoto igual que trataría cualquier otro nodo del clúster. Por ejemplo, puede crear un recurso y colocar una restricción de recursos para que se ejecute en el nodo remoto, como en los siguientes comandos, que se ejecutan desde un nodo del clúster.

```
# pcs resource create webserver apache configfile=/etc/httpd/conf/httpd.conf op
monitor interval=30s
# pcs constraint location webserver prefers remote1
```



AVISO

Nunca incluya un recurso de conexión de nodo remoto en un grupo de recursos, restricción de colocación o restricción de orden.

7. Configure los recursos de cercado para el nodo remoto. Los nodos remotos se cercan de la misma manera que los nodos de cluster. Configure los recursos de esgrima para su uso con los nodos remotos de la misma manera que lo haría con los nodos del clúster. Sin embargo, tenga en cuenta que los nodos remotos nunca pueden iniciar una acción de esgrima. Sólo los nodos del clúster son capaces de ejecutar una operación de cercado contra otro nodo.

28.4. CAMBIAR LA UBICACIÓN DEL PUERTO POR DEFECTO

Si necesita cambiar la ubicación del puerto por defecto para Pacemaker o **pacemaker_remote**, puede establecer la variable de entorno **PCMK_remote_port** que afecta a ambos demonios. Esta variable de entorno se puede habilitar colocándola en el archivo **/etc/sysconfig/pacemaker** de la siguiente manera.

```
====# Pacemaker Remote
...
#
# Specify a custom port for Pacemaker Remote connections
PCMK_remote_port=3121
```

Cuando se cambia el puerto por defecto utilizado por un nodo invitado o un nodo remoto en particular, la variable **PCMK_remote_port** debe establecerse en el archivo **/etc/sysconfig/pacemaker** de ese nodo, y el recurso del clúster que crea la conexión del nodo invitado o del nodo remoto también debe configurarse con el mismo número de puerto (utilizando la opción de metadatos **remote-port** para los nodos invitados, o la opción **port** para los nodos remotos).

28.5. ACTUALIZACIÓN DE SISTEMAS CON NODOS PACEMAKER_REMOTE

Si el servicio **pacemaker_remote** se detiene en un nodo activo de Pacemaker Remote, el clúster migrará con gracia los recursos del nodo antes de detenerlo. Esto le permite realizar actualizaciones de software y otros procedimientos de mantenimiento rutinarios sin retirar el nodo del clúster. Sin embargo, una vez que se apague **pacemaker_remote**, el clúster intentará reconectarse inmediatamente. Si **pacemaker_remote** no se reinicia dentro del tiempo de espera de monitorización del recurso, el clúster considerará la operación de monitorización como fallida.

Si desea evitar fallos de monitorización cuando el servicio **pacemaker_remote** se detiene en un nodo activo de Pacemaker Remote, puede utilizar el siguiente procedimiento para sacar el nodo del clúster antes de realizar cualquier administración del sistema que pueda detener **pacemaker_remote**

1. Detenga el recurso de conexión del nodo con el botón **pcs resource disable *resourcename*** que moverá todos los servicios fuera del nodo. Para los nodos invitados, esto también detendrá la VM, por lo que la VM debe iniciarse fuera del clúster (por ejemplo, utilizando **virsh**) para realizar cualquier mantenimiento.
2. Realice el mantenimiento necesario.
3. Cuando esté listo para devolver el nodo al clúster, vuelva a habilitar el recurso con el botón **pcs resource enable**.

CAPÍTULO 29. REALIZAR EL MANTENIMIENTO DEL CLÚSTER

Para realizar el mantenimiento de los nodos de su clúster, es posible que tenga que detener o mover los recursos y servicios que se ejecutan en ese clúster. O puede que tenga que detener el software del clúster dejando los servicios intactos. Pacemaker proporciona una variedad de métodos para realizar el mantenimiento del sistema.

- Si necesita detener un nodo de un clúster mientras continúa proporcionando los servicios que se ejecutan en ese clúster en otro nodo, puede poner el nodo del clúster en modo de espera. Un nodo que está en modo de espera ya no puede alojar recursos. Cualquier recurso actualmente activo en el nodo se moverá a otro nodo, o se detendrá si ningún otro nodo es elegible para ejecutar el recurso. Para obtener información sobre el modo de espera, consulte [Poner un nodo en modo de espera](#).
- Si necesita mover un recurso individual fuera del nodo en el que se está ejecutando actualmente sin detener ese recurso, puede utilizar el comando **pcs resource move** para mover el recurso a un nodo diferente. Para obtener información sobre el comando **pcs resource move**, consulte [Mover manualmente los recursos del clúster](#).
Cuando se ejecuta el comando **pcs resource move**, se añade una restricción al recurso para evitar que se ejecute en el nodo en el que se está ejecutando actualmente. Cuando esté listo para volver a mover el recurso, puede ejecutar el comando **pcs resource clear** o **pcs constraint delete** para eliminar la restricción. Sin embargo, esto no necesariamente mueve los recursos de vuelta al nodo original, ya que el lugar donde los recursos pueden ejecutarse en ese momento depende de cómo hayas configurado tus recursos inicialmente. Puede reubicar un recurso en su nodo preferido con el comando **pcs resource relocate run**, como se describe en [Mover un recurso a su nodo preferido](#).
- Si necesita detener un recurso en ejecución por completo y evitar que el clúster lo vuelva a iniciar, puede utilizar el comando **pcs resource disable**. Para obtener información sobre el comando **pcs resource disable**, consulte [Activación, desactivación y prohibición de recursos del clúster](#).
- Si desea evitar que Pacemaker realice alguna acción para un recurso (por ejemplo, si desea desactivar las acciones de recuperación mientras se realiza el mantenimiento del recurso, o si necesita recargar la configuración de `/etc/sysconfig/pacemaker`), utilice el comando **pcs resource unmanage**, como se describe en [Cómo poner un recurso en modo no gestionado](#). Los recursos de conexión de Pacemaker Remote nunca deben estar sin gestionar.
- Si necesita poner el clúster en un estado en el que no se inicie ni se detenga ningún servicio, puede establecer la propiedad de clúster **maintenance-mode**. Al poner el clúster en modo de mantenimiento, se deshabilitan automáticamente todos los recursos. Para obtener información sobre cómo poner el clúster en modo de mantenimiento, consulte [Poner un clúster en modo de mantenimiento](#).
- Si necesita actualizar los paquetes que componen los complementos RHEL High Availability y Resilient Storage, puede actualizar los paquetes en un nodo a la vez o en todo el cluster como un todo, como se resume en [Actualización de un cluster de alta disponibilidad de Red Hat Enterprise Linux](#).
- Si necesita realizar tareas de mantenimiento en un nodo remoto de Pacemaker, puede eliminar ese nodo del clúster desactivando el recurso del nodo remoto, como se describe en [Actualización de nodos remotos y nodos invitados](#).

29.1. PONER UN NODO EN MODO DE ESPERA

Cuando un nodo del clúster está en modo de espera, el nodo ya no puede albergar recursos. Cualquier recurso actualmente activo en el nodo será trasladado a otro nodo.

El siguiente comando pone el nodo especificado en modo de espera. Si se especifica el **--all**, este comando pone todos los nodos en modo de espera.

Puedes utilizar este comando cuando actualices los paquetes de un recurso. También se puede utilizar este comando al probar una configuración, para simular la recuperación sin apagar realmente un nodo.

```
pcs nodo standby node | --all
```

El siguiente comando elimina el nodo especificado del modo de espera. Después de ejecutar este comando, el nodo especificado podrá alojar recursos. Si se especifica el **--all**, este comando elimina todos los nodos del modo de espera.

```
pcs node unstandby node | --all
```

Tenga en cuenta que cuando ejecuta el comando **pcs node standby**, esto impide que los recursos se ejecuten en el nodo indicado. Cuando se ejecuta el comando **pcs node unstandby**, se permite que los recursos se ejecuten en el nodo indicado. Esto no mueve necesariamente los recursos de vuelta al nodo indicado; dónde pueden ejecutarse los recursos en ese momento depende de cómo hayas configurado tus recursos inicialmente.

29.2. MOVER MANUALMENTE LOS RECURSOS DEL CLÚSTER

Puede anular el clúster y forzar que los recursos se muevan de su ubicación actual. Hay dos ocasiones en las que querrás hacer esto:

- Cuando un nodo está en mantenimiento, y necesita mover todos los recursos que se ejecutan en ese nodo a un nodo diferente
- Cuando hay que mover recursos especificados individualmente

Para trasladar todos los recursos que se ejecutan en un nodo a otro nodo, se pone el nodo en modo de espera.

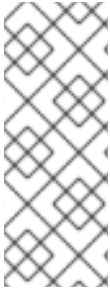
Puede mover los recursos especificados individualmente de cualquiera de las siguientes maneras.

- Puede utilizar el comando **pcs resource move** para mover un recurso fuera de un nodo en el que se está ejecutando actualmente.
- Puede utilizar el comando **pcs resource relocate run** para mover un recurso a su nodo preferido, según el estado actual del clúster, las restricciones, la ubicación de los recursos y otras configuraciones.

29.2.1. Mover un recurso de su nodo actual

Para mover un recurso fuera del nodo en el que se está ejecutando actualmente, utilice el siguiente comando, especificando el *resource_id* del recurso tal y como está definido. Especifique la dirección **destination_node** si desea indicar en qué nodo debe ejecutarse el recurso que está moviendo.

```
pcs resource move resource_id [destination_node] [--master] [lifetime=lifetime]
```



NOTA

Cuando se ejecuta el comando **pcs resource move**, se añade una restricción al recurso para evitar que se ejecute en el nodo en el que se está ejecutando actualmente. Puede ejecutar el comando **pcs resource clear** o **pcs constraint delete** para eliminar la restricción. Esto no necesariamente mueve los recursos de vuelta al nodo original; dónde pueden ejecutarse los recursos en ese momento depende de cómo hayas configurado tus recursos inicialmente.

Si especifica el parámetro **--master** del comando **pcs resource move**, el alcance de la restricción se limita al rol maestro y debe especificar *master_id* en lugar de *resource_id*.

Opcionalmente, puede configurar un parámetro **lifetime** para el comando **pcs resource move** para indicar un período de tiempo que la restricción debe permanecer. Las unidades de un parámetro **lifetime** se especifican según el formato definido en la norma ISO 8601, que requiere que se especifique la unidad como una letra mayúscula como Y (para años), M (para meses), W (para semanas), D (para días), H (para horas), M (para minutos) y S (para segundos).

Para distinguir una unidad de minutos(M) de una unidad de meses(M), debe especificar PT antes de indicar el valor en minutos. Por ejemplo, un parámetro **lifetime** de 5M indica un intervalo de cinco meses, mientras que un parámetro **lifetime** de PT5M indica un intervalo de cinco minutos.

El parámetro **lifetime** se comprueba a intervalos definidos por la propiedad de cluster **cluster-recheck-interval**. Por defecto este valor es de 15 minutos. Si su configuración requiere que compruebe este parámetro con más frecuencia, puede restablecer este valor con el siguiente comando.

```
pcs property set cluster-recheck-interval=value
```

Opcionalmente se puede configurar un parámetro **--wait[=*n*]** para el comando **pcs resource move** para indicar el número de segundos que se debe esperar para que el recurso se inicie en el nodo de destino antes de devolver 0 si el recurso se ha iniciado o 1 si el recurso aún no se ha iniciado. Si no se especifica *n*, se utilizará el tiempo de espera del recurso por defecto.

El siguiente comando mueve el recurso **resource1** al nodo **example-node2** y evita que vuelva al nodo en el que se estaba ejecutando originalmente durante una hora y treinta minutos.

```
pcs resource move resource1 example-node2 lifetime=PT1H30M
```

El siguiente comando mueve el recurso **resource1** al nodo **example-node2** y evita que vuelva al nodo en el que se estaba ejecutando originalmente durante treinta minutos.

```
pcs resource move resource1 example-node2 lifetime=PT30M
```

29.2.2. Mover un recurso a su nodo preferido

Después de que un recurso se haya movido, ya sea debido a una conmutación por error o a que un administrador haya movido manualmente el nodo, no necesariamente se moverá de nuevo a su nodo original incluso después de que las circunstancias que causaron la conmutación por error se hayan corregido. Para reubicar los recursos en su nodo preferido, utilice el siguiente comando. El nodo preferido está determinado por el estado actual del clúster, las restricciones, la ubicación de los recursos y otras configuraciones, y puede cambiar con el tiempo.

```
pcs resource relocate run [resource1] [resource2] ...
```


Si no se especifica ningún recurso, todos los recursos se reubican en sus nodos preferidos.

Este comando calcula el nodo preferido para cada recurso sin tener en cuenta la rigidez de los recursos. Después de calcular el nodo preferido, crea restricciones de ubicación que harán que los recursos se muevan a sus nodos preferidos. Una vez que los recursos se han movido, las restricciones se eliminan automáticamente. Para eliminar todas las restricciones creadas por el comando **pcs resource relocate run**, puede introducir el comando **pcs resource relocate clear**. Para mostrar el estado actual de los recursos y su nodo óptimo ignorando la adherencia de los recursos, introduzca el comando **pcs resource relocate show**.

29.3. DESACTIVACIÓN, ACTIVACIÓN Y PROHIBICIÓN DE LOS RECURSOS DEL CLÚSTER

Además de los comandos **pcs resource move** y **pcs resource relocate**, hay una variedad de otros comandos que puede utilizar para controlar el comportamiento de los recursos del clúster.

Desactivación de un recurso del clúster

Puede detener manualmente un recurso en ejecución y evitar que el clúster lo inicie de nuevo con el siguiente comando. Dependiendo del resto de la configuración (restricciones, opciones, fallos, etc.), el recurso puede permanecer iniciado. Si se especifica la opción **--wait**, **pcs** esperará hasta 'n' segundos para que el recurso se detenga y luego devolverá 0 si el recurso está detenido o 1 si el recurso no se ha detenido. Si no se especifica 'n', el valor por defecto es de 60 minutos.

```
pcs resource disable resource_id [--wait[=n]]
```

A partir de Red Hat Enterprise Linux 8.2, puede especificar que un recurso sea desactivado sólo si la desactivación del recurso no tendría un efecto sobre otros recursos. Asegurar que este sea el caso puede ser imposible de hacer a mano cuando se establecen relaciones complejas de recursos.

- El comando **pcs resource disable --simulate** muestra los efectos de desactivar un recurso sin cambiar la configuración del cluster.
- El comando **pcs resource disable --safe** deshabilita un recurso sólo si no hay otros recursos que se vean afectados de alguna manera, como por ejemplo, si se migra de un nodo a otro. El comando **pcs resource safe-disable** es un alias del comando **pcs resource disable --safe**.
- El comando **pcs resource disable --safe --no-strict** desactiva un recurso sólo si no se detienen o degradan otros recursos

Habilitación de un recurso de clúster

Utilice el siguiente comando para permitir que el clúster inicie un recurso. Dependiendo del resto de la configuración, el recurso puede permanecer detenido. Si especifica la opción **--wait**, **pcs** esperará hasta 'n' segundos para que el recurso se inicie y luego devolverá 0 si el recurso se ha iniciado o 1 si el recurso no se ha iniciado. Si no se especifica 'n', el valor por defecto es de 60 minutos.

```
pcs resource enable resource_id [--wait[=n]]
```

Impedir que un recurso se ejecute en un nodo concreto

Utilice el siguiente comando para evitar que un recurso se ejecute en un nodo especificado, o en el nodo actual si no se especifica ningún nodo.

```
pcs resource ban resource_id [node] [--master] [[lifetime=lifetime] [--wait[=n]]]
```

Tenga en cuenta que cuando ejecuta el comando **pcs resource ban**, éste añade una restricción de

ubicación `-INFINITY` al recurso para evitar que se ejecute en el nodo indicado. Puede ejecutar el comando **pcs resource clear** o **pcs constraint delete** para eliminar la restricción. Esto no necesariamente mueve los recursos de vuelta al nodo indicado; dónde pueden ejecutarse los recursos en ese momento depende de cómo haya configurado sus recursos inicialmente.

Si especifica el parámetro `--master` del comando **pcs resource ban**, el alcance de la restricción se limita al rol maestro y debe especificar `master_id` en lugar de `resource_id`.

Opcionalmente, puede configurar un parámetro **lifetime** para el comando **pcs resource ban** para indicar un período de tiempo que la restricción debe permanecer.

Opcionalmente se puede configurar un parámetro `--wait[=n]` para el comando **pcs resource ban** para indicar el número de segundos que se debe esperar para que el recurso se inicie en el nodo de destino antes de devolver 0 si el recurso se ha iniciado o 1 si el recurso aún no se ha iniciado. Si no se especifica `n`, se utilizará el tiempo de espera del recurso por defecto.

Forzar el inicio de un recurso en el nodo actual

Utilice el parámetro **debug-start** del comando **pcs resource** para forzar el inicio de un recurso especificado en el nodo actual, ignorando las recomendaciones del clúster e imprimiendo la salida del inicio del recurso. Esto se utiliza principalmente para depurar recursos; el arranque de recursos en un cluster se realiza (casi) siempre por Pacemaker y no directamente con un comando **pcs**. Si su recurso no se inicia, normalmente se debe a una mala configuración del recurso (que se depura en el registro del sistema), a restricciones que impiden que el recurso se inicie o a que el recurso está deshabilitado. Puede utilizar este comando para probar la configuración de los recursos, pero normalmente no debería utilizarse para iniciar recursos en un clúster.

El formato del comando **debug-start** es el siguiente.

```
pcs resource debug-start resource_id
```

29.4. PONER UN RECURSO EN MODO NO GESTIONADO

Cuando un recurso está en modo **unmanaged**, el recurso sigue en la configuración pero Pacemaker no gestiona el recurso.

El siguiente comando establece los recursos indicados en el modo **unmanaged**.

```
pcs resource unmanage resource1 [resource2] ...
```

El siguiente comando pone los recursos en modo **managed**, que es el estado por defecto.

```
pcs resource manage resource1 [resource2] ...
```

Puede especificar el nombre de un grupo de recursos con el comando **pcs resource manage** o **pcs resource unmanage**. El comando actuará sobre todos los recursos del grupo, de modo que puede poner todos los recursos de un grupo en modo **managed** o **unmanaged** con un solo comando y luego gestionar los recursos contenidos individualmente.

29.5. PONER UN CLÚSTER EN MODO DE MANTENIMIENTO

Cuando un clúster está en modo de mantenimiento, el clúster no inicia ni detiene ningún servicio hasta que se le indique lo contrario. Cuando el modo de mantenimiento finaliza, el clúster realiza una comprobación del estado actual de los servicios y, a continuación, detiene o inicia los que lo necesiten.

Para poner un clúster en modo de mantenimiento, utilice el siguiente comando para establecer la propiedad del clúster **maintenance-mode** en **true**.

```
# pcs property set maintenance-mode=true
```

Para eliminar un clúster del modo de mantenimiento, utilice el siguiente comando para establecer la propiedad del clúster **maintenance-mode** en **false**.

```
# pcs property set maintenance-mode=false
```

Puede eliminar una propiedad del clúster de la configuración con el siguiente comando.

```
propiedad pcs unset property
```

Alternativamente, puede eliminar una propiedad del clúster de una configuración dejando en blanco el campo de valor del comando **pcs property set**. Esto restablece esa propiedad a su valor por defecto. Por ejemplo, si ha establecido previamente la propiedad **symmetric-cluster** a **false**, el siguiente comando elimina el valor que ha establecido de la configuración y restaura el valor de **symmetric-cluster** a **true**, que es su valor por defecto.

```
# pcs property set symmetric-cluster=
```

29.6. ACTUALIZACIÓN DE UN CLÚSTER DE ALTA DISPONIBILIDAD RHEL

La actualización de los paquetes que componen los complementos RHEL High Availability y Resilient Storage, ya sea de forma individual o en su conjunto, puede realizarse de dos formas generales:

- *Rolling Updates*: Retirar del servicio un nodo a la vez, actualizar su software y volver a integrarlo en el clúster. Esto permite que el clúster siga prestando servicio y gestionando recursos mientras se actualiza cada nodo.
- *Entire Cluster Update*: Detenga todo el clúster, aplique las actualizaciones a todos los nodos y vuelva a iniciar el clúster.



AVISO

Es crítico que cuando se realicen procedimientos de actualización de software para los clusters de Red Hat Enterprise Linux High Availability y Resilient Storage, se asegure de que cualquier nodo que vaya a sufrir actualizaciones no sea un miembro activo del cluster antes de que se inicien dichas actualizaciones.

Para obtener una descripción completa de cada uno de estos métodos y los procedimientos a seguir para las actualizaciones, consulte [Prácticas recomendadas para aplicar actualizaciones de software a un clúster de alta disponibilidad o de almacenamiento resistente de RHEL](#).

29.7. ACTUALIZACIÓN DE NODOS REMOTOS Y NODOS INVITADOS

Si el servicio **pacemaker_remote** se detiene en un nodo remoto activo o en un nodo invitado, el clúster migrará con gracia los recursos del nodo antes de detenerlo. Esto permite realizar actualizaciones de software y otros procedimientos de mantenimiento rutinarios sin necesidad de retirar el nodo del clúster. Sin embargo, una vez que se apague **pacemaker_remote**, el clúster intentará reconectarse inmediatamente. Si **pacemaker_remote** no se reinicia dentro del tiempo de espera de monitorización del recurso, el clúster considerará la operación de monitorización como fallida.

Si desea evitar fallos de monitorización cuando el servicio **pacemaker_remote** se detiene en un nodo activo de Pacemaker Remote, puede utilizar el siguiente procedimiento para sacar el nodo del clúster antes de realizar cualquier administración del sistema que pueda detener **pacemaker_remote**

1. Detenga el recurso de conexión del nodo con el botón **pcs resource disable** **resourcenam** que moverá todos los servicios fuera del nodo. Para los nodos invitados, esto también detendrá la VM, por lo que la VM debe iniciarse fuera del clúster (por ejemplo, utilizando **virsh**) para realizar cualquier mantenimiento.
2. Realice el mantenimiento necesario.
3. Cuando esté listo para devolver el nodo al clúster, vuelva a habilitar el recurso con el botón **pcs resource enable**.

29.8. MIGRACIÓN DE MÁQUINAS VIRTUALES EN UN CLÚSTER RHEL

La información sobre las políticas de soporte para los clusters de alta disponibilidad de RHEL con miembros de clusters virtualizados se puede encontrar en [Políticas de soporte para clusters de alta disponibilidad de RHEL - Condiciones generales con miembros de clusters virtualizados](#). Como se ha indicado, Red Hat no soporta la migración en vivo de nodos de cluster activos entre hipervisores o hosts. Si necesita realizar una migración en vivo, primero tendrá que detener los servicios de cluster en la VM para eliminar el nodo del cluster, y luego volver a iniciar el cluster después de realizar la migración. Los siguientes pasos describen el procedimiento para eliminar una máquina virtual de un clúster, migrar la máquina virtual y restaurar la máquina virtual en el clúster.

Este procedimiento se aplica a las máquinas virtuales que se utilizan como nodos de clúster completos, no a las máquinas virtuales gestionadas como recursos de clúster (incluidas las máquinas virtuales utilizadas como nodos invitados) que pueden migrarse en vivo sin precauciones especiales. Para obtener información general sobre el procedimiento más completo necesario para actualizar los paquetes que componen los complementos RHEL High Availability y Resilient Storage, ya sea individualmente o en conjunto, consulte [Prácticas recomendadas para aplicar actualizaciones de software a un clúster RHEL High Availability o Resilient Storage](#).



NOTA

Antes de realizar este procedimiento, considere el efecto que tiene la eliminación de un nodo del clúster sobre el quórum del mismo. Por ejemplo, si tiene un clúster de tres nodos y elimina un nodo, su clúster sólo puede soportar el fallo de un nodo más. Si un nodo de un clúster de tres nodos ya está caído, al eliminar un segundo nodo se perderá el quórum.

1. Si es necesario realizar alguna preparación antes de detener o mover los recursos o el software que se ejecuta en la VM para migrar, realice esos pasos.
2. Ejecute el siguiente comando en la VM para detener el software de cluster en la VM.

```
# pcs cluster stop
```

3. Realice la migración en vivo de la VM.
4. Inicie los servicios de cluster en la VM.

```
# pcs cluster start
```

CAPÍTULO 30. CONFIGURACIÓN DE CLUSTERS DE RECUPERACIÓN DE DESASTRES

Un método para proporcionar recuperación de desastres para un clúster de alta disponibilidad es configurar dos clústeres. Puede configurar un clúster como su clúster de sitio primario, y el segundo clúster como su clúster de recuperación de desastres.

En circunstancias normales, el clúster primario está ejecutando los recursos en modo de producción. El clúster de recuperación de desastres también tiene todos los recursos configurados y los ejecuta en modo degradado o no los ejecuta en absoluto. Por ejemplo, puede haber una base de datos que se ejecute en el clúster primario en modo promocionado y que se ejecute en el clúster de recuperación de desastres en modo degradado. La base de datos en esta configuración estaría configurada para que los datos se sincronicen desde el sitio primario al de recuperación de desastres. Esto se hace a través de la propia configuración de la base de datos y no a través de la interfaz de comandos **pcs**.

Cuando el clúster primario se cae, los usuarios pueden utilizar la interfaz de comandos **pcs** para transferir manualmente los recursos al sitio de recuperación de desastres. A continuación, pueden iniciar sesión en el sitio de desastre y promover e iniciar los recursos allí. Una vez que el clúster primario se haya recuperado, los usuarios pueden utilizar la interfaz de comandos **pcs** para trasladar manualmente los recursos de vuelta al sitio primario.

A partir de Red Hat Enterprise Linux 8.2, puede utilizar el comando **pcs** para mostrar el estado del cluster del sitio primario y del de recuperación de desastres desde un solo nodo en cualquiera de los dos sitios.

30.1. CONSIDERACIONES SOBRE LOS CLUSTERS DE RECUPERACIÓN DE DESASTRES

Cuando planifique y configure un sitio de recuperación de desastres que gestionará y supervisará con la interfaz de comandos **pcs**, tenga en cuenta las siguientes consideraciones.

- El sitio de recuperación de desastres debe ser un clúster. Esto permite configurarlo con las mismas herramientas y procedimientos similares a los del sitio primario.
- Los clústeres primario y de recuperación de desastres se crean mediante comandos independientes de **pcs cluster setup**.
- Los clusters y sus recursos deben estar configurados para que los datos estén sincronizados y sea posible la conmutación por error.
- Los nodos del cluster en el sitio de recuperación no pueden tener los mismos nombres que los nodos del sitio primario.
- El usuario **pcs hacluster** debe estar autenticado para cada nodo en ambos clusters en el nodo desde el que se ejecutarán los comandos de **pcs**.

30.2. VISUALIZACIÓN DEL ESTADO DE LOS CLUSTERS DE RECUPERACIÓN (RHEL 8.2 Y POSTERIORES)

Para configurar un clúster primario y uno de recuperación de desastres de manera que pueda visualizar el estado de ambos clústeres, realice el siguiente procedimiento.



NOTA

La configuración de un clúster de recuperación de desastres no configura automáticamente los recursos ni replica los datos. Esos elementos deben ser configurados manualmente por el usuario.

En este ejemplo:

- El cluster primario se llamará **PrimarySite** y estará formado por los nodos **z1.example.com** y **z2.example.com**.
- El clúster del sitio de recuperación de desastres se llamará **DRsite** y estará formado por los nodos **z3.example.com** y **z4.example.com**.

Este ejemplo configura un clúster básico sin recursos ni cercas configuradas.

1. Autenticar todos los nodos que se utilizarán para ambos clústeres.

```
[root@z1 ~]# pcs host auth z1.example.com z2.example.com z3.example.com
z4.example.com -u hacluster -p password
z1.example.com: Authorized
z2.example.com: Authorized
z3.example.com: Authorized
z4.example.com: Authorized
```

2. Cree el clúster que se utilizará como clúster primario e inicie los servicios de clúster para el clúster.

```
[root@z1 ~]# pcs cluster setup PrimarySite z1.example.com z2.example.com --start
{...}
Cluster has been successfully set up.
Starting cluster on hosts: 'z1.example.com', 'z2.example.com'...
```

3. Cree el clúster que se utilizará como clúster de recuperación de desastres e inicie los servicios de clúster para el clúster.

```
[root@z1 ~]# pcs cluster setup DRSite z3.example.com z4.example.com --start
{...}
Cluster has been successfully set up.
Starting cluster on hosts: 'z3.example.com', 'z4.example.com'...
```

4. Desde un nodo del clúster primario, configure el segundo clúster como sitio de recuperación. El sitio de recuperación se define con el nombre de uno de sus nodos.

```
[root@z1 ~]# pcs dr set-recovery-site z3.example.com
Sending 'disaster-recovery config' to 'z3.example.com', 'z4.example.com'
z3.example.com: successful distribution of the file 'disaster-recovery config'
z4.example.com: successful distribution of the file 'disaster-recovery config'
Sending 'disaster-recovery config' to 'z1.example.com', 'z2.example.com'
z1.example.com: successful distribution of the file 'disaster-recovery config'
z2.example.com: successful distribution of the file 'disaster-recovery config'
```

5. Compruebe la configuración de la recuperación de desastres.

```
[root@z1 ~]# pcs dr config
```

```
Local site:
```

```
Role: Primary
```

```
Remote site:
```

```
Role: Recovery
```

```
Nodes:
```

```
z1.example.com
```

```
z2.example.com
```

6. Compruebe el estado del clúster primario y del clúster de recuperación de desastres desde un nodo del clúster primario.

```
[root@z1 ~]# pcs dr status
```

```
--- Local cluster - Primary site ---
```

```
Cluster name: PrimarySite
```

```
WARNINGS:
```

```
No stonith devices and stonith-enabled is not false
```

```
Cluster Summary:
```

```
* Stack: corosync
```

```
* Current DC: z2.example.com (version 2.0.3-2.el8-2c9cea563e) - partition with quorum
```

```
* Last updated: Mon Dec 9 04:10:31 2019
```

```
* Last change: Mon Dec 9 04:06:10 2019 by hacluster via crmd on z2.example.com
```

```
* 2 nodes configured
```

```
* 0 resource instances configured
```

```
Node List:
```

```
* Online: [ z1.example.com z2.example.com ]
```

```
Full List of Resources:
```

```
* No resources
```

```
Daemon Status:
```

```
corosync: active/disabled
```

```
pacemaker: active/disabled
```

```
pcsd: active/enabled
```

```
--- Remote cluster - Recovery site ---
```

```
Cluster name: DRSite
```

```
WARNINGS:
```

```
No stonith devices and stonith-enabled is not false
```

```
Cluster Summary:
```

```
* Stack: corosync
```

```
* Current DC: z4.example.com (version 2.0.3-2.el8-2c9cea563e) - partition with quorum
```

```
* Last updated: Mon Dec 9 04:10:34 2019
```

```
* Last change: Mon Dec 9 04:09:55 2019 by hacluster via crmd on z4.example.com
```

```
* 2 nodes configured
```

```
* 0 resource instances configured
```

```
Node List:
```

```
* Online: [ z3.example.com z4.example.com ]
```


Full List of Resources:

* No resources

Daemon Status:

corosync: active/disabled

pacemaker: active/disabled

pcsd: active/enabled

Para conocer otras opciones de visualización de una configuración de recuperación de desastres, consulte la pantalla de ayuda del comando **pcs dr**.