Red Hat Enterprise Linux 6

Administra๛๛o de Cluster

Configurando e Gerenciando o Componente de Alta Disponibilidade

Last Updated: 2017-10-13

Red Hat Enterprise Linux 6 Administra꜠꜠o de Cluster

Configurando e Gerenciando o Componente de Alta Disponibilidade

Red Hat Serviços de Conteúdo de Engenharia
docs-need-a-fix@redhat.com

Nota Legal

Resumo

Configurando e Gerenciando o Complemento de Alta Disponibilidade descreve a configuração e gerenciamento do Complemento de Alta Disponibilidade para o Red Hat Enterprise Linux 6.

Índice

INTRODUÇÃO

Este documento fornece informações sobre a instalação, configuração e gerenciamento dos componentes do Complemento de Alta Disponibilidade da Red Hat. Os componentes do Complemento de Alta Disponibilidade da Red Hat lhe permitem conectar a um grupo de computadores (chamados *nós* ou *membros*) para trabalhar juntos como um cluster. Neste documento, o uso da palavra *cluster* ou *clusters* é para se referir a um grupo de computadores rodando o Complemento de Alta Disponibilidade da Red Hat.

O público deste documento deve possuir conhecimento avançado de trabalho do Red Hat Enterprise Linux e entender conceitos de cluster, armazenamento e computação de servidores.

Para mais informações sobre o Red Hat Enterprise Linux 6, consulte os seguintes recursos:

- *Guia de Instalação do Red Hat Enterprise Linux*— Fornece informações sobre a instalação do Red Hat Enterprise Linux 6.
- *Guia de Implantação do Red Hat Enterprise Linux*— Fornece informações sobre a implantação, configuração e administração do Red Hat Enterprise Linux 6.

Para mais informações sobre o Complemento de Alta Disponibilidade e produtos relacionados para o Red Hat Enterprise Linux 6, consulte os seguintes recursos:

- *Visão Geral do Complemento de Alta Disponibilidade*— Fornece uma visão geral de alto nível do Complemento de Alta Disponibilidade da Red Hat.
- *Administração do Gerenciador de Volume Lógico*— Fornece uma descrição do Gerenciador de Volume Lógico (LVM), incluindo informações sobre rodar o LVM em um ambiente de cluster.
- *Global File System 2: Configuração e Administração*— Fornece informações sobre instalação, configuração e como manter o Red Hat GFS2 (Red Hat Global File System 2), que é incluido na Complemento de Armazenamento Resiliente.
- *DM Multipath* — Fornece informações sobre o uso do recurso Mapeador de Dispositivos Multipath do Red Hat Enterprise Linux 6.
- *Administração do Balanceador de Carga*(Load Balancer) — Fornece informações sobre configuração de sistemas de alta performance e serviços com o Complemento do Balanceador de Carga, um conjunto de componentes de software integrados que fornece Linux Virtual Servers (LVS) para balanceamento de carga de IP por todo um conjunto de servidores reais.
- *Notas de Lançamento*— Fornece informações sobre os lançamentos atuais dos produtos da Red Hat.

Documentação do Complemento de Alta Disponibilidade e outros documentos da Red Hat estão disponíveis em HTML, PDF e versões RPM no CD de documentação do Red Hat Enterprise Linux e online em http://docs.redhat.com/docs/en-US/index.html.

1. FEEDBACK

Se você encontrar um erro de digitação ou se você acha que é possível melhorar este manual, nós gostariamos de saber. Por favor envie um relatório no Bugzilla (http://bugzilla.redhat.com/bugzilla/) sobre o componente **doc-Cluster_Administration**.

Certifique-se de mencionar o identificador do manual:

```
Cluster_Administration(EN)-6 (2013-2-15T16:26)
```

Mencionando o identificador do manual, nós saberemos exatamente a versão do guia que você possui.

Se você tiver uma sugestão para melhorar a documentação, tente ser o mais específico possível. Se você encontrar um erro, por favor inclua o número da seção e parte do texto para podermos identificar com mais facilidade.

CAPÍTULO 1. CONFIGURAÇÃO DO COMPLEMENTO DE ALTA DISPONIBILIDADE DA RED HAT E VISÃO GERAL DO GERENCIAMENTO

O Complemento de Alta Disponibilidade Red Hat lhe permite conectar a um grupo de computadores (chamados *nós* ou *membros*) para trabalhar juntos como um cluster. Você pode usar o Complemento de Alta Disponibilidade Red Hat para atender suas necessidades de cluster (por exemplo, configurar um cluster para compartilhar arquivos em um sistema de arquivos GFS2 ou configurar um serviço de failover).

NOTA

Para obter informações sobre as melhores práticas de implementação e atualização dos clusters do Red Hat Enterprise Linux, utilizando o Complemento de Alta Disponiblidade e Sistema de Arquivo Global da Red Hat 2 (GFS2), consulte o artigo Red Hat Enterprise Linux Cluster, High Availability, e GFS Deployment Best Practices" no Red Hat Customer Portal em . https://access.redhat.com/kb/docs/DOC-40821.

Este capítulo fornece um resumo dos recursos de documentação e atualizações que foram adicionados ao Complemento de Alta Disponibilidade Red Hat desde o lançamento inicial do Red Hat Enterprise Linux 6, seguido por uma visão geral de configuração e gerenciamento do Complemento de Alta Disponibilidade Red Hat.

1.1. RECURSOS NOVOS E MODIFICADOS

Esta seção lista os recursos novos e modificados da documentação do Complemento de Alta Disponibilidade da Red Hat que foram adicionados desde o lançamento inicial do Red Hat Enterprise Linux 6.

1.1.1. Recursos Novos e Alterados para o Red Hat Enterprise Linux 6.1

O Red Hat Enterprise Linux 6.1 inclui a seguinte documentação e atualização de recursos e mudanças.

- A partir do lançamento do Red Hat Enterprise Linux 6.1 e posteriores, o Complemento de Alta Disponibilidade Red Hat fornece suporte para SNMP traps (sinais). Para informações sobre configurar SNMP traps com o Complemento de Alta Disponibilidade Red Hat, consulte o Capítulo 10, *Configuração do SNMP com Complemento de Alta Disponibilidade da Red Hat*.
- A partir do lançamento do Red Hat Enterprise Linux 6.1 e posteriores, o Complemento de Alta Disponibilidade Red Hat fornece suporte para o comando de configuração de cluster **ccs**. Para informações sobre o comando **ccs**, consulte o Capítulo 5, *Configurando o Complemento de Alta Disponibilidade da Red Hat com o comando* **ccs**.
- A documentação para configurar e gerenciar o software Complemento de Alta Disponibilidade Red Hat usando o Conga foi atualizado para refletir as telas do Conga e suporte de recursos.
- Para o lançamento do Red Hat Enterprise Linux 6.1 e posteriores, usar o **ricci** requer uma senha para a primeira vez que você propagar configurações atualizadas do cluster a partir de qualquer nó especifico. Para informações sobre o **ricci** consulte a Seção 2.13, "Considerações para o **ricci**".
- Você pode agora especificar uma política de falha *Restart-Disable* para um serviço, indicando que um sistema deveria tentar reiniciar o serviço se este falhar, mas se a reinicialização do

serviço falhar, o serviço será desabilitado em vez de ser movido para outro host no cluster. Este recurso está documentado na Seção 3.10, "Adicionar um Serviço de Cluster ao Cluster" e no Apêndice B, *Parâmetros dos Recursos de Alta Disponibilidade*.

- Agora você pode configurar uma sub árvore independente como não crítica, indicando que se o recurso falhar então somente aquele recurso é desabilitado. Para informações sobre este recurso veja a Seção 3.10, "Adicionar um Serviço de Cluster ao Cluster" e a Seção C.4, "Recuperação de Falhas e Sub Árvores Independentes".

- Este documento agora inclui o novo capítulo, Capítulo 9, *Diagnosticando e Corrigindo Problemas em um Cluster*.

Além disso, pequenas correções e clarificações foram feitas através de todo o documento.

1.1.2. Recursos Novos e Modificados para o Red Hat Enterprise Linux 6.2.

O Red Hat Enterprise Linux 6.2 inclui a seguinte documentação e atualização de recursos e mudanças.

- O Red Hat Enterprise Linux agora fornece suporte para o Samba em Cluster em execução em uma configuração ativa/ativa. Para obter mais informações sobre a configuração do Samba em cluster, consulte o Capítulo 11, *Configurações do Samba em Cluster*.

- Embora qualquer usuário capaz de autenticar no sistema que esteja hospedando o **luci** possa se autenticar no **luci**, desde o Red Hat Enterprise Linux 6.2, somente o usuário root no sistema que esteja sendo executado o **luci** poderá acessar qualquer um dos componentes do **luci** até que um administrador (o usuário root ou um usuário com permissão de administrador) defina permissões para aquele usuário. Para informações sobre como configurar permissões do **luci** para usuários, consulte o Seção 3.3, "Controlando o Acesso ao luci".

- Os nós em um cluster podem se comunicar entre si, utilizando o mecanismo de transporte unicast UDP. Para mais informações sobre como configurar o unicast UDP, consulte o Seção 2.12, "Tráfego do Unicast UDP".

- Você pode agora configurar alguns aspectos do comportamento do **luci** através do arquivo **/etc/sysconfig/luci**. Por exemplo, você pode configurar especialmente o único endereço de IP onde o **luci** está sendo servido. Para obter mais informações sobre como configurar o único endereço IP onde o **luci**está sendo servido, consulte o Tabela 2.2, "Porta IP habilitada em um computador que roda o **luci**". Para obter informações sobre o arquivo **/etc/sysconfig/luci** em geral, consulte o Seção 2.4, "Configurando **luci** com **/etc/sysconfig/luci**".

- O comando **ccs** agora inclui a opção **--lsfenceopts** a qual imprime uma lista de dispositivos de fence disponíveis, **--lsfenceopts** e a opção *fence_type* que imprime cada tipo de fence disponível. Para informações sobre estas opções, consulte o Seção 5.6, "Lista de Dispositivos de Fence e Opções de Dispositivos de Fence".

- O comando **ccs** agora inclui a opção **--lsserviceopts** que imprime uma lista de serviços de cluster disponíveis atualmente, e a opção **--lsserviceopts** *service_type* que imprime uma lista de opções que você pode especificar para um tipo de serviço específico. Para informações sobre estas opções, consulte o Seção 5.11, "Listando Serviços de Cluster Disponíveis".

- O lançamento Red Hat Enterprise Linux 6.2 fornece suporte para o agente de fence VMware (SOAP Interface). Para mais informações sobre os parâmetros de dispositivo do fence, consulte o Apêndice A, *Parâmetros de Dispositos Fence*.

- O lançamento Red Hat Enterprise Linux 6.2 fornece suporte para o agente de fenceRHEV-M REST API no RHEV 3.0 ou posteriores. Para mais informações sobre os parâmetros de dispositivo do fence, consulte o Apêndice A, *Parâmetros de Dispositos Fence*.

- A partir do lançamento do Red Hat Enterprise Linux 6.2 , quando você configura uma máquina virtual em um cluster com o comando **ccs**, você pode usar a opção **--addvm** (ao invés da opção **addservice**). Isto assegura que o recurso **vm** está definido diretamente sob o nó de configuração **rm** no arquivo de configuração do cluster. Para informações sobre recursos de máquina virtual de configuração com o comando **ccs** consulte o Seção 5.12, "Recursos de Máquina Virtual".

- Este documento inclui um novo apêndice Apêndice D, *Checagem de Recursos de Serviço de Cluster e Expiração de Failover*. Ele descreve como o **rgmanager** monitora o estado dos recursos do cluster, e como modificar o estado do intervalo de verificação. O apêndice também descreve o parâmetro do serviço **__enforce_timeouts**, o qual indica que um timeout para uma operação deve causar falha no serviço.

- O documento inclui uma nova seção, Seção 2.3.3, "Configurando o Firewall iptables para Permitir Componentes do Cluster.". Esta seção exibe o filtro que você pode utilizar para permitir tráfego de multicast através do firewall do **iptables** para diversos componentes de cluster.

Além disso, pequenas correções e clarificações foram feitas através de todo o documento.

1.1.3. Recursos Novos e Alterados para o Red Hat Enterprise Linux 6.3

O Red Hat Enterprise Linux 6.3 inclui a seguinte documentação e atualização de recursos e mudanças.

- O lançamento do Red Hat Enterprise Linux 6.3 fornece suporte para o agente de recurso **condor**. Para mais informações sobre os parâmetros de recursos HA, consulte Apêndice B, *Parâmetros dos Recursos de Alta Disponibilidade*.

- Este documento agora inclui o novo capítulo, Apêndice F, *Alta Disponibilidade LVM (HA-LVM)*.

- Informações neste documento explicam quais mudanças de configuração requerem um reinício de cluster. Para um resumo destas mudanças, consulte Seção 9.1, "Mudança de configuração Não é efetuada".

- A documentação agora observa que existe um timeout de espera para o **luci** que o retira depois de 15 minutos de inatividade. Para mais informações sobre iniciar um **luci**, consulte Seção 3.2, "Iniciando o **luci**".

- O dispositivo de fence **fence_ipmilan** suporta um parâmetro de nível de privilégio. Para informações sobre parâmetros fence, consulte o Apêndice A, *Parâmetros de Dispositos Fence*.

- Este documento agora inclui o novo capítulo, Seção 2.14, "Configurando as Máquinas Virtuais em um Ambiente Cluster".

- Este documento agora inclui o novo capítulo, Seção 4.6, "Fazendo um backup e Recuperando a Configuração do luci".

- Este documento agora inclui o novo capítulo, Seção 9.4, "O daemon do Cluster trava" .

- Este documento fornece informações sobre as opções de depuração de configuração em Seção 5.14.4, "Autenticando", Seção 7.7, "Configuração das Opções de Depuração" , e Seção 9.13, "Autenticação de Depug para o Gerenciador de Bloqueio Distribuído (DLM) Precisa

ser Habilitada".

- Desde o Red Hat Enterprise Linux 6.3, o usuário root ou um usuário que tenha recebido permissões de um administrador do **luci** também poderão utilizar a interface do **luci** para adicionar usuários ao sistema, como descrito em Seção 3.3, "Controlando o Acesso ao luci" .

- Desde o Red Hat Enterprise Linux 6.3, o comando **ccs** valida a configuração de acordo com o esquema do cluster em **/usr/share/cluster/cluster.rng** no nó que você espeficar com a opção **-h**. Anteriormente, o comando **ccs** sempre usava o esquema do cluster que era empacotado com o próprio comando **ccs** , **/usr/share/ccs/cluster.rng** no sistema local. Para informações sobre a validação de configuração, consulte Seção 5.1.6, "Validação de Configuração".

- As tabelas que descrevem os parâmetros de dispositivo do fence em Apêndice A, *Parâmetros de Dispositos Fence* e tabelas de descrevem os parâmetros de recurso do HA em Apêndice B, *Parâmetros dos Recursos de Alta Disponibilidade* agora incluem os nomes daqueles parâmetros como aparecem no arquivo **cluster.conf**.

Além disso, pequenas correções e clarificações foram feitas através de todo o documento.

1.1.4. Recursos Novos e Modificados para o Red Hat Enterprise Linux 6.4

Red Hat Enterprise Linux 6.4 inclui a seguinte documentação e atualizações de recursos e modificações.

- O lançamento Red Hat Enterprise Linux 6.4 fornece suporte para o agente de fence Eaton Network Power Controller (SNMP Interface), HP BladeSystem, e o IBM iPDU. Para mais informações sobre os parâmetros de dispositivo do fence, consulte o Apêndice A, *Parâmetros de Dispositos Fence*.

- Apêndice B, *Parâmetros dos Recursos de Alta Disponibilidade* agora fornece uma descrição de agente de recurso do Servidor NFS.

- Desde Red Hat Enterprise Linux 6.4, o usuário root ou um usuário que recebeu permissões de administrador do **luci** também podem utilizar a interface do **luci** para remover usuários de sistemas. Isto é documentado em Seção 3.3, "Controlando o Acesso ao luci" .

- Apêndice B, *Parâmetros dos Recursos de Alta Disponibilidade* fornece uma descrição do novo parâmetro **nfsrestart** para o Filesystem e recursos do GFS2 HA.

- Esta documentação inclui uma nova seção Seção 5.1.5, "Comandos que Sobrescrevem Configurações Anteriores".

- Seção 2.3, "Habilitando Portas IP" agora inclui informações sobre filtragem do firewall **iptables** para **igmp**.

- O agente de fence IPMI LAN agora suporta um parâmetro para configurar o nível de privilégio no dispositivos IPMI, como documentado em Apêndice A, *Parâmetros de Dispositos Fence*

- Além do modo de vinculação Ethernet, os modos 0 e 2 são agora suportados pela comunicação entre nós em um cluster. A sugestão de troubleshooting neste documenta que sugere que você se certifique se está usando somente modos de vinculação suportada agora observa isto.

- Os dispositivos de rede marcados com VLAN são agora suportados para a comunicação de pulsação do coração do cluster. A nota sobre troubleshooting indica que a frase "isto não é suportado" foi removida deste documennto.

- O Red Hat High Availability Add-On agora suporta a configuração de protocolo de anel redundante. Para informações em geral sobre como utilizar o recurso e configuração do arquivo de configuração **cluster.conf** consulte o Seção 7.6, "Configurando o Protocolo de Anel Redundante". Para informações sobre o protocolo de anel redundante com **luci**, consulte o Seção 3.5.4, "Configurando, Protocolo de Anel Redundante" . Para informações sobre como configurar o protocolo de anel redundante com o comando **ccs** consulte o Seção 5.14.5, "Configurando o Protocolo de Anel Redundante".

Além disso, pequenas correções e clarificações foram feitas através de todo o documento.

1.2. CONFIGURAÇÕES BÁSICAS

Para configurar um cluster, você deve conectar os nós a certos hardwares de cluster e configurar os nós dentro de um ambiente de cluster. Configurando e gerenciando o Complemento de Alta Disponibilidade consiste dos seguintes passos básicos:

1. Definir um hardware. Consulte a Seção 1.3, "Configurando o Hardware".
2. Instalando o software Complemento de Alta Disponibilidade. Consulte a Seção 1.4, "Instalando o software de Alta Disponibilidade da Red Hat".
3. Configurando o software Complemento de Alta Disponibilidade. Consulte a Seção 1.5, "Configurando o software do Complemento de Alta Disponibilidade da Red Hat".

1.3. CONFIGURANDO O HARDWARE

Configurar o hardware consiste em conectar nós do cluster a outros hardwares requeridos para rodar o Complemento de Alta Disponibilidade da Red Hat. A quantidade e tipo de hardware varia de acordo com o propósito e requerimentos de disponibilidade do cluster. Tipicamente, um cluster de nível corporativo requer os seguintes tipos de hardware (veja a Figura 1.1, "Visão Geral do Hardware do Complemento de Alta Disponibilidade da Red Hat"). Para considerações sobre o hardware e outras referências de configuração de cluster, consulte Capítulo 2, *Antes de configurar o Complemento de Alta Disponibilidade da Red Hat* ou consulte um representante autorizado Red Hat.

- Nós em cluster — Os computadores são capázeis de executar um Red Hat Enterprise Linux 6 software, com o mínimo de 6GB de RAM.
- Switch Ethernet ou hub para rede pública — Isto é requerido para o cliente acessar o cluster.
- Switch Ethernet ou hub para rede privada — Isto é requerido para comunicação entre os nós do cluster e outros hardwares de cluster tais como switches de energia de rede e switches Fibre Channel.
- Switch de Energia de Rede — É recomendado para realizar fences em um cluster de nível corporativo.
- Fibre Channel switch — Fornece acesso a armazenamento Fibre Channel. Outras opções estão disponíveis para armazenamento de acordo com o tipo de interface de armazenamento; por exemplo, iSCSI. Um switch Fibre Channel pode ser configurado para realizar fencing.
- Armazenamento — Algum tipo de armazenamento é requerido para um cluster. O tipo requerido depende do propósito do cluster.

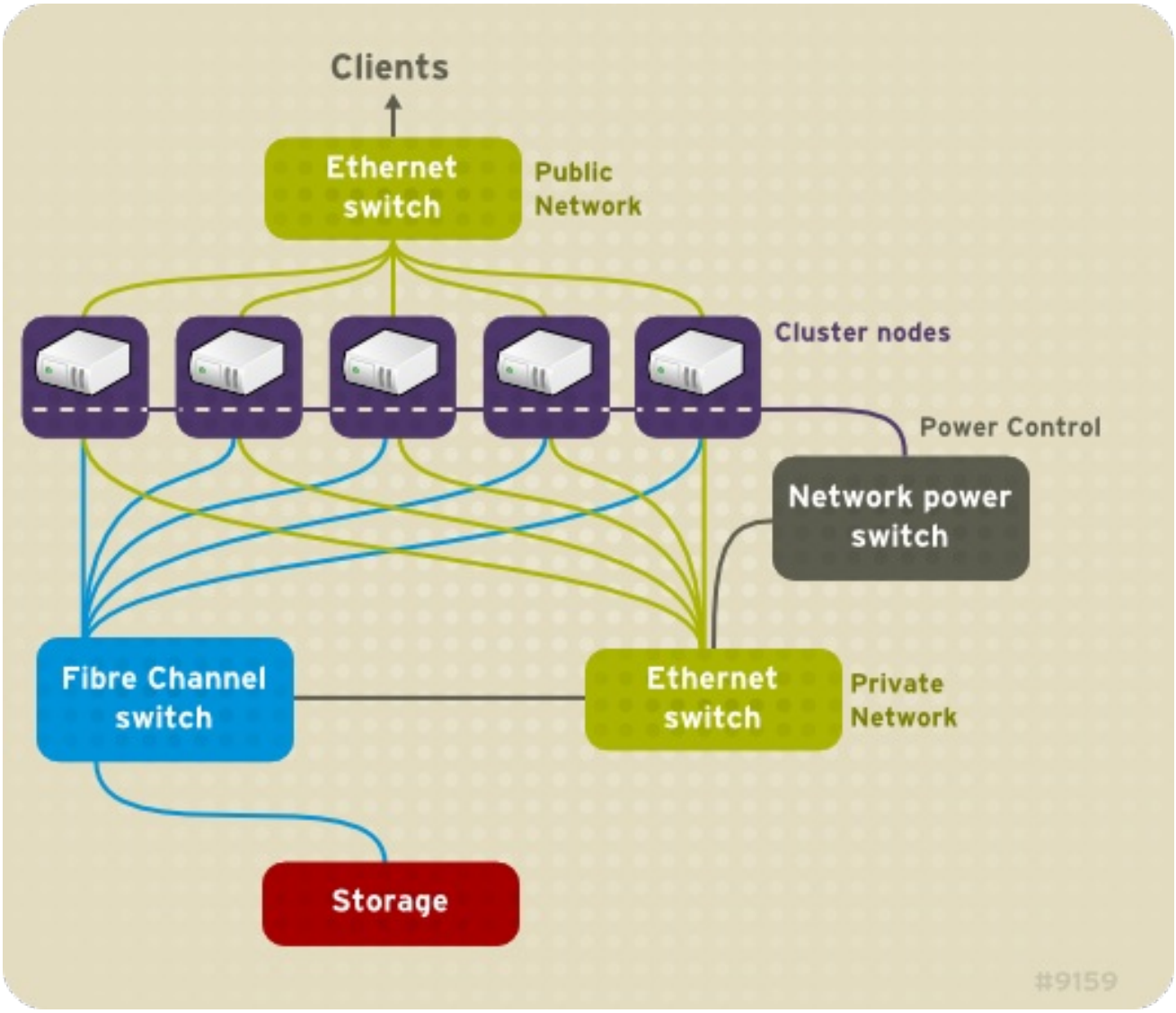


Figura 1.1. Visão Geral do Hardware do Complemento de Alta Disponibilidade da Red Hat

1.4. INSTALANDO O SOFTWARE DE ALTA DISPONIBILIDADE DA RED HAT

Para instalar o Red Hat o software do High Availability Add-On, você precisa ter direitos para o software. Se você estiver utilizando o GUI de configuração do **luci**, você pode deixá-lo instalar o software de cluster. Se você estiver utilizando outras ferramentas para configurar o cluster, assegure e instale o software como você faria com o Red Hat Enterprise Linux software.

VocÊ pode usar o seguinte comando **`yum install`** para instalar os pacotes de software do High Availability Add-On:

```
# yum install rgmanager lvm2-cluster gfs2-utils
```

Note que somente instalar o **`rgmanager`** colocará todas as dependências necessárias para criar um cluster de HA a partir do canal do HighAvailability. Os pacotes **`lvm2-cluster`** e o **`gfs2-utils`** são parte do canal ResilientStorage e pode não ser requisitado pelo seu site.

1.4.1. Atualizando o software de Alta Disponibilidade Red Hat

É possível atualizar o software de cluster em um lançamento maior do Red Hat Enterprise Linux sem tirar o cluster do ar. Para isso é requerida a desabilitação do software de cluster em um nó do host por vez, atualizando o software e reiniciando o software de cluster neste host.

1. Desligue todos os serviços de cluster em um nó do cluster único. Para instruções de como parar o software de cluster em um nó, consulte a Seção 8.1.2, "Parando um Software de Cluster". Pode ser desejável realocar manualmente os serviços gerenciados de cluster e máquinas virtuais fora do host antes de para-los.

2. Execute o comando **`yum update`** para atualizar pacotes instalados.

3. Reinicie o nó no cluster ou reinicie os serviços de cluster manualmente. Para instruções sobre reiniciar os software de cluster em um nó, consulte a Seção 8.1.1, "Iniciar o Software do Cluster".

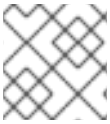
1.5. CONFIGURANDO O SOFTWARE DO COMPLEMENTO DE ALTA DISPONIBILIDADE DA RED HAT

Configurando o software do Complemento de Alta Disponibilidade da Red Hat consiste em usar ferramentas para especificar o relacionamento entre os componentes do cluster. As seguintes ferramentas de configuração de cluster estão disponíveis com o Complemento de Alta Disponibilidade da Red Hat:

- **Conga** — Esta é um compreensiva interface de usuário para instalar, configurar e gerenciar o Complemento de Alta Disponibilidade da Red Hat. Consulte o Capítulo 3, *Configurando o Complemento de Alta Disponibilidade da Red Hat com o **Conga*** e o Capítulo 4, *Gerenciando o Complemento de Alta Disponibilidade Red Hat com o **Conga*** para informações sobre configurar e gerenciar o Complemento de Alta Disponibilidade com o **Conga**.

- O comando **`ccs`** — Este comando configura e gerencia o Complemento de Alta Disponibilidade da Red Hat. Consulte o Capítulo 5, *Configurando o Complemento de Alta Disponibilidade da Red Hat com o comando **ccs*** e o Capítulo 6, *Gerenciando o Complemento de Alta Disponibilidade da Red Hat com o **ccs*** para informações sobre configurar e gerenciar o Complemento de Alta Disponibilidade com o comando **`ccs`**.

- Ferramentas da Linha de Comando — Este é um conjunto de ferramentas de linha de comando para configuração e gerenciamento do Complemento de Alta Disponibilidade da Red Hat. Consulte o Capítulo 7, *Configurando o Complemento de Alta Disponibilidade da Red Hat com as Ferramentas da Linha de Comando* e o Capítulo 8, *Gerenciando o Complemento de Alta Disponibilidade da Red Hat com Ferramentas da Linha de Comando.* para informações sobre configurar e gerenciar um cluster com as ferramentas de linha de comando. Consulte o Apêndice E, *Resumo das Ferramentas da Linha de Comando*para um resumo das ferramentas de linha de comando preferidas.



NOTA

O **`system-config-cluster`** não está disponível no RHEL 6.

CAPÍTULO 2. ANTES DE CONFIGURAR O COMPLEMENTO DE ALTA DISPONIBILIDADE DA RED HAT

Este capítulo descreve as tarefas a serem realizadas e as considerações a serem feitas antes de instalar e configurar o Complemento de Alta Disponibilidade Red Hat e consiste das seguintes seções.

IMPORTANTE

Certifique-se que sua implantação do Complemento de Alta Disponibilidade atenda suas necessidades e possa ser suportada. Consulte um representante autorizado Red Hat para verificar suas configurações antes da implementação. Além disso, separe um tempo para testar as configurações para testar modos de falhas.

2.1. CONSIDERAÇÕES GERAIS DE CONFIGURAÇÃO

Você pode configurar o Complemento de Alta Disponibilidade em diversas maneiras que se adequem às suas necessidades. Leve em conta as seguintes considerações gerais quando você planejar, configurar e implementar sua instalação.

Número de nós em clusters suportados

O número máximo de nós de clusters suportados pelo Complemento de Alta Disponibilidade é 16.

Cluster de locais únicos

Somente clusters de locais únicos são suportados neste momento. Clusters espalhados por múltiplas localidades físicas não são formalmente suportados. Para mais detalhes e para discutir sobre cluster em múltiplas locações, por favor converse com seu representante de vendas ou

suporte Red Hat.

GFS2

Apesar de que um sistema GFS2 possa ser implementado em um sistema isolado ou como parte de uma configuração de cluster, a Red Hat não suporta o uso do GFS2 como um sistema de arquivos de nó único. A Red Hat suporta um número de sistemas de arquivos de nó único de alto desempenho que são otimizados para nó único, e portanto tem um custo geralmente menor que um sistema de arquivos em cluster. A Red Hat recomenda usar em preferência estes sistemas de arquivos do que o GFS2 em casos onde somente um nó único precisa ser montado no sistema de arquivos. A Red Hat continuará a dar suporte à sistemas de arquivos de nós únicos GFS2 para clientes existentes.

Quando você configura um sistema de arquivos GFS2 como um sistema de arquivos de cluster, você deve assegurar que todos os nós no cluster têm acesso ao sistema de arquivos compartilhados. Configurações de cluster assimétricas nos quais os nós tem acesso ao sistema de arquivos e outros não são suportados. Isto não requer que todos os nós montem o sistema de arquivos GFS2.

Configuração de hardware sem ponto único de falha

Os clusters podem incluir uma Matriz RAID de controlador duplo, múltiplos canais de redes ligadas, múltiplos caminhos entre membros do cluster e armazenamento e sistemas de fonte de energia ininterrupta redundante (UPS - uninterruptible power supply) para assegurar que não haja uma falha única que resulte na queda das aplicações ou perda de dados.

Alternativamente, um cluster de baixo custo pode ser configurado para fornecer menos disponibilidade do que um cluster sem ponto único de falha. Por exemplo, você pode configurar um cluster com uma matriz RAID de controlador único e somente um canal único Ethernet.

Certas alternativas de baixo custo, tal como controladores de host RAID, RAID software sem suporte de cluster e configurações SCSI paralelas multi-initiator não são compatíveis ou apropriadas para uso como armazenamento de cluster compartilhado.

Garantia de Integridade de Dados

Para garantir integridade de dados, somente um nó pode rodar um serviço de cluster e acessar os dados do serviço de cluster por vez. O uso de switches de energia na configuração de hardware do cluster permite a um nó fazer um ciclo de energia em outro nó antes de iniciar os serviços de Alta Disponibilidade em outro nó durante um processo de failover. Isso previne que dois nós acessem simultaneamente os mesmos dados e os tornem corrompidos. *Dispositivos Fence* (soluções de hardware e software que remotamente ligam, desligam e reinicializam nós no cluster) são usados para garantir integridade dos dados quando em condições de falha.

Ligação de Canal Ethernet

O quorum de cluster e a saúde do nó são determinados pela comunicação de mensagens entre nós do cluster via Ethernet. Além disso, nós do cluster usam a Ethernet para uma variedade de outras funções críticas do cluster (por exemplo, fencing). Com a ligação de canal Ethernet, múltiplas interfaces Ethernet são configuradas para se comportar como uma só, reduzindo o risco de falha em ponto único na conexão típica Ethernet no switch entre nós do cluster e outros hardware de cluster.

Desde o Red Hat Enterprise Linux 6.4, os modos de vinculação 0, 1, e 2 são suportados.

IPv4 e IPv6

O Complemento de Alta Disponibilidade suporta ambos protocolos de internet IPv4 e IPv6. O suporte ao IPv6 no Complemento de Alta Disponibilidade é nova para o Red Hat Enterprise Linux 6.

2.2. HARDWARE COMPATÍVEIS

Antes de configurar o software do Complemento de Alta Disponibilidade da Red Hat, tenha certeza que seu cluster usa um hardware apropriado (por exemplo, dispositivo fence suportados e switches de Canal Fibre). Consulte o guia de configuração de hardware em http://www.redhat.com/cluster_suite/hardware/ para as informações mais atuais sobre a compatibilidade de hardware.

2.3. HABILITANDO PORTAS IP

Antes de implementar o Complemento de Alta Disponibilidade Red Hat, você deve habilitar certas portas IP nos nós do cluster e em computadores que rodam **luci** (o servidor de interface de usuário **Conga**). As seguintes seções identificam as portas IPs a serem habilitadas:

- Seção 2.3.1, “Habilitando Portas IP em nós de Cluster”
- Seção 2.3.2, “Habilitando portas IP para **luci**”

A seção a seguir fornece as regras do `iptables` para habilitar as necessidades das portas IP pelo Red Hat High Availability Add-On:

- Seção 2.3.3, “Configurando o Firewall iptables para Permitir Componentes do Cluster.”

2.3.1. Habilitando Portas IP em nós de Cluster

Para permitir que os nós em um cluster se comuniquem entre si, você precisa habilitar as portas IP atribuídas a certos componentes do Red Hat High Availability Add-On. O Tabela 2.1, “Portas IP habilitadas em nós com o Complemento de Alta Disponibilidade” lista os números de porta IP, seus protocolos e os componentes para o qual os números de porta são atribuídos. Em cada nó de cluster, habilite as portas IP de acordo com o Tabela 2.1, “Portas IP habilitadas em nós com o Complemento de Alta Disponibilidade”. Você pode utilizar o `system-config-firewall` para habilitar as portas IP.

Tabela 2.1. Portas IP habilitadas em nós com o Complemento de Alta Disponibilidade

Número de Porta IP	Protocolo	Componente
5404, 5405	UDP	`corosync/cman` (Gerenciador de Cluster)
11111	TCP	`ricci` (propaga as informações atualizadas do cluster)
21064	TCP	`dlm` (Gerenciador de Bloqueio Distribuído)
16851	TCP	`modclusterd`

2.3.2. Habilitando portas IP para luci

Para permitir computadores clientes se comunicarem com um computador que roda **luci** (o servidor

de interface de usuário **Conga**), você deve habilitar a porta IP atribuída ao **luci**. Em cada computador que roda o **luci**, habilite a porta IP de acordo com a Tabela 2.2, "Porta IP habilitada em um computador que roda o **luci**".

NOTA

Se um nó de cluster estiver rodando o **luci**, a porta 11111 ja deveria ter sido habilitada.

Tabela 2.2. Porta IP habilitada em um computador que roda o luci

Número de Porta IP	Protocolo	Componente
8084	TCP	**luci** (**Conga** servidor de interface de usuário)

Desde o Red Hat Enterprise Linux release 6.1, o qual habilitou a configuração através do arquivo **/etc/sysconfig/luci**, você pode configurar especificamente o único endereço iP onde **luci** está sendo servido. Você pode utilizar esta capacidade se sua infraestrutura de servidor incorporar mais do que uma rede e você quiser acessar o **luci** de uma rede interna. Para fazer isto, descomente e edite a linha no arquivo que especifia o **host**. Por exemplo, para mudar a configuração do **host** no arquivo 10.10.10.10, edite a linha do **host** como se segue:

```
host = 10.10.10.10
```

Para informações sobre o arquivo **/etc/sysconfig/luci** consulte o Seção 2.4, "Configurando **luci** com **/etc/sysconfig/luci**".

2.3.3. Configurando o Firewall iptables para Permitir Componentes do Cluster.

Abaixo segue uma lista de regras iptable para habilitar as portas IP necessárias pelo Red Hat Enterprise Linux 6 (com High Availability Add-on). Por favor, note que estes exemplos utilizam o 192.168.1.0/24 como subnet, mas você precisará substituir o 192.168.1.0/24 pela rede apropriada se você utilizar estas regras.

Para **cman** (Cluster Manager), utilize o seguinte filtro.

```
$ iptables -I INPUT -m state --state NEW -m multiport -p udp -s
192.168.1.0/24 -d 192.168.1.0/24 --dports 5404,5405 -j ACCEPT
$ iptables -I INPUT -m addrtype --dst-type MULTICAST -m state --state NEW
-m multiport -p udp -s 192.168.1.0/24 --dports 5404,5405 -j ACCEPT
```

Para **dlm** (Distributed Lock Manager):

```
$ iptables -I INPUT -m state --state NEW -p tcp -s 192.168.1.0/24 -d
192.168.1.0/24 --dport 21064 -j ACCEPT
```

Para **ricci** (parte do agente remoto do Conga):

```
$ iptables -I INPUT -m state --state NEW -p tcp -s 192.168.1.0/24 -d
192.168.1.0/24 --dport 11111 -j ACCEPT
```

Para **modclusterd** (parte do agente remoto do Conga):

```
$ iptables -I INPUT -m state --state NEW -p tcp -s 192.168.1.0/24 -d
192.168.1.0/24 --dport 16851 -j ACCEPT
```

Para **luci** (O servidor de Interface do Usuário Conga):

```
$ iptables -I INPUT -m state --state NEW -p tcp -s 192.168.1.0/24 -d
192.168.1.0/24 --dport 16851 -j ACCEPT
```

Para **igmp** (Internet Group Management Protocol):

```
$ iptables -I INPUT -p igmp -j ACCEPT
```

Depois de executar estes comandos, execute o seguinte comando para salvar a configuração atual para que as mudanças sejam consistentes durante a reinicialização.

```
$ service iptables save ; service iptables restart
```

2.4. CONFIGURANDO LUCI COM /ETC/SYSCONFIG/LUCI

Desde o Red Hat Enterprise Linux release 6.1, você pode configurar alguns aspectos do comportamento do **luci** através do arquivo **/etc/sysconfig/luci**. Os parâmetros que você pode modificar com este arquivo inclui as configurações auxiliares do ambiente em execução usado pelo script init assim como a configuração do servidor. Alem disso, você pode editar este arquivo para modificar alguns parâmetros de configuração de aplicativo. Estas são instruções dentro do arquivo que descrevem quais parâmetros de configuração você pode mudar ao editar este arquivo.

Para proteger o formato pretendido, você não deveria mudar as linhas de não configuração do arquivo **/etc/sysconfig/luci** quando você editar o arquivo. Além disso, você deve tomar cuidado para seguir a sintaxe requerida para este arquivo, especialmente para a seção **INITSCRIPT** que não permite espaços em branco perto do sinal de igual e requer que você utilize as marcações de quotação para colocar as faixas que contém espaços em branco entre parênteses.

Os seguintes exemplos mostram como mudar a porta onde o **luci** está sendo servido, editando o arquivo **/etc/sysconfig/luci**.

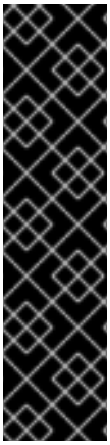
1. Descomente a seguinte linha no arquivo **/etc/sysconfig/luci**:

   ```
   #port = 4443
   ```

2. Substitua o 4443 pelo número de porta desejado, que deve ser maior do que ou igual a 1024 (não uma porta privilegiada). Por exemplo, você pode editar aquela linha de arquivos como se segue para definir a porta onde o **luci** está sendo servido no 8084.

   ```
   port = 8084
   ```

3. Reiniciar o serviço do **luci** para as mudanças tomarem efeito.



IMPORTANTE

Quando você modificar um parâmetro de configuração no arquivo `/etc/sysconfig/luci` para redefinir um valor padrão, você deve tomar cuidado para usar um novo valor no lugar do valor padrão documentado. Por exemplo, quando você modificar a porta onde o **luci** está sendo servido, tenha a certeza de que você especificou o valor modificado ao habilitar uma porta IP para o **luci** como descrito em Seção 2.3.2, "Habilitando portas IP para **luci**".

A porta modificada e parâmetros de host refletirão automaticamente na URL exibida quando o serviço **luci** iniciar, como descrito em Seção 3.2, "Iniciando o **luci**". Você precisa utilizar esta URL para acessar o **luci**.

Para informações completas sobre parâmetros que você pode configurar com o arquivo `/etc/sysconfig/luci` consulte a própria documentação dentro do próprio arquivo.

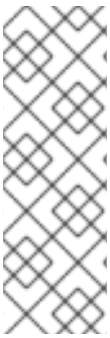
2.5. CONFIGURANDO O ACPI PARA USO COM DISPOSITIVOS FENCE INTEGRADOS

Se seu cluster usa dispositivos fence integrados, você deve configurar o ACPI (Advanced Configuration and Power Interface) para garantir um fencing imediato e completo.

NOTA

Para as informações mais atuais sobre dispositivos fence integrados suportados pelo Complemento de Alta Disponibilidade da Red Hat, consulte http://www.redhat.com/cluster_suite/hardware/.

Se um nó do cluster estiver configurado para ter um fence por um dispositivo fence integrado, desabilite o ACPI Soft-Off para este nó. Desabilitando o ACPI Soft-Off permite que um dispositivo de fence integrado desligar um nó imediatamente e completamente em vez de tentar um desligamento normal (por exemplo, **`shutdown -h now`**). Senão, se o ACPI Soft-Off estiver habilitado, um dispositivo fence integrado pode levar quatro ou mais segundos para desligar um nó. Além disso, se o ACPI Soft-Off estiver habilitado e um nó entra em pânico ou congela durante o desligamento, um dispositivo fence integrado pode não ser capaz de desligar o nó. Dentro destas circunstâncias, a ação do fence é atrasada ou realizada sem sucesso. Consequentemete, quando um nó estiver em fence com um dispositivo fence integrado e o ACPI Soft-Off estiver habilitado, um cluster se recupera lentamente ou requer intervenção administrativa para se recuperar.



NOTA

A quantidade de tempo requerida para fazer um fence em um nó depende do dispositivo fence integrado usado. Alguns dispositivos realizam o equivalente a pressionar e segurar o botão Liga/Desliga; portanto, o dispositivo fence desliga em quatro ou cinco segundos. Outros dispositivos fence integrados realizam o equivalente a pressionar o botão liga/desliga uma vez, confiando no sistema operacional para desligar o nó; portanto, o dispositivo fence desliga o nó num período de tempo muito maior do que quatro ou cinco segundos.

Para desabilitar o ACPI Soft-Off, use o gerenciador **`chkconfig`** e verifique que o nó se desligue imediatamente quando em fence. A maneira preferencial para desabilitar o ACPI Soft-Off é com o gerenciador **`chkconfig`**: entretanto, se este método não é satisfatório para seu cluster, você pode

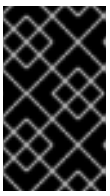
desabilitar o ACPI Soft-Off com um dos seguintes comandos alternativos:

- Alterar a configuração da BIOS para "instant-off" ou uma configuração equivalente que desliga um nó sem atraso.

NOTA

Desabilitar o ACPI Soft-Off com a BIOS pode não ser possível em alguns computadores.

- Anexando **`acpi=off`** na linha de comando de inicialização do kernel no arquivo **`/boot/grub/grub.conf`**



IMPORTANTE

Este método desabilita completamente o ACPI; alguns computadores não inicializam corretamente se o ACPI estiver completamente desabilitado. Use este método *somente* se os outros métodos não forem efetivos para seu cluster.

As seções seguintes fornecem procedimentos para o método preferido e alternativos para desabilitar o ACPI Soft-Off:

- Seção 2.5.1, “Desabilitando o ACPI Soft-Off com o gerenciador **`chkconfig`**” — Método preferido
- Seção 2.5.2, “Desabilitando o ACPI Soft-Off com a BIOS” — Primeiro método alternativo
- Seção 2.5.3, “Desabilitar completamente o ACPI no arquivo **`grub.conf`**.” — Segundo método alternativo

2.5.1. Desabilitando o ACPI Soft-Off com o gerenciador `chkconfig`

Você pode usar o gerenciador **`chkconfig`** para desabilitar o ACPI Soft-Off tanto removendo o ACPI daemon (**`acpid`**) do gerenciador **`chkconfig`** ou desligando o **`acpid`**.

NOTA

Este é o método preferido para desabilitar o ACPI Soft-Off.

Desabilite o ACPI Soft-Off com o gerenciador **`chkconfig`** em cada nó do cluster como a seguir:

1. Rode qualquer dos seguintes comandos:
   - **`chkconfig --del acpid`** — Este comando remove o **`acpid`** do gerenciador **`chkconfig`**.

     — OU —

   - **`chkconfig --level 2345 acpid off`** — Este comando desliga o **`acpid`**.
2. Reinicialize o nó.
3. Quando o cluster estiver configurado e rodando, verifique de que o nó se desligue imediatamente quando estiver em fence.

NOTA

Você pode fazer um fence em um nó com o comando **fence_node** ou **Conga**.

2.5.2. Desabilitando o ACPI Soft-Off com a BIOS

O método preferido para desabilitar o ACPI Soft Off com o gerenciador **chkconfig** (Seção 2.5.1, "Desabilitando o ACPI Soft-Off com o gerenciador **chkconfig**"). Entretanto, se o método preferido não for efetivo para seu cluster, siga o procedimento desta seção.

NOTA

Desabilitar o ACPI Soft-Off com a BIOS pode não ser possível em alguns computadores.

Você pode desabilitar o ACPI Soft-Off configurando a BIOS de cada nó no cluster como a seguir:

1. Reinicialize o nó e inicie o programa **Utilitário de Configuração da BIOS CMOS**
2. Vá até o menu **Power** (ou menu equivalente de gerenciamento de energia).
3. No menu **Power**, configure a função **Soft-Off by PWR-BTTN** (ou equivalente) para **Instant-Off** (ou configuração equivalente que desligue o nó pelo botão liga/desliga sem atraso). O Exemplo 2.1, "**Utilitário de Configuração da BIOS CMOS**: Soft-Off by PWR-BTTN definido para **Instant-Off**" exibe o menu **Power** com **a função ACPI** definida para **Habilitada** e **Soft-Off pelo PWR-BTTN** definido para **Instant Off**.

   NOTA

   O equivalente a **Função ACPI**, **Soft-Off pelo PWR-BTTN**, e **Instant-Off** pode variar entre computadores. Entretanto, o objetivo deste procedimento é configurar a BIOS para que o computador desligue pelo botão de energia sem atraso.

4. Saia do **Utilitário de configuração da BIOS CMOS** e salve as configurações.
5. Quando o cluster estiver configurado e rodando, verifique de que o nó se desligue imediatamente quando estiver em fence.

   NOTA

   Você pode fazer um fence em um nó com o comando **fence_node** ou **Conga**.

Exemplo 2.1. Utilitário de Configuração da BIOS CMOS: Soft-Off by PWR-BTTN definido para Instant-Off

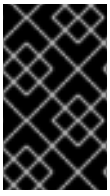
```
+---------------------------------------------|-------------------+
|    ACPI Function             [Enabled]      |    Item Help      |
|    ACPI Suspend Type         [S1(POS)]      |-------------------|
|  x Run VGABIOS if S3 Resume   Auto          |   Menu Level   *  |
|    Suspend Mode              [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 exemplo mostra a **função ACPI** configurada para **Habilitada**, e o **Soft-Off pelo PWR-BTTN** configurada para **Instant-Off**.

2.5.3. Desabilitar completamente o ACPI no arquivo `grub.conf`.

O método preferido para desabilitar o ACPI Soft-Off é com o gerenciador **`chkconfig`** (Seção 2.5.1, "Desabilitando o ACPI Soft-Off com o gerenciador **`chkconfig`**"). Caso o método preferido não seja efetivo em seu cluster, você pode desabilitar o ACPI Soft-Off com o gerenciamento de energia do BIOS (Seção 2.5.2, "Desabilitando o ACPI Soft-Off com a BIOS"). Se nenhum destes métodos for efetivo em seu cluster, você pode desabilitar o ACPI totalmente, adicionando o **`acpi=off`** na linha de comando de inicialização do kernel no arquivo **`grub.conf`**.



IMPORTANTE

Este método desabilita completamente o ACPI; alguns computadores não inicializam corretamente se o ACPI estiver completamente desabilitado. Use este método *somente* se os outros métodos não forem efetivos para seu cluster.

Você pode desabilitar o ACPI completamente editando o arquivo **`grub.conf`** em cada nó do cluster como a seguir:

1. Abra **`/boot/grub/grub.conf`** com um editor de textos.
2. Adicione **`acpi=off`** à linha de comando de inicialização do kernel em **`/boot/grub/grub.conf`** (consulte Exemplo 2.2, "A linha de comando de inicialização do kernel com **`acpi=off`**").
3. Reinicialize o nó.
4. Quando o cluster estiver configurado e rodando, verifique de que o nó se desligue imediatamente quando estiver em fence.

NOTA

Você pode fazer um fence em um nó com o comando **`fence_node`** ou **Conga**.

Exemplo 2.2. A linha de comando de inicialização do kernel com `acpi=off`

```
# grub.conf generated by anaconda
#
# Note that you do not have to rerun grub after making changes to this
file
# NOTICE:  You have a /boot partition.  This means that
#          all kernel and initrd paths are relative to /boot/, eg.
#          root (hd0,0)
#          kernel /vmlinuz-version ro root=/dev/mapper/vg_doc01-lv_root
#          initrd /initrd-[generic-]version.img
#boot=/dev/hda
default=0
timeout=5
serial --unit=0 --speed=115200
terminal --timeout=5 serial console
title Red Hat Enterprise Linux Server (2.6.32-193.el6.x86_64)
        root (hd0,0)
        kernel /vmlinuz-2.6.32-193.el6.x86_64 ro
root=/dev/mapper/vg_doc01-lv_root console=ttyS0,115200n8 acpi=off
        initrd /initramrs-2.6.32-131.0.15.el6.x86_64.img
```

Neste exemplo, o **acpi=off** foi inserido à linha de comando de inicialização do kernel — a linha iniciando com "kernel /vmlinuz-2.6.32-193.el6.x86_64.img".

2.6. CONSIDERAÇÕES PARA CONFIGURAR SERVIÇOS HA

Você pode criar um cluster para suprir suas necessidades de alta disponibilidade configurando os serviços de alta disponibilidade (HA: high-availability). O componente chave para o gerenciamento de serviço HA dentro do Complemento de Alta Disponibilidade Red Hat, **rgmanager**, implementa failover para aplicações comuns. No Complemento de Alta Disponibilidade, uma aplicação é configurada com outros recursos de cluster para formar um serviço HA que pode ocorrer failover de um nó no cluster para outro sem interrupção aparente para clientes do cluster. O failover em serviços HA podem ocorrer se um nó no cluster falhar ou se um administrador de sistemas do cluster move o serviço de um nó do cluster para outro (por exemplo, para uma manutenção planejada de um nó no cluster).

Para criar um serviço HA, você deve configura-lo no arquivo de configuração do cluster. Um serviço HA abrange *recursos* de cluster. Recursos de cluster são blocos de construção que você cria e gerencia no arquivo de configuração do cluster — por exemplo, um endereço de IP, um script de inicialização de aplicativo ou uma partição Red Hat GFS2.

Um serviço HA pode rodar somente em um nó no cluster por vez para manter a integridade dos dados. Você pode especificar a prioridade de failover em um domínio de failover. Especificar prioridades de failover consiste em atribuir níveis de prioridade para cada nó em um domínio de failiover. O nível de prioridade determina a ordem de failover — determinando para qual nó que um serviço HA deveria fazer o failover. Se você não especificar uma prioridade da failover, um serviço HA pode fazer o failover para qualquer nó em seu domínio. Também, você pode especificar se um serviço HA é restringido para rodar somente em nós deste domínio de failover associado. (Quando associado a um irrestrito domínio de failover, um serviço HA pode iniciar em qualquer nó no cluster em um evento de que nenhum membro do domínio de failover esteja disponível).

A Figura 2.1, “Exemplo de Serviço de Cluster Servidor Web” mostra um exemplo de um serviço HA que é um servidor web chamado "content-webserver". Ele está rodando no nó do cluster B e está dentro de um domínio de failover que consiste dos nós A, B e D. Além disso, o domínio de failover está

configurado com uma prioridade para fazer um failover para o nó D antes do nó A e para restringir o failover somente em nós dentro daquele domínio de failover. O serviço HA abrange estes recursos de cluster:

- recurso de endereço de IP — endereço de IP 10.10.10.201.
- Um recurso de aplicação chamado "http-content" — um script de inicialiação de aplicação de servidor web **`/etc/init.d/httpd`** (especificando o **`httpd`**).
- Um recurso de sistema de arquivos — Red Hat GFS 2 chamado "gfs2-content-webserver".

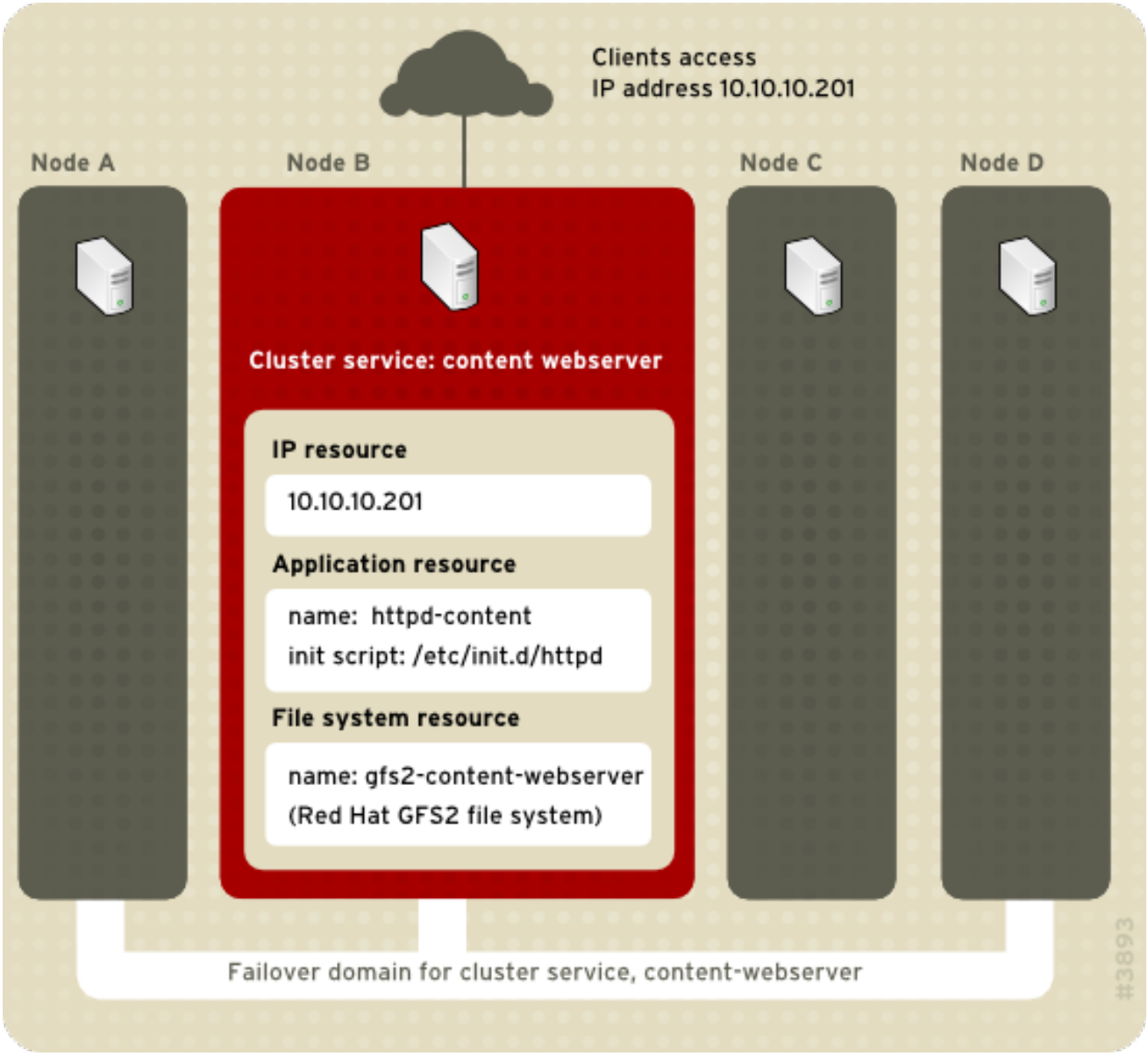
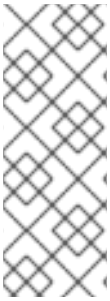


Figura 2.1. Exemplo de Serviço de Cluster Servidor Web

Os clientes acessam o serviço HA através do endereço de IP 10.10.10.201, habilitando interação com a aplicação de servidor web, httpd-content. O aplicativo httpd-content usa o sistema de arquivos gfs2-content-webserver. Se o nodo B falhar, o serviço HA content-webserver faria o failover para o nó D. Se o nó D não estivesse disponível ou também falhasse, o serviço faria o failover para o nó A. O failover ocorreria com mínima interrupção de serviços aos clientes do cluster. Por exemplo, em um serviço HTTP, certas informações sobre o estado podem ser perdidas (como dados da seção). O serviço HA estaria disponível a partir de um outro nó no cluster pelo o mesmo endereço de IP como se fosse antes do failover.



NOTA

Para mais informações sobre serviços HA e domínios failover, consulte a *Visão Geral do Complemento de Alta Disponibilidade*. Para informações sobre configurar domínios failover, consulte o Capítulo 3, *Configurando o Complemento de Alta Disponibilidade da Red Hat com o **Conga*** (usando o **Conga**) ou o Capítulo 7, *Configurando o Complemento de Alta Disponibilidade da Red Hat com as Ferramentas da Linha de Comando* (usando utilitários da linha de comando).

Um serviço HA é um grupo de recursos de cluster configurados em uma entidade coerente que fornece serviços especializados à clientes. Um serviço HA é representado como uma árvore de recursos em um arquivo de configuração de cluster, **/etc/cluster/cluster.conf** (em cada nó no cluster). No arquivo de configuração do cluster, cada árvore de recursos é uma representação XML que especifica cada recurso, seus atributos e seu relacionamento entre outros recursos na árvore de recursos (relacionamentos pai, filhos e irmãos).

NOTA

Por causa que um serviço HA consiste de recursos organizados em uma árvore hierárquica, um serviço é as vezes referido como um *recurso de árvore* ou *grupo de recursos*. Ambos são sinônimos em um *serviço HA*.

Na raíz de cada árvore de recurso há um especial tipo de recurso – um *recurso de serviço*. Outros tipos de recursos compreendem do resto de um serviço, determinando suas características. Configurando em serviço HA consiste em criar um recurso de serviço, criando recursos de cluster subordinados e organiza-los em uma entidade coerente que tem conformidade com às restrições hierárquicas do serviço.

Existem duas maiores considerações a serem levadas em conta quando configurar um serviço HA:

- Os tipos de recursos necessários para criar um serviço
- Relacionamentos de Pai, filhos e irmãos entre recursos

Os tipos de recursos e a hierarquia de recursos dependem do tipo de serviço que você está configurando.

Os tipos de recursos de cluster estão listados no Apêndice B, *Parâmetros dos Recursos de Alta Disponibilidade*. Informações sobre relacionamentos de categorias pai, filhos e irmãos entre recursos estão descritas no Apêndice C, *Comportamento do Recurso de Alta Disponibilidade*.

2.7. VALIDAÇÃO DE CONFIGURAÇÃO

A configuração do cluster é automaticamente validada de acordo com o esquema de cluster no **/usr/share/cluster/cluster.rng** durante o tempo de inicialização e quando uma configuração é recarregada. Também, você pode validar uma configuração de cluster em qualquer momento usando o comando **ccs_config_validate**. Para mais informações sobre validação de configuração quando usar o comando **ccs** veja Seção 5.1.6, “Validação de Configuração”.

Um esquema anotado é disponível para vizualização em **/usr/share/doc/cman-X.Y.ZZ/cluster_conf.html** (por exemplo **/usr/share/doc/cman-3.0.12/cluster_conf.html**).

A validação de configuração checa pelos seguintes erros básicos:

- Validade XML — Checa que o arquivo de configuração é um arquivo XML válido.
- Opções de configuração — Verifica para ter certeza que opções (elementos e atributos XML) são valídos.
- Valores de Opção — Verifica que as opções contém dados válidos (limitados).

Os seguintes exemplos mostram uma configuração válida e configurações inválidas que ilustram as verificações de validação:

- Configurações válidas — Exemplo 2.3, "`cluster.conf` Exemplo de Configuração: Arquivo Válido"
- XML Inválidos — Exemplo 2.4, "`cluster.conf` Exemplo de Configuração: XML Inválido"
- Opções inválidas — Exemplo 2.5, "`cluster.conf` Exemplo de configuração: Opção Inválida"
- Valor de opção inválida — Exemplo 2.6, "`cluster.conf` Exemplo de configuração: Valor de opção inválida"

Exemplo 2.3. `cluster.conf` Exemplo de Configuração: Arquivo Válido

```
<cluster name="mycluster" config_version="1">
  <logging debug="off"/>
   <clusternodes>
     <clusternode name="node-01.example.com" nodeid="1">
         <fence>
         </fence>
     </clusternode>
     <clusternode name="node-02.example.com" nodeid="2">
         <fence>
         </fence>
     </clusternode>
     <clusternode name="node-03.example.com" nodeid="3">
         <fence>
         </fence>
     </clusternode>
   </clusternodes>
   <fencedevices>
   </fencedevices>
   <rm>
   </rm>
</cluster>
```

Exemplo 2.4. `cluster.conf` Exemplo de Configuração: XML Inválido

```
<cluster name="mycluster" config_version="1">
  <logging debug="off"/>
   <clusternodes>
     <clusternode name="node-01.example.com" nodeid="1">
```

```
          <fence>
          </fence>
      </clusternode>
      <clusternode name="node-02.example.com" nodeid="2">
          <fence>
          </fence>
      </clusternode>
      <clusternode name="node-03.example.com" nodeid="3">
          <fence>
          </fence>
      </clusternode>
    </clusternodes>
    <fencedevices>
    </fencedevices>
    <rm>
    </rm>
<cluster>         <----------------INVALID
```

Neste exemplo, a última linha de configuração (anotada como "INVALID" aqui) está faltando uma barra — mostrando **<cluster>** ao invés do correto **</cluster>**.

Exemplo 2.5. `cluster.conf` Exemplo de configuração: Opção Inválida

```
<cluster name="mycluster" config_version="1">
  <loging debug="off"/>         <----------------INVALID
    <clusternodes>
      <clusternode name="node-01.example.com" nodeid="1">
          <fence>
          </fence>
      </clusternode>
      <clusternode name="node-02.example.com" nodeid="2">
          <fence>
          </fence>
      </clusternode>
      <clusternode name="node-03.example.com" nodeid="3">
          <fence>
          </fence>
      </clusternode>
    </clusternodes>
    <fencedevices>
    </fencedevices>
    <rm>
    </rm>
<cluster>
```

Neste exemplo, a segunda linha da configuração (anotada como "INVALID") contém um elemento XML inválido — mostrando **loging** ao invés do correto **logging**.

Exemplo 2.6. `cluster.conf` Exemplo de configuração: Valor de opção inválida

```
<cluster name="mycluster" config_version="1">
  <loging debug="off"/>
   <clusternodes>
     <clusternode name="node-01.example.com" nodeid="-1">  <--------
INVALID
         <fence>
         </fence>
     </clusternode>
     <clusternode name="node-02.example.com" nodeid="2">
         <fence>
         </fence>
     </clusternode>
     <clusternode name="node-03.example.com" nodeid="3">
         <fence>
         </fence>
     </clusternode>
   </clusternodes>
   <fencedevices>
   </fencedevices>
   <rm>
   </rm>
<cluster>
```

Neste exemplo, a quarta linha da configuração (anotada como "INVALID") contém um valor inválido para o atributo XML, **nodeid** na linha **clusternode** para **node-01.example.com**. O valor é negativo ("-1") em vez de positivo ("1"). Para o atributo **nodeid** o valor deve ser um positivo.

2.8. CONSIDERAÇÕES PARA O NETWORKMANAGER

O uso do **NetworkManager** não é suportado em nós do cluster. Se você tiver instalado o **NetworkManager** em seus nós do cluster, você deveria tanto remove-lo ou desabilita-lo.

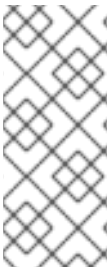
NOTA

O serviço **cman** não iniciará se o **NetworkManager** estiver tanto rodando ou foi configurado para rodar com o comando **chkconfig**.

2.9. CONSIDERAÇÕES PARA USAR O DISCO DE QUORUM

O Disco de Quorum é um daemon quorum baseado em disco, **qdiskd**, que fornece heurísticas complementares para determinar adequação do nó. Com heurísticas você pode determinar fatores que são importantes para a operação do nós em um evento de partição da rede. Por exemplo, em um cluster de quatro nós com uma divisão 3:1, ordinariamente, os três nós automaticamente "ganham" por causa da maioria três pra um. Sob essas circunstâncias, o nó possui um fence. Com o **qdiskd** entretanto, você pode configurar as heurísticas que permitem um nó a vencer baseado no acesso a um

recurso crítico (por exemplo, um caminho de rede crítico). Se seu cluster requer métodos adicionais para determinar a saúde do nó, então você deveria configurar o **qdiskd** para atender essas necessidades.



NOTA

Configurando o **qdiskd** não é requerido ao menos que você tenha requerimentos especiais para a saúde do nó. Um exemplo de um requerimento especial é uma configuração "todos-menos-um" Em uma configuração "todos-menos-um", o **qdiskd** está configurado para fornecer suficientes votos de quorum para manter quorum mesmo que somente um nó esteja funcionando.

IMPORTANTE

Em geral, heurísticas e outros parâmetros **qdiskd** para sua implementação dependem do ambiente local e requerimentos especiais necessários. Para entender o uso de heurísticas e outros parâmetros **qdiskd**, consulte a página man qdisk(5). Se você requer assistencia para entender sobre o uso do **qdiskd** para seu local, contate um representante de suporte Red Hat autorizado.

Se você precisar usar o comando **qdiskd**, você deveria levar em conta as seguintes considerações:

Votos de nós do Cluster

Quando usar o Disco de Quorum, cada nó do cluster deve ter um voto.

O valor de expiração de sessão da afiliação CMAN

O valor de expiração CMAN (o tempo que um nó precisa de estar sem resposta antes que o CMAN considere que o nó precisa ser eliminado e não mais um membro) deve ser ao menos duas vezes o valor de expiração de afiliação do **qdiskd**. A razão é por causa que o daemon quorum deve detectar nós com falha sozinho, e pode levar muito mais tempo para fazer isso do que o CMAN. O valor padrão para a expiração de afiliação do CMAN é 10 segundos. Outras condições específicas do lugar podem afetar o relacionamento entre a valores de expiração de afiliação do CMAN e o **qdiskd**. Para assistência em ajustar o valor de expiração de afiliação do CMAN, contate um representante de suporte autorizado Red Hat.

Fencing

Para garantir um fencing confiável quando usar o **qdiskd**, use o power fencing. Enquanto outros tipos de fencing podem ser confiáveis para cluster não configurados com o **qdiskd**, eles não são confiáveis para um cluster configurado com o **qdiskd**.

Nós máximos

Um cluster configurado com o **qdiskd** suporta um máximo de 16 nós. A razão para o limite é por causa da escalabilidade; aumentando a contagem de nós, aumenta o número de contenções de E/S síncronas para o dispositivo de disco de quorum compartilhado.

Dispositivo de Disco de Quorum

Um dispositivo de disco de quorum deve ser um dispositivo de bloco compartilhado com acesso de leitura e escritas simultâneos por todos os nós no cluster. O tamanho mínimo do dispositivo de bloco é 10 Megabytes. Exemplos de dispositivos de bloco compartilhados que podem ser usados pelo **qdiskd** são uma multi-port SCSI RAID array, Fibre Channel RAID SAN, ou um RAID-configured

iSCSI target. Você pode criar um dispositivo de disco de quorum com o **mkqdiskd**, o Utilitário de Disco de Quorum de Cluster. Para informações sobre o uso deste utilitário, consulte a página man mkqdiskd(8).

NOTA

Usando o JBOD como disco de quorum não é recomendado. Um JBOD não pode fornecer desempenho dependente e então não pode permitir que um nó escreva nele rápido o suficiente. Se um nó é incapaz de escrever em um dispositivo de disco de quorum rapido o bastante, o nó é falsamente expulso de um cluster.

2.10. COMPLEMENTO DE ALTA DISPONIBILIDADE RED HAT E O SELINUX

A Complemento de Alta Disponibilidade para o Red Hat Enterprise Linux 6 suporta SELinux no estado **enforcing** com o tipo de política SELinux definida para **targeted**.

Para mais informações sobre SELinux, consulte o *Guia de Implementação* para o Red Hat Enterprise Linux 6.

2.11. ENDEREÇOS MULTICAST

Os nós em um cluster se comunicam entre si utilizando os endereços multicast. Portanto, cada interruptor de rede e equipamentos de rede associados no Red Hat High Availability Add-On deve ser configurado para habilitar os endereços do multicast e suportar o IGMP (Internet Group Management Protocol). Assegure-se de que cada interruptor de rede e equipamentos associados no Red Hat High Availability Add-On é capaz de suportar os endereços de multicast e IGMP; caso sejam, certifique de que o endereçamento do multicast e o IGMP estão habilitados. Sem o multicast e IGMP, nem todos os nós podem participar em um cluster, causando falha; use o unicast UDP nestes ambientes, como descrito em Seção 2.12, “Tráfego do Unicast UDP” .

NOTA

Procedimentos para configurar switches de rede e equipamentos de rede associados variam de acordo com cada produto. Consulte a documentação apropriada ou outras informações sobre como configurar switches de rede e equipamentos de rede associados para habilitar endereços multicast e IGMP.

2.12. TRÁFEGO DO UNICAST UDP

Desde o lançamento do Red Hat Enterprise Linux 6.2, os nós em um cluster podem se comunicar entre si utilizando o mecanismo de transporte do Unicast UDP. Recomenda-se no entanto, que você utilize o multicasting do IP para a rede de cluster. O Unicast UDP é uma alternativa que pode ser utilizada quando o musticast IP não estiver disponível.

Você pode configurar o Complemento de Alta Disponibilidade da Red Hat para utilizar o Unicast UDP para configurar o parâmetro **cman transport="udpu"** no arquivo de configuração **cluster.conf**. Você pode especificar Unicast a partir da página **Network Configuration** da interface do usuário, **Conga** como descrito em Seção 3.5.3, “Configuração de Rede”.

2.13. CONSIDERAÇÕES PARA O RICCI

Para o Red Hat Enterprise Linux 6, o **ricci** substitui o **ccsd**. Portanto, é necessário que o **ricci** esteja rodando em cada nó do cluster para ser capaz de propagar as configurações do cluster atualizadas seja pelo comando **cman_tool -r**, o comando **ccs** ou pelo servidor de interface do usuário **luci**. Você pode iniciar o **luci** usando **service ricci start** ou habilitando-o para iniciar durante o boot pelo **chkconfig**. Para informações sobre habilitar portas IP através do **ricci**, consulte a Seção 2.3.1, “Habilitando Portas IP em nós de Cluster” .

Para o lançamento do Red Hat Enterprise Linux 6.1 e posteriores, usar o **ricci** requer uma senha na primeira vez que você propaga a configuração de cluster atualizada a partir de algum nó específico. Você define a senha **ricci** como root depois de instalar o **ricci** em seu sistema com o comando **passwd ricci**, para o usuário **ricci**.

2.14. CONFIGURANDO AS MÁQUINAS VIRTUAIS EM UM AMBIENTE CLUSTER

Quando você configurar seu cluster com os recursos de máquina virtual você precisa utilizar as ferramentas do **rgmanager** para iniciar e interromper as máquinas virtuais. O uso do **virsh** para iniciar a máquina pode fazer com que a máquina seja executada em mais do que um local, que pode causar danos de dados na máquina virtual.

Para reduzir as chances de o administrador acidentalmente "iniciar duplamente" as máquinas virtuais utilizando ferramentas cluster e não cluster em um ambiente em cluster, você pode configurar seu sistema armazenando os arquivos de configuração da máquina virtual em um local não padrão. Armazenar os arquivos de configuração da máquina virtual em algum local a não ser seu local padrão, torna mais difícil a tarefa de iniciar uma máquina virtual acidentalmente utilizando o **virsh**, como o arquivo de configuração ficará desconhecido fora da caixa para o **virsh**.

O local não padrão para arquivos de máquina virtual pode ser qualquer um. A vantagem de utilizar um compartilhamento NFS ou um sistema de arquivo compartilhado GFS2, é que o administrador não precisa manter os arquivos de configuração em sincronização entre os membros do cluster. No entanto, é também permissível utilizar um diretório local pelo tempo que o administrador mantiver o conteúdo sincronizado em cluster de alguma forma.

Na configuração do cluster, as máquinas virtuais podem referenciar este local não padrão utilizando o atributo **path** de um recurso da máquina virtual. Observe que o atributo **path** é um diretório ou conjunto de diretórios separados pelo caractere dois pontos ':', não um caminho para o arquivo específico.

ATENÇÃO

O serviço **libvirt-guests** deve estar disabilitado em todos os nós que estiverem executando o **rgmanager**. Se uma máquina virtual iniciar automaticamente ou parar, isto pode resultar na máquina virtual ser executada em mais do que um local, o que pode causar danos de dados na máquina virtual.

Para informações sobre os atributos dos recursos de máquina virtual, consulte Tabela B.24, “Máquina Virtual”.

CAPÍTULO 3. CONFIGURANDO O COMPLEMENTO DE ALTA DISPONIBILIDADE DA RED HAT COM O CONGA

Este capítulo descreve como configurar software do Complemento de Alta Disponibilidade da Red Hat usando o **Conga**. Para Informações sobre o uso do **Conga** para gerenciar um cluster em execução, veja a Capítulo 4, *Gerenciando o Complemento de Alta Disponibilidade Red Hat com o* ***Conga***.

NOTA

O Conga é uma interface de usuário gráfica que você pode usar para administrar o Complemento de Alta Disponibilidade da Red Hat. Note, entretanto, que para usar esta interface efetivamente você precisa ter um bom e claro entendimento dos conceitos fundamentais. Aprender sobre a configuração de cluster, explorando os recursos disponíveis na interface do usuário não é recomendado, já que isso pode resultar em um sistema que não é robusto o bastante para manter todos os serviços rodando quando componentes falham.

Este capítulo consiste das seguintes seções:

- Seção 3.1, “Tarefas de Configuração”
- Seção 3.2, “Iniciando o **luci**”
- Seção 3.3, “Controlando o Acesso ao luci”
- Seção 3.4, “Criando um Cluster”
- Seção 3.5, “Propriedades de Cluster Globais”
- Seção 3.6, “Configurando Dispositivos Fence”
- Seção 3.7, “Configurar Fence para Membros do Cluster”
- Seção 3.8, “Configurando um Domínio de Failover”
- Seção 3.9, “Configurar Recursos de Cluster Globais”
- Seção 3.10, “Adicionar um Serviço de Cluster ao Cluster”

3.1. TAREFAS DE CONFIGURAÇÃO

Configurar o software do Complemento de Alta Disponibilidade da Red Hat com o **Conga** consiste dos seguintes passos:

1. Configurar e rodar a interface de usuário de configuração do **Conga** – o servidor **luci**. Consulte a Seção 3.2, “Iniciando o **luci**”.
2. Criando um cluster. Consulte a Seção 3.4, “Criando um Cluster” .
3. Configurando propriedades de cluster globais. Consulte a Seção 3.5, “Propriedades de Cluster Globais”.
4. Configurando dispositivos fence. Consulte a Seção 3.6, “Configurando Dispositivos Fence”.

5. Configurando o fencing para membros do cluster. Consulte a Seção 3.7, “Configurar Fence para Membros do Cluster”.
6. Criando domínios failover. Consulte a Seção 3.8, “Configurando um Domínio de Failover” .
7. Criando recursos. Consulte a Seção 3.9, “Configurar Recursos de Cluster Globais” .
8. Criando serviços de cluster. Consulte a Seção 3.10, “Adicionar um Serviço de Cluster ao Cluster”.

3.2. INICIANDO O LUCI

NOTA

Usando o **ricci** para configurar um cluster requer que o **ricci** seja instalado e rodado nos nós do cluster, conforme descrito na Seção 2.13, “Considerações para o **ricci**”. Conforme observado nessa seção, usando o **ricci** requer uma senha que o **luci** pede para você digitar para cada nó no cluster quando você criar um cluster, conforme descrito em Seção 3.4, “Criando um Cluster” .

Antes de iniciar o **luci**, assegure-se que as portas IP em seus nós do cluster permitem conexões à porta 11111 do servidor **luci** em quaisquer nós com que o **luci** estará se comunicando. Para informações sobre habilitar portas IP em nós no cluster, veja a Seção 2.3.1, “Habilitando Portas IP em nós de Cluster” .

Para administrar o Complemento de Alta Disponibilidade da Red Hat com o **Conga**, instale e execute o **luci** conforme a seguir:

1. Selecione um computador para ter o **luci** e instale o software **luci** nesse computador. Por exemplo:

```
# yum install luci
```

NOTA

Tipicamente, um computador no papel de servidor ou um data center hospeda o **luci**; entretanto, um computador em cluster pode hospedar o **luci**.

2. Inicie o **luci** usando **service luci start**. Por exemplo:

```
# service luci start
Starting luci: generating https SSL certificates...  done
                                                           [  OK
]

Please, point your web browser to https://nano-01:8084 to access
luci
```

NOTA

Desde o Red Hat Enterprise Linux release 6.1, você pode configurar alguns aspectos do comportamento do **luci** através do arquivo **`/etc/sysconfig/luci`**, incluindo os parâmetros de porta e máquina, como descrito em Seção 2.4, "Configurando **luci** com **`/etc/sysconfig/luci`**". Os parâmetros de porta e máquina modificados irão refletir automaticamente no URL exibido quando o serviço **luci** iniciar.

3. Em um navegador da Web, coloque a URL do servidor **luci** em sua barra de endereços de URL e clique em **Go** (ou o equivalente). A sintáxe da URL para o servidor **luci** é **`https://`*`luci_server_hostname`*`:`*`luci_server_port`***. O valor padrão de *luci_server_port* é **8084**.

   A primeira vez que você acessar o **luci**, o navegador web exibe uma pergunta sobre o certificado auto assinado de SSL (do servidor **luci**). Após confirmação o navegador exibe a página de login do **luci**.

4. Embora qualquer usuário capaz de autenticar no sistema que esteja hospedando o **luci** possa se autenticar no **luci**, desde o Red Hat Enterprise Linux 6.2, somente o usuário root no sistema que esteja sendo executado o **luci** poderá acessar qualquer um dos componentes do **luci** até que um administrador (o usuário root ou um usuário com permissão de administrador) defina permissões para aquele usuário. Para informações sobre como configurar permissões do **luci** para usuários, consulte o Seção 3.3, "Controlando o Acesso ao luci" .

   Depois de logar, o **luci** exibe a página **Homebase**, como mostrada na Figura 3.1, "Página Homebase do luci".

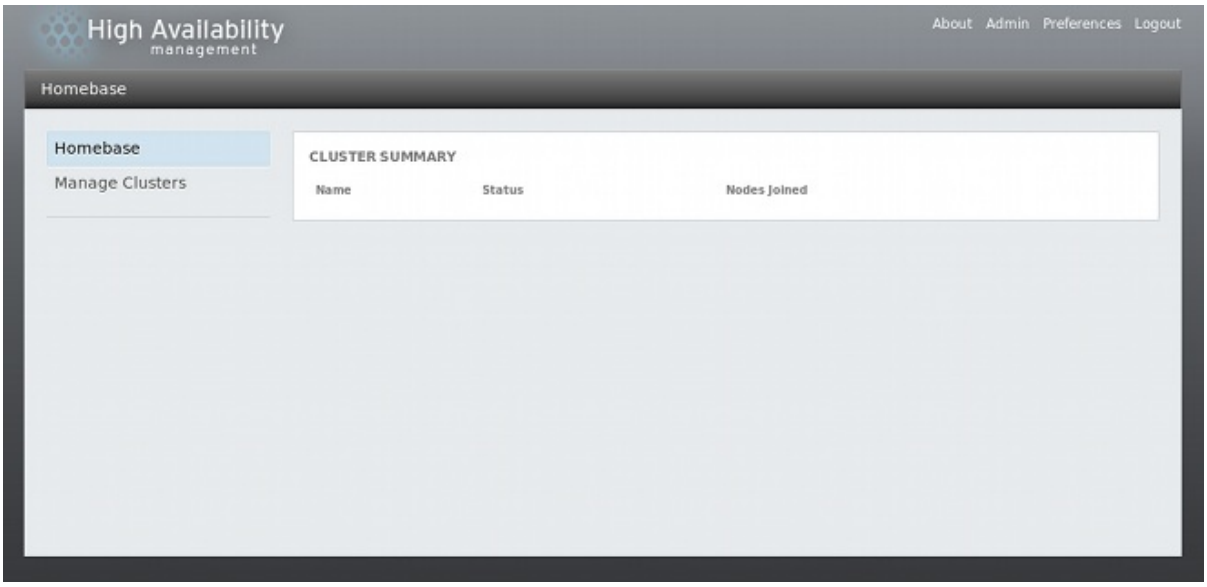


Figura 3.1. Página Homebase do luci

NOTA

Existe um tempo limite em espera para o **luci** que o desautentica após 15 minutos de inatividade.

3.3. CONTROLANDO O ACESSO AO LUCI

Como o lançamento inicial do Red Hat Enterprise Linux 6, os recursos a seguir foram adicionados à página **Usuários e Permissões**

- Desde o Red Hat Enterprise Linux 6.2, o usuário root ou um usuário que recebeu permissões do administrador do **luci** em um sistema executando o **luci** pode controlar o acesso à diversos componentes do **luci** definindo permissões para os usuários individuais em um sistema.
- Desde o Red Hat Enterprise Linux 6.3, o usuário root ou um usuário que recebeu permissões de administrador do **luci** também poderão utilizar a interface do **luci** para adicionar usuários ao sistema.
- Desde o Red Hat Enterprise Linux 6.4, o usuário root ou um usuário que recebeu permissões de administrador do **luci** também poderão utilizar a interface do **luci** para remover usuários do sistema.

Para adicionar usuários, remover usuários ou definir permissões de usuário, autentique-se no **luci** como **root** ou então como um usuário que recebeu permissões de administrador e clique na seleção **Admin** no canto superior direito da tela do **luci**. Isto exibirá a página do **Usuários e Permissões**, que exibe os usuários existentes.

Para remover usuários, selecione o usuário ou usuários e clique em **Remover Selecionado**.

Para adicionar um usuário, clique em **Adicionar um Usuário** e insira o nome do usuário a adicionar.

Para definir ou modificar permissões para um usuário, selecione o usuário a partir do menu suspenso sob **Permissões de Usuário**. Isto permite que você defina as seguintes permissões:

Luci Administrator

Fornece as mesmas permissões ao usuário e ao usuário root com permissão total em todos os clusters e a habilidade de definir ou remover permissões em todos os outros usuários exceto do root, cujas permissões não podem ser restringidas.

Pode Criar Clusters

Permite que o usuário crie novos clusters, como descrito em Seção 3.4, “Criando um Cluster” .

Pode Importar Clusters Existentes

Permite que usuários adicionem um cluster existente à interface do **luci**, como descrito em Seção 4.1, “Adicionar um Cluster Existente à interface do luci” .

Para cada cluster que for criado ou importado para o **luci**, você pode definir as seguintes permissões para o usuário indicado:]

Pode visualizar este Cluster

Permite que o usuário visualize o cluster especificado:

Pode mudar a configuração do Cluster

Permite que o usuário modifique a configuração para o cluster especificado, com a exceção de adicionar e remover nós de cluster.

Pode Habilitar, Desabilitar, Realocar, e Migrar Grupos de Serviços

Permite que o usuário gerencie serviços de alta disponibilidade como descrito em Seção 4.5, “Gerenciando Serviços de Alta Disponibilidade”.

Pode Interromper, Iniciar e Reiniciar os Nós de Cluster

Pemite que o usuário gerencie os nós individuais de um cluster, como descrito em Seção 4.3, "Gerenciando Nós no Cluster".

Pode Adicionar e Remover Nós

Permite que o usuário adicione e remova nós de um cluster, como descrito em Seção 3.4, "Criando um Cluster".

Pode Remover este Cluster do Luci

Permite que o usuário remova um cluster da interface do **luci**, como descrito em Seção 4.4, "Iniciando, Parando, Reinicializando e Deletando Clusters".

Clique em **Submeter** para que as permissões sejam efetuadas ou clique em **Reset** para retornar aos valores iniciais.

3.4. CRIANDO UM CLUSTER

Criando um cluster com o **luci** consiste em nomear um cluster, adicionar nós no cluster, digitar senhas do **ricci** para cada nó e enviar o pedido para criar um cluster. Se as informações e senhas dos nós estiverem corretas, o **Conga** automaticamente instala o software nos nós do cluster (se o pacote de software apropriado não estiver atualmente instalado) e inicia o cluster. Crie um cluster como a seguir:

1. Clique em **Gerenciar Clusters** (Manage Clusters) no menu do lado esquerdo da página **Homebase** do **luci**. A página **Clusters** aparecerá, como mostrado na Figura 3.2, "Página de gerenciamento de cluster do luci".

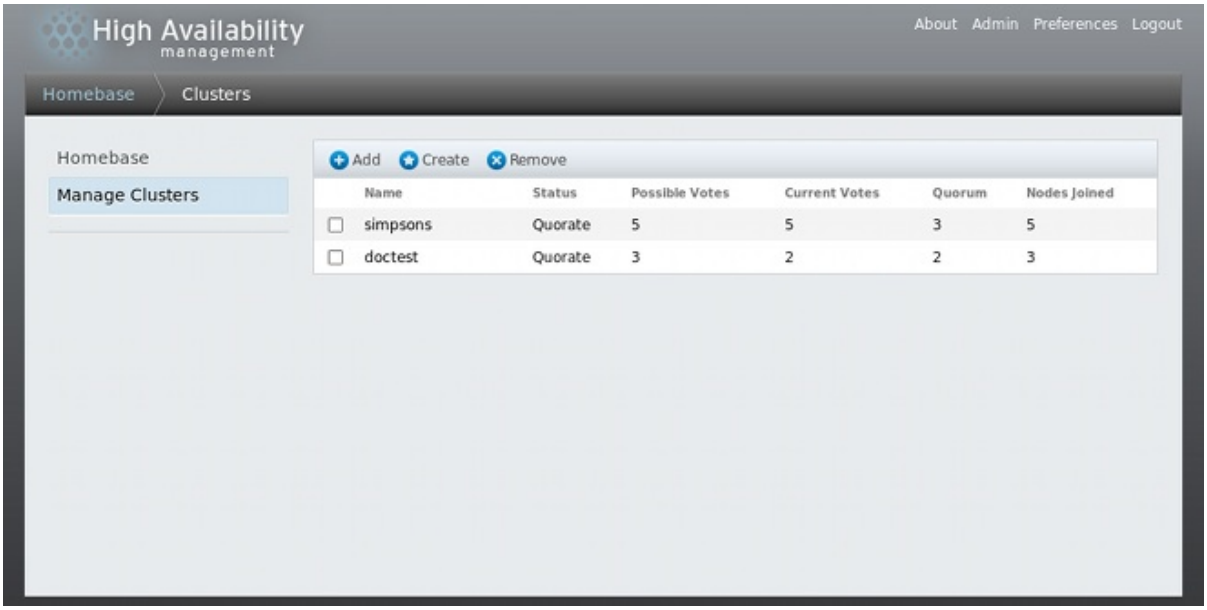


Figura 3.2. Página de gerenciamento de cluster do luci

2. Clique em **Criar** (Create). A caixa de diálogo aparece **Criar novo Cluster** (Create New Cluster), como mostrado na Figura 3.3, "Caixa de diálogo de criação de cluster do luci" .

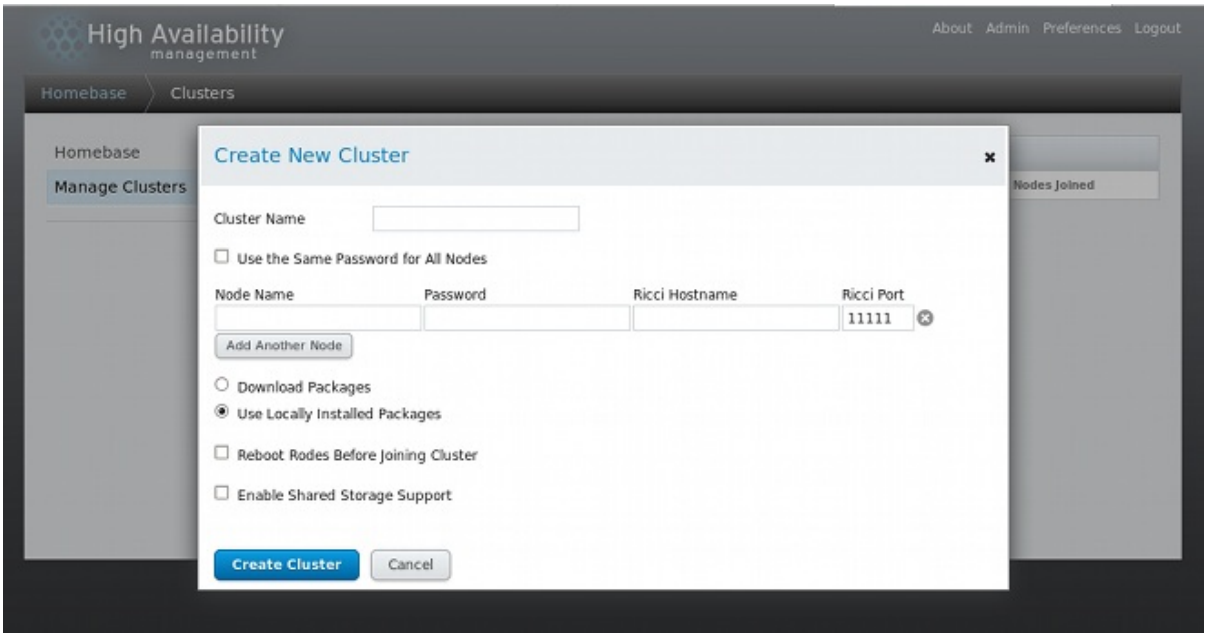


Figura 3.3. Caixa de diálogo de criação de cluster do luci

3. Entre com os seguintes parâmetros na caixa de diálogo **Criar Novo Cluster**, conforme necessário:

   - Na caixa de texto **Nome do Cluster** (Cluster Name), digite um nome de cluster. O nome do cluster não pode ultrapassar 15 caractéres.

   - Se cada nó no cluster tiver a mesma senha **ricci**, você pode marcar **Use a mesma senha para todos os nós** (Use the same password for all nodes) para autocompletar o campo o campo de **senha** conforme você adiciona nós.

   - Entre o nome do nó para um nó no cluster na coluna **Nome do Nó** (Node Name) e digite a senha **ricci** para o nó na coluna **Senha** (Password).

   - Se seu sistema estiver configurado com uma rede privada dedicada que é usada somente para tráfego do cluster, você poderá configurar o **luci** para se comunicar com o **ricci** em um endereço que é diferente do endereço no qual o nome do nó no cluster é resolvido. Você pode fazer isso digitando esse endereço como o **Ricci Hostname**.

   - Se você está usando uma porta diferente para o agente **ricci** do que a padrão 11111, você pode mudar esse parâmetro.

   - Clique em **Adicionar outro Nó** (Add Another Node) e entre o nome do nó e senha **ricci** para cada nó adicional no cluster.

   - Se você não quiser atualizar os pacotes de software do cluster que já estão instalados nos nós quando você criar um cluster, deixe selecionada a opção **Usar pacotes localmente instalados** (Use locally installed packages). Se você quiser atualizar todos os pacotes de software do cluster, selecione a opção **Baixar Pacotes** (Download Packages).

NOTA

Se você selecionar **Usar pacotes localmente instalados** (Use locally installed packages) ou a opção **Baixar Pacotes** (Download Packages), se qualquer dos componentes base do cluster estiverem faltando (**cman**, **rgmanager**, **modcluster** e suas dependências), eles serão instalados. Se eles não podem ser instalados, a criação do nó falhará.

- Selecione **Reinicializar nós antes de se juntar ao cluster** (Reboot nodes before joining cluster) se desejado.
- Selecione **Habilitar suporte a armazenamento compartilhado** se um armazenamento clusterizado é requerido; isto baixa os pacotes que suportam armazenamento clusterizado e habilita o LVM clusterizado. Você deveria selecionar isto somente quando você tiver acesso ao Complemento de Armazenamento Resiliente ou Complemento de Sistema de Arquivos Escalável.

4. Clique em **Criar Cluster** (Create Cluster). Clicando em **Criar Cluster** causa as seguintes ações:
   1. Se você selecionou **Baixar Pacotes** (Download Packages), os pacotes de software do cluster são baixados nos nós.
   2. O software de Cluster está instalado nos nós (ou é checado que os pacotes de software apropriados estão instalados).
   3. O arquivo de configuração do cluster está atualizado e propagado em cada nó no cluster.
   4. Os nós adicionados se juntam ao cluster.

   Uma mensagem é exibida indicando que o cluster está sendo criado. Quando um cluster estiver pronto, a exibição mostra o estado do cluster recém criado, conforme mostrado na Figura 3.4, “Exibição do nó no cluster” . Note que se o **ricci** não estiver rodando em qualquer um dos nós, a criação do cluster falhará.

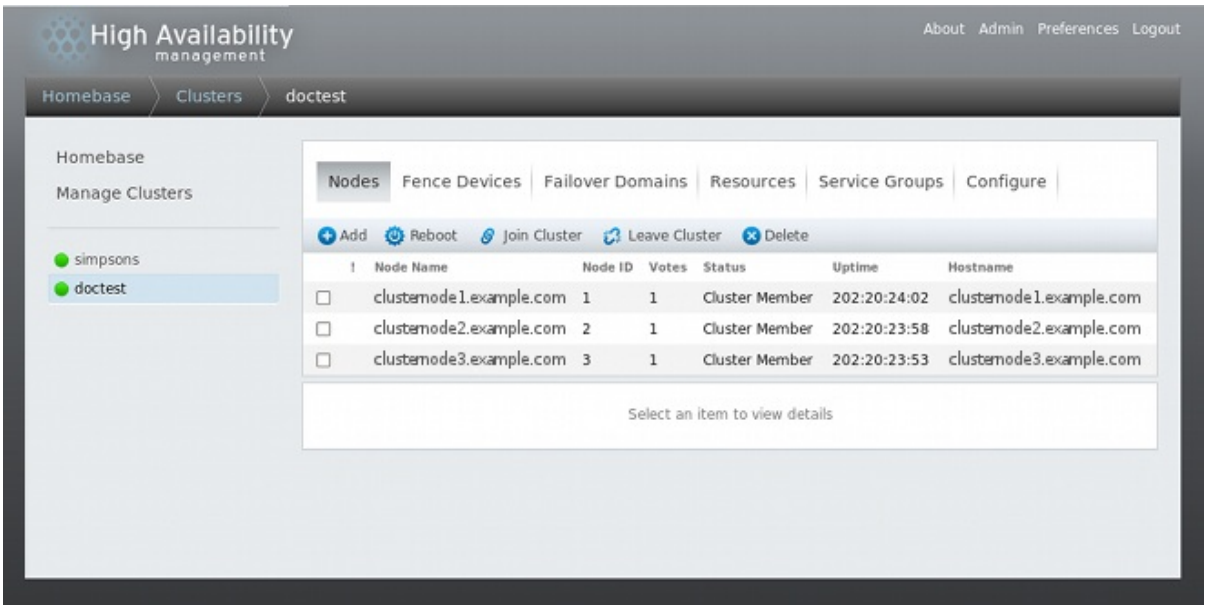


Figura 3.4. Exibição do nó no cluster

5. Depois de clicar em **Criar Cluster** (Create Cluster), você pode adicionar ou deletar nós do

cluster clicando nas funções **Adicionar** (Add) ou **Deletar** (Delete) no menu no topo da página de exibição do nó no cluster. A menos que você esteja deletando um cluster inteiro, nós devem ser parados antes de serem deletados. Para informações sobre deletar um nó de um cluster existente que está atualmente em operação, veja a Seção 4.3.4, “Excluindo um Membro de um Cluster”.

NOTA

Remover um nó de cluster de um cluster é uma operação destrutiva que não pode ser desfeita.

3.5. PROPRIEDADES DE CLUSTER GLOBAIS

Quando você selecionar um cluster para configurar, uma página de cluster específico será exibida. A página fornece uma interface para a configuração de propriedades de todo o cluster. Você pode configurar as propriedades de todo o cluster clicando em **Configurar** no topo da exibição do cluster. Isto gera uma interface marcada que provê as seguintes abas: **Geral**, **Fence Daemon**, **Network**, **Anel Redundante**, **QDisk** e **Logging**. Para configurar parâmetros nestas abas, siga os passos nas seções a seguir. Se você não precisar configurar parâmetros em uma aba, pule a seção desta aba.

3.5.1. Propriedades Gerais de Configuração

Clicando na aba **Geral** (General), é exibida a página **Propriedades Gerais** (General Properties), que fornece uma interface para modificar a versão da configuração.

- A caixa de texto **Nome do Cluster** (Cluster Name) exibe o nome do cluster; ela não aceita mudança de nome de cluster. A única maneira de mudar o nome do cluster é criar uma nova configuração de cluster com um novo nome.
- O valor **Versão da Configuração** (Configuration Version) é definido para **1** no momento da criação de cluster e é automaticamente incrementado cada vez que você modifica suas configurações de cluster. Entretanto, se você precisar definir isso para um outro valor, você pode especificar na caixa de texto **Versão da Configuração** (Configuration Version).

Se você mudou o valor da **Versão da Configuração** (Configuration Version), clique em **Aplicar** (Apply) para que as mudanças tenham efeito.

3.5.2. Propriedades de Configuração do Daemon Fence

Clicar na aba **Fence Daemon** exibe a página **Propriedades do Daemon Fence** (Fence Daemon Properties), que fornece uma interface para configurar **Post fail delay** e **Post join delay**. Os valores que você configurar para estes parâmetros são propriedades gerais de fence para o cluster. Para configurar os dispositivos de fence específicos para o nós do cluster, use o item do menu **Dispositivos Fence** (Fence Devices) na exibição de cluster, conforme descrito em Seção 3.6, “Configurando Dispositivos Fence”.

- O parâmetro **Post Fail Delay** é o número de segundos que o fence daemon (**fenced**) espera antes de fazer um fence em um nó (um membro de domínio fence) depois que um nó tiver falhado. O valor padrão do **Post fail delay** é **0**. Seu valor pode ser variado para se adequar ao desempenho da rede e do cluster.
- O parâmetro **Post Fail Delay** é o número de segundos que o fence daemon (**fenced**) espera antes de fazer um fence em um nó depois que o nó se unir ao domínio do fence. O valor padrão do **Post fail delay** é **6**. Uma configuração típica para **Post Join Delay** é entre 20 e 30

segundos, mas pode ser variado para se adequar ao desempenho da rede e do cluster.

Digite os valores requeridos e clique em **Aplicar** (Apply) para as mudanças terem efeito.

NOTA

Para informações sobre o **Post join delay** e **Post fail delay**, consulte a página man fenced(8).

3.5.3. Configuração de Rede

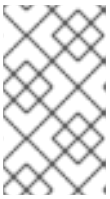
Clicando na aba **Network**, é exibido a página **Configuração de Rede** (Network Configuration), que fornece uma interface para configurar o tipo de transporte de rede.

Você pode usar esta aba para selecionar as seguintes opções:

- **multicast UDP e deixe o cluster escolher o endereço de multicast** (UDP multicast and let cluster choose the multicast address)

  Esta é a configuração padrão. Com esta opção selecionada, o software do Complemento de Alta Disponibilidade da Red Hat cria um endereço multicast baseado no ID do cluster. Ela gera os 16 bits mais baixos do endereço e os acrescenta à porção mais alta do endereço de acordo se o protocolo IP é IPV4 ou IPV6:

  - Para o IPV4 — O endereço formado é 239.192. mais os 16 bits mais baixos gerados pelo software do Complemento de Alta Disponibilidade da Red Hat.
  - Para o IPV6 — O endereço formado é FF15:: mais os 16 bits mais baixos gerados pelo software do Complemento de Alta Disponibilidade da Red Hat.

  NOTA

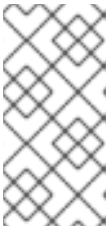
  O ID do cluster é um identificador único que gera o **cman** para cada cluster. Para vizualizar o ID do cluster, rode o comando **cman_tool status** em um nó do cluster.

- **Multicast UDP e especifique o endereço multicast manualmente** (UDP multicast and specify the multicast address manually)

  Se você precisar usar um endereço multicast específico, selecione esta opção e digitie um endereço multicast na caixa de texto do **Multicast Address**.

  Se você especificar um endereço multicast, você deveria usar as séries 239.192.x.x (ou FF15:: para IPv6) que o **cman** usa. Senão, usar um endereço multicast fora deste alcance pode causar resultados imprevisíveis. Por exemplo, usando 224.0.0.x (que é "Todos os hosts na rede") podem não ser roteados corretamente ou mesmo não serem roteados completamente em alguns hardwares.

  Se você especificar ou modificar um endereço multicast, você precisa reiniciar o clusterpara que seja efetuado. Para instruções sobre reiniciar os software de cluster com o **Conga**, consulte a Seção 4.4, “Iniciando, Parando, Reinicializando e Deletando Clusters” .



NOTA

Se você especificar um endereço multicast, certifique-se que você verificou a configuração dos roteadores para os pacotes do cluster passarem. Alguns roteadores podem levar um longo tempo para aprender os endereços, impactando seriamente no desempenho do cluster.

- **UDP Unicast (UDPU)**

  Desde o lançamento do Red Hat Enterprise Linux 6.2, os nós em um cluster podem se comunicar entre si utilizando o mecanismo de transporte do Unicast UDP. Recomenda-se no entanto, que você utilize o multicasting do IP para a rede de cluster. O Unicast UDP é uma alternativa que pode ser utilizada quando o musticast IP não estiver disponível. Para as implementações do GFS2 não recomendamos o uso do Unicast UDP.

Clique em **`Aplicar`** (Apply). Quando alterar o tipo de transporte, uma reinicialização de cluster é necessária para as mudanças terem efeito.

3.5.4. Configurando, Protocolo de Anel Redundante

Desde o Red Hat Enterprise Linux 6.4, o Red Hat High Availability Add-On suporta a configuração do protocolo de anel redundante. Ao utilizar o protoloco de anel redundante, existem diversas considerações que você deve levar em conta, como descrito em Seção 7.6, “Configurando o Protocolo de Anel Redundante”.

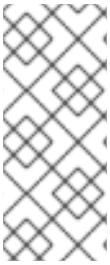
Ao clicar na aba **Anel Redundante**, será exibido a página do **Configuração do Protocolo de Anel Redundante**. Esta página exibe todos os nós que são configurados atualmente para o cluster. Se você estiver configurando um sistema para usar o protocolo de anel redundante, você precisa especificar o **Nome Alternado** para cada nó para o segundo anel.

A página **Configuração de Protocolo de Anel Redundante** permite que você especifique o **Endereço do Multicast de Anel Alternado**, a **Porta CMAN do Anel Alternado** , e o **TTL do Pacote Multicast do Anel Alternado** para o segundo anel.

Se você especificar um endereço multicast para o segundo anel, tanto o endereço de multicast alternado quanto a porta alternada devem ser diferentes do endereço do multicast para o primeiro anel. Se você especificar uma porta alternada, os números de porta do primeiro anel e segundo anel devem ser diferentes em pelo menos dois, pois o próprio sistema usa a porta e porta-1 para realizar as operações. Se você não especificar um endereço e multicast alternado, o sistema irá utilizar automaticamente um endereço de multicast diferente para o segundo anel.

3.5.5. Configuração de Disco de Quorum

Clicando na aba **Qdisk**, é exibida a página **Configuração de Disco de Quorum**, que fornece uma interface para configuração dos parâmetros de disco de quorum para configuração se você precisar.



NOTA

Os parâmetros de disco de quorum e heurísticas dependem do ambiente do local e os requerimentos especiais necessários. Para entender o uso dos parâmetros de quorum de disco e heurísticas, consulte a página man qdisk(5). Se você precisar de assistência para entender e usar o disco de quorum, contate um representante de suporte autorizado Red Hat.

O parâmetro **Não use um Quorum de Disco** (Do not use a Quorum Disk) é habilitado por padrão. Se você precisar usar um disco de quorum, clique em **Usar um Disco de Quorum** (Use a Quorum Disk), digite os parâmetros de disco de quorum, clique **Aplicar** (Apply) e reinicie o cluster para as mudanças terem efeito.

Tabela 3.1, "Parâmetros de Disco de Quorum" descreve os parâmetros de disco de quorum.

Tabela 3.1. Parâmetros de Disco de Quorum

Parâmetros	Descrição
Especificar dispositivo físico: Por rótulo de dispositivo (Specify physical device: By device label)	Especifica o rótulo de disco de quorum criado pelo utilitário `mkqdisk`. Se este campo é usado, o daemon quorum lê o `/proc/partitions` e verifica por assinaturas qdisk em cada dispositivo de bloco encontrado, comparando o rótulo com o rótulo especificado. Isto é útil em configurações onde o nome de dispositivo de quorum difere entre nós.
Heurísticas (Heuristics)	**Path to Program** — O programa utilizado para determinar se esta eurística está disponível. Isto pode ser qualquer coisa que possa ser executada pelo `/bin/sh -c`. Um valor de retorno de 0 indica sucesso; qualquer outra coisa indica falha. Este campo é requerido. **Intervalo** (Interval) — A frequência (em segundos) na qual a heurística é consultada. O intervalo padrão para cada heurística é 2 segundos. **Contagem** (Score) — O peso desta heurística. Seja cuidadoso quando determinar estes valores para heurística. A contagem padrão para cada heurística é 1. **TKO** — O número de falhas consecutivas requeridas antes desta heurística ser declarada indisponível.
Contagem Mínima Total (Minimum total score)	A contagem mínima para um nó ser considerado "vivo". Se omitido ou definido para 0, a função padrão `floor((n+1)/2)`, é usada, onde *n* é a soma das contagens de heurísticas. O valor **Contagem Mínima** nunca deve exceder a soma das contagens das heurísticas; caso contrário, o disco de quorum não pode estar disponível.

NOTA

Clicando em **Aplicar** (Apply) na aba **Configuração QDisk** (QDisk Configuration) propaga mudanças no arquivo de configuração de cluster (`/etc/cluster/cluster.conf`) em cada nó do cluster. Entretanto, para o disco de quorum operar, você deve reiniciar o cluster (consulte a Seção 4.4, "Iniciando, Parando, Reinicializando e Deletando Clusters").

3.5.6. Configuração de Log

Clicando na aba **Logging** será exibida a página de **Configuração de Log**, que fornece uma interface para configurar definições de log.

Você pode definir as seguintes configurações de log globais:

- Marcando **Mensagens de Depuração de Log** (Log debugging messages) habilita as mensagens de depuração no arquivo de log.
- Marcando **Logar mensagens no syslog** (Log messages to syslog) habilita mensagens no **syslog**. Você pode selecionar a **facilidade de mensagens do syslog** (syslog message facility) e o **prioridade de mensagens do syslog** (syslog message priority). Este último indica que mensagens de um nível determinado e maior sejam enviados ao **syslog**.
- Marcando **Logar mensagens ao arquivo de log** (Log messages to log file) habilita mensagens ao arquivo de log. Você pode especificar o nome do caminho de arquivo de log. A configuração **prioridade de mensagens do arquivo de log** (logfile message priority) indica que mensagens a um nível selecionado e maior sejam escritos no arquivo de log.

Você pode sobrescrever as configurações de log globais para específicos daemons selecionando um dos daemons na parte inferior da página **Configuração de Log** (Logging Configuration). Depois de selecionar o daemon, você pode marcar se dever logar mensagens de depuração para aquele daemon em particular. Você pode também especificar o **syslog** e as configurações de arquivo de log para esse daemon.

Clique em **Aplicar** (Apply) para as mudanças de configuração de log que você especificou terem efeito.

3.6. CONFIGURANDO DISPOSITIVOS FENCE

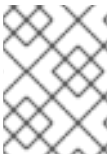
Configurar dispositivos fence consiste na criação, atualização e exclusão de dispositivos fence para o cluster. Você deve configurar os dispositivos fence em um cluster antes que você possa configurar o fence para os nós no cluster.

Criando um dispositivo de fence consiste em selecionar um tipo de dispositivo fence e digitar parâmetros para aquele dispositivo fence (por exemplo, nome, endereço IP, login e senha). Atualizar um dispositivo fence consistem em selecionar um dispositivo de fence existente e alterar os parâmetros para aquele dispositivo fence. Excluir um dispositivo fence consiste em selecionar um dispositivo fence existente e exclui-lo.

Esta seção fornece procedimentos para as seguintes tarefas:

- Criando dispositivos fence — Consulte a Seção 3.6.1, “Criando um Dispositivo Fence” . Uma vez criado e nomeado um dispositivo fence, você pode configurar os dispositivos fence para cada nó no cluster, conforme descrito na Seção 3.7, “Configurar Fence para Membros do Cluster”.
- Atualizando dispositivos fence — Consulte a Seção 3.6.2, “Modificando um Dispositivo Fence” .
- Excluir dispositivos fence — Consulte a Seção 3.6.3, “Deletando um Dispositivo Fence” .

A partir da página específica do cluster, você pode configurar dispositivos fence para aquele cluster clicando em **Dispositivos Fence** no topo da exibição do cluster. Isso exibe os dispositivos fence para o cluster e exibe os ítens do menu para configuração de dispositivos fence: **Adicionar** (Add), e **Remover** (Delete). Este é o ponto de partida para cada procedimento descrito nas seguintes seções.



NOTA

Se esta é uma configuração inicial de cluster, nenhum dispositivo fence foi criado e portanto nenhum é exibido.

A Figura 3.5, “Página de configuração de dispositivos fence do luci” mostra a tela de configuração dos dispositivos fence antes de qualquer dispositivo ter sido criado.

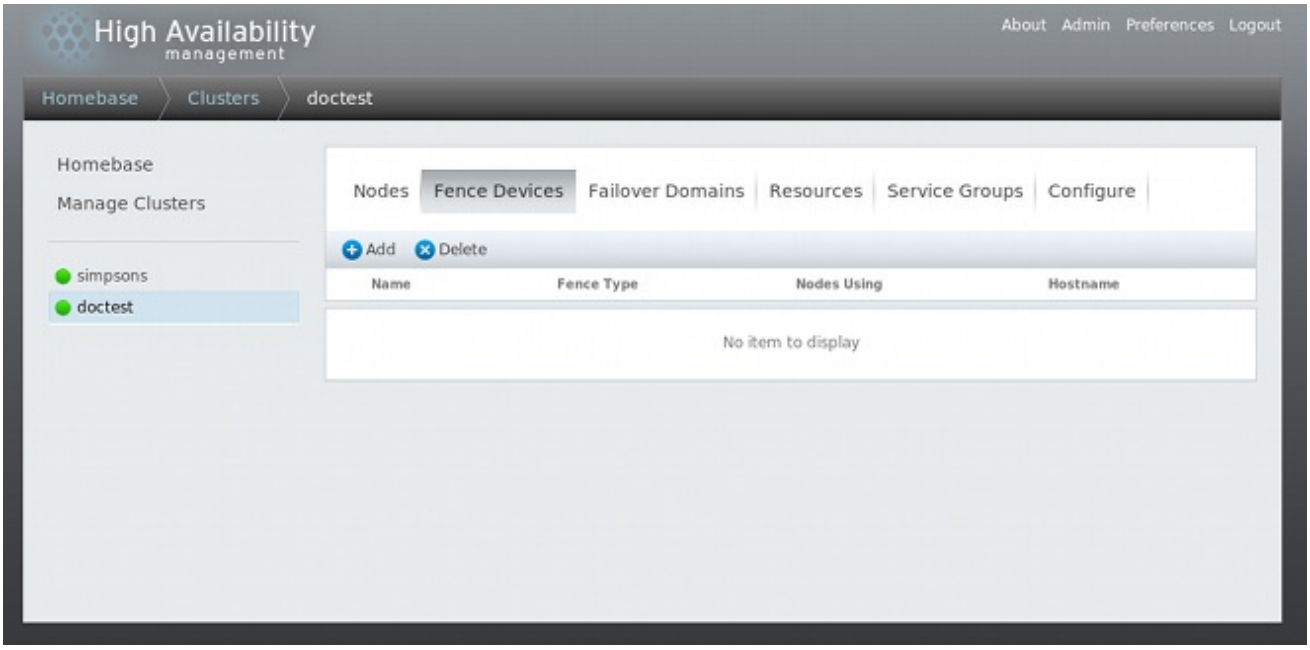


Figura 3.5. Página de configuração de dispositivos fence do luci

3.6.1. Criando um Dispositivo Fence

Para criar um dispositivo fence, siga os seguintes passos:

1. Da página de configuração de **Dispositivos Fence**, clique em **Adicionar** (Add). Clicando em **Adicionar** é exibida a caixa de diálogo **`Adicionar Dispositivo Fence (Instância)`** (Add Fence Device (Instance)). Desta caixa de diálogo, selecione o tipo de dispositivo fence para configurar.

2. Especifique a informação na caixa de diálogo de acordo **`Adicionar Dispositivo Fence (Instância)`** (Add Fence Device (Instance)) de acordo com o tipo de dispositivo fence. Consulte o Apêndice A, *Parâmetros de Dispositos Fence*para mais informações sobre parâmetros de dispositivo fence. Em alguns casos você precisará especificar parâmetros de nós específicos para o dispositivo fence quando configurar o fence para nós individuais, conforme descrito na Seção 3.7, “Configurar Fence para Membros do Cluster”.

3. Clique em **`Enviar`** (Submit)

Depois do dispositivo fence ter sido adicionado, ele aparece na página de configuração **Dispositivos Fence** (Fence Devices).

3.6.2. Modificando um Dispositivo Fence

Para modificar um dispositivo fence, siga estes passos:

1. Da página de configuração **Dispositivos Fence** (Fence Devices), clique no nome do dispositivo fence a ser modificado. Isto mostrará a caixa de diálogo para aquele dispositivo fence, com os valores que foram configurados para o dispositivo.

2. Para modificar o dispositivo fence, digite as mudanças aos parâmetros exibidos. Consulte o Apêndice A, *Parâmetros de Dispositos Fence*para mais informações.

3. Clique em **Aplicar** (Apply) e aguarde a configuração ser atualizada.

3.6.3. Deletando um Dispositivo Fence

NOTA

Dispositivos fence que estão em uso não podem ser deletados. Para deletar um dispositivo fence que um nó está usando atualmente, primeiro atualize a configuração de fence do nó para qualquer nó usando o dispositivo e então delete o dispositivo.

Para deletar um dispositivo fence, siga estes passos:

1. Da página de configuração **Dispositivos Fence** (Fence Devices), marque a caixa na esquerda do ou dos dispositivos para selecionar quais deletar.

2. Clique em **Delete** e aguarde pela configuração ser atualizada. Uma mensagem aparece indicando quais dispositivos estão sendo deletados.

Quando a configuração for atualizada, o dispositivo fence deletado não aparece mais na exibição.

3.7. CONFIGURAR FENCE PARA MEMBROS DO CLUSTER

Uma vez que você completou os passos iniciais sobre criar um cluster e criar dispositivos fence, você precisa configurar o fence para nós no cluster. Para configurar o fence para os nós depois de criar um novo cluster e configurar os dispositivos fence para o cluster, siga os passos desta seção. Note que você deve configurar o fence para cada nó no cluster.

As seções seguintes fornecem procedimentos para configurar um dispositivo fence único para um nó, configurando um nó com um dispositivo fence de backup e configurar um nó com fonte de alimentação redundante:

- Seção 3.7.1, “Configurar um Dispositivo Fence Único para um Nó”
- Seção 3.7.2, “Configurando um Dispositivo Fence de Backup”
- Seção 3.7.3, “Configurando um nó com energia redundante”

3.7.1. Configurar um Dispositivo Fence Único para um Nó

Use o seguinte procedimento para configurar um nó com um dispositivo fence único.

1. Da página do cluster específico, você pode configurar o fence para os nós nos cluster clicando em **Nós** (Nodes) no topo da exibição de cluster. Ela exibe os nós que constituem um cluster. Ela é também a página padrão que aparece quando você clica no nome do cluster abaixo de **Manage Clusters** (Gerenciar Clusters) no menu do lado esquerdo da página **Homebase** do **luci**.

2. Clique no nome do nó. Clicar em um link para um nó leva à exibição de como o nó é configurado.

   A página específica do nó exibe quaisquer serviços que estão atualmente rodando no nó, tanto como quaisquer domínios failover de qual o nó é um membro. Você pode modificar um domínio failover existente clicando em seu nome. Para informações sobre configurar domínios failover, veja a Seção 3.8, “Configurando um Domínio de Failover” .

3. Na página específica do nó, sob **Dispositivos Fence** (Fence Devices), clique em **Adicionar Método Fence** (Add Fence Method).

4. Digite o **Nome do Método** (Method Name) para o método fence que você está configurando para este nó. Isto é um nome arbitrário que será usado pelo Complemento de Alta Disponibilidade da Red Hat; este não é o mesmo como o nome de DNS para o dispositivo.

5. Clique em **Enviar** (Submit). Isto exibe a tela do nó específico que agora mostra o método que você acabou de adicionar sob **Dispositivos Fence** (Fence Devices).

6. Configure uma instância fence para este método clicando no botão **Adicionar Instância Fence** (Add Fence Instance) que aparece abaixo do método fence. Isto exibe o menu suspenso do qual você pode selecionar um dispositivo fence que você configurou anteriormente, conforme descrito na Seção 3.6.1, “Criando um Dispositivo Fence” .

7. Selecione um dispositivo fence para este método. Se este dispositivo fence requer que você configure parâmetros do nó específico, que exibe os parâmentros para configurar. Para informações sobre parâmetros fence, consulte o Apêndice A, *Parâmetros de Dispositos Fence*

NOTA

Para métodos fence sem energia (que é SAN/Armazenamento Fencing), o **Unfencing** é selecionado por padrão na exibição de parâmetros do nó específico. Isto garante que o acesso de fence do nó ao armazenamento não é reabilitado até que o nó foi reiniciado. Para informações sobre "unfencing" em um nó, consulte a página man **fence_node**(8).

8. Clique em **Enviar** (Submit). Isto retornará à tela do nó especifico com o método fence e a instância fence exibida.

3.7.2. Configurando um Dispositivo Fence de Backup

Você pode definir múltiplos métodos fence para um nó. Se o fence falhar usando o primeiro método, o sistema tentará fazer o fence no nó usando o segundo método, seguido por quaisquer métodos adicionais que você configurou.

Use o seguinte procedimento para configurar um dispositivo fence de backup para um nó.

1. Use o procedimento fornecido na Seção 3.7.1, “Configurar um Dispositivo Fence Único para um Nó” para configurar o fence primário para um nó.

2. Abaixo da exibição do primeiro método que você definiu, clique **Adicionar Método Fence** (Add Fence Method).

3. Digite um nome para o método fence de backup que você está configurando para este nó e clique em **Enviar** (Submit). Isto exibe a tela do nó específico que agora exibe o método que você acabou de adicionar, abaixo o método primário de fence.

4. Configure uma instância fence para este método clicando em **Adicionar Instância Fence** (Add Fence Instance). Isto exibe um menu suspenso do qual você pode selecionar um dispositivo fence que você configurou previamente, conforme descrito na Seção 3.6.1, “Criando um Dispositivo Fence”.

5. Selecione um dispositivo fence para este método. Se este dispositivo fence requer que você configure parâmetros do nó específico, que exibe os parâmentros para configurar. Para informações sobre parâmetros fence, consulte o Apêndice A, *Parâmetros de Dispositos Fence*.

6. Clique em **Enviar** (Submit). Isto retornará à tela do nó especifico com o método fence e a instância fence exibida.

Você pode continuar a adicionar métodos fence conforme necessário. Você pode rearranjar a ordem dos métodos fence que serão usados para este nó clicando em **Mover para cima** (Move Up) e **Mover para baixo** (Move Down).

3.7.3. Configurando um nó com energia redundante

Se seu cluster estiver configurado com fonte de alimentação redundante para seus nós, você deve se certificar de configurar o fence para que então seus nós desliguem completamente quando eles precisam ter um fence. Se você configurar cada fonte de alimentação como um método fence separado, cada fonte de alimentação terá um fence separadamente; a segunda fonte de alimentação permitirá ao sistema continuar rodando quando a primeira fonte de alimentação estiver com fence e o sistema não terá um fence em geral. Para configurar um sistema com duas fontes de alimentação, você deve configurar seus dispositivos fence para que ambas fontes de alimentação sejam desligadas e o sistema é completamente desativado. Quando configurar seus sistema usando o **Conga**, isto requer que você configure duas instâncias dentro de um método fence único.

Para configurar o fence para um nó com duas fontes de alimentação, siga os passos desta seção.

1. Antes de você poder configurar o fence para um nó com energia redundante, você deve configurar cada um dos switches de energia como um dispositivo fence para o cluster. Para informações sobre configurar dispositivos fence, veja a Seção 3.6, “Configurando Dispositivos Fence”.

2. Da página especifica do cluster, clique em **Nós** no topo da exibição do cluster. Isto mostra os nós que constituem o cluster. Ela também é a página padrão que aparece quando você clica no nome do cluster abaixo do **Gerenciar Clusters** (Manage Clusters) do menu do lado esquerdo da página **Homebase** do **luci**.

3. Clique no nome do nó. Clicar em um link para um nó leva à exibição de como o nó é configurado.

4. Na página especifica do nó, clique em **Adicionar Método Fence** (Add Fence Method).

5. Digite um nome para o método fence que você está configurando para este nó.

6. Clique em **Enviar** (Submit). Isto exibe a tela do nó específico que agora mostra o método que você acabou de adicionar sob **Dispositivos Fence** (Fence Devices).

7. Configure a primeira fonte de energia como uma instância de fence para este método clicando em **Adicionar Instância Fence** (Add Fence Instance). Isto exibe um menu suspenso no qual você pode selecionar um dos dispositivos fence de energia que você configurou previamente, conforme descrito na Seção 3.6.1, “Criando um Dispositivo Fence” .

8. Selecione um dos dispositivos fence de energia para este método e digite os parâmetros apropriados para este dispositivo.

9. Clique em **Enviar** (Submit). Isto retornará à tela do nó especifico com o método fence e a instância fence exibida.

10. Sob o mesmo método fence do qual você configurou o primeiro dispositivo fence de energia, clique em **`Adicionar Instância Fence`** (Add Fence Instance). Isto mostra um menu suspenso do qual você pode selecionar o segundo dispositivo fence de energia que você configurou anteriormente, conforme descrito na Seção 3.6.1, "Criando um Dispositivo Fence" .

11. Selecione o segundo dos dispositivos fence de energia para este método e digite os parâmetros apropriados para este dispositivo.

12. Clique em **`Enviar`** (Submit). Isto lhe retorna à tela específica do nó com os métodos fence e as instâncias fence exibidas, mostrando que cada dispositivo desligará o sistema em sequencia e ligará o sistema em sequencia. Isto é mostrado na Figura 3.6, "Configuração Fence de Fonte Dupla".

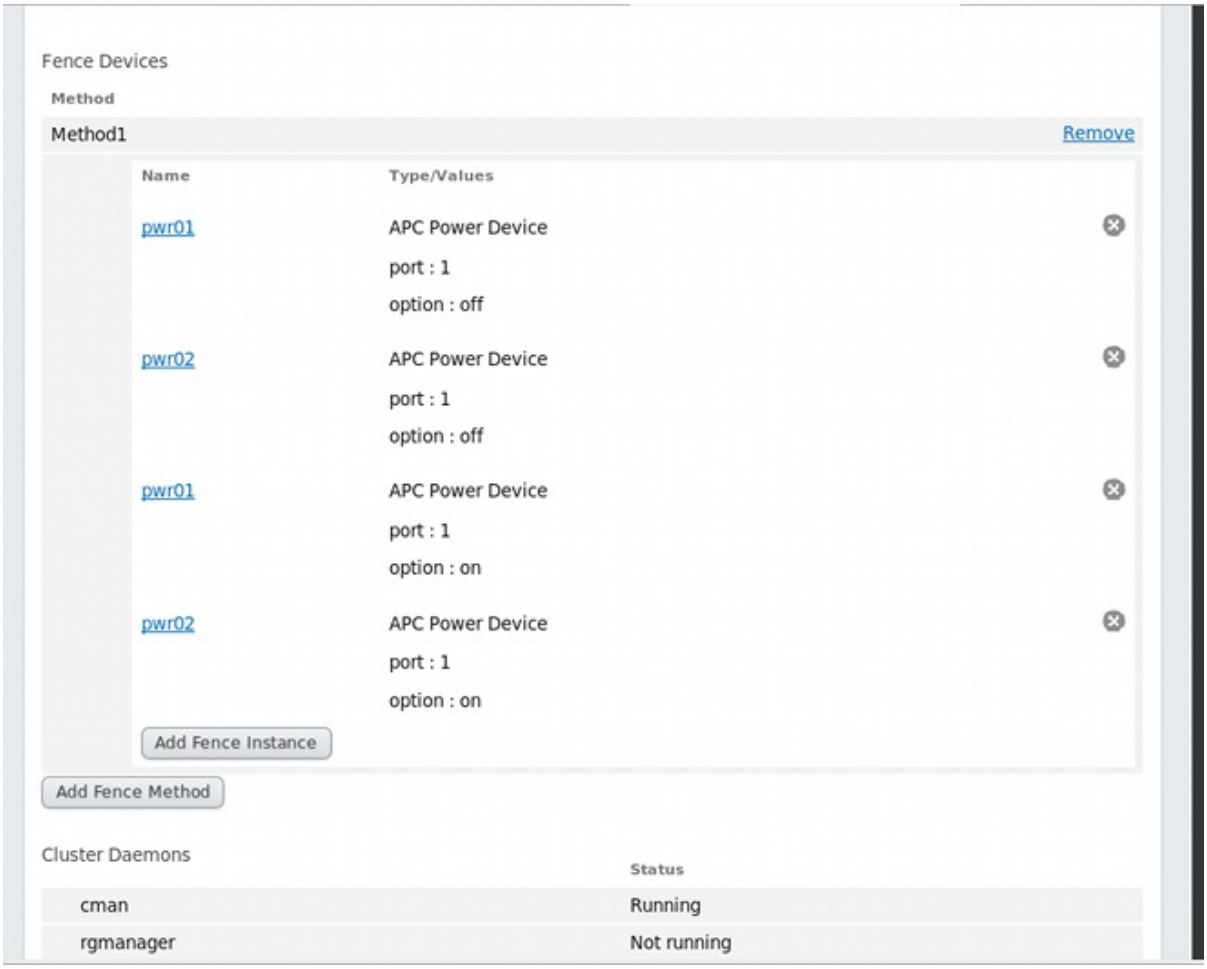


Figura 3.6. Configuração Fence de Fonte Dupla

3.8. CONFIGURANDO UM DOMÍNIO DE FAILOVER

Um domínio de failover é um sub conjunto de nós do cluster que são elegíveis para rodar um serviço de cluster em um evento de falha de um nó. Um domínio failover pode ter as seguintes características:

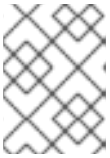
- Irrestrito (Unrestricted) — Permite especificar que um sub conjunto de membros são preferidos mas que um serviço de cluster atribuído a este domínio possa rodar em qualquer membro disponível.

- Restringido (Restricted) — Permite restringir os membros que podem rodar um serviço de cluster em particular. Se nenhum dos membros em um domínio de failover estiverem disponíveis, o serviço de cluster não pode ser iniciado (tanto manualmente ou pelo software de cluster).

- Desordenado (Unordered) — Quando um serviço de cluster é atribuído a um domínio de failover desordenado, o membro no qual o serviço de cluster roda é escolhido a partir dos membros do domínio de failover disponíveis sem prioridade de ordem.

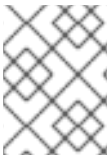
- Ordenados (Ordered) — Permite especificar a ordem de preferência entre os membros de um domínio de failover. O membro no topo da lista é o preferido, seguido pelo segundo e assim por diante.

- Failback — Permite especificar se um serviço no domínio de failover deveria fazer um fail back no nó que estava originalmente rodando antes desse nó falhar. Configurar esta característica é útil em circunstâncias onde um nó falha repetidamente e é parte de um domínio de failover ordenado. Nesta circunstância, se o nó é o preferido no domínio de failover, é possível para um serviço fazer fail over e fail back repetidamente entre o nó preferido e o outro nó, causando um impacto severo no desempenho.



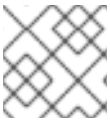
NOTA

A característica failback é aplicável somente se o failover ordenado é configurado.



NOTA

Alterar a configuração de um domínio de failover não possui efeito em serviços atualmente em execução.



NOTA

Domínios failover *não* são requeridos para operação.

Por padrão, domínios failover são irrestritos e desordenados.

Em um cluster com diversos membros, usar um domínio failover restringido pode minimizar o trabalho de configurar o cluster para executar o serviço de cluster (tal como **`httpd`**), que requer que você defina identicamente a configuração em todos os membros que rodam o serviço de cluster. Em vez de definir o cluster inteiro para rodar o serviço de cluster, você pode definir somente os membros no domínio de failover restringidos que você associa com o serviço de cluster.

NOTA

Para configurar um membro preferido, você pode criar um domínio de failover irrestrito compreendendo de somente um membro do cluster. Isso faz que um serviço de cluster rode naquele membro de cluster primariamente (o membro preferido) mas permite que o serviço de cluster faça um fail over em qualquer um dos outros membros.

As seguintes seções descrevem a adição, modificação e exclusão de um domínio failover:

- Seção 3.8.1, “Adicionando um Domínio Failover”
- Seção 3.8.2, “Modificando um Domínio de Failover”
- Seção 3.8.3, “Excluir um Domínio de Failover”

3.8.1. Adicionando um Domínio Failover

Para adicionar um domínio failover, siga os seguintes passos nesta seção.

1. A partir da página específica do cluster, você pode configurar domínios failover para o cluster clicando em **Domínios Failover** (Failover Domains) no topo da tela de exibição do cluster. Ela mostra domínios failover que foram configurados para este cluster.

2. Clique em **Adicionar** (Add) para mostrar a caixa de diálogo **Adicionar Domínio Failover ao Cluster** (Add Failover Domain to Cluster), como mostrado na Figura 3.7, "Caixa de Diálogo de Configuração do domínio failover do luci".

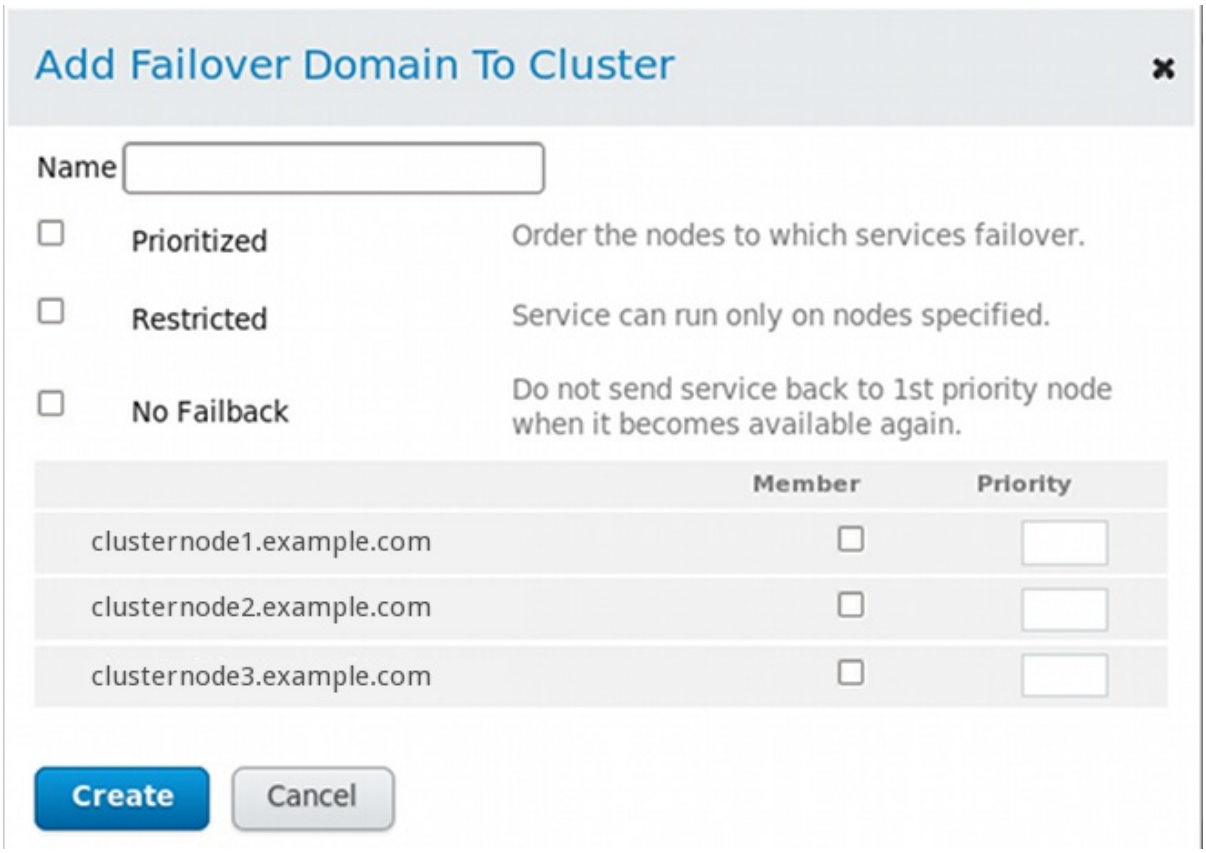


Figura 3.7. Caixa de Diálogo de Configuração do domínio failover do luci

3. Na caixa de diálogo **`Adicionar Domínio de Failover ao Cluster`** (Add Failover Domain to Cluster), especifique um nome de domínio failover na caixa de texto **Name**.

NOTA

O nome deve ser descritivo o suficiente para distinguir seu relativo propósito de outros nomes usados no seu cluster.

4. Para habilitar a configuração de prioridade do failover dos membros do domínio failover, marque a caixa **Priorizado** (Prioritized). Com esta caixa marcada, você pode definir o valor da prioridade **Priority** (Prioridade), para cada nó selecionado como membros do domínio failover.

5. Para restringir o failover para membros neste domínio de failover, marque a caixa **Restringido** (Restricted). Dessa maneira os serviços atribuídos a este domínio failover farão o failover somente nos nós dentro deste domínio de failover.

6. Para especificar que um nó não tenha um fail back neste domínio failover, marque a opção **Sem Failback** (No Failback). Desta maneira, se um serviço tiver um fail over a partir de um nó preferencial, o serviço não faz o fail back para o nó original uma vez que foi recuperado.

7. Configure os membros para este domínio failover. Marque a caixa **Membro** (Member) para cada nó que é para ser um membro do domínio de failover. Se **Priorizado** (Prioritized) for marcado, defina a prioridade na caixa de texto **Prioridade** (Priority) para cada membro do domínio de failover.

8. Clique em **`Criar`** (Create). Isto exibe a página do **`Domínios de Failover`** (Failover Domains) com o recém criado domínio de failover exibido. Uma mensagem indica que o novo domínio está sendo criado. Atualize a página para atualizar o estado.

3.8.2. Modificando um Domínio de Failover

Para modificar um domínio de failover, siga os seguintes passos nesta seção.

1. A partir da página específica do cluster, você pode configurar os Domínios de Failover para o cluster, clicando em **Domínios de Failover** (Failover Domains) no topo da exibição do cluster. Isto mostra domínios de failover que foram configurados para este cluster.

2. Clique no nome do domínio de failover. Isto exibe a página de configuração para o domínio de failover.

3. Para modificar as propriedades **Priorizado** (Prioritized), **Restringido** (Restricted) ou **Sem Failback** (No Failback) para o domínio de failover, marque ou desmarque a opção próxima à propriedade e clique **`Atualizar Propriedades`** (Update Properties).

4. Para modificar a afiliação de domínio de failover, marque ou desmarque a opção próxima ao membro de cluster. Se um domínio de failover é priorizado, você pode também modificar a definição de prioridade para o membro do cluster. Clique **`Atualizar Configurações`**.

3.8.3. Excluir um Domínio de Failover

Para excluir um domínio de failover, siga os seguintes passos desta seção.

1. A partir da página específica do cluster, você pode configurar os Domínios de Failover para o cluster, clicando em **Domínios de Failover** (Failover Domains) no topo da exibição do cluster. Isto mostra domínios de failover que foram configurados para este cluster.

2. Selecione a opção do domínio de failover a ser deletado.

3. Clique em **Delete**.

3.9. CONFIGURAR RECURSOS DE CLUSTER GLOBAIS

Você pode configurar recursos globais que podem ser usados por qualquer serviço em execução no cluster e você pode configurar recursos que estão disponíveis somente a um serviço especificado.

Para adicionar um recurso de cluster global, siga os passos nesta seção. Você pode adicionar um recurso que é local a um determinado serviço quando você configurar o serviço, como descrito na Seção 3.10, “Adicionar um Serviço de Cluster ao Cluster” .

1. A partir da página específica do cluster, você pode adicionar recursos ao cluster clicando em **Recursos** (Resources) no topo da exibição do cluster. Isso mostra os recursos que foram configurados para o cluster.

2. Clique em **Adicionar** (Add). Isso exibe o menu suspenso **Adicionar Recurso ao Cluster** (Add Resource to Cluster).

3. Clique na caixa suspensa sobre **Adicionar Recurso ao Cluster** (Add Resource to Cluster) e selecione o tipo de recurso para configurar.

4. Digite os parâmetros para o recurso que você está adicionando. O Apêndice B, *Parâmetros dos Recursos de Alta Disponibilidade* descreve os parâmetros do recurso.

5. Clique em **Enviar** (Submit). Clicando em **Enviar** se retorna à página de recursos que exibe os **Recursos** (Resources), onde está o recurso adicionar (e outros recursos).

Para modificar um recurso existente, realize os seguintes passos.

1. Da página **Recursos** (Resources) do **luci**, clique no nome do recurso para modificar. Isto exibe os parâmetros para o recurso.

2. Edite os parâmetros do recurso.

3. Clique em **Aplicar** (Apply).

Para excluir um recurso existente, realize os seguintes passos.

1. Da página **Resources** do **luci**, marque a caixa para os recursos a deletar.

2. Clique em **Delete**.

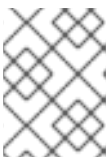
3.10. ADICIONAR UM SERVIÇO DE CLUSTER AO CLUSTER

Para adicionar um serviço de cluster ao cluster, siga os passos desta seção.

1. A partir da página específica do cluster, você pode adicionar serviços ao cluster clicando em **Grupos de Serviço** (Service Groups) no topo de exibição do cluster. Isto exibe os serviços que foram configurados para esse cluster. (A partir da página **Grupos de Serviço**, você pode também iniciar, reiniciar e desabilitar um serviço conforme descrito na Seção 4.5, "Gerenciando Serviços de Alta Disponibilidade".)

2. Clique em **Adicionar** (Add). Isso exibe a caixa de diálogo **Adicionar Serviço ao Cluster** (Add Service do Cluster).

3. Na caixa de diálogo **Adicionar Serviço ao Cluster** (Add Service to Cluster), na caixa de texto **Nome do Serviço** (Service Name), digite o nome do serviço.



NOTA

Use um nome descritivo que claramente distingue o serviço de outros serviços no cluster.

4. Marque a caixa de seleção **Iniciar este serviço automaticamente** (Automatically start this service) se você quiser que o serviço inicie automaticamente quando um cluster é iniciado e em execução. Se a caixa *não* estiver marcada, o serviço deve ser iniciado manualmente em qualquer momento após o cluster sair do estado parado.

5. Marque a caixa de seleção **Rodar exclusivo** (Run exclusive) para definir um política onde o serviço somente roda em nós que não possuem outros serviços rodando neles.

6. Se você configurou domínios de failover para o cluster, você pode usar o menu suspenso do parâmetro **Domínio Failover** (Failover Domain) para selecionar um domínio para este serviço.

Para informações sobre configurar domínios de failover veja a Seção 3.8, “Configurando um Domínio de Failover”.

7. Use a caixa suspensa **Política de Recuperação** (Recovery Policy) para selecionar uma política para o serviço. As opções são **Realocar** (Relocate), **Reiniciar** (Restart), **Desabilitar Reiniciar** (Restart-Disable), ou **Desabilitar** (Disable).

   Selecionar a opção **Reiniciar** (Restart) indica que o sistema deve tentar reiniciar o serviço com falha antes de realocar o serviço. Selecionar a opção **Desabilitar Reiniciar** (Restart-Disable) indica que o sistema deve tentar reiniciar o serviço em questão se ele falhar, mas se a reinicialização do serviço falhar, o serviço será desabilitado em vez de ser movido para outro host no cluster.

   Se você selecionar **Reiniciar** (Restart) ou **Desabilitar Reiniciar** (Restart-Disable) como a política de recuperação para o serviço, você pode especificar o número máximo de falhas de reinicializações antes de realocar ou desabilitar o serviço, você pode especificar o período de tempo em segundos depois em que se deve ignorar uma reinicialização.

8. Para adicionar um recurso ao serviço, clique em **`Adicionar Recurso`** (Add resource). Isto leva à exibição da caixa de seleção **`Adicionar Recurso ao Serviço`** (Add Resource To Service) que permite que você adicione um recurso global existente ou adicione um novo recurso que está disponível **`somente`** a este serviço.

   - Para adicionar um recurso global existente, clique no nome do recurso existente na caixa de seleção **`Adicionar Recurso ao Serviço`** (Add Resource To Service). Isto exibe o recurso e seus parâmetros na página **Grupos de Serviços** (Services Groups) para o serviço que você está configurando. Para informações sobre adicionar ou modificar recursos globais, veja Seção 3.9, “Configurar Recursos de Cluster Globais”).

   - Para adicionar um novo recurso que está disponível somente neste serviço, selecione o tipo de recurso para configurar a partir da caixa de seleção **`Adicionar um recurso`** (Add a resource) e digite os parâmetros do recurso para o recurso que você está adicionando. O Apêndice B, *Parâmetros dos Recursos de Alta Disponibilidade*descreve os parâmetros do recurso.

   - Quando adicionar um recurso a um serviço, seja ele um recurso global existente ou um recurso disponível somente a este serviço, você pode especificar se o recurso é uma **Sub árvore Independente** (Independent subtree) ou um **Recurso não crítico** (Non-critical resource).

     Se você especificar que um recurso é uma sub árvore independente, então se o recurso falhar, somente ele é reiniciado (ao invés de todo o serviço) antes do sistema tentar uma recuperação normal. Você pode especificar o número máximo de reinicializações a serem tentadas para aquele recurso em um nó antes de implementar a política de recuperação para o serviço. Você pode também especificar o período de tempo em segundos depois que o sistema implementará a política de recuperação para o serviço.

     Se você especificar que o recurso é um recurso não crítico, então se o recurso falhar, somente aquele recurso é reiniciado e se o serviço continuar a falhar, então somente este serviço é desabilitado ao invés de todo o serviço. Você pode especificar o número máximo de reinicializações a serem tentadas para o recurso em um nó antes de desabilitar o recurso. Você pode também especificar a quantidade de tempo em segundos depois que o sistema desabilitará aquele recurso.

9. Se você quiser adicionar recursos filhos ao recurso que você está definindo, clique em **`Adicionar recurso filho`** (Add a child resource). Isto exibe a caixa suspensa **Adicionar Recurso ao Serviço** (Add Resource To Service), da qual você pode adicionar um recurso

existente global ou adicionar um novo recurso que está disponível somente a este serviço. Você pode continuar adicionando recursos filhos ao recurso para atender suas necessidades.

NOTA

Se você estiver adicionando um recurso de serviço Samba, adicione-o diretamente ao serviço, *não* como um filho de outro recurso.

10. Quando você tiver completado a adição de recursos ao serviço e tiver completado a adição de recursos filhos aos recursos, clique em **Enviar** (Submit). Clicando em **Enviar** retornará à página **Grupos de Serviços** (Service Groups), mostrando os serviços adicionados (e outros serviços).

NOTA

Para verificar a existência do recurso de serviço IP usado em um serviço de cluster, você pode usar o comando **/sbin/ip addr show** em um nó de cluster (ao invés do comando obsoleto **ifconfig**). O resultado a seguir demonstra o comando **/sbin/ip addr show** executado em um nó executando um serviço de cluster:

```
1: lo: <LOOPBACK,UP> mtu 16436 qdisc noqueue
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
    inet6 ::1/128 scope host
       valid_lft forever preferred_lft forever
2: eth0: <BROADCAST,MULTICAST,UP> mtu 1356 qdisc pfifo_fast
qlen 1000
    link/ether 00:05:5d:9a:d8:91 brd ff:ff:ff:ff:ff:ff
    inet 10.11.4.31/22 brd 10.11.7.255 scope global eth0
    inet6 fe80::205:5dff:fe9a:d891/64 scope link
    inet 10.11.4.240/22 scope global secondary eth0
       valid_lft forever preferred_lft forever
```

Para modificar um serviço existente, realize os seguintes passos.

1. Da caixa de diálogo **Grupos de Serviços** (Service Groups), clique no nome do serviço a ser modificado. Isto exibe os parâmetros e recursos que foram configurados para o serviço.

2. Edite os parâmetros de serviço.

3. Clique em **Enviar** (Submit)

Para deletar um serviço existente, realize os seguintes passos.

1. A partir da página **Grupos de Serviço** (Service Groups) do **luci**, marque a caixa para quaisquer serviços a serem deletados.

2. Clique em **Delete**.

3. Desde o Red Hat Enterprise Linux 6.3, antes do **luci** remover qualquer serviço, aparecerá uma mensagem pedindo que você confirme que você pretende remover os grupos de serviço ou grupo, que interrompe os recursos que comprometem-no. Clique em **Cancelar** para fechar a caixa de diálogo sem remover qualquer serviço, ou clique em **Proceder** para remover o serviço selecionado ou serviços.

CAPÍTULO 4. GERENCIANDO O COMPLEMENTO DE ALTA DISPONIBILIDADE RED HAT COM O CONGA

Este capítulo descreve as várias tarefas administrativas para gerenciar o Complemento de Alta Disponibilidade da Red Hat e consiste das seguintes seções:

4.1. ADICIONAR UM CLUSTER EXISTENTE À INTERFACE DO LUCI

Se você criou anteriormente um cluster Complemento de Alta Disponibilidade você pode facilmente adicionar o cluster à interface do **luci** para que você possa gerenciar o cluster com o **Conga**.

Para adicionar um cluster existente à interface do **luci**, siga os seguintes passos:

1. Clique em **Gerenciar Clusters** (Manager Clusters) a partir do menu do lado esquerdo da página **Homebase** do **luci**. A tela **Clusters** aparecerá.
2. Clique em **Adicionar** (Add). A tela **Adicionar Cluster Existente** (Add Existing Cluster) aparecerá.
3. Digite o hostname do nó e a senha **ricci** para quaisquer dos nós no cluster existente. Já que cada nó no cluster contém toda a informação de configuração para o cluster, isto deve fornecer informações suficientes para adicionar o cluster à interface **luci**.
4. Clique em **Conectar** (Connect). A tela **Adicionar Cluster Existente** (Add Existing Cluster) então mostra o nome do cluster e os nós remanescentes no cluster.
5. Digite as senhas **ricci** individuais para cada nó no cluster ou digite uma senha e selecione **Use a mesma senha para todos os nós** (Use same password for all nodes).
6. Clique em **Adicionar Cluster** (Add Cluster). O cluster anteriormente configurado agora mostra a tela **Gerenciar Clusters** (Manage Clusters).

4.2. REMOVENDO UM CLUSTER DA INTERFACE DO LUCI

Você pode remover um cluster do GUI de gerenciamento do **luci** sem afetar serviços de cluster ou registro de cluster. Se você remover um cluster, você pode adicionar mais tarde o cluster novamente, ou você pode adicioná-lo a outra instância do **luci** como descrito em Seção 4.1, “Adicionar um Cluster Existente à interface do luci”.

Para remover um cluster do GUI de gerenciamento **luci**sem interferir nos serviços do cluster ou registro do cluster, siga os seguintes passos:

1. Clique em **Gerenciar Clusters** (Manager Clusters) a partir do menu do lado esquerdo da página **Homebase** do **luci**. A tela **Clusters** aparecerá.

2. Selecione o cluster ou clusters se você quiser remover.

3. Clique em **Delete**.

Para instruções sobre como remover um cluster completamente, interromper todos os serviços de cluster removendo as informações de configuração do cluster dos próprios nós, consulte a Seção 4.4, “Iniciando, Parando, Reinicializando e Deletando Clusters”.

4.3. GERENCIANDO NÓS NO CLUSTER

Esta seção documenta como realizar as seguintes funções de gerenciamento de nó através do componente do servidor **luci** do **Conga**:

- Seção 4.3.1, “Reinicializando um Nó no Cluster”
- Seção 4.3.2, “Faz um nó sair ou se juntar a um Cluster”
- Seção 4.3.3, “Adicionar um Membro a um Cluster em Execução”
- Seção 4.3.4, “Excluindo um Membro de um Cluster”

4.3.1. Reinicializando um Nó no Cluster

Para reinicializar um nó em um cluster, realize os seguintes passos:

1. Da página especifica do cluster, clique em **Nós** no topo da exibição do cluster. Isto mostra os nós que constituem o cluster. Ela também é a página padrão que aparece quando você clica no nome do cluster abaixo do **Gerenciar Clusters** (Manage Clusters) do menu do lado esquerdo da página **Homebase** do **luci**.

2. Selecione o nó para reinicializar clicando na caixa de marcação deste nó.

3. Selecione a função **Reinicializar** (Reboot) do menu no topo da página. Isto faz que o nó selecionado reinicializa e uma mensagem aparece no topo da página indicando que o nó está sendo reinicializado.

4. Atualize a página para ver o estado atualizado do nó.

Também é possível reinicializar mais do que um nó por vez selecionando todos os nós que você deseja reinicializar antes de clicar em **Reinicializar**.

4.3.2. Faz um nó sair ou se juntar a um Cluster

Você pode usar o componente do servidor **luci** do **Conga** para fazer um nó sair de um cluster ativo parando todos os serviços de cluster naquele nó. Você pode também usar o componente de servidor **luci** do **Conga** para fazer um nó que saiu do cluster a se rejuntar ao Cluster.

Fazer um nó sair do cluster não remove a informação de configuração do cluster daquele nó, e o nó ainda aparece na exibição de nó do cluster com o estado **`Não é um membro do cluster`**. Para informações sobre excluir totalmente um nó da configuração do cluster, veja Seção 4.3.4, “Excluindo um Membro de um Cluster”.

Para fazer um nó sair do cluster, realize os seguintes passos. Isto desliga o software de cluster do nó. Fazer um nó sair de um cluster previne o nó de se ligar automaticamente a um cluster quando ele é reinicializado.

1. Da página especifica do cluster, clique em **Nós** no topo da exibição do cluster. Isto mostra os nós que constituem o cluster. Ela também é a página padrão que aparece quando você clica no nome do cluster abaixo do **Gerenciar Clusters** (Manage Clusters) do menu do lado esquerdo da página **Homebase** do **luci**.

2. Selecione o nó que você quer que saia do cluste marcando a opção para esse nó.

3. Selecione a função **Sair do Cluster** (Leave Cluster) no menu no topo da página. Isto faz uma mensagem aparecer no topo da página indicando que o nó está sendo parado.

4. Atualize a página para ver o estado atualizado do nó.

Também é possível fazer que mais de um nó por vez saia do cluster selecionando todos os nós que devem sair do cluster antes de clicar em **Sair do Cluster** (Leave Cluster).

Para fazer um nó se rejuntar a um cluster, selecione os nós que você quer se rejuntem ao cluster marcando a caixa para estes nós e selecionando **Juntar ao Cluster** (Join Cluster). Isto faz que os nós selecionados se liguem ao cluster e permite aos nós selecionados se unirem ao cluster quando forem reinicializados.

4.3.3. Adicionar um Membro a um Cluster em Execução

Para adicionar um membro a um cluster, siga os passos desta seção.

1. Da página especifica do cluster, clique em **Nós** (Nodes) no topo da exibição do cluster. Isto mostra os nós que constituem o cluster. Isto também é a página padrão que aparece quando você clica no nome do cluster abaixo do **Gerenciar Clusters** do menu do lado esquerdo da página **Homebase** do **luci**.

2. Clique em **Adicionar** (Add) para exibir a caixa de diálogo **Adicionar Nós ao Cluster** (Add Nodes to Cluster).

3. Digite o nome do nó na caixa de texto **Hostname do Nó** (Node Hostname); digite a senha **ricci** na caixa de texto **Senha** (Password). Se você estiver usando uma porta diferente para o agente **ricci** que não seja 11111, você pode mudar esse parâmetro.

4. Marque a opção **Habilitar Suporte a Armazenamento Compartilhado** (Enable Shared Storage Support) se a armazenagem em cluster é requerida para baixar os pacotes que suportam armazenagem clusterizada e habilitam LVM clusterizado; você deveria selecionar isto somente quando você tem acesso ao Complemento de Armazenamento Resiliente ou Complemento de Sistema de Arquivo Escalável.

5. Se você quiser adicionar mais nós, clique em **`Adicionar Outro Nó`** e digite o nome e senha do nó para cada nó adicional.

6. Clique em **`Adicionar Nós`** (Add Nodes) para fazer as seguintes ações:

    1. Se você selecionou **Baixar Pacotes** (Download Packages), os pacotes de software do cluster são baixados nos nós.

    2. O software de Cluster está instalado nos nós (ou é checado que os pacotes de software apropriados estão instalados).

3. O arquivo de configuração de cluster é atualizado e propagado para cada nó no cluster — incluindo o nó adicionado.

4. O nó adicionado se junta ao cluster.

A página **Nós** aparece com a mensagem indicando que o nó está sendo adicionado ao cluster. Atualize a página para ver o novo estado.

7. Quando o processo de adicionar um nó estiver completo, clique no nome do nó para o nó recém adicionado para configurar um fence para este nó, conforme descrito na Seção 3.6, "Configurando Dispositivos Fence".

4.3.4. Excluindo um Membro de um Cluster

Para remover um membro de um cluster existente que está atualmente em operação, siga os passos desta seção. Note que os nós devem estar parados antes de serem excluídos ao menos que você remova todos os nós no cluster de uma vez.

1. Da página especifica do cluster, clique em **Nós** (Nodes) no topo da exibição do cluster. Isto mostra os nós que constituem o cluster. Isto também é a página padrão que aparece quando você clica no nome do cluster abaixo do **Gerenciar Clusters** do menu do lado esquerdo da página **Homebase** do **luci**.

NOTA

Para permitir que serviços em execução em um nó façam fail over quando o nó é excluído, pule o próximo passo.

2. Desabilite ou realoque cada serviço que esteja em execução no nó a ser excluído. Para informações sobre desabilitar e realocar serviços, veja a Seção 4.5, "Gerenciando Serviços de Alta Disponibilidade".

3. Selecione o nó ou nós para deletar.

4. Clique em **Delete**. A página dos **Nós** indica que o nó está sendo removido. Atualize a página para ver o estado atual.

IMPORTANTE

Remover um nó de cluster de um cluster é uma operação destrutiva que não pode ser desfeita.

4.4. INICIANDO, PARANDO, REINICIALIZANDO E DELETANDO CLUSTERS

Você pode iniciar, parar e reiniciar um cluster realizando estas ações em nós individuais no cluster. Da página especifica do cluster, clique em **Nós** (Nodes) no topo da exibição do cluster. Isto exibe os nós que constituem o cluster.

Operações de início ou reinício para nós de cluster ou para um cluster todo, permite que você crie serviços de cluster curtos se um serviço de cluster precisar ser movido para outro membro de cluster pois ele está rodando em um nó que está sendo interrompido ou reiniciado.

Para parar um cluster, realize os seguintes passos. Isto fecha o software de cluster nos nós, mas não remove a informação de configuração do cluster dos nós e os nós ainda aparecem na exibição do nó no cluster com um estado **`Não é um membro do Cluster`** (Not a cluster member).

1. Selecione todos os nós no cluster marcando a caixa próxima a cada nó.
2. Selecione a função **Sair do Cluster** (Leave Cluster) no menu no topo da página. Isto faz uma mensagem aparecer no topo da página indicando que o cada nó está sendo parado.
3. Atualize a página para ver o estado atualizado dos nós.

Para iniciar um cluster, realize os seguintes passos:

1. Selecione todos os nós no cluster marcando a caixa próxima a cada nó.
2. Selecione a função **Juntar ao Cluster** (Join Cluster) no menu no topo da página.
3. Atualize a página para ver o estado atualizado dos nós.

Para reiniciar um cluster em execução, primeiro pare todos os nós no cluster, então inicie todos os nós no cluster, conforme descrito acima.

Para remover um cluster por completo, realize os seguintes passos. Isto fecha o software de cluster nos nós, mas não remove a informação de configuração do cluster dos nós e os nós ainda aparecem na exibição do nó no cluster. Caso você adicione mais tarde, um cluster existente usando qualquer um dos nós que você removeu, o **luci** irá indicar que o nós não é um membro de nenhum dos clusters.

IMPORTANTE

Excluir um cluster é uma operação destrutiva que não pode ser desfeita. Para restaurar um cluster depois que você o excluiu, é requerido que você recrie e redefina o cluster a partir do zero.

1. Selecione todos os nós no cluster marcando a caixa próxima a cada nó.
2. Selecione a função **Delete** do menu no topo da página.

Se você deseja remover um cluster de uma interface **luci** sem parar qualquer um dos serviços cluster ou mudar o registro do cluster, você pode utilizar a opção **Remove** na página **Manage Clusters**, como descrito em Seção 4.2, “Removendo um Cluster da interface do luci” .

4.5. GERENCIANDO SERVIÇOS DE ALTA DISPONIBILIDADE

Além de adicionar e modificar um serviço, conforme descrito na Seção 3.10, “Adicionar um Serviço de Cluster ao Cluster”, você pode realizar as seguintes funções de gerenciamento para serviços de alta disponibilidade através do componente de servidor **luci** do **Conga**:

- Iniciar um serviço
- Reiniciar um serviço
- Desabilitar um serviço
- Deletar um serviço
- Realocar um serviço

Da página específica do cluster, você pode gerenciar serviços para aquele cluster clicando em **Grupos de Serviço** (Service Groups) no topo da exibição do cluster. Isto exibe os serviços que foram modificados para esse cluster.

- **Iniciando um serviço** — Para iniciar quaisquer serviços que não estão atualmente rodando, selecione quaisquer serviços que queira iniciar marcando a caixa para esse serviço e clicando em **Iniciar** (Start).

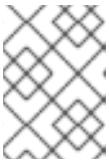
- **Reiniciar um serviço** — Para reiniciar quaisquer serviços que estão atualmente em execução, selecione quaisquer serviços que você queira reiniciar marcando a caixa de seleção do serviço e clicando em **Reiniciar** (Restart).

- **Desabilitar um serviço** — Para desabilitar qualquer serviço que está atualmente em execução, selecione quaisquer serviços que você queira desabilitar marcando a caixa de seleção para esse serviço e clique em **Desabilitar** (Disable).

- **Deletando um serviço** — Para excluir quaisquer serviços que não estão atualmente em execução, selecione quaisquer serviços que você queira desabilitar clicando na caixa de seleção para esse serviço e clique em **Delete**.

- **Realocar um serviço** — Para realocar um serviço em execução, clique no nome do serviço na exibição dos serviços. Isto faz com que a página de configuração dos serviços seja mostrada, com uma exibição indicando em qual nó o serviço está atualmente em execução.

  Da caixa suspensa **`Iniciar no Nó...`** (Start on node...), selecione o nó no qual você quer realocar o serviço, e clique no ícone **Iniciar** (Start). Um mensagem aparece no topo da tela indicando que o serviço está sendo iniciado. Você pode precisar atualizar a tela para ver a nova indicação de que o serviço está rodando no nó que você selecionou.



NOTA

Caso o serviço em execução que você escolheu for um serviço **`vm`**, a caixa de menu suspenso mostrará uma opção **`migrar`** ao invés de uma opção **`relocar`**

NOTA

Você pode iniciar, reinicializar, desabilitar ou deletar um serviço individual clicando no nome do serviço na página **Serviços** (Services). Isto exibe a página de configuração do serviço. No canto superior direito da página de configuração do serviço estão os mesmo ícones para **Iniciar** (Start), **Reiniciar** (Restart), **Desabilitar** (Disable) e **Deletar** (Delete).

4.6. FAZENDO UM BACKUP E RECUPERANDO A CONFIGURAÇÃO DO LUCI

Desde o lançamento do Red Hat Enterprise Linux 6.2, você pode usar o seguinte procedimento para fazer um backup do banco de dados do **luci**, o qual é armazenado no arquivo **`/var/lib/luci/data/luci.db`**. Esta não é exatamente uma configuração do cluster, a qual é armazenada no arquivo **`cluster.conf`**. Ao invés disso, ele contém a lista de usuários e clusters e propriedades relacionadas que o **luci** mantém. Por padrão, o backup que este procedimento cria será gravado no mesmo diretório que o arquivo **`luci.db`**.

1. Execute **`service luci stop`**.

2. Execute **`service luci backup-db`**.

Como forma alternativa, você pode especificar um nome de arquivo como um parâmetro para o comando **backup-db** o qual irá gravar o banco de dados do **luci** naquele arquivo. Por exemplo, para gravar o banco de dados do **luci** no arquivo **/root/luci.db.backup**, você pode executar o comando **service luci backup-db /root/luci.db.backup**. Note, no entanto, que os arquivos de backup que são gravados em locais ao invés de **/var/lib/luci/data/** (Para backups cujos nomes de arquivos você especifica ao usar **service luci backup-db**) não aparecerão no resultado do comando **list-backups**

3. Execute **service luci start**.

Use o seguinte procedimento para restaurar um banco de dados do **luci**.

1. Execute **service luci stop**.
2. Execute **service luci list-backups** e observe o nome do arquivo a ser recuperado.
3. Execute **service luci restore-db /var/lib/luci/data/*lucibackupfile*** onde *lucibackupfile* é o backup do arquivo a ser recuperado.

   Por exemplo, o comando a seguir recupera as informaçºoes de configuração do **luci** armazenadas no arquivo de backup **luci-backup20110923062526.db**:

   ```
   service luci restore-db /var/lib/luci/data/luci-
   backup20110923062526.db
   ```

4. Execute **service luci start**.

Se você precisar recuperar um banco de dados do **luci** mas você perdeu o arquivo **host.pem** da máquina onde você criou o backup porcausa de uma reinstalação completa, por exemplo, você irá precisar adicionar seus clusters de volta ao **luci** manualmente para reautenticar os nós de cluster.

Use o procedimento a seguir para recuperar um banco de dados do **luci** para uma máquina diferente desta na qual o backup foi criado. Observe que além de recuperar o próprio banco de dados, você também precisará copiar um arquivo de certificado SSL para garantir que o **luci** foi autenticado nos nós do **ricci**. Neste exemplo, o backup é criado na máquina **luci1** e o backup é recuperado na máquina **luci2**.

1. Execute a seguinte sequência de comandos para criar um backup do **luci** no **luci1** e copiar o arquivo de certificado SSL e o backup do **luci** para **luci2**.

   ```
   [root@luci1 ~]# service luci stop
   [root@luci1 ~]# service luci backup-db
   [root@luci1 ~]# service luci list-backups
   /var/lib/luci/data/luci-backup20120504134051.db
   [root@luci1 ~]# scp /var/lib/luci/certs/host.pem
   /var/lib/luci/data/luci-backup20120504134051.db root@luci2:
   ```

2. Na máquina **luci2** certifique-se de que o **luci** foi instalado e não está em execução. Instale o pacote, caso ainda não esteja instalado.
3. Execute a seguinte sequência de comandos para certificar-se de que as autenticações estão no local e para recuperar o banco de dados do **luci** do **luci1** para **luci2**.

   ```
   [root@luci2 ~]# cp host.pem /var/lib/luci/certs/
   ```

```
[root@luci2 ~]# chown luci: /var/lib/luci/certs/host.pem
[root@luci2 ~]# /etc/init.d/luci restore-db ~/luci-
backup20120504134051.db
[root@luci2 ~]# shred -u ~/host.pem ~/luci-backup20120504134051.db
[root@luci2 ~]# service luci start
```

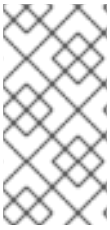
CAPÍTULO 5. CONFIGURANDO O COMPLEMENTO DE ALTA DISPONIBILIDADE DA RED HAT COM O COMANDO CCS

A partir do lançamento do Red Hat Enterprise Linux 6.1 e posteriores, o Complemento de Alta Disponibilidade da Red Hat fornece suporte para o comando de configuração de cluster **ccs**. O comando **ccs** permite um administrador criar, modificar e vizualizar o arquivo de configuração **cluster.conf**. Você pode usar o comando **ccs** para configurar um arquivo de configuração de cluster em um sistema de arquivo local ou em um nódo remoto. Usando o comando **ccs**, um administrador pode também iniciar e parar os serviços de cluster em um ou todos os nódos em um cluster configurado.

Este capítulo descreve como configurar o arquivo de configuração de cluster do Complemento de Alta Disponibilidade da Red Hat usando o comando **ccs**. Para informações sobre usar o comando **css** para gerenciar um cluster em execução, veja o Capítulo 6, *Gerenciando o Complemento de Alta Disponibilidade da Red Hat com o **ccs***.

Este capítulo consiste das seguintes seções:

- Seção 5.1, “Visão Geral Operacional”
- Seção 5.2, “Tarefas de Configuração”
- Seção 5.3, “Iniciando o **ricci**”
- Seção 5.4, “Criando um Cluster”
- Seção 5.5, “Configurando Dispositivos Fence”
- Seção 5.7, “Configurando o Fence para Membros do Cluster”
- Seção 5.8, “Configurando um Domínio de Failover”
- Seção 5.9, “Configurando Recursos de Cluster Globais”
- Seção 5.10, “Adicionando um Serviço de Cluster ao Cluster”
- Seção 5.13, “Configurando um Disco de Quorum”
- Seção 5.14, “Configurações de Cluster Diversas”
- Seção 5.14, “Configurações de Cluster Diversas”
- Seção 5.15, “Propagar o Arquivo de Configuração aos Nós do Cluster”



NOTA

Certifique-se que sua implementação do Complemento de Alta Disponibilidade atenda suas necessidades e possa ser suportada. Consulte um representante autorizado Red Hat para verificar suas configurações antes da implementação. Além disso, disponibilize tempo para um período de testes de falha.

NOTA

Este capítulo referencia elementos e atributos **cluster.conf** comumente usados. Para uma lista compreensiva e a descrição dos elementos e atributos do **cluster.conf**, consulte o esquema de cluster em **/usr/share/cluster/cluster.rng** e o esquema anotado em **/usr/share/doc/cman-X.Y.ZZ/cluster_conf.html** (por exemplo **/usr/share/doc/cman-3.0.12/cluster_conf.html**).

5.1. VISÃO GERAL OPERACIONAL

Esta seção descreve os seguintes aspectos operacionais gerais do uso do comando **ccs** para configurar um cluster:

- Seção 5.1.1, “Criando um arquivo de Configuração de Cluster em um Sistema Local”
- Seção 5.1.2, “Vizualizar a Configuração de Cluster Atual”
- Seção 5.1.3, “Especificando Senhas ricci com o comando ccs”
- Seção 5.1.4, “Modificando Componentes de Configuração de Cluster”

5.1.1. Criando um arquivo de Configuração de Cluster em um Sistema Local

Usar o comando **ccs**, você pode criar um arquivo de configuração do cluster em um nó de cluster, ou você pode criar um arquivo de configuração do cluster em um sistema de arquivo local e depois enviar aquele arquivo para uma máquina em um cluster. Isto permite que você trabalhe em um arquivo de uma máquina local, onde você pode mantê-lo sob o controle da versão ou então marcar o arquivo de acordo com suas necessidades. Utilizando o comando **ccs** não requer o privilégio root.

Quando você criar e editar um arquivo de configuração de cluster em um nódo em cluster com o comando **ccs**, você usa a opção **-h** para especificar o nome do host. Isto cria e edita o arquivo **cluster.conf** no host:

```
ccs -h host [options]
```

Para criar e editar um arquivo de configuração de cluster em um sistema local, use a opção **-f** do comando **ccs** para especificar o nome do arquivo de configuração quando você realizar uma operação no cluster. Você pode nomear este arquivo do jeito que quiser.

```
ccs -f file [options]
```

Depois de você ter criado o arquivo localmente você pode envia-lo a um nódo do cluster usando a opção **--setconf** do comando **ccs**. Em uma máquina host em um cluster, o arquivo que você envia será nomeado **cluster.conf** e será colocado no diretório **/etc/cluster**.

```
ccs -h host -f file --setconf
```

Para informações sobre o uso da opção **--setconf** do comando **ccs**, veja Seção 5.15, “Propagar o Arquivo de Configuração aos Nós do Cluster”.

5.1.2. Vizualizar a Configuração de Cluster Atual

Se em qualquer momento a criação de um arquivo de configuração de cluster, você desejar imprimir o arquivo atual, use o segunte comando, especificando o nó em um cluster como host:

```
ccs -h host --getconf
```

Se você estiver criando seu arquivo de configuração de cluster em um sistema local, você pode especificar a opção **-f** ao invés da opção **-h**, como descrito na Seção 5.1.1, "Criando um arquivo de Configuração de Cluster em um Sistema Local".

5.1.3. Especificando Senhas ricci com o comando ccs

Executar o comando **ccs** que distribui cópias do arquivo **cluster.conf** aos nódos de um cluster requer que o **ricci** seja instalado e esteja rodando nos nódos do cluster, como descrito na Seção 2.13, "Considerações para o **ricci**". Usar o **ricci** requer uma senha na primeira vez que você interagir com o **ricci** de qualquer máquina específica.

Caso não tenha inserido uma senha para uma instância do **ricci** em uma máquina específica a partir da máquina que você está utilizando, lhe será solicitado uma senha quando o comando **ccs** precisar. Como forma alternativa, você poderá utilizar a opção **-p** para especificar uma senha **ricci** na linha de comando.

```
ccs -h host -p password --sync --activate
```

Quando você propagar o arquivo **cluster.conf** para todos os nódos no cluster com a opção **--sync** do comando **ccs** e você especificar uma senha **ricci** para o comando, o **ricci** usará essa senha para cada nódo no cluster. Se você precisar definir diferentes senhas para o **ricci** em nódos individuais, você pode usar o comando **--setconf** com o **-p** para distribuir o arquivo de configuração a um nódo por vez.

5.1.4. Modificando Componentes de Configuração de Cluster

Você usa o comando **ccs** para configurar componentes do cluster e seus atributos no arquivo de configuração do cluster. Depois de ter adicionado um componente de cluster ao arquivo, para modificar os atributos deste componente você deve remover o componente que definiu e adicionar o componente novamente, com os atributos modificados. Informações sobre como fazer isso em cada componente são fornecidas nas seções individuais deste capítulo.

Os atributos do componente do cluster **cman** fornece uma exceção à este procedimento para modificar componentes de cluster. Para modificar estes atributos, você pode executar a opção **--setcman** do comando **ccs**, especificando os novos atributos. Note que ao especificar esta opção, você irá redefinir todos os valores que você não especifica explicitamente para seus valores padrão, como descrito em Seção 5.1.5, "Comandos que Sobrescrevem Configurações Anteriores".

5.1.5. Comandos que Sobrescrevem Configurações Anteriores

Existem diversas opções do comando **ccs** que implementa semânticas sobrescritas ao definir as propriedades. Isto significa que você pode emitir um comando **ccs** com uma destas opções sem especificar qualquer configuração, e ele irá redefinir todas as configurações para seus valores padrão. Estas opções são estas a seguir:

- **--settotem**
- **--setdlm**

- **--setrm**
- **--setcman**
- **--setmulticast**
- **--setaltmulticast**
- **--setfencedaemon**
- **--setlogging**
- **--setquorumd**

Por exemplo, para redefinir todas as propriedades do daemon do fence, você pode executar este comando:

```
# ccs -h hostname --setfencedaemon
```

Note, no entanto, que se você utilizar um destes comandos para redefinir um propriedade, as outras propriedades do comando serão redefinidas para seus valores padrão. Por exemplo, você pode usar o seguinte comando para definir a propriedade do **post_fail_delay** para 5:

```
# ccs -h hostname --setfencedaemon post_fail_delay=5
```

Se após executar este comando, você executar o seguinte comando para redefinir a propriedade de **post_join_delay** para 10, a propriedade do **post_fail_delay** será recuperada para seu valor padrão:

```
# ccs -h hostname --setfencedaemon post_join_delay=10
```

Para redefinir ambos os **post_fail_delay** e as propriedades **post_join_delay**, você os indica no mesmo comando, como no exemplo a seguir:

```
# ccs -h hostname --setfencedaemon post_fail_delay=5 post_join_delay=10
```

Para mais informações sobre como configurar dispositivos de fence, consulte o Seção 5.5, “Configurando Dispositivos Fence”.

5.1.6. Validação de Configuração

Quando você utilizar o comando **ccs** para criar e editar o arquivo de configuração do cluster, a configuração é automaticamente validada de acordo com o esquema de cluster. Desde o Red Hat Enterprise Linux 6.3, o comando **ccs** valida a configuração de acordo com o esquema do cluster em **/usr/share/cluster/cluster.rng** no nó que você espeficar com a opção **-h**. Anteriormente, o comando **ccs** sempre usava o esquema do cluster que era empacotado com o próprio comando **ccs**, **/usr/share/ccs/cluster.rng** no sistema local. Quando você utilizar a opçaõ **-f** para especificar o sistema local, o comando **ccs** ainda utiliza o esquema cluster **/usr/share/ccs/cluster.rng** que foi empacotada com o comando **ccs**.

5.2. TAREFAS DE CONFIGURAÇÃO

Configurando o software Complemento Alta Disponibilidade da Red Hat com o **ccs** consiste dos seguintes passos:

1. Certifique-se que o **ricci** está rodando em todos os nódos no cluster. Consulte a Seção 5.3, "Iniciando o **ricci**".
2. Criando um cluster. Consulte a Seção 5.4, "Criando um Cluster"
3. Configurando dispositivos fence. Consulte a Seção 5.5, "Configurando Dispositivos Fence".
4. Configurando o fence para membros do cluster. Consulte a Seção 5.7, "Configurando o Fence para Membros do Cluster".
5. Criando domínios failover. Consulte a Seção 5.8, "Configurando um Domínio de Failover" .
6. Criando recursos. Consulte a Seção 5.9, "Configurando Recursos de Cluster Globais" .
7. Criando dispositivos de cluster. Consulte a Seção 5.10, "Adicionando um Serviço de Cluster ao Cluster".
8. Configurando um disco quorum, se necessário. Consulte a Seção 5.13, "Configurando um Disco de Quorum".
9. Configurando propriedades de cluster globais. Consulte a Seção 5.14, "Configurações de Cluster Diversas".
10. Propagando o arquivo de configuração de cluster em todos os nódos do cluster. Consulte a Seção 5.15, "Propagar o Arquivo de Configuração aos Nós do Cluster" .

5.3. INICIANDO O RICCI

Para criar e distribuir arquivos de configuração de cluster em nódos do cluster, o serviço **ricci** deve estar rodando em cada nódo. Antes de iniciar o **ricci**, você deve ter certeza que tem seu sistema configurado como a seguir:

1. As portas IP em seu nódos no cluster devem estar habilitadas para o **ricci**. Para informações sobre habilitar portas IP nos nódos do cluster, veja a Seção 2.3.1, "Habilitando Portas IP em nós de Cluster".
2. O serviço **ricci** está instalado em todos os nódos no cluster e atribuído uma senha **ricci**, como descrito na Seção 2.13, "Considerações para o `ricci`".

Depois do **ricci** tiver sido instalado e configurado em cada nódo, inicie o serviço **ricci** em cada nódo:

```
# service ricci start
Starting ricci:                                            [  OK  ]
```

5.4. CRIANDO UM CLUSTER

Esta seção descreve como criar, modificar e deletar uma configuração de esqueleto de cluster com o comando `ricci` sem usar fence, domínios failover e serviços de Alta Disponibilidade (HA - High Availability). As seções subsequentes descrevem como configurar aquelas partes da configuração.

Para criar um arquivo de configuração de esqueleto de cluster, primeiro crie e nomeie o cluster e então adicione os nódos ao cluster, como no procedimento seguinte:

1. Crie um arquivo de configuração de cluster em um dos nódos no cluster executando o comando **ricci** usando o parâmetro **-h** para especificar o nódo no qual criar o arquivo e a opção **createcluster** para especificar um nome para o cluster:

```
ccs -h host --createcluster clustername
```

Por exemplo, o comando seguinte criar um arquivo de configuração no **node-01.example.com** chamado **mycluster**:

```
ccs -h node-01.example.com --createcluster mycluster
```

O nome do cluster não pode ultrapassar 15 caractéres.

Se um arquivo **cluster.conf** já existe no host que você especificar, executar este comando substituirá o arquivo existente.

Se você deseja criar um arquivo de configuração de cluster em seu sistema local, você pode especificar a opção **-f** ao invés da opção **-h**. Para mais informações sobre como criar o arquivo localmente, consulte Seção 5.1.1, “Criando um arquivo de Configuração de Cluster em um Sistema Local”.

2. Para configurar os nódos que o cluster possui, execute o seguinte comando para cada nódo no cluster:

```
ccs -h host --addnode node
```

Por exemplo, os seguintes três comandos adicionam os nódos **node-01.example.com**, **node-02.example.com**, e **node-03.example.com** ao arquivo de configuração no **node-01.example.com**:

```
ccs -h node-01.example.com --addnode node-01.example.com
ccs -h node-01.example.com --addnode node-02.example.com
ccs -h node-01.example.com --addnode node-03.example.com
```

Para vizualizar uma lista de nódos que foram configurados para um cluster, execute o seguinte comando:

```
ccs -h host --lsnodes
```

Exemplo 5.1, “O arquivo cluster.conf depois de adicionar três nódos” exibe um arquivo de configuração **cluster.conf** depois de você ter criado o cluster **mycluster** que contém os nodos **node-01.example.com** **node-02.example.com** e **node-03.example.com**.

Exemplo 5.1. O arquivo cluster.conf depois de adicionar três nódos

```
<cluster name="mycluster" config_version="2">
   <clusternodes>
     <clusternode name="node-01.example.com" nodeid="1">
         <fence>
         </fence>
     </clusternode>
     <clusternode name="node-02.example.com" nodeid="2">
```

```
            <fence>
            </fence>
        </clusternode>
        <clusternode name="node-03.example.com" nodeid="3">
            <fence>
            </fence>
        </clusternode>
    </clusternodes>
    <fencedevices>
    </fencedevices>
    <rm>
    </rm>
</cluster>
```

Quando você adicionar um nódo ao um cluster, você pode especificar o número de votos que o nódo contribui para determinar se existe um quorum. Para definir o número de votos para um nódo do cluster, use o seguinte comando:

```
ccs -h host --addnode host --votes votes
```

Quando você adicionar um nódo, o **ccs** atribui ao nódo um número inteiro único que é usado como o identificador do nódo. Se você quiser especificar o identificador de nódo manualmente quando criar um nódo, use o seguinte comando:

```
ccs -h host --addnode host --nodeid nodeid
```

Para remover um nódo de um cluster, execute o seguinte comando:

```
ccs -h host --rmnode node
```

Quando você terminar de configurar todos os componentes de seu cluster, você precisará sincronizar o arquivo de configuração do cluster para todos os nodos, como descrito na Seção 5.15, “Propagar o Arquivo de Configuração aos Nós do Cluster”.

5.5. CONFIGURANDO DISPOSITIVOS FENCE

Configurar dispositivos fence consiste na criação, atualização e exclusão de dispositivos fence para o cluster. Você deve criar e nomear os dispositivos fence em um cluster antes que você possa configurar o fence para os nós no cluster. Para mais informações sobre como configurar o fencing para os nós individuais em um cluster, consulte Seção 5.7, “Configurando o Fence para Membros do Cluster”.

Antes de configurar seus dispositivos fence, você pode querer modificar algumas das propriedades daemon do fence para seus sistemas a partir dos valores padrões. Os valores que você configurar para o daemon fence são valores gerais para o cluster. As propriedades gerais de fence para o cluster que você possa querer modificar estão resumidas como a seguir

- O atributo **post_fail_delay** é o número de segundos que o daemon fence (**fenced**) espera antes de executar um fence em um nódo (um membro do domínio fence) depois que o nódo tiver falhado. O valor padrão **post_fail_delay** é **0**. O seu valor pode ser variado para adequar o desempenho de rede e cluster.

- O atributo **post-join_delay** é o número de segundos que o fence daemon (**fenced**) espera

antes de fazer um fence em um nó depois que o nó se unir ao domínio do fence. O valor padrão do **post-join_delay** é **6**. Uma configuração típica para **post-join_delay** é entre 20 e 30 segundos, mas pode ser variado para se adequar ao desempenho da rede e do cluster.

Você pode redefinir os valores dos atributos **post_fail_delay** e **post_join_delay** com a opção **--setfencedaemon** do comando **ccs**. Note, no entanto que ao executar o comando **ccs --setfencedaemon**, ele sobrescreve todas as propriedades de daemon de fence existentes que foram explicitamente definidas e recupera-os para seus valores padrão.

Por exemplo, para configurar um valor para o atributo **post_fail_delay**, execute o seguinte comando. Este comando irá sobrescrever os valores de todas as propriedades do daemon de fence existentes que você já definiu com este comando e recuperá-los para seus valores padrão.

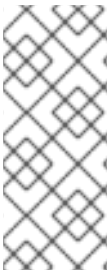
```
ccs -h host --setfencedaemon post_fail_delay=value
```

Para configurar um valor para o atributo **post_join_delay**, execute o seguinte comando. Este comando irá sobrescrever os valores de todas as propriedades de daemon de fence existentes que você já definiu com este comando e recuperá-los para seus valores padrão.

```
ccs -h host --setfencedaemon post_join_delay=value
```

Para configurar um valor para ambos atributos **post_join_delay** e **post_fail_delay** execute o seguinte comando:

```
ccs -h host --setfencedaemon post_fail_delay=value post_join_delay=value
```



NOTA

Para mais informações sobre os atributos **post_join_delay** e **post_fail_delay** tanto como propriedades de fence daemon adicionais que você pode modificar, consulte a página man fenced(8) e consulte o esquema de cluster em **/usr/share/cluster/cluster.rng**, e o esquema anotado em **/usr/share/doc/cman-X.Y.ZZ/cluster_conf.html**.

Para configurar um dispositivo fence para um cluster, execute o seguinte comando:

```
ccs -h host --addfencedev devicename [fencedeviceoptions]
```

Por exemplo, para configurar um dispositivo fence apc no arquivo de configuração no nódo do cluster **node1** chamado **myfence** com o endereço de IP de **apc_ip_example**, um login de **login_example**, e uma senha de **password_example**, execute o seguinte comando:

```
ccs -h node1 --addfencedev myfence agent=fence_apc ipaddr=apc_ip_example
login=login_example passwd=password_example
```

O exemplo seguinte exibe a seção **fence devices** no arquivo de configuração **cluster.conf** depois de você ter adicionado este dispositivo fence APC:

```
<fencedevices>
      <fencedevice agent="fence_apc" ipaddr="apc_ip_example"
```

```
login="login_example" name="myfence" passwd="password_example"/>
</fencedevices>
```

Ao configurar dispositivos fence para um cluster, você pode achar útil ver uma lista dos dispositivos disponíveis para seu cluster e as opções que estão disponíveis para cada dispositivo. Você também pode querer ver uma lista dos dispositivos de fence atualmente configuradas para seu cluster. Para informações sobre como utilizar o comando **ccs** para imprimir uma lista de dispositivos de fence disponíveis e opções ou para imprimir uma lista de dispositivos de fence atualmente configuradas para seu cluster, veja Seção 5.6, “Lista de Dispositivos de Fence e Opções de Dispositivos de Fence” .

Para remover um dispositivo fence da configuração de seu cluster, execute o seguinte comando:

```
ccs -h host --rmfencedev fence_device_name
```

Por exemplo, para remover um dispositivo fence que você nomeou como **myfence** do arquivo de configuração do cluster no nódo **node1**, execute o seguinte comando:

```
ccs -h node1 --rmfencedev myfence
```

Se você precisar modificar os atributos de um dispositivo fence que você já configurou, você deve primeiro remover este dispositivo fence e então adiciona-lo novamento com os atributos modificados.

Observe que quando você tiver terminado de configurar todos os componentes de seu cluster, você precisará sincronizar o arquivo de configuração do cluster em todos os nódos, como descrito na Seção 5.15, “Propagar o Arquivo de Configuração aos Nós do Cluster” .

5.6. LISTA DE DISPOSITIVOS DE FENCE E OPÇÕES DE DISPOSITIVOS DE FENCE

Você pode usar o **ccs** para imprimir uma lista de dispositivos de fence disponíveis e para imprimir uma lista de opções para cada tipo de fence disponível. Você também pode utilizar o comando **ccs** para imprimir uma lista de dispositivos de fence atualmente configuradas para seu cluster.

Para imprimir uma lista de dispositivos fence atualmente configurados para seu cluster, execute o seguinte comando:

```
ccs -h host --lsfenceopts
```

Por exemplo, o seguinte comando lista os disposivitos de fence disponíveis no nó de cluster **node1**, mostrando o resultado de exemplo.

```
[root@ask-03 ~]# ccs -h node1 --lsfenceopts
fence_rps10 - RPS10 Serial Switch
fence_vixel - No description available
fence_egenera - No description available
fence_xcat - No description available
fence_na - Node Assassin
fence_apc - Fence agent for APC over telnet/ssh
fence_apc_snmp - Fence agent for APC over SNMP
fence_bladecenter - Fence agent for IBM BladeCenter
fence_bladecenter_snmp - Fence agent for IBM BladeCenter over SNMP
fence_cisco_mds - Fence agent for Cisco MDS
fence_cisco_ucs - Fence agent for Cisco UCS
```

```
fence_drac5 - Fence agent for Dell DRAC CMC/5
fence_eps - Fence agent for ePowerSwitch
fence_ibmblade - Fence agent for IBM BladeCenter over SNMP
fence_ifmib - Fence agent for IF MIB
fence_ilo - Fence agent for HP iLO
fence_ilo_mp - Fence agent for HP iLO MP
fence_intelmodular - Fence agent for Intel Modular
fence_ipmilan - Fence agent for IPMI over LAN
fence_kdump - Fence agent for use with kdump
fence_rhevm - Fence agent for RHEV-M REST API
fence_rsa - Fence agent for IBM RSA
fence_sanbox2 - Fence agent for QLogic SANBox2 FC switches
fence_scsi - fence agent for SCSI-3 persistent reservations
fence_virsh - Fence agent for virsh
fence_virt - Fence agent for virtual machines
fence_vmware - Fence agent for VMware
fence_vmware_soap - Fence agent for VMware over SOAP API
fence_wti - Fence agent for WTI
fence_xvm - Fence agent for virtual machines
```

Para imprimir uma lista de opções que você pode especificar para um tipo de fence específico, execute o seguinte comando:

```
ccs -h host --lsfenceopts fence_type
```

Por exemplo, o seguinte comando lista as opções de fence parao agente de fence **`fence_wti`**.

```
[root@ask-03 ~]# ccs -h node1 --lsfenceopts fence_wti
fence_wti - Fence agent for WTI
  Required Options:
  Optional Options:
    option: No description available
    action: Fencing Action
    ipaddr: IP Address or Hostname
    login: Login Name
    passwd: Login password or passphrase
    passwd_script: Script to retrieve password
    cmd_prompt: Force command prompt
    secure: SSH connection
    identity_file: Identity file for ssh
    port: Physical plug number or name of virtual machine
    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
    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
```

Para exibir uma lista de dispositivos fence atualmente configurados para seu cluster, execute o seguinte comando:

```
ccs -h host --lsfencedev
```

5.7. CONFIGURANDO O FENCE PARA MEMBROS DO CLUSTER

Uma vez que você completou os passos iniciais para criar um cluster e criar dispositivos fence, você precisa configurar o fence para os nódos do cluster. Para configurar o fence para os nódos depois de criar um novo cluster e configurar os dispositivos fence para o cluster, siga os passos nesta seção. Note que você deve configurar o fence para cada nódo no cluster.

Esta seção documenta os seguintes procedimentos:

- Seção 5.7.1, “Configurando um Dispositivo Fence Baseado em Energia Única para um Nódo”
- Seção 5.7.2, “Configurando um Dispositivo Fence Baseado em Armazenamento para um nódo”
- Seção 5.7.3, “Configurando um dispositivo Fence de Backup”
- Seção 5.7.4, “Configurando um Nódo com energia Redundante”
- Seção 5.7.5, “Remover Métodos Fence e Instâncias Fence”

5.7.1. Configurando um Dispositivo Fence Baseado em Energia Única para um Nódo

Use o seguinte procedimento para configurar um nódo com um dispositivo fence de energia única que usa um dispositivo de fence chamado **apc**, que usa o agente fence **fence_apc**.

1. Adicionar um método fence para o nódo, fornecendo um nome para o método fence.

   ```
   ccs -h host --addmethod method node
   ```

   Por exemplo, para configurar um método fence chamado **APC** para o nódo **node-01.example.com** no arquivo de configuração no nódo do cluster **node-01.example.com**, execute o seguinte comando:

   ```
   ccs -h node01.example.com --addmethod APC node01.example.com
   ```

2. Adicionar uma instância de fence para o método. Você deve especificar o dispositivo fence a ser usado para o nódo, o nódo que esta instância se aplica, o nome do método e quaisquer opções para este método que são específicas para este nódo:

   ```
   ccs -h host --addfenceinst fencedevicename node method [options]
   ```

   Por exemplo, para configurar uma instância fence no arquivo de configuração no nódo de cluster **node-01.example.com** que usa o switch de energia APC porta 1 no dispositivo fence chamado **apc** para criar um fence no nódo do cluster **node-01.example.com** usando o método chamado **APC**, execute o seguinte comando:

   ```
   ccs -h node01.example.com --addfenceinst apc node01.example.com APC
   port=1
   ```

Você precisará adicionar um método fence para cada nódo no cluster. Os seguintes comandos configuram um método fence para cada nó com o nome de método **APC**. O dispositivo para o método fence especifica **apc** como o nome do dispositivo, que é um dispositivo previamente configurado com a opção **--addfencedev**, conforme descrito na Seção 5.5, "Configurando Dispositivos Fence". Cada nó é configurado com um único número de porta do switch de energia APC: O número da porta para o **node-01.example.com** é **1**, o número da porta para o **node-02.example.com** é **2**, e o número da porta para o **node-03.example.com** é **3**.

```
ccs -h node01.example.com --addmethod APC node01.example.com
ccs -h node01.example.com --addmethod APC node02.example.com
ccs -h node01.example.com --addmethod APC node03.example.com
ccs -h node01.example.com --addfenceinst apc node01.example.com APC port=1
ccs -h node01.example.com --addfenceinst apc node02.example.com APC port=2
ccs -h node01.example.com --addfenceinst apc node03.example.com APC port=3
```

Exemplo 5.2, "**cluster.conf** Depois de Adicionar Métodos Fence Baseados em Energia" exibe um arquivo de configuração **cluster.conf** depois de você ter adicionado estes métodos fencing e instâncias para cada nódo no cluster.

Exemplo 5.2. cluster.conf Depois de Adicionar Métodos Fence Baseados em Energia

```
<cluster name="mycluster" config_version="3">
   <clusternodes>
     <clusternode name="node-01.example.com" nodeid="1">
         <fence>
            <method name="APC">
              <device name="apc" port="1"/>
             </method>
         </fence>
     </clusternode>
     <clusternode name="node-02.example.com" nodeid="2">
         <fence>
            <method name="APC">
              <device name="apc" port="2"/>
            </method>
         </fence>
     </clusternode>
     <clusternode name="node-03.example.com" nodeid="3">
         <fence>
            <method name="APC">
              <device name="apc" port="3"/>
            </method>
         </fence>
     </clusternode>
   </clusternodes>
   <fencedevices>
         <fencedevice agent="fence_apc" ipaddr="apc_ip_example"
login="login_example" name="apc" passwd="password_example"/>
   </fencedevices>
   <rm>
   </rm>
</cluster>
```

-

Observe que quando você tiver terminado de configurar todos os componentes de seu cluster, você precisará sincronizar o arquivo de configuração do cluster em todos os nódos, como descrito na Seção 5.15, “Propagar o Arquivo de Configuração aos Nós do Cluster” .

5.7.2. Configurando um Dispositivo Fence Baseado em Armazenamento para um nódo

Quando usar métodos fencing sem energia (que é, SAN/armazenamento fencing) para fazer um fence em um nódo, você deve configurar o *unfencing* para o dispositivo fence. Isto assegura que um nódo com fence não está reabilitado até que o nódo tiver sido reinicializado. Quando você configurar o unfencing para um nódo, você especifica um dispositivo que espelha o dispositivo fence correspondente que você configurou para o nódo com a adição notável da ação explícita do **on** ou **enable**.

Para mais informações sobre fazer unfencing em um nódo, consulte a página man **fence_node**(8).

Use o seguinte procedimento para configurar um nódo com um dispositivo fence baseado em armazenamento único que usa um disposito fence chamado **sanswitch1**, que usa o agente fencing **fence_sanbox2**.

1. Adicionar um método fence para o nódo, fornecendo um nome para o método fence.

   ```
   ccs -h host --addmethod method node
   ```

   Por exemplo, para configurar um método fence chamado **SAN** para o nódo **node-01.example.com** no arquivo de configuração no nódo do cluster **node-01.example.com**, execute o seguinte comando:

   ```
   ccs -h node01.example.com --addmethod SAN  node01.example.com
   ```

2. Adicionar uma instância de fence para o método. Você deve especificar o dispositivo fence a ser usado para o nódo, o nódo que esta instância se aplica, o nome do método e quaisquer opções para este método que são específicas para este nódo:

   ```
   ccs -h host --addfenceinst fencedevicename node method [options]
   ```

   Por exemplo, para configurar uma instância fence no arquivo de configuração no nódo do cluster **node-01.example.com** que usa o porta de energia 11 do switch SAN no dispositivo fence chamado **sanswitch1** para fazer um fence no nódo do cluster **node-01.example.com** usando o método chamado **SAN**, execute o seguinte comando:

   ```
   ccs -h node01.example.com --addfenceinst sanswitch1
   node01.example.com SAN port=11
   ```

3. Para configurar o unfencing para o dispositivo fence baseado em armazenamento neste nódo execute o seguinte comando:

   ```
   ccs -h host --addunfence fencedevicename node action=on|off
   ```

Você precisará adicionar um método fence para cada nódo no cluster. Os seguintes comandos configuram um método fence para cada nódo com nome de método **SAN**. O dispositivo para o método fence especifica o **sanswitch** como nome de dispositivo, que é um dispositivo previamente

configurado com a opção --addfencedev, como descrito na Seção 5.5, "Configurando Dispositivos Fence". Cada nódo é configurado com único número de porta física SAN: O número da porta para **node-01.example.com** é **11**, o número da porta para **node-02.example.com** é **12**, e o número da porta para **node-03.example.com** é **13**.

```
ccs -h node01.example.com --addmethod SAN node01.example.com
ccs -h node01.example.com --addmethod SAN node02.example.com
ccs -h node01.example.com --addmethod SAN node03.example.com
ccs -h node01.example.com --addfenceinst sanswitch1 node01.example.com SAN
port=11
ccs -h node01.example.com --addfenceinst sanswitch1 node02.example.com SAN
port=12
ccs -h node01.example.com --addfenceinst sanswitch1 node03.example.com SAN
port=13
ccs -h node01.example.com --addunfence sanswitch1 node01.example.com
port=11 action=on
ccs -h node01.example.com --addunfence sanswitch1 node02.example.com
port=12 action=on
ccs -h node01.example.com --addunfence sanswitch1 node03.example.com
port=13 action=on
```

Exemplo 5.3, "**cluster.conf** Depois de Adicionar Métodos Fence baseados em Armazenamento" exibe uma configuração do arquivo **cluster.conf** depois de você ter adicionado métodos fencing, instâncias fencing e unfencing para cada nódo no cluster.

Exemplo 5.3. cluster.conf Depois de Adicionar Métodos Fence baseados em Armazenamento

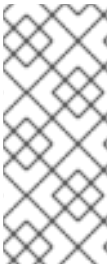
```
<cluster name="mycluster" config_version="3">
   <clusternodes>
     <clusternode name="node-01.example.com" nodeid="1">
         <fence>
            <method name="SAN">
       <device name="sanswitch1" port="11"/>
             </method>
         </fence>
         <unfence>
             <device name="sanswitch1" port="11" action="on"/>
         </unfence>
     </clusternode>
     <clusternode name="node-02.example.com" nodeid="2">
         <fence>
            <method name="SAN">
       <device name="sanswitch1" port="12"/>
             </method>
         </fence>
         <unfence>
             <device name="sanswitch1" port="12" action="on"/>
         </unfence>
     </clusternode>
     <clusternode name="node-03.example.com" nodeid="3">
         <fence>
            <method name="SAN">
       <device name="sanswitch1" port="13"/>
             </method>
```

```
        </fence>
        <unfence>
            <device name="sanswitch1" port="13" action="on"/>
        </unfence>
      </clusternode>
   </clusternodes>
   <fencedevices>
        <fencedevice agent="fence_sanbox2" ipaddr="san_ip_example"
login="login_example" name="sanswitch1" passwd="password_example"/>
   </fencedevices>
   <rm>
   </rm>
</cluster>
```

Observe que quando você tiver terminado de configurar todos os componentes de seu cluster, você precisará sincronizar o arquivo de configuração do cluster em todos os nódos, como descrito na Seção 5.15, “Propagar o Arquivo de Configuração aos Nós do Cluster” .

5.7.3. Configurando um dispositivo Fence de Backup

Você pode definir múltiplos métodos fence para um nódo. Se o fence falhar ao usar o primeiro método, o sistema tentará fazer o fence do nódo usando o segundo método, seguido por qualquer método adicional que você configurou. Para configurar um método fencing de backup para um nódo, você configura dois métodos para um nódo, configurando uma instância fence para cada método.



NOTA

A ordem na qual o sistema usará os métodos fencing que você configurou, segue a ordem no arquivo de configuração do cluster. O primeiro método que você configurar com o comando **ccs** será o primeiro método, o segundo método que você configurar é o backup do fencing. Para alterar a ordem, você pode remover o primeiro método do arquivo de configuração, então adicionar o método backup.

Note que a qualquer momento você pode exibir uma lista de métodos fence e instâncias atualmente configuradas para um nódo executando o seguinte comando. Se você não especificar um nódo, este comando listará os métodos fence e instâncias atualmente configurados para todos os nódos.

```
ccs -h host --lsfenceinst [node]
```

Use o seguinte procedimento para configurar um nódo com um método fence primário que usa um dispositivo fence chamado **apc**, que usa o agente fencing **fence_apc**, e um dispositivo fence de backup que usa um dispositivo fence chamado **sanswitch1**, que usa o agente fencing **fence_sandbox2**. Já que o dispositivo **sanswitch1** é um agente fencing baseado em armazenamento, você precisará configurar o unfencing para este dispositivo também.

1. Adicione um método fence primário para o nódo, fornecendo um nome para o método fence.

   ```
   ccs -h host --addmethod method node
   ```

Por exemplo, para configurar um método fence chamado **APC** como o método primário para o nódo **node-01.example.com** no arquivo de configuração no nódo do cluster **node-01.example.com**, execute o seguinte comando:

```
ccs -h node01.example.com --addmethod APC node01.example.com
```

2. Adicione uma instância fence para o método primário. Você deve especificar o dispositivo fence para usar no nódo, o nódo que esta instância se aplica, o nome do método e qualquer opções para este método que são específicas para este nódo:

```
ccs -h host --addfenceinst fencedevicename node method [options]
```

Por exemplo, para configurar uma instância fence no arquivo de configuração no nódo de cluster **node-01.example.com** que usa o switch de energia APC porta 1 no dispositivo fence chamado **apc** para criar um fence no nódo do cluster **node-01.example.com** usando o método chamado **APC**, execute o seguinte comando:

```
ccs -h node01.example.com --addfenceinst apc node01.example.com APC
port=1
```

3. Adicione um método fence de backup para o nódo, fornecendo um nome para o método fence.

```
ccs -h host --addmethod method node
```

Por exemplo, para configurar um método fence de backup chamado **SAN** para o nódo **node-01.example.com** no arquivo de configuração no nódo do cluster **node-01.example.com**, execute este comando:

```
ccs -h node01.example.com --addmethod SAN  node01.example.com
```

4. Adicione uma instância fence para método de backup. Você deve especificar o dispositivo fence para usar no nódo, o nódo que a instância se aplica, o nome do método e quaisquer opções para este método que são específicas a este nódo:

```
ccs -h host --addfenceinst fencedevicename node method [options]
```

Por exemplo, para configurar uma instância fence no arquivo de configuração no nódo do cluster **node-01.example.com** que usa o porta de energia 11 do switch SAN no dispositivo fence chamado **sanswitch1** para fazer um fence no nódo do cluster **node-01.example.com** usando o método chamado **SAN**, execute o seguinte comando:

```
ccs -h node01.example.com --addfenceinst sanswitch1
node01.example.com SAN port=11
```

5. Já que o dispositivo **sanswitch1** é um dispositivo baseado em armazenamento, você deve configurar o unfencing para este dispositivo.

```
ccs -h node01.example.com --addunfence sanswitch1 node01.example.com
port=11 action=on
```

Você pode continuar adicionar métodos fencing conforme necessário.

Este procedimento configura um dispositivo fence e um dispositivo fence de backup para um nódo no cluster. Você precisará configurar o fencing para outros nódos no cluster também.

Exemplo 5.4, "`cluster.conf` Depois de Adicionar Métodos Fence de Backup" exibe o arquivo de configuração `cluster.conf` depois de você ter adicionado um método fencing primário baseado em energia e um método fencing de backup baseado em armazenamento em cada nódo no cluster.

Exemplo 5.4. `cluster.conf` Depois de Adicionar Métodos Fence de Backup

```
<cluster name="mycluster" config_version="3">
   <clusternodes>
     <clusternode name="node-01.example.com" nodeid="1">
         <fence>
            <method name="APC">
              <device name="apc" port="1"/>
             </method>
            <method name="SAN">
       <device name="sanswitch1" port="11"/>
             </method>
         </fence>
         <unfence>
             <device name="sanswitch1" port="11" action="on"/>
         </unfence
     </clusternode>
     <clusternode name="node-02.example.com" nodeid="2">
         <fence>
            <method name="APC">
              <device name="apc" port="2"/>
            </method>
            <method name="SAN">
       <device name="sanswitch1" port="12"/>
             </method>
         </fence>
         <unfence>
             <device name="sanswitch1" port="12" action="on"/>
         </unfence
     </clusternode>
     <clusternode name="node-03.example.com" nodeid="3">
         <fence>
            <method name="APC">
              <device name="apc" port="3"/>
            </method>
            <method name="SAN">
       <device name="sanswitch1" port="13"/>
             </method>
         </fence>
         <unfence>
             <device name="sanswitch1" port="13" action="on"/>
         </unfence
     </clusternode>
   </clusternodes>
   <fencedevices>
        <fencedevice agent="fence_apc" ipaddr="apc_ip_example"
login="login_example" name="apc" passwd="password_example"/>
        <fencedevice agent="fence_sanbox2" ipaddr="san_ip_example"
```

```
login="login_example" name="sanswitch1" passwd="password_example"/>
   </fencedevices>
   <rm>
   </rm>
</cluster>
```

Observe que quando você tiver terminado de configurar todos os componentes de seu cluster, você precisará sincronizar o arquivo de configuração do cluster em todos os nódos, como descrito na Seção 5.15, “Propagar o Arquivo de Configuração aos Nós do Cluster” .

NOTA

A ordem no qual o sistema usará os métodos fencing que você criou segue sua ordem no arquivo de configuração do cluster. O primeiro método que você configura é o primeiro método fencing, e o segundo método que configura é o método fencing de backup. Para alterar a ordem, você pode remover o método primário do arquivo de configuração, então adicionar o método de volta.

5.7.4. Configurando um Nódo com energia Redundante

Se seu cluster está configurado com fontes de energia redundantes, você deve ter certeza de configurar o fencing para que seus nódos desliguem totalmente quando eles precisarem terem fence. Se você configurar cada fonte de energia com um método fence separado, cada fonte terá um fence separadamente; a segunda fonte de energia permitirá que o sistema continue rodando quando a primeira fonte houver fence e o sistema não terá fence num todo. Para configurar um sistema com duas fontes de alimentação, você deve configurar seus dispositivos fence para que ambas fontes de energia sejam desligadas e o sistema é totalmente desconectado. Isto requer que você configure duas instâncias dentro de um método fence único e que para cada instância que você configure ambos dispositivos com um atributo de **`action`** de **`off`** antes de configurar cada um dos dispositivos com um atributo **`action`** de **on**.

Para configurar um fence para um nó com duas fontes de energia, siga os passos desta seção.

1. Antes que você possa configurar um fence para um nó com energia redundante, você deve configurar cada um dos switches de energia como um dispositivo fence para o cluster. Para informações sobre configurar dispositivos fence, veja Seção 5.5, “Configurando Dispositivos Fence”.

   Para exibir uma lista de dispositivos fence atualmente configurados para seu cluster, execute o seguinte comando:

   ```
   ccs -h host --lsfencedev
   ```

2. Adicionar um método fence para o nódo, fornecendo um nome para o método fence.

   ```
   ccs -h host --addmethod method node
   ```

   Por exemplo, para configurar um método fence chamado **`APC-dual`** para o nó **`node-01.example.com`** no arquivo de configuração no nó do cluster **`node-01.example.com`**, execute o seguinte comando:

   ```
   ```

```
ccs -h node01.example.com --addmethod APC-dual node01.example.com
```

3. Adicione uma instância fence para a primeira fonte de energia para o método fence. Você deve especificar o dispositivo fence a ser usado para o nó, o nó nesta instância se aplica ao nome do método e quaisquer opções para este método que são específicas a este nó. Neste momento você configura o atributo **action** como **off**.

```
ccs -h host --addfenceinst fencedevicename node method [options]
action=off
```

Por exemplo, para configurar uma instância de fence no arquivo de configuração no nó do cluster **node-01.example.com** que usa o switch de energia APC porta 1 no dispositivo fence chamado **apc1** para fazer um fence no nó do cluster **node-01.example.com** usando o método chamado **APC-dual** e configurar o atributo **action** para **off**, execute o seguinte comando:

```
ccs -h node01.example.com --addfenceinst apc1 node01.example.com
APC-dual port=1 action=off
```

4. Adicione uma instância fence para a segunda fonte de energia para o método fence. Você deve especificar o dispositivo fence a ser usado para o nó. O nó nesta instância se aplica ao nome do método e quaisquer opções para este método que são específicos ao nó. Neste ponto, você configura o atributo **action** como **off** para esta instância também.

```
ccs -h host --addfenceinst fencedevicename node method [options]
action=off
```

Por exemplo, para configurar uma segunda instância fence no arquivo de configuração no nó do cluster **node-01.example.com** que usa o switch de energia APC porta 1 no dispositivo fence chamado **apc2** para fazer um fence no nó do cluster **node-01.example.com** usando o mesmo método conforme você especificou para a primeira instância chamada **APC-dual** e defina o atributo **action** para **on**, execute o seguinte comando:

```
ccs -h node01.example.com --addfenceinst apc2 node01.example.com
APC-dual port=1 action=off
```

5. Neste ponto, adicione uma outra instância fence à primeira fonte de energia ao método fence, configure o atributo **action** como **on**. Você deve especificar o dispositivo fence para usar para o nó. O nó nesta instância se aplica ao nome do método e quaisquer opções para o método que são específicas a este nó e especifique o atributo **action** como **on**:

```
ccs -h host --addfenceinst fencedevicename node method [options]
action=on
```

Por exemplo, para configurar uma instância fence no arquivo de configuração no nó do cluster **node-01.example.com** que usa o switch de energia APC porta 1 no dispositivo fence chamado **apc1** ao **apc1** para fazer um fence no nó do cluster **node-01.example.com** usando o método chamado **APC-dual** e defina o atributo **action** para **on**, execute o seguinte comando:

```
ccs -h node01.example.com --addfenceinst apc1 node01.example.com
APC-dual port=1 action=on
```

6. Adicione uma outra instância fence para a segunda fonte de energia ao método fence especificando o atributo **action** como **on** para esta instância. Você deve especificar o disposito fence a ser usado para o nó. O nó nesta instância se aplica ao nome do método e quaisquer opções para este método que são específicas a este nó tanto quando o atributo **action** para **on**.

```
ccs -h host --addfenceinst fencedevicename node method [options]
action=on
```

Por exemplo, para configurar uma segunda instância fence no arquivo de configuração no nó do cluster **node-01.example.com** que usa o switch de energia APC porta 1 no dispositivo fence chamado **apc2** para fazer um fence no nó do cluster **node-01.example.com** usando o mesmo método conforme você especificou para a primeira instância chamada **APC-dual** e defina o atributo **action** para **on**, execute o seguinte comando:

```
ccs -h node01.example.com --addfenceinst apc2 node01.example.com
APC-dual port=1 action=on
```

O Exemplo 5.5, "**cluster.conf** Depois de Adicionar um Fence de Duas Forças" mostra o arquivo de configuração **cluster.conf** depois de você ter adicionado um fence para duas fontes de energia para cada nó no cluster.

Exemplo 5.5. cluster.conf Depois de Adicionar um Fence de Duas Forças

```
<cluster name="mycluster" config_version="3">
   <clusternodes>
     <clusternode name="node-01.example.com" nodeid="1">
         <fence>
            <method name="APC-dual">
              <device name="apc1" port="1"action="off"/>
              <device name="apc2" port="1"action="off"/>
              <device name="apc1" port="1"action="on"/>
              <device name="apc2" port="1"action="on"/>
             </method>
         </fence>
     </clusternode>
     <clusternode name="node-02.example.com" nodeid="2">
         <fence>
            <method name="APC-dual">
              <device name="apc1" port="2"action="off"/>
              <device name="apc2" port="2"action="off"/>
              <device name="apc1" port="2"action="on"/>
              <device name="apc2" port="2"action="on"/>
            </method>
         </fence>
     </clusternode>
     <clusternode name="node-03.example.com" nodeid="3">
         <fence>
            <method name="APC-dual">
              <device name="apc1" port="3"action="off"/>
              <device name="apc2" port="3"action="off"/>
              <device name="apc1" port="3"action="on"/>
              <device name="apc2" port="3"action="on"/>
```

```
            </method>
          </fence>
      </clusternode>
    </clusternodes>
    <fencedevices>
        <fencedevice agent="fence_apc" ipaddr="apc_ip_example"
login="login_example" name="apc1" passwd="password_example"/>
        <fencedevice agent="fence_apc" ipaddr="apc_ip_example"
login="login_example" name="apc2" passwd="password_example"/>
    </fencedevices>
    <rm>
    </rm>
</cluster>
```

Observe que quando você tiver terminado de configurar todos os componentes de seu cluster, você precisará sincronizar o arquivo de configuração do cluster em todos os nódos, como descrito na Seção 5.15, “Propagar o Arquivo de Configuração aos Nós do Cluster” .

5.7.5. Remover Métodos Fence e Instâncias Fence

Para remover um método fence da sua configuração de cluster, execute o seguinte comando:

```
ccs -h host --rmmethod method node
```

Por exemplo, para remover um método fence que você nomeou **APC** e você configurou para o **node01.example.com** do arquivo de configuração no nó do cluster **node01.example.com**, execute o seguinte comando:

```
ccs -h node01.example.com  --rmmethod APC node01.example.com
```

Para remover todas as instâncias fence de um dispositivo fence a partir de um método fence, execute o seguinte comando:

```
ccs -h host --rmfenceinst fencedevicename node method
```

Por exemplo, para remover todas as instâncias do dispositivo fence chamado **apc1** a partir do método nomeado **APC-dual** configurado para o **node01.example.com** a partir do arquivo de configuração no nó do cluster **node01.example.com**, execute o seguinte comando:

```
ccs -h node01.example.com --rmfenceinst apc1 node01.example.com APC-dual
```

5.8. CONFIGURANDO UM DOMÍNIO DE FAILOVER

Um domínio de failover é um subconjunto nomeado de nós no cluster que são elegíveis a executar um serviço de cluster em um evento de falha no nó. Um domínio de failover pode ter as seguintes características:

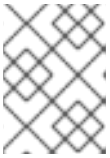
- Irrestrito (Unrestricted) — Permite especificar que um subconjunto de membros são preferidos mas que um serviço de cluster atribuído a este domínio pode rodar em qualquer membro disponível.

- Restringido (Restricted) — Permite restringir os membros que podem rodar um serviço de cluster em particular. Se nenhum dos membros de um domínio de failover restringido estiverem disponíveis, o serviço de cluster não pode ser iniciado (tanto manualmente ou pelo software do cluster).

- Desordenado (Unordered) — Quando um serviço de cluster é atribuído a um domínio de failover desordenado, o membro no qual o serviço de cluster roda é escolhido a partir dos membros do domínio de failover disponíveis sem ordem de prioridade.

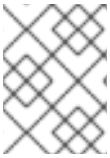
- Ordenado (Ordered) — Permite especificar uma ordem de preferência entre os membros de um domínio de failover. O membro no topo da lista é o mais preferido, seguido do segundo e assim por diante.

- Failback — Permite especificar se um serviço do domínio de failover deveria fazer um fail back no nó que estava originalmente rodando antes da falha do nó. Configurando esta característica é útil em circunstâncias onde um nó repetidamente falha e é parte de um domínio de failover ordenado.



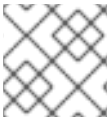
NOTA

A característica de failback é aplicável somente se um failover ordenado é configurado.



NOTA

Alterando um configuração de domínio de failover não possui efeito em serviços atualmente em execução.



NOTA

Domínio de failover *não* são requeridos para a operação.

Por padrão, domínios de failover são irrestritos e desordenados.

Em um cluster com diversos membros, usando um domínio de failover restringido pode minimizar o trabalho de configurar o cluster para executar um serviço de cluster (tal como **`httpd`**), que requer que você defina a configuração identicamente em todos os membros que rodam o serviço de cluster. Ao invés de configurar o cluster inteiro para rodar o serviço de cluster, você pode definir somente os membros do domínio de failover restringidos que você associou com o serviço de cluster.

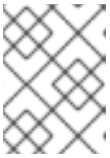
NOTA

Para configurar um membro preferido, você pode criar um domínio de failover que compreende somente de um membro no cluster. Isso faz que um serviço de cluster rode nesse membro do cluster primariamente (o membro preferido), mas permite que um serviço de cluster fazer um failover para quaisquer dos outros membros.

Para configurar um domínio de failover, realize o seguinte procedimento:

1. Para adicionar um domínio de failover, execute o seguinte comando:

```
ccs -h host --addfailoverdomain name [restricted] [ordered]
[nofailback]
```



NOTA

O nome deve ser descritivo o bastante para distinguir seu propósito relativo a outros nome usados em seu cluster.

Por exemplo, o comando seguinte configura um arquivo de domínio de failover chamado **example_pri** no **node-01.example.com** que é irrestrito, ordenado e permite failback:

```
ccs -h node-01.example.com --addfailoverdomain example_pri ordered
```

2. Para adicionar um nó a um domínio de failover, execute o seguinte comando:

```
ccs -h host --addfailoverdomainnode failoverdomain node priority
```

Por exemplo, para configurar um domínio de failover **example_pri** no arquivo de configuração no **node-01.example.com** que contenha **node-01.example.com** com prioridade 1, **node-02.example.com** com prioridade 2, e **node-03.example.com** com prioridade 3, execute os seguintes comandos:

```
ccs -h node-01.example.com --addfailoverdomainnode example_pri node-
01.example.com 1
ccs -h node-01.example.com --addfailoverdomainnode example_pri node-
02.example.com 2
ccs -h node-01.example.com --addfailoverdomainnode example_pri node-
03.example.com 3
```

Você pode listar todos os domínios de failover e nós do domínio de failover configurados em um cluster com o seguinte comando:

```
ccs -h host --lsfailoverdomain
```

Para remover um domínio de failover, execute o seguinte comando:

```
ccs -h host --rmfailoverdomain name
```

Para remover um nó de um domínio de failover, execute o seguinte comando:

```
ccs -h host --rmfailoverdomainnode failoverdomain node
```

Observe que quando você tiver terminado de configurar todos os componentes de seu cluster, você precisará sincronizar o arquivo de configuração do cluster em todos os nódos, como descrito na Seção 5.15, “Propagar o Arquivo de Configuração aos Nós do Cluster” .

5.9. CONFIGURANDO RECURSOS DE CLUSTER GLOBAIS

Você pode configurar dois tipos de recursos:

- Global — Recursos que estão disponíveis a qualquer serviço no cluster.
- Serviço específico — Recursos que são disponíveis a somente um serviço.

Para ver uma lista de recursos atualmente configurados e serviços no cluster, execute o seguinte comando:

```
ccs -h host --lsservices
```

Para adicionar um recurso de cluster global, execute o seguinte comando. Você pode adicionar um recurso que é local a um determinado serviço quando você configurar o serviço, conforme descrito na Seção 5.10, “Adicionando um Serviço de Cluster ao Cluster” .

```
ccs -h host --addresource resourcetype [resource options]
```

Por exemplo, o comando seguinte adiciona um recurso de sistema de arquivo global ao arquivo de configuração do cluster no **node01.example.com**. O nome do recurso é **web_fs**, o dispositivo do sistema de arquivo é **/dev/sdd2**, o ponto de montagem do sistema de arquivo é **/var/www**, e o tipo do sistema de arquivo é **ext3**.

```
ccs -h node01.example.com --addresource fs name=web_fs device=/dev/sdd2
mountpoint=/var/www fstype=ext3
```

Para informações sobre os tipos de recursos disponíveis e opções de recursos, veja Apêndice B, *Parâmetros dos Recursos de Alta Disponibilidade*.

Para remover um recurso global, execute o seguinte comando:

```
ccs -h host --rmresource resourcetype [resource options]
```

Se você precisar modificar os parâmetros de um recurso global existente, você pode remover o recurso e configura-lo novamente.

Observe que quando você tiver terminado de configurar todos os componentes de seu cluster, você precisará sincronizar o arquivo de configuração do cluster em todos os nódos, como descrito na Seção 5.15, “Propagar o Arquivo de Configuração aos Nós do Cluster” .

5.10. ADICIONANDO UM SERVIÇO DE CLUSTER AO CLUSTER

Para configurar um serviço de cluster em um cluster, realize os seguintes passos:

1. Adicione um serviço ao cluster com o seguinte comando:

   ```
   ccs -h host --addservice servicename [service options]
   ```

   NOTA

   Use um nome descritivo que claramente distingua o serviço de outros serviços no cluster.

   Quando você adicionar um serviço a uma configuração do cluster, você configura os seguintes atributos

   - **autostart** – Especifica se faz inicialização automática do serviço quando o cluster inicia. Use '1' para ativar e '0' para desativar; o padrão é ativado.

- **domain** — Especifica um domínio de failover (se requerido).
- **exclusive** — Especifica uma política onde o serviço somente roda em nós que não possuem outros serviços rodando neles.
- **recovery** — Especificar uma política de recuperação para o serviço. As opções são relocar, reiniciar, desabilitar ou recuperar padrão do serviço. A política de reiniciar a recuperação indica que o sistema deve tentar reiniciar o serviço com falha antes de realocar o serviço. A política Recuperar indica que o sistema deve tentar reiniciar o serviço em um nó diferente. A política Desabilitar indica que o sistema deve desasbilitar o grupo de recurso se algum componente falhar. A política Reiniciar Desabilitar indica que o sistema deve tentar reiniciar o serviço em questão se ele falhar, mas se a reinicialização do serviço falhar, o serviço será desabilitado em vez de ser movido para outro host no cluster.

  Se você selecionar **Reiniciar** (Restart) ou **Desabilitar Reiniciar** (Restart-Disable) como a política de recuperação para o serviço, você pode especificar o número máximo de falhas de reinicializações antes de realocar ou desabilitar o serviço, você pode especificar o período de tempo em segundos depois em que se deve ignorar uma reinicialização.

Por exemplo, para adicionar um serviço ao arquivo de configuração no nó do cluster **node-01.example.com** chamado **example_apache** que usa o domínio de failover **example_pri**, e possui a política de recuperação **relocate**, execute o seguinte comando:

```
ccs -h node-01.example.com --addservice example_apache
domain=example_pri recovery=relocate
```

Ao configurar os serviços para um cluster, você pode achar útil ver uma lista dos serviços disponíveis para seu cluster e as opções que estão disponíveis para cada serviço. Para informações sobre como utilizar o comando **ccs** para imprimir uma lista de serviços disponíveis e suas opções, veja Seção 5.11, “Listando Serviços de Cluster Disponíveis” .

2. Adicione recursos ao serviço com o seguinte comando:

```
ccs -h host --addsubservice servicename subservice [service options]
```

Dependendo do tipo de recurso que você quer usar, preencha o serviço com recursos globais ou específicos. Para adicionar um recurso global, use a opção **--addsubservice** do **ccs** para adicionar um recurso. Por exemplo, para adicionar um sistema de arquivos global chamado **web_fs** ao serviço chamado **example_apache** no arquivo de configuração do cluster no **node-01.example.com**, execute o seguinte comando:

```
ccs -h node01.example.com --addsubservice example_apache fs
ref=web_fs
```

Para adicionar um recurso de serviço específico ao serviço, você precisa especificar todas as opções de serviço. Por exemplo, se você não tivesse definido anteriormente o **web_fs** como um serviço global, você poderia adiciona-lo como um recurso de serviço específico com o seguinte comando:

```
ccs -h node01.example.com --addsubservice example_apache fs
name=web_fs device=/dev/sdd2 mountpoint=/var/www fstype=ext3
```

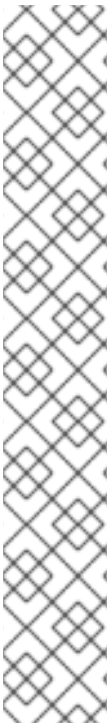
3. Para adicionar um serviço filh0 ao serviço, você também pode usar a opção **--addsubservice** do comando **ccs**, especificando as opções de serviço.

Se você precisar adicionar serviços dentro de uma estrutura de árvore de dependências, use dois pontos (":") para separar elementos e identificar sub serviços do mesmo tipo. O exemplo seguinte adiciona um terceiro serviço **nfsclient** como um subserviço do serviço **nfsclient** que é também um subserviço de um serviço **nfsclient** que é um subserviço de um serviço chamado **service_a**:

```
ccs -h node01.example.com --addsubservice service_a
nfsclient[1]:nfsclient[2]:nfsclient
```

NOTA

Se você estiver adicionando um recurso de serviço Samba, adicione-o diretamente ao serviço, *não* como um filho de outro recurso.



NOTA

Para verificar a existência do recurso de serviço IP usado em um serviço de cluster, você pode usar o comando **/sbin/ip addr show** em um nó de cluster (ao invés do comando obsoleto **ifconfig**). O resultado a seguir demonstra o comando **/sbin/ip addr show** executado em um nó executando um serviço de cluster:

```
1: lo: <LOOPBACK,UP> mtu 16436 qdisc noqueue
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
    inet6 ::1/128 scope host
       valid_lft forever preferred_lft forever
2: eth0: <BROADCAST,MULTICAST,UP> mtu 1356 qdisc pfifo_fast
qlen 1000
    link/ether 00:05:5d:9a:d8:91 brd ff:ff:ff:ff:ff:ff
    inet 10.11.4.31/22 brd 10.11.7.255 scope global eth0
    inet6 fe80::205:5dff:fe9a:d891/64 scope link
    inet 10.11.4.240/22 scope global secondary eth0
       valid_lft forever preferred_lft forever
```

Para remover um serviço e todos seus subserviços, execute o seguinte comando:

```
ccs -h host --rmservice servicename
```

Para remover um subserviço, execute o seguinte comando:

```
ccs -h host --rmsubservice servicename subservice [service options]
```

Observe que quando você tiver terminado de configurar todos os componentes de seu cluster, você precisará sincronizar o arquivo de configuração do cluster em todos os nódos, como descrito na Seção 5.15, “Propagar o Arquivo de Configuração aos Nós do Cluster” .

5.11. LISTANDO SERVIÇOS DE CLUSTER DISPONÍVEIS

Você pode usar o **ccs** para imprimir uma lista de serviços disponíveis para um cluster. Você também pode utilizar o comando **ccs** para imprimir uma lista de opções que você pode especificar para um tipo de serviço específico.

Para imprimir uma lista dos serviços de cluster atualmente disponíveis para seu cluster execute o seguinte comando:

```
ccs -h host --lsserviceopts
```

Por exemplo, o seguinte comando lista os serviços de cluster disponíveis no nó de cluster **node1**, mostrando o resultado de exemplo.

```
[root@ask-03 ~]# ccs -h node1 --lsserviceopts
service - Defines a service (resource group).
ASEHAagent - Sybase ASE Failover Instance
SAPDatabase - SAP database resource agent
SAPInstance - SAP instance resource agent
apache - Defines an Apache web server
clusterfs - Defines a cluster file system mount.
fs - Defines a file system mount.
ip - This is an IP address.
lvm - LVM Failover script
mysql - Defines a MySQL database server
named - Defines an instance of named server
netfs - Defines an NFS/CIFS file system mount.
nfsclient - Defines an NFS client.
nfsexport - This defines an NFS export.
nfsserver - This defines an NFS server resource.
openldap - Defines an Open LDAP server
oracledb - Oracle 10g Failover Instance
orainstance - Oracle 10g Failover Instance
oralistener - Oracle 10g Listener Instance
postgres-8 - Defines a PostgreSQL server
samba - Dynamic smbd/nmbd resource agent
script - LSB-compliant init script as a clustered resource.
tomcat-6 - Defines a Tomcat server
vm - Defines a Virtual Machine
action - Overrides resource action timings for a resource instance.
```

Para imprimir uma lista de opções que você pode especificar para um tipo de serviço específico, execute o seguinte comando:

```
ccs -h host --lsserviceopts service_type
```

Por exemplo, o seguinte comando lista as opções de serviço para o serviço **vm**.

```
[root@ask-03 ~]# ccs -f node1 --lsserviceopts vm
vm - Defines a Virtual Machine
  Required Options:
    name: Name
  Optional Options:
    domain: Cluster failover Domain
    autostart: Automatic start after quorum formation
    exclusive: Exclusive resource group
    recovery: Failure recovery policy
    migration_mapping: memberhost:targethost,memberhost:targethost ..
    use_virsh: If set to 1, vm.sh will use the virsh command to manage
virtual machines instead of xm. This is required when using non-Xen
virtual machines (e.g. qemu / KVM).
```

```
    xmlfile: Full path to libvirt XML file describing the domain.
    migrate: Migration type (live or pause, default = live).
    path: Path to virtual machine configuration files.
    snapshot: Path to the snapshot directory where the virtual machine
image will be stored.
    depend: Top-level service this depends on, in service:name format.
    depend_mode: Service dependency mode (soft or hard).
    max_restarts: Maximum restarts for this service.
    restart_expire_time: Restart expiration time; amount of time before a
restart is forgotten.
    status_program: Additional status check program
    hypervisor: Hypervisor
    hypervisor_uri: Hypervisor URI (normally automatic).
    migration_uri: Migration URI (normally automatic).
    __independent_subtree: Treat this and all children as an independent
subtree.
    __enforce_timeouts: Consider a timeout for operations as fatal.
    __max_failures: Maximum number of failures before returning a failure
to a status check.
    __failure_expire_time: Amount of time before a failure is forgotten.
    __max_restarts: Maximum number restarts for an independent subtree
before giving up.
    __restart_expire_time: Amount of time before a failure is forgotten
for an independent subtree.
```

5.12. RECURSOS DE MÁQUINA VIRTUAL

Recursos de máquina virtual são configurados de forma diferente de outros recursos de cluster, eles não são agrupados em definições de serviços. Desde o lançamento do Red Hat Enterprise Linux 6.2, quando você configura uma máquina virtual em um cluster com o comando **ccs** você pode usar a opção **--addvm** (ao invés da opção **addservice**). Isto assegura que o recurso **vm** é definido diretamente sob o nó de configuração **rm** no arquivo de confiugração do cluster.

Um recurso de máquina virtual requer ao menos um atributo de **name** e um **path**. O atributo **name** deve coincidir com o nome do domínio **libvirt** e o atributo **path** deve especificar o diretório onde as definições da máquina virtual compartilhada são armazenadas.

NOTA

O atributo **path** no arquivo de configuração do cluster é uma especificação de caminho ou um nome de diretório, não um caminho para um arquivo individual.

Se as definições de máquina virtual são armazenadas em um diretório compartilhado chamado **/mnt/vm_defs**, o comando a seguir definirá uma máquina virtual chamada **guest1**:

```
# ccs -h node1.example.com --addvm guest1 path=/mnt/vm_defs
```

Executar este comando adiciona a seguinte linha ao nó de configuração no arquivo **cluster.conf**:

```
<vm name="guest1" path="/mnt/vm_defs"/>
```

5.13. CONFIGURANDO UM DISCO DE QUORUM

NOTA

Os parâmetros do disco de quorum e heurísticas dependem do ambiente local e os requerimentos especiais requeridos. Para entender o uso dos parâmetros do disco de quorum e heurísticas, consulte a man page qdisk(5). Se você requer assistência para entender e usar o disco de quorum, contate um representante autorizado de suporte da Red Hat.

Use o seguinte comando para configurar seu sistema para usar um disco de quorum:

```
ccs -h host --setquorumd [quorumd options]
```

Note que este comando redefine todas as propriedades que você pode definir com a opção **--setquorumd** para seus valores padrão, como descrito em Seção 5.1.5, “Comandos que Sobrescrevem Configurações Anteriores”.

A Tabela 5.1, “Opções do Disco de Quorum” resume o significado das opções de disco de quorum que você pode precisar definir. Para uma lista completa de parâmetros do disco de quorum, consulte o esquema de cluster em **/usr/share/cluster/cluster.rng**, e o esquema anotado em **/usr/share/doc/cman-X.Y.ZZ/cluster_conf.html**.

Tabela 5.1. Opções do Disco de Quorum

Parâmetro	Descrição
interval	A frequencia de ciclos de leitura/escrita, em segundos.
votes	O número de votos de quorum anuncia ao **cman** quando ele possui uma contagem alta o bastante.
tko	O número de ciclos que um nó deve perder para ser declarado morto.
min_score	A contagem mínima para um nó ser considerado "vivo". Se omitido ou ajustado para 0, a função padrão **floor((*n*+1)/2)**, é usada, onde *n* é a soma da contagem de heurísticas. O valor da **Contagem Mínima** nunca deve exceder a soma da da contagem de heurísticas; caso contrário, o disco de quorum pode não estar disponível.
device	O dispositivo de armazenamento que o daemon quorum usa. O dispositivo deve ser o mesmo em todos os nós.
label	Especifica o rótulo do disco de quorum criado pelo utilitário **mkqdisk**. Se este campo contém uma entrada, o rótulo sobrescreve o campo do **Dispositivo**. Se este campo é usado, o daemon do quorum le o **/proc/partitions** e verifica por assinaturas qdisk em cada dispositivo de bloco encontrado, comparando o rótulo com o rótulo especificado. Isto é útil em configurações onde o nome de dispositivo de quorum difere entre os nós.

Use o seguinte comando para configurar as heurísticas para um disco de quorum:

```
ccs -h host --addheuristic [heuristic options]
```

A Tabela 5.2, “Heurísticas do Disco de Quorum” resume o significado das heuristicas do disco de quorum que você poderá precisar definir.

Tabela 5.2. Heurísticas do Disco de Quorum

Parâmetro	Descrição
program	O caminho para o programa utilizado para determinar se esta eurística está disponível. Isto pode ser qualquer coisa que possa ser executada pelo **/bin/sh -c**. Um valor de retorno de 0 indica sucesso; qualquer outra coisa indica falha. Este parâmetro é necessário para utilizar o disco do quorum.
interval	A frequencia (em segundos) na qual a heurística é consultada. O intervalo padrão para cada heurística é de 2 segundos.
score	O peso desta heurística. Seja cuidadoso quando determinar contagem para as heurísticas. A contagem padrão para cada heurística é 1.
tko	O número consecutivo de falhas requeridas antes que esta heuristica seja declarada indisponível.

Para ver uma lista das opções de disco de quorum e heurísticas que estão configuradas em um sistema, você pode executar o seguinte comando:

```
ccs -h host --lsquorum
```

Para remover um heurística especificada por uma opção da heurística, você pode executar o seguinte comando:

```
ccs -h host rmheuristic [heuristic options]
```

Observe que quando você tiver terminado de configurar todos os componentes de seu cluster, você precisará sincronizar o arquivo de configuração do cluster em todos os nódos, como descrito na Seção 5.15, “Propagar o Arquivo de Configuração aos Nós do Cluster” .

NOTA

A sincronização e ativação propaga e ativa o atualizado arquivo de configuração do cluster atualizado. Entretanto, para o disco de quorum operar, você deve reinicializar o cluster (consulte a Seção 6.2, “Iniciando e Parando um Cluster”), certificando-se de que você reiniciou o daemon do **qdiskd** em cada nó.

5.14. CONFIGURAÇÕES DE CLUSTER DIVERSAS

Esta seção descreve o uso do comando **ccs** para configurar o seguinte:

- Seção 5.14.1, “Versão de Configuração do Cluster”
- Seção 5.14.2, “Configuração Multicast”
- Seção 5.14.3, “Configurando um Cluster de Dois Nós”

- Seção 5.14.4, “Autenticando”
- Seção 5.14.5, “Configurando o Protocolo de Anel Redundante”

Você pode também usar o comando **ccs** para definir parâmetros de configuração de cluster avançados, incluindo opções **totem**, opções **dlm**, opções **rm** e opções **cman**. Para informações sobre definir parâmetros veja a página man **ccs**(8) e o esquema de arquivo de configuração de cluster anotada **/usr/share/doc/cman-X.Y.ZZ/cluster_conf.html**.

Para ver uma lista de atributos de cluster diversos que foram configuradas para um cluster, execute o seguinte comando:

```
ccs -h host --lsmisc
```

5.14.1. Versão de Configuração do Cluster

Um arquivo de configuração de cluster inclui o valor da versão de configuração de cluster. O valor da versão da configuração é definida para **1** por padrão quando você cria um arquivo de configuração de cluster e é automaticamente incrementado cada vez que você modificar sua configuração de cluster. Entretanto, se você precisar definir isso para um outro valor, você pode especificar isso com o seguinte comando:

```
ccs -h host --setversion n
```

Você pode obter o valor da versão de configuração atual em um arquivo de configuração de cluster existente com o seguinte comando:

```
ccs -h host --getversion
```

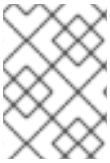
Para incrementar o valor da versão da configuração atual para 1 no arquivo de configuração do cluster em cada nó no cluster, execute o seguinte comando:

```
ccs -h host --incversion
```

5.14.2. Configuração Multicast

Se você não especificar um endereço multicast no arquivo de configuração do cluster, o software Complemento de Alta Disponibilidade da Red Hat cria um baseado no ID do cluster. Ele gera os 16 bits menores do endereço e anexa-os à porção maior do endereço de acordo se o protocolo é IPV4 ou IPV6:

- Para o IPV4 — O endereço formado é 239.192. mais os 16 bits mais baixos gerados pelo software do Complemento de Alta Disponibilidade da Red Hat.
- Para o IPV6 — O endereço formado é FF15:: mais os 16 bits mais baixos gerados pelo software do Complemento de Alta Disponibilidade da Red Hat.



NOTA

O ID do cluster é um identicado único que o **cman** gera para cada cluster. Para vizualizar o ID do cluster, execute o comando **cman_tool status** em um nó do cluster.

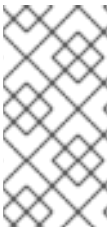
Você pode especificar manualmente um endereço multicast no arquivo de configuração do cluster com o seguinte comando:

```
ccs -h host --setmulticast multicastaddress
```

Note que este comando redefine todas as propriedades que você pode definir com a opção **--setmulticast** para seus valores padrão, como descrito em Seção 5.1.5, "Comandos que Sobrescrevem Configurações Anteriores".

Se você especificar um endereço multicast, você deve usar a série 239.192.x.x (ou FF15:: para IPv6) que usa o **cman**. Caso contrário, usando um endereço multicast fora desta abrangência pode causar resultados imprevísiveis. Por exemplo, usando 224.0.0.x (que é "Todos os hosts na rede") pode não ser roteado corretamente ou mesmo não ser roteado por algum hardware.

Se você especificar ou modificar um endereço multicast, você precisa reiniciar o cluster para que seja efetuado. Para instruções sobre iniciar ou interromper um cluster com o comando **ccs** consulte o Seção 6.2, "Iniciando e Parando um Cluster" .



NOTA

Se você especificar um endereço multicast, certifique-se que você checou a configuração dos roteadores que os pacotes do cluster passam. Alguns roteadores pode levar um longo tempo para aprender os endereços, impactando seriamente a performance do cluster.

Para remover um endereço multicast de um arquivo de configuração, use a opção **--setmulticast** do **ccs**, mas não especifica um endereço multicast:

```
ccs -h host --setmulticast
```

5.14.3. Configurando um Cluster de Dois Nós

Se você estiver configurando um cluster de dois nós, você pode executar o seguinte comando para permitir um nó único para manter quorum (por exemplo, se um nó falhar):

```
ccs -h host --setcman two_node=1 expected_votes=1
```

Note que este comando redefine todas as propriedades que você pode definir com a opção **--setcman** para seus valores padrão, como descrito em Seção 5.1.5, "Comandos que Sobrescrevem Configurações Anteriores".

Quando você utilizar o comando **ccs --setcman** para adicionar, remover ou modificar a opção **two_node**, você precisa reiniciar o cluster para esta mudança tomar efeito. Para informações sobre iniciar ou interromper um cluster com o comando **ccs** consulte o Seção 6.2, "Iniciando e Parando um Cluster".

5.14.4. Autenticando

Você pode habilitar a depuração para todos os daemons em um cluster, ou você pode habilitar a autenticação para o processamento do cluster específico.

Para habilitar a depuração em todos os daemons, execute o seguinte comando. Por padrão, o logging é direcionado para o arquivo **/var/log/cluster/*daemon*.log**.

```
ccs -h host --setlogging [logging options]
```

Por exemplo, o seguinte comando habilita a depuração de todos os daemons.

```
# ccs -h node1.example.com --setlogging debug=on
```

Note que este comando redefine todas as propriedades que você pode definir com a opção **--setlogging** para seus valores padrão, como descrito em Seção 5.1.5, “Comandos que Sobrescrevem Configurações Anteriores”.

Para habilitar a depuração para processos de cluster individuais, execute o seguinte comando. A configuração de logging per-daemon sobrescreve as configurações globais.

```
ccs -h host --addlogging [logging daemon options]
```

Por exemplo, os seguintes comandos habilitam a depuração para os daemons **corosync** e **fenced** daemons.

```
# ccs -h node1.example.com --addlogging name=corosync debug=on
# ccs -h node1.example.com --addlogging name=fenced debug=on
```

Para remover configurações de log para daemons individuais, use o seguinte comando;

```
ccs -h host --rmlogging name=clusterprocess
```

Por exemplo, o seguinte comando remove o log de daemon específico para os daemons **fenced**.

```
ccs -h host --rmlogging name=fenced
```

Para uma lista de daemons de log para os quais você pode habilitar o logging assim como as opções de logging adicionais que você pode configurar para ambos logging global e per-daemon, consulte a página man **cluster.conf**(5).

Observe que quando você tiver terminado de configurar todos os componentes de seu cluster, você precisará sincronizar o arquivo de configuração do cluster em todos os nódos, como descrito na Seção 5.15, “Propagar o Arquivo de Configuração aos Nós do Cluster” .

5.14.5. Configurando o Protocolo de Anel Redundante

Desde o Red Hat Enterprise Linux 6.4, o Red Hat High Availability Add-On suporta a configuração do protocolo de anel redundante. Ao utilizar o protoloco de anel redundante, existem diversas considerações que você deve levar em conta, como descrito em Seção 7.6, “Configurando o Protocolo de Anel Redundante”.

Para especificar uma segunda interface de rede para usar para protoloco de anel redundante, você adiciona um nome alternado para o nó utilizando a opção **addalt** do comando **ccs**.

```
ccs -h host --addalt node_name alt_name
```

Por exemplo, o seguinte comando configura o nome alternado **clusternet-node1-eth2** para o nó de cluster **clusternet-node1-eth1**:

```
# ccs -h clusternet-node1-eth1 --addalt clusternet-node1-eth1 clusternet-
node1-eth2
```

Opcionalmente, você pode especificar manualmente um endereço de multicast, uma porta e um TTL para o segundo anel. Se você especificar um multicast para o segundo anel, tanto o endereço de multicast alternado quanto a porta alternada devem ser diferentes do endereço do multicast para o primeiro anel. Se você especificar uma porta alternada, os números de porta do primeiro anel e segundo anel devem ser diferentes em pelo menos dois, pois o próprio sistema usa a porta e porta-1 para realizar as operações. Se você não especificar um endereço e multicast alternado, o sistema irá utilizar automaticamente um endereço de multicast diferente para o segundo anel.

Para especificar um endereço multicast alternado, porta ou TTL para o segundo anel, você utiliza a opção **--setaltmulticast** do comando **ccs**:

```
ccs -h host --setaltmulticast [alt_multicast_address]
[alt_multicast_options].
```

Por exemplo, o seguinte comando define um endereço multicast alternado de 239.192.99.88, uma porta de 888, e um TTL de 3 para o cluster definido no arquivo **cluster.conf**no nó **clusternet-node1-eth1**:

```
ccs -h clusternet-node1-eth1 --setaltmulticast 239.192.99.88 port=888
ttl=3
```

Para remover um endereço de multicast alternado, especifique a opção **--setaltmulticast** do comando**ccs** mas não especifique o endereço multicast. Note que ao executar este comando, ele redefinirá todas as outras propriedades que você pode definir com a opção **--setaltmulticast** para seus valores padrão, como descrito em Seção 5.1.5, “Comandos que Sobrescrevem Configurações Anteriores”.

Quando você terminar de configurar todos os componentes de seu cluster, você precisará sincronizar o arquivo de configuração do cluster para todos os nodos, como descrito na Seção 5.15, “Propagar o Arquivo de Configuração aos Nós do Cluster”.

5.15. PROPAGAR O ARQUIVO DE CONFIGURAÇÃO AOS NÓS DO CLUSTER

Depois de você ter criado ou editado o arquivo de configuração do cluster em um dos nós no cluster, você precisa propagar o mesmo arquivo para todos os nós no cluster e ativar a configuração.

Use o seguinte comando para propagar e ativar o arquivo de configuração do cluster:

```
ccs -h host --sync --activate
```

Para verificar que todos os nós especificados no arquivo de configuração de cluster dos hosts possuem arquivos idênticos, execute o seguinte comando:

```
ccs -h host --checkconf
```

Se você tiver criado ou editado um arquivo de configuração em um nó local, use o seguinte comando para enviar esse arquivo a um dos nós no cluster:

```
ccs -f file -h host --setconf
```

- Para verificar que todos os nós especificados no arquivo local possuem o arquivo de configuração de cluster idênticos, execute o seguinte comando:

```
ccs -f file --checkconf
```

CAPÍTULO 6. GERENCIANDO O COMPLEMENTO DE ALTA DISPONIBILIDADE DA RED HAT COM O CCS

Este capítulo descreve as várias tarefas administrativas para gerenciar o Complemento de Alta Disponibilidade da Red Hat por meio do comando **ccs**, que é suportado a partir do lançamento do Red Hat Enterprise Linux 6.1 e posteriores. Este capítulo consiste das seguintes seções:

- Seção 6.1, “Gerenciando Nós no Cluster”
- Seção 6.2, “Iniciando e Parando um Cluster”
- Seção 6.3, “Diagnosticando e Corrigindo Problemas em um Cluster”

6.1. GERENCIANDO NÓS NO CLUSTER

Esta seção documenta como realizar as seguintes funções de gerenciamento de nós com o comando **ccs**:

- Seção 6.1.1, “Faz um nó sair ou se juntar a um Cluster”
- Seção 6.1.2, “Adicionar um Membro a um Cluster em Execução”

6.1.1. Faz um nó sair ou se juntar a um Cluster

Você pode usar o comando **ccs** para fazer um nó sair de um cluster parando os serviços de cluster nesse nó. Fazer um nó deixar um cluster não remove a informação de configuração de cluster desse nó. Fazer um nó sair de um cluster previne que nó se junte ao cluster quando for reinicializado.

Para fazer um nó sair de um cluster, execute o seguinte comando, o qual pára os serviços de cluster no nó especificado com a opção **-h**:

```
ccs -h host --stop
```

Quando você pára os serviços de cluster em um nó, qualquer serviço que estiver rodando nesse nó falhará.

Para apagar um nó inteiramente de uma configuração de cluster, use a opção **--rmnode** do comando **ccs**, como descrito na Seção 5.4, “Criando um Cluster” .

Para fazer um nó se rejuntar a um cluster, execute o seguinte comando, que reinicia os serviços de cluster no nó especificado com a opção **-h**:

```
ccs -h host --start
```

6.1.2. Adicionar um Membro a um Cluster em Execução

Para adicionar um membro a um cluster em execução, adicione um nó ao cluster conforme descrito na Seção 5.4, “Criando um Cluster” . Depois de atualizar o arquivo de configuração, propague o arquivo a todos os nós no cluster e certifique-se de ativar o novo arquivo de configuração do cluster, conforme descrito na Seção 5.15, “Propagar o Arquivo de Configuração aos Nós do Cluster” .

6.2. INICIANDO E PARANDO UM CLUSTER

Você pode usar o **ccs** para parar um cluster usando o seguinte comando para parar serviços de cluster em todos os nós no cluster:

```
ccs -h host --stopall
```

Você pode usar o **ccs** para parar um cluster que não estiver rodando usando o seguinte comando para iniciar serviços de cluster em todos os nós do cluster:

```
ccs -h host --startall
```

6.3. DIAGNOSTICANDO E CORRIGINDO PROBLEMAS EM UM CLUSTER

Para informações sobre diagnosticar e corrigir problemas em um cluster, veja o Capítulo 9, *Diagnosticando e Corrigindo Problemas em um Cluster*. Existem algumas checagens simples que você pode fazer com o comando **ccs** entretanto.

Para verificar que todos os nós especificados no arquivo de configuração de cluster dos hosts possuem arquivos de configuração idênticos, execute o seguinte comando:

```
ccs -h host --checkconf
```

Se você criou ou editou um arquivo de configuração em um nó local, você pode verificar se todos os nós especificados no arquivo local possuem arquivos de configuração idênticos com o seguinte comando:

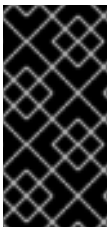
```
ccs -f file --checkconf
```

CAPÍTULO 7. CONFIGURANDO O COMPLEMENTO DE ALTA DISPONIBILIDADE DA RED HAT COM AS FERRAMENTAS DA LINHA DE COMANDO

Este capítulo descreve como configurar o software Complemento de Alta Disponibilidade da Red Hat editando diretamente o arquivo de configuração do cluster (**/etc/cluster/cluster.conf**) e usando as ferramentas da linha de comando. O capítulo fornece procedimentos sobre montar um arquivo de configuração, uma seção por vez, iniciando com uma amostra disponibilizada neste capítulo. Como uma alternativa para iniciar com um arquivo de exemplo fornecido aqui, você pode copiar o esqueleto do arquivo de configuração da página man **cluster.conf**. Entretanto, fazendo isso não alinharia necessariamente com a informação fornecida nos procedimentos subsequentes deste capítulo. Existem outras maneiras para criar e configurar um arquivo de configuração de cluster; este capítulo fornece procedimentos sobre montar um arquivo de configuração, uma seção por vez. Também, tenha em mente que isto é apenas um ponto inicial para desenvolver um arquivo de configuração que atenda suas necessidades de clusterização.

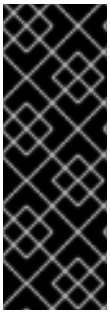
Este capítulo consiste das seguintes seções:

- Seção 7.1, “Tarefas de Configuração”
- Seção 7.2, “Criando um arquivo de Configuração de Cluster Básica”
- Seção 7.3, “Configuração de Fence”
- Seção 7.4, “Configurar Domínios de Failover”
- Seção 7.5, “Configurando Serviços de Alta Disponibilidade”
- Seção 7.7, “Configuração das Opções de Depuração”
- Seção 7.6, “Configurando o Protocolo de Anel Redundante”
- Seção 7.8, “Verificando uma Configuração”



IMPORTANTE

Certifique-se que sua implementação do Complemento de Alta Disponibilidade atenda suas necessidades e possa ser suportada. Consulte um representante autorizado Red Hat para verificar suas configurações antes da implementação. Além disso, disponibilize tempo para um período de testes de falha.

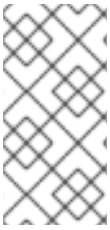


IMPORTANTE

Este capítulo referencia elementos e atributos **cluster.conf** comumente usados. Para uma lista compreensiva e a descrição dos elementos e atributos do **cluster.conf**, consulte o esquema de cluster em **/usr/share/cluster/cluster.rng** e o esquema anotado em **/usr/share/doc/cman-X.Y.ZZ/cluster_conf.html** (por exemplo **/usr/share/doc/cman-3.0.12/cluster_conf.html**).

IMPORTANTE

Certos procedimentos neste capítulo indicam o uso do comando **`cman_tool version -r`** para propagar a configuração de cluster por todo um cluster. Usar esse comando requer que o **`ricci`** esteja rodando. Usar o **ricci** requer uma senha na primeira vez que você interagir com o **ricci** a partir de qualquer máquina especifica. Para informações sobre o serviço **`ricci`**, consulte a Seção 2.13, “Considerações para o **`ricci`**”.



NOTA

Procedimentos neste capítulo, podem incluir comandos especificos para algumas das ferramentas da linha de comando listada no Apêndice E, *Resumo das Ferramentas da Linha de Comando*. Para mais informações sobre todos os comandos e variáveis, consulte a página man para cada ferramenta da linha de comando.

7.1. TAREFAS DE CONFIGURAÇÃO

Configurando o software do Complemento de Alta Disponibilidade da Red Hat com as ferramentas da linha de comando consiste dos seguintes passos:

1. Criando um cluster. Consulte a Seção 7.2, “Criando um arquivo de Configuração de Cluster Básica”.
2. Configurando o fence. Consulte a Seção 7.3, “Configuração de Fence”.
3. Configurando domínios de failover. Consulte a Seção 7.4, “Configurar Domínios de Failover”.
4. Configurando serviços de Alta Disponibilidade. Consulte a Seção 7.5, “Configurando Serviços de Alta Disponibilidade”.
5. Verificando a configuração. Consulte a Seção 7.8, “Verificando uma Configuração”.

7.2. CRIANDO UM ARQUIVO DE CONFIGURAÇÃO DE CLUSTER BÁSICA

Desde que o hardware de cluster, o Red Hat Enterprise Linux e o software do Complemento de Alta Disponibilidade estão instalados, você pode criar um arquivo de configuração (**`/etc/cluster/cluster.conf`**) e inicie a execução do Complemento de Alta Disponibilidade. Como um ponto inicial somente, esta seção descreve como criar um esqueleto do arquivo de configuração de cluster sem fence, domínios failover e serviços de Alta Disponibilidade. Seções subsequentes descrevem como configurar estas partes do arquivo de configuração.

IMPORTANTE

Este é somente um passo para criar um arquivo de configuração de cluster; o arquivo resultante não possui qualquer fence e não é considerado ser uma configuração suportada.

Os seguintes passos descrevem como criar e configurar um esqueleto do arquivo de configuração do cluster. Finalmente, o arquivo de configuração para seu cluster irá variar de acordo com o número de nós, o tipo de fence, o tipo e número de serviços de Alta Disponibilidade e outros requerimentos especificos do local.

1. Em qualquer nó no cluster, crie o **/etc/cluster/cluster.conf**, usando o modelo do exemplo em Exemplo 7.1, "**cluster.conf** Exemplo: Configuração Básica".

2. **Opcional** se você está configurando um cluster de dois nós, você pode adicionar a seguinte linha ao arquivo de configuração para permitir que um nó único mantenha quorum (por exemplo, se um nó falhar):

   ```
   <cman two_node="1" expected_votes="1"/>
   ```

   Quando você adicionar ou remover a opção **two_node** do arquivo **cluster.conf**, você precisa reiniciar o cluster para que esta mudança seja efetuada ao atualizar a confiugração. Para obter informações sobre como atualizar uma configuração de cluster, consulte o Seção 8.4, "Atualizando uma Configuração". Para um exemplo de especificação da opção **two_node** consulte o Exemplo 7.2, "**cluster.conf** Exemplo: Configuração de Nós Básica".

3. Especifique o nome do cluster e o número da versão da configuração usando os atributos do **cluster**: **name** e **config_version** (consulte o Exemplo 7.1, "**cluster.conf** Exemplo: Configuração Básica" ou Exemplo 7.2, "**cluster.conf** Exemplo: Configuração de Nós Básica").

4. Na seção **clusternodes**, especifique o nome do nó e a ID do nó para cada nó usando os atributos do **clusternode**: **name** e **nodeid**.

5. Salve o **/etc/cluster/cluster.conf**.

6. Valide o arquivo no esquema de cluster (**cluster.rng**) rodando o comando **ccs_config_validate**. Por exemplo:

   ```
   [root@example-01 ~]# ccs_config_validate
   Configuration validates
   ```

7. Propague o arquivo de configuração no **/etc/cluster/** em cada nó no cluster. Por exemplo, você poderia propagar o arquivo a outros nós no cluster usando o comando **scp**.

   NOTA

   Propagando o arquivo de configuração do cluster desta maneira é necessário na primeira vez que o cluster é criado. Uma vez o cluster é instalado e rodando, o arquivo de configuração do cluster pode ser propagado usando o **cman_tool version -r**. É possível usar o comando **scp**. Além disso, você deve rodar o **ccs_config_validate** se você propagar um arquivo de configuração atualizado pelo **scp**.

   NOTA

   Enquanto existem outros elementos e atributos presentes no exemplo do arquivo de configuração (por exemplo o **fence** e o **fencedevices**, não há necessidade de popula-los agora.) Procedimentos subsequentes neste capítulo fornecem informações sobre especificar outros elementos e atributos.

8. Inicie o cluster. Em cada nó no cluster, execute o seguinte comando:

   ```
   service cman start
   ```

Por exemplo:

```
[root@example-01 ~]# service cman start
Starting cluster:
   Checking Network Manager...                             [  OK
]
   Global setup...                                         [  OK
]
   Loading kernel modules...                               [  OK
]
   Mounting configfs...                                    [  OK
]
   Starting cman...                                        [  OK
]
   Waiting for quorum...                                   [  OK
]
   Starting fenced...                                      [  OK
]
   Starting dlm_controld...                                [  OK
]
   Starting gfs_controld...                                [  OK
]
   Unfencing self...                                       [  OK
]
   Joining fence domain...                                 [  OK
]
```

9. Em qualquer nó no cluster, rode o **`cman_tools nodes`** para verificar que os nós estão funcionando como membros no cluster (mostrados como "M" na coluna de estado "Sts"). Por exemplo:

```
[root@example-01 ~]# cman_tool nodes
Node  Sts   Inc   Joined               Name
   1   M    548   2010-09-28 10:52:21  node-01.example.com
   2   M    548   2010-09-28 10:52:21  node-02.example.com
   3   M    544   2010-09-28 10:52:21  node-03.example.com
```

10. Se o cluster estiver rodando, vá para Seção 7.3, “Configuração de Fence”.

7.2.1. Exemplos de Configurações Básicas

Exemplo 7.1, “**`cluster.conf`** Exemplo: Configuração Básica” e Exemplo 7.2, “**`cluster.conf`** Exemplo: Configuração de Nós Básica” (para um cluster de dois nós) cada fornece um exemplo muito básico de arquivo de configuração como um ponto de início. Procedimentos subsequentes neste capítulos fornecem informações sobre configurar o fence dos serviços de Alta Disponibilidade.

Exemplo 7.1. `cluster.conf` Exemplo: Configuração Básica

```
<cluster name="mycluster" config_version="2">
   <clusternodes>
     <clusternode name="node-01.example.com" nodeid="1">
         <fence>
         </fence>
```

```
      </clusternode>
      <clusternode name="node-02.example.com" nodeid="2">
          <fence>
          </fence>
      </clusternode>
      <clusternode name="node-03.example.com" nodeid="3">
          <fence>
          </fence>
      </clusternode>
    </clusternodes>
    <fencedevices>
    </fencedevices>
    <rm>
    </rm>
</cluster>
```

Exemplo 7.2. `cluster.conf` Exemplo: Configuração de Nós Básica

```
<cluster name="mycluster" config_version="2">
    <cman two_node="1" expected_votes="1"/>
    <clusternodes>
      <clusternode name="node-01.example.com" nodeid="1">
          <fence>
          </fence>
      </clusternode>
      <clusternode name="node-02.example.com" nodeid="2">
          <fence>
          </fence>
      </clusternode>
    </clusternodes>
    <fencedevices>
    </fencedevices>
    <rm>
    </rm>
</cluster>
```

7.2.2. O valor concensus para o `totem` em um cluster de dois nós.

Quando você cria um cluster de dois nós e você não pretende adicionar nós adicionais ao cluster depois, então você pode omitir o valor **consensus** na tag `totem` no arquivo `cluster.conf` para que o valor **consensus** seja calculado automaticamente. Quando o valor **consensus** é calculadoautomaticamente, as seguintes regras são usadas:

- Se existem dois nós ou menos, o valor **consensus** será (token * 0.2), com um teto de 2000 milisegundos e um piso de 200 milisegundos.
- Se existem três ou mais nós, o valor **consensus** será (token + 2000 milisegundos)

Se você permitir que o utilitário **cman** configure a expiração de seu consensus desta maneira e então

mudar em um outro momento de dois para três nós (ou mais) isso irá requerer a reinicialização de um cluster, já que a expiração do consensus precisará mudar para um valor maior baseado na expiração do token.

Se você estiver configurando um cluster de dois nós e pretende atualizar no futuro para um número maior de dois nós, você pode sobrescrever a expiração consensus para que então uma reinicialização de cluster não seja necessária quando mover de dois para três (ou mais) nós. Isto pode ser feito no **cluster.conf** conforme se segue:

```
<totem token="X" consensus="X + 2000" />
```

Note que o análise da configuração não calcula X + 2000 automaticamente. Um número inteiro deve ser usado em vez de uma equação.

A vantagem de usar uma expiração de consensus otimizada para clusters de dois nós é que o período de failover geral é reduzido para o processo de dois nós, já que o consensus não é uma função de expiração de token.

Note que para a auto detecção de dois nós no **cman**, o número de nós físicos é o que importa e não a presença da directiva **two_node=1** no arquivo **cluster.conf**.

7.3. CONFIGURAÇÃO DE FENCE

A configuração de fence consiste de (a) especificar um ou mais dispositivos fence em um cluster e (b) especificar um ou mais métodos fence para cada nó (usando um dispositivo fence ou dispositivos fence especificados).

Baseados no tipo de dispositivos fence e métodos fence requeridos para sua configuração, configure o **cluster.conf** conforme a seguir:

1. Na seção **fencedevices**, especifique cada dispositivo fence, usando um elemento **fencedevice** e atributos dependentes de dispositivos fence. O Exemplo 7.3, “O Dispositivo Fence APC Adicionado ao **cluster.conf**” exibe um exemplo de arquivo de configuração com um dispositivo fence APC adicionado a ele.

2. Na seção **clusternodes**, dentro do elemento **fence** de cada seção do **clusternode**, especifique cada método fence do nó. Especifique o nome do método fence, usando o atributo **method**, **name**. Especifique o dispositivo fence para cada método fence, usando o elemento **device** e seus atributos, **name** e parâmetros específicos de dispositivos fence. O Exemplo 7.4, “Métodos Fence adicionados ao **cluster.conf**” mostra um exemplo de um método fence com um dispositivo fence para nó no cluster.

3. Para métodos fence sem energia (que é SAN/fence de armazenamento) na seção **clusternodes**, adicione uma seção **unfence**. Isto garante que um nó com fence não seja rehabilitado até que o nó seja reinicializado. Para mais informações sobre tirar um fence (unfencing) de um nó, consulte a página man **fence_node**(8).

   A seção **unfence** não contém seções **method** como a seção **fence** possui. Ela contém referências **device** diretamente, que espelha as seções dos dispositivos correspondentes para o **fence**, com a adição notável da ação explícita (**action**) do "on" ou "enable". O mesmo **fencedevice** é referenciado por ambas linhas de **dispositivo fence** e **unfence** e os mesmos argumentos por nó devem ser repetidos.

Especifique o atributo **action** como "on" ou "enable" habilita o nó quando reinicializado. O Exemplo 7.4, "Métodos Fence adicionados ao **cluster.conf**" e Exemplo 7.5, "**cluster.conf**: Múltiplos Métodos Fence por Nó" inclui exemplos dos elementos **unfence** e atribuídos.

Para mais informações sobre o **unfence** consulte a página man **fence_node**.

4. Atualize o atributo **config_version** incrementando seu valor (por exemplo, mudando de **config_version="2"** para **config_version="3">**).

5. Salve o **/etc/cluster/cluster.conf**.

6. **(Opcional)** Valide o arquivo atualizado contra o esquema de cluster (**cluster.rng**) rodando o comando **ccs_config_validate**. Por exemplo:

```
[root@example-01 ~]# ccs_config_validate
Configuration validates
```

7. Rode o comando **cman_tool version -r** para propagar a configuração e o resto dos nós do cluster. Isto também executará validação adicional. É necessário que o **ricci** esteja rodando em cada nó no cluster para ser capaz de propagar a informação de configuração de cluster atualizada.

8. Verifique que o arquivo de configuração atualizado foi propagado.

9. Consulte a Seção 7.4, "Configurar Domínios de Failover".

Se requerido, você pode fazer configurações complexas com múltiplos métodos fence por nó e com múltiplos dispositivos fence por método fence. Quando especificar múltiplos métodos fence por nó, se o fence falhar usando o primeiro método, **fenced**, o daemon fence tenta o próximo método e continua a rotacionar os métodos até que um tenha sucesso.

As vezes, fazer o fence em um nó requer que se desabilite dois caminhos E/S ou duas portas de energia. Isto é feito especificando dois ou mais dispositivos dentro de um método fence. O **fence** roda o agente fence para cada linha de dispositivo fence; todos devem ser bem sucedidos para o fence ser considerado bem sucedido.

Mais configurações complexas são exibidas em "Exemplos de Configuração Fence".

Você pode encontrar mais informações sobre configurar dispositivos fence específicos a partir da página man de um agente de dispositivo fence (por exemplo, a página man do **fence_apc**). Além disso, você pode obter mais informações sobre parâmetros fence no Apêndice A, *Parâmetros de Dispositos Fence*, os agentes fence em **/usr/sbin/**, o esquema de cluster em **/usr/share/cluster/cluster.rng**, e o esquema anotado em **/usr/share/doc/cman-X.Y.ZZ/cluster_conf.html** (por exemplo, **/usr/share/doc/cman-3.0.12/cluster_conf.html**).

7.3.1. Exemplos de Configuração Fence

Os seguintes exemplos mostram uma configuração simples com um método fence por nó e um dispositivo fence por método fence:

- Exemplo 7.3, "O Dispositivo Fence APC Adicionado ao **cluster.conf**"
- Exemplo 7.4, "Métodos Fence adicionados ao **cluster.conf**"

Os seguintes exemplos mostram configurações mais complexas:

- Exemplo 7.5, "**cluster.conf**: Múltiplos Métodos Fence por Nó"
- Exemplo 7.6, "**cluster.conf**: Fencing, Múltiplas Portas Multipath"
- Exemplo 7.7, "**cluster.conf**: Nós Fence com Duas Fontes de Energia"

NOTA

Os exemplos nesta seção não são profundos; onde podem haver outras maneiras para configurar o fence dependendo de seus requerimentos.

Exemplo 7.3. O Dispositivo Fence APC Adicionado ao cluster.conf

```
<cluster name="mycluster" config_version="3">
   <clusternodes>
     <clusternode name="node-01.example.com" nodeid="1">
         <fence>
         </fence>
     </clusternode>
     <clusternode name="node-02.example.com" nodeid="2">
         <fence>
         </fence>
     </clusternode>
     <clusternode name="node-03.example.com" nodeid="3">
         <fence>
         </fence>
     </clusternode>
   </clusternodes>
   <fencedevices>
         <fencedevice agent="fence_apc" ipaddr="apc_ip_example"
login="login_example" name="apc" passwd="password_example"/>
   </fencedevices>
   <rm>
   </rm>
</cluster>
```

Neste exemplo, um dispositivo fence (**fencedevice**) foi adicionado ao elemento **fencedevices**, especificando o agente fence (**agent**) como o **fence_apc**, o endereço de IP (**ipaddr**) como **apc_ip_example**, o login (**login**) como **login_example**, o nome do dispositivo fence (**name**) como **apc**, e a senha (**passwd**) como **password_example**.

Exemplo 7.4. Métodos Fence adicionados ao cluster.conf

```
<cluster name="mycluster" config_version="3">
   <clusternodes>
     <clusternode name="node-01.example.com" nodeid="1">
         <fence>
            <method name="APC">
```

```
                <device name="apc" port="1"/>
               </method>
           </fence>
      </clusternode>
      <clusternode name="node-02.example.com" nodeid="2">
           <fence>
              <method name="APC">
                <device name="apc" port="2"/>
              </method>
           </fence>
      </clusternode>
      <clusternode name="node-03.example.com" nodeid="3">
           <fence>
              <method name="APC">
                <device name="apc" port="3"/>
              </method>
           </fence>
      </clusternode>
   </clusternodes>
   <fencedevices>
         <fencedevice agent="fence_apc" ipaddr="apc_ip_example"
login="login_example" name="apc" passwd="password_example"/>
   </fencedevices>
   <rm>
   </rm>
</cluster>
```

Neste exemplo, um método fence (**method**) foi adicionado a cada nó. O nome do método fence (**name**) para cada nó é **APC**. O dispositivo (**device**) para o método fence em cada nó especifica o nome (**name**) como **apc** e um único switch de energia APC porta número (**port**) para cada nó. Por exemplo, a porta número para o node-01.example.com é **1** (**port="1"**). O nome do dispositivo para cada pontos de nó (**device name="apc"**) ao dispositivo fence pelo nome (**name**) do **apc** nesta linha do **fencedevices** element: **fencedevice agent="fence_apc" ipaddr="apc_ip_example" login="login_example" name="apc" passwd="password_example"/**.

Exemplo 7.5. `cluster.conf`: Múltiplos Métodos Fence por Nó

```
<cluster name="mycluster" config_version="3">
   <clusternodes>
     <clusternode name="node-01.example.com" nodeid="1">
         <fence>
            <method name="APC">
              <device name="apc" port="1"/>
             </method>
            <method name="SAN">
       <device name="sanswitch1" port="11"/>
             </method>
         </fence>
         <unfence>
             <device name="sanswitch1" port="11" action="on"/>
         </unfence
```

```
      </clusternode>
      <clusternode name="node-02.example.com" nodeid="2">
          <fence>
             <method name="APC">
               <device name="apc" port="2"/>
             </method>
             <method name="SAN">
        <device name="sanswitch1" port="12"/>
              </method>
          </fence>
          <unfence>
              <device name="sanswitch1" port="12" action="on"/>
          </unfence
      </clusternode>
      <clusternode name="node-03.example.com" nodeid="3">
          <fence>
             <method name="APC">
               <device name="apc" port="3"/>
             </method>
             <method name="SAN">
        <device name="sanswitch1" port="13"/>
              </method>
          </fence>
          <unfence>
              <device name="sanswitch1" port="13" action="on"/>
          </unfence
      </clusternode>
   </clusternodes>
   <fencedevices>
        <fencedevice agent="fence_apc" ipaddr="apc_ip_example"
login="login_example" name="apc" passwd="password_example"/>
        <fencedevice agent="fence_sanbox2" ipaddr="san_ip_example"
login="login_example" name="sanswitch1" passwd="password_example"/>
   </fencedevices>
   <rm>
   </rm>
</cluster>
```

Exemplo 7.6. `cluster.conf`: Fencing, Múltiplas Portas Multipath

```
<cluster name="mycluster" config_version="3">
   <clusternodes>
      <clusternode name="node-01.example.com" nodeid="1">
          <fence>
             <method name="SAN-multi">
        <device name="sanswitch1" port="11"/>
        <device name="sanswitch2" port="11"/>
      </method>
          </fence>
          <unfence>
              <device name="sanswitch1" port="11" action="on"/>
```

```
            <device name="sanswitch2" port="11" action="on"/>
        </unfence>
    </clusternode>
    <clusternode name="node-02.example.com" nodeid="2">
        <fence>
           <method name="SAN-multi">
      <device name="sanswitch1" port="12"/>
      <device name="sanswitch2" port="12"/>
           </method>
        </fence>
        <unfence>
            <device name="sanswitch1" port="12" action="on"/>
            <device name="sanswitch2" port="12" action="on"/>
        </unfence
    </clusternode>
    <clusternode name="node-03.example.com" nodeid="3">
        <fence>
           <method name="SAN-multi">
      <device name="sanswitch1" port="13"/>
      <device name="sanswitch2" port="13"/>
           </method>
        </fence>
        <unfence>
            <device name="sanswitch1" port="13" action="on"/>
            <device name="sanswitch2" port="13" action="on"/>
        </unfence
    </clusternode>
  </clusternodes>
  <fencedevices>
       <fencedevice agent="fence_sanbox2" ipaddr="san_ip_example"
login="login_example" name="sanswitch1" passwd="password_example"/>
       <fencedevice agent="fence_sanbox2" ipaddr="san_ip_example"
login="login_example" name="sanswitch2" passwd="password_example"/>
  </fencedevices>
  <rm>
  </rm>
</cluster>
```

Exemplo 7.7. `cluster.conf`: Nós Fence com Duas Fontes de Energia

```
<cluster name="mycluster" config_version="3">
  <clusternodes>
    <clusternode name="node-01.example.com" nodeid="1">
        <fence>
           <method name="APC-dual">
             <device name="apc1" port="1"action="off"/>
             <device name="apc2" port="1"action="off"/>
             <device name="apc1" port="1"action="on"/>
             <device name="apc2" port="1"action="on"/>
           </method>
        </fence>
```

```
      </clusternode>
      <clusternode name="node-02.example.com" nodeid="2">
          <fence>
             <method name="APC-dual">
               <device name="apc1" port="2"action="off"/>
               <device name="apc2" port="2"action="off"/>
               <device name="apc1" port="2"action="on"/>
               <device name="apc2" port="2"action="on"/>
             </method>
          </fence>
      </clusternode>
      <clusternode name="node-03.example.com" nodeid="3">
          <fence>
             <method name="APC-dual">
               <device name="apc1" port="3"action="off"/>
               <device name="apc2" port="3"action="off"/>
               <device name="apc1" port="3"action="on"/>
               <device name="apc2" port="3"action="on"/>
             </method>
          </fence>
      </clusternode>
    </clusternodes>
    <fencedevices>
        <fencedevice agent="fence_apc" ipaddr="apc_ip_example"
login="login_example" name="apc1" passwd="password_example"/>
        <fencedevice agent="fence_apc" ipaddr="apc_ip_example"
login="login_example" name="apc2" passwd="password_example"/>
    </fencedevices>
    <rm>
    </rm>
</cluster>
```

Quando usar switches de energia para nós fence com duas fontes de energia, os agentes devem ser configurados para desligar ambas portas de energia antes de restaurar energia em cada das portas. O padrão de comportamento on-off do agente pode resultar na porta nunca sendo totalmente desabilitada no nó.

7.4. CONFIGURAR DOMÍNIOS DE FAILOVER

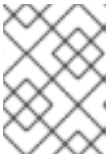
Um domínio de failover é um sub conjunto de nós do cluster que são elegíveis para rodar um serviço de cluster em um evento de falha de um nó. Um domínio failover pode ter as seguintes características:

- Irrestrito (Unrestricted) — Permite especificar que um sub conjunto de membros são preferidos mas que um serviço de cluster atribuído a este domínio possa rodar em qualquer membro disponível.
- Restringido (Restricted) — Permite restringir os membros que podem rodar um serviço de cluster em particular. Se nenhum dos membros em um domínio de failover estiverem disponíveis, o serviço de cluster não pode ser iniciado (tanto manualmente ou pelo software de cluster).
- Desordenado (Unordered) — Quando um serviço de cluster é atribuído a um domínio de failover desordenado, o membro no qual o serviço de cluster roda é escolhido a partir dos membros do

domínio de failover disponíveis sem prioridade de ordem.

- Ordenado (Ordered) — Permite especificar uma ordem de preferência entre os membros de um domínio de failover. Domínios de failover ordenados selecionam o nó com o número prioridade mais baixo primeiro. Onde o domínio failover com uma prioridade de número "1" significa a prioridade mais alta, e portanto é o nó mais preferido em um domínio de failover. Depois desse nó, o próximo nó preferido seria o nó com próximo número de prioridade e assim pode diante.

- Failback — Permite especificar se um serviço no domínio de failover deveria fazer um fail back no nó que estava originalmente rodando antes desse nó falhar. Configurar esta característica é útil em circunstâncias onde um nó falha repetidamente e é parte de um domínio de failover ordenado. Nesta circunstância, se o nó é o preferido no domínio de failover, é possível para um serviço fazer fail over e fail back repetidamente entre o nó preferido e o outro nó, causando um impacto severo no desempenho.

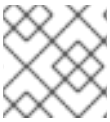


NOTA

A característica failback é aplicável somente se o failover ordenado é configurado.

NOTA

Alterar a configuração de um domínio de failover não possui efeito em serviços atualmente em execução.

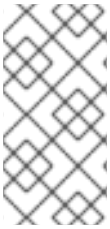


NOTA

Domínios failover *não* são requeridos para operação.

Por padrão, domínios failover são irrestritos e desordenados.

Em um cluster com diversos membros, usar um domínio failover restringido pode minimizar o trabalho de configurar o cluster para executar o serviço de cluster (tal como **`httpd`**), que requer que você defina identicamente a configuração em todos os membros que rodam o serviço de cluster. Em vez de definir o cluster inteiro para rodar o serviço de cluster, você pode definir somente os membros no domínio de failover restringidos que você associa com o serviço de cluster.



NOTA

Para configurar um membro preferido, você pode criar um domínio de failover irrestrito compreendendo de somente um membro do cluster. Isso faz que um serviço de cluster rode naquele membro de cluster primariamente (o membro preferido) mas permite que o serviço de cluster faça um fail over em qualquer um dos outros membros.

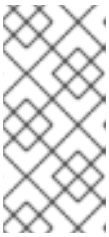
Para configurar um domínio de failover, use os seguintes procedimentos:

1. Abra o **`/etc/cluster/cluster.conf`** em qualquer nó do cluster.

2. Adicione a seguinte estrutura de seção dentro do elemento **`rm`** para cada domínio de failover a ser usado:

```
        <failoverdomains>
```

```
            <failoverdomain name="" nofailback="" ordered=""
restricted="">
                <failoverdomainnode name="" priority=""/>
                <failoverdomainnode name="" priority=""/>
                <failoverdomainnode name="" priority=""/>
            </failoverdomain>
        </failoverdomains>
```



NOTA

O número de atributos de **failoverdomainnode** depende do número de nós no domínio failover. A estrutura da seção **failoverdomain** no texto anterior mostra três elementos **failoverdomainnode** (sem nomes de nós especificados), mostrando que existem três nós no domínio de failover.

3. Na seção **failoverdomain**, forneça os valores dos elementos e atributos. Para descriçoes dos elementos e atributos, consulte a seção *failoverdomain* do esquema de cluster anotados. O esquema de cluster anotado está disponível em **/usr/share/doc/cman-X.Y.ZZ/cluster_conf.html** (por exemplo **/usr/share/doc/cman-3.0.12/cluster_conf.html**) em qualquer dos nós no cluster. Para um exemplo de uma seção **failoverdomains**, consulte Exemplo 7.8, “Um Domínio de Failover Adicionado ao **cluster.conf**”.

4. Atualize o atributo **config_version** incrementando seu valor (por exemplo, mudando de **config_version="2"** para **config_version="3">**).

5. Salve o **/etc/cluster/cluster.conf**.

6. **(Opcional)** Valide o arquivo no esquema de cluster (**cluster.rng**) rodando o comando **ccs_config_validate**. Por exemplo:

```
[root@example-01 ~]# ccs_config_validate
Configuration validates
```

7. Rode o comando **cman_tool version -r** para propagar a configuração ao resto dos nós no cluster.

8. Vá para Seção 7.5, “Configurando Serviços de Alta Disponibilidade” .

O Exemplo 7.8, “Um Domínio de Failover Adicionado ao **cluster.conf**” mostra um exemplo de configuração com um domínio de failover ordenado e irrestrito.

Exemplo 7.8. Um Domínio de Failover Adicionado ao cluster.conf

```
<cluster name="mycluster" config_version="3">
   <clusternodes>
     <clusternode name="node-01.example.com" nodeid="1">
         <fence>
            <method name="APC">
              <device name="apc" port="1"/>
             </method>
```

```
            </fence>
        </clusternode>
        <clusternode name="node-02.example.com" nodeid="2">
            <fence>
               <method name="APC">
                 <device name="apc" port="2"/>
               </method>
            </fence>
        </clusternode>
        <clusternode name="node-03.example.com" nodeid="3">
            <fence>
               <method name="APC">
                 <device name="apc" port="3"/>
               </method>
            </fence>
        </clusternode>
    </clusternodes>
    <fencedevices>
            <fencedevice agent="fence_apc" ipaddr="apc_ip_example"
login="login_example" name="apc" passwd="password_example"/>
    </fencedevices>
    <rm>
        <failoverdomains>
            <failoverdomain name="example_pri" nofailback="0"
ordered="1" restricted="0">
                <failoverdomainnode name="node-01.example.com"
priority="1"/>
                <failoverdomainnode name="node-02.example.com"
priority="2"/>
                <failoverdomainnode name="node-03.example.com"
priority="3"/>
            </failoverdomain>
        </failoverdomains>
    </rm>
</cluster>
```

A seção **failoverdomains** contém uma seção **failoverdomain** para cada domínio de failover no cluster. Este exemplo possui um domínio de failover. Na linha **failoverdomain** , o nome (**name**) é especificado como **example_pri**. Além disso, ele não especifica failback (**failback="0"**), este failover é ordenado (**ordered="1"**), e que o domínio failover é irrestrito (**restricted="0"**).

7.5. CONFIGURANDO SERVIÇOS DE ALTA DISPONIBILIDADE

Configurando os serviços de Alta Disponibilidade consiste em configurar recursos e atribui-los a serviços.

As seguintes seções descrevem como editar o **/etc/cluster/cluster.conf** para adicionar recursos e serviços.

- Seção 7.5.1, “Adicionando Recursos de Cluster”
- Seção 7.5.2, “Adicionar um Serviço de Cluster ao Cluster”

IMPORTANTE

Poderão haver uma grande variedade de configurações possíveis com os recursos e serviços de Alta Disponibilidade. Para um melhor entendimento sobre parâmetros de recursos e comportamento de recursos, consulte o Apêndice B, *Parâmetros dos Recursos de Alta Disponibilidade* e o Apêndice C, *Comportamento do Recurso de Alta Disponibilidade*. Para um desempenho otimizado e para garantir que sua configuração possa ser suportada, contacte um representante autorizado Red Hat.

7.5.1. Adicionando Recursos de Cluster

Você pode configurar dois tipos de recursos:

- Global — Recursos que estão disponíveis a qualquer serviço no cluster. Eles são configurados na seção **resources** do arquivo de configuração (dentro do elemento **rm**).
- Serviço especifico (Service-specific) — Recursos que estão disponíveis somente a um serviço. Eles são configurados em cada seção de **service** do arquivo de configuração (dentro do elemento **rm**).

Esta seção descreve como adicionar um recurso global. Para procedimentos sobre configurar serviços específicos, consulte a Seção 7.5.2, “Adicionar um Serviço de Cluster ao Cluster” .

Para adicionar um recurso de cluster global, siga os passos nesta seção.

1. Abra o **/etc/cluster/cluster.conf** em qualquer nó do cluster.
2. Adicione a seção **recursos** dentro do elemento **rm**. Por exemplo:

```
<rm>
    <resources>

    </resources>
</rm>
```

3. Preencha-o com recursos de acordo com os serviços que você quer criar. Por exemplo, aqui estão os recursos que estão para serem usados em um serviço Apache. Eles consistem de um recurso de sistema de arquivo (**fs**) e um recurso Apache (**apache**).

```
    <rm>
        <resources>
           <fs name="web_fs" device="/dev/sdd2"
mountpoint="/var/www" fstype="ext3"/>
           <ip address="127.143.131.100" monitor_link="on"
sleeptime="10"/>
           <apache config_file="conf/httpd.conf"
name="example_server" server_root="/etc/httpd" shutdown_wait="0"/>
        </resources>
    </rm>
```

O Exemplo 7.9, “Arquivo **cluster.conf** com Recursos Adicionados” exibe um exemplo de um arquivo **cluster.conf** com a seção **resources** adicionada.

4. Atualize o atributo **config_version** incrementando seu valor (por exemplo, mudando de **config_version="2"** para **config_version="3"**).

5. Salve o **/etc/cluster/cluster.conf**.

6. **(Opcional)** Valide o arquivo no esquema de cluster (**cluster.rng**) rodando o comando **ccs_config_validate**. Por exemplo:

```
[root@example-01 ~]# ccs_config_validate
Configuration validates
```

7. Rode o comando **cman_tool version -r** para propagar a configuração ao resto dos nós no cluster.

8. Verifique que o arquivo de configuração atualizado foi propagado.

9. Vá para Seção 7.5.2, “Adicionar um Serviço de Cluster ao Cluster” .

Exemplo 7.9. Arquivo cluster.conf com Recursos Adicionados

```
<cluster name="mycluster" config_version="3">
   <clusternodes>
     <clusternode name="node-01.example.com" nodeid="1">
         <fence>
            <method name="APC">
              <device name="apc" port="1"/>
             </method>
         </fence>
     </clusternode>
     <clusternode name="node-02.example.com" nodeid="2">
         <fence>
            <method name="APC">
              <device name="apc" port="2"/>
            </method>
         </fence>
     </clusternode>
     <clusternode name="node-03.example.com" nodeid="3">
         <fence>
            <method name="APC">
              <device name="apc" port="3"/>
            </method>
         </fence>
     </clusternode>
   </clusternodes>
   <fencedevices>
         <fencedevice agent="fence_apc" ipaddr="apc_ip_example"
login="login_example" name="apc" passwd="password_example"/>
   </fencedevices>
   <rm>
       <failoverdomains>
           <failoverdomain name="example_pri" nofailback="0"
```

```
ordered="1" restricted="0">
                <failoverdomainnode name="node-01.example.com"
priority="1"/>
                <failoverdomainnode name="node-02.example.com"
priority="2"/>
                <failoverdomainnode name="node-03.example.com"
priority="3"/>
            </failoverdomain>
        </failoverdomains>
        <resources>
            <fs name="web_fs" device="/dev/sdd2" mountpoint="/var/www"
fstype="ext3"/>
            <ip address="127.143.131.100" monitor_link="yes"
sleeptime="10"/>
            <apache config_file="conf/httpd.conf" name="example_server"
server_root="/etc/httpd" shutdown_wait="0"/>
         </resources>

    </rm>
</cluster>
```

7.5.2. Adicionar um Serviço de Cluster ao Cluster

Para adicionar um serviço de cluster ao cluster, siga os passos desta seção.

1. Abra o **/etc/cluster/cluster.conf** em qualquer nó do cluster.
2. Adicione uma seção **service** dentro do elemento **rm** para cada serviço. Por exemplo:

```
    <rm>
        <service autostart="1" domain="" exclusive="0" name=""
recovery="restart">

        </service>
    </rm>
```

3. Configure os seguintes parâmetros (atributos) no elemento **service**:
   - **autostart** — Especifica se faz inicialização automática do serviço quando o cluster inicia. Use '1' para ativar e '0' para desativar; o padrão é ativado.
   - **domain** — Especifica um domínio de failover (se requerido).
   - **exclusive** — Especifica uma política onde o serviço somente roda em nós que não possuem outros serviços rodando neles.
   - **recovery** — Especifica uma política de recuperação para o serviço. As opções são realocar, reiniciar ou desabilitar o serviço.
4. Dependendo do tipo de recursos que você quer usar, preencha o serviço com recursos globais ou serviços especificos.

Por exemplo, aqui está um serviço Apache que usa recursos globais:

```
    <rm>
        <resources>
                <fs name="web_fs" device="/dev/sdd2"
mountpoint="/var/www" fstype="ext3"/>
                <ip address="127.143.131.100" monitor_link="yes"
sleeptime="10"/>
                <apache config_file="conf/httpd.conf"
name="example_server" server_root="/etc/httpd" shutdown_wait="0"/>
        </resources>
        <service autostart="1" domain="example_pri" exclusive="0"
name="example_apache" recovery="relocate">
                <fs ref="web_fs"/>
                <ip ref="127.143.131.100"/>
                <apache ref="example_server"/>
        </service>
    </rm>
```

Por exemplo, aqui está um serviço Apache que usa recursos de serviços especificos:

```
    <rm>
        <service autostart="0" domain="example_pri" exclusive="0"
name="example_apache2" recovery="relocate">
                <fs name="web_fs2" device="/dev/sdd3"
mountpoint="/var/www2" fstype="ext3"/>
                <ip address="127.143.131.101" monitor_link="yes"
sleeptime="10"/>
                <apache config_file="conf/httpd.conf"
name="example_server2" server_root="/etc/httpd" shutdown_wait="0"/>
        </service>
    </rm>
```

O Exemplo 7.10, “**cluster.conf** com Serviços Adicionados. Um usando Recursos Globais e Um usando Recursos de Serviços Especificos ” exibe um exemplo de arquivo **cluster.conf** com dois serviços:

- **example_apache** – Este serviço usa os recursos globais **web_fs**, **127.143.131.100**, e **example_server**.
- **example_apache2** – Este serviço usa os recursos de serviços especificos **web_fs2**, **127.143.131.101**, e **example_server2**.

5. Atualize o atributo **config_version** incrementando seu valor (por exemplo, mudando de **config_version="2"** para **config_version="3">**).
6. Salve o **/etc/cluster/cluster.conf**.
7. **(Opcional)** Valide o arquivo atualizado contra o esquema de cluster (**cluster.rng**) rodando o comando **ccs_config_validate**. Por exemplo:

```
[root@example-01 ~]# ccs_config_validate
Configuration validates
```

8. Rode o comando **`cman_tool version -r`** para propagar a configuração ao resto dos nós no cluster.

9. Verifique que o arquivo de configuração atualizado foi propagado.

10. Siga para Seção 7.8, “Verificando uma Configuração”.

Exemplo 7.10. `cluster.conf` com Serviços Adicionados. Um usando Recursos Globais e Um usando Recursos de Serviços Especificos

```
<cluster name="mycluster" config_version="3">
   <clusternodes>
     <clusternode name="node-01.example.com" nodeid="1">
         <fence>
            <method name="APC">
              <device name="apc" port="1"/>
             </method>
         </fence>
     </clusternode>
     <clusternode name="node-02.example.com" nodeid="2">
         <fence>
            <method name="APC">
              <device name="apc" port="2"/>
            </method>
         </fence>
     </clusternode>
     <clusternode name="node-03.example.com" nodeid="3">
         <fence>
            <method name="APC">
              <device name="apc" port="3"/>
            </method>
         </fence>
     </clusternode>
   </clusternodes>
   <fencedevices>
         <fencedevice agent="fence_apc" ipaddr="apc_ip_example"
login="login_example" name="apc" passwd="password_example"/>
   </fencedevices>
   <rm>
       <failoverdomains>
           <failoverdomain name="example_pri" nofailback="0"
ordered="1" restricted="0">
               <failoverdomainnode name="node-01.example.com"
priority="1"/>
               <failoverdomainnode name="node-02.example.com"
priority="2"/>
               <failoverdomainnode name="node-03.example.com"
priority="3"/>
           </failoverdomain>
       </failoverdomains>
       <resources>
```

```
            <fs name="web_fs" device="/dev/sdd2" mountpoint="/var/www"
fstype="ext3"/>
            <ip address="127.143.131.100" monitor_link="yes"
sleeptime="10"/>
            <apache config_file="conf/httpd.conf" name="example_server"
server_root="/etc/httpd" shutdown_wait="0"/>
        </resources>
        <service autostart="1" domain="example_pri" exclusive="0"
name="example_apache" recovery="relocate">
            <fs ref="web_fs"/>
            <ip ref="127.143.131.100"/>
            <apache ref="example_server"/>
        </service>
        <service autostart="0" domain="example_pri" exclusive="0"
name="example_apache2" recovery="relocate">
            <fs name="web_fs2" device="/dev/sdd3" mountpoint="/var/www2"
fstype="ext3"/>
            <ip address="127.143.131.101" monitor_link="yes"
sleeptime="10"/>
            <apache config_file="conf/httpd.conf" name="example_server2"
server_root="/etc/httpd" shutdown_wait="0"/>
        </service>
    </rm>
</cluster>
```

7.6. CONFIGURANDO O PROTOCOLO DE ANEL REDUNDANTE

Desde o Red Hat Enterprise Linux 6.4, o Red Hat High Availability Add-On suporta a configuração do protocolo de anel redundante.

Ao configurar um sistema para usar o protocolo de anel redundante, você precisa levar as seguintes considerações em conta:

- Não especifique mais do que dois anéis.
- Cada anel deve utilizar o mesmo protocolo; não confunda IPv4 e IPv6.
- Caso seja necessário, você poderá especificar um endereço de multicast manualmente para o segundo anel. Se você especificar um endereço multicast para o segundo anel, tanto o endereço de multicast alternado quanto a porta alternada devem ser diferentes do endereço do multicast para o primeiro anel. Se você não especificar um endereço e multicast alternado, o sistema irá utilizar automaticamente um endereço de multicast diferente para o segundo anel.

  Se você especificar uma porta alternada, os números de porta do primeiro anel e o segundo anel devem se diferir por ao menos duas, pois o sistema usa a porta e porta -1 para realizar operações.
- Não use duas interfaces diferentes no mesmo subnet.
- Em geral, é uma boa prática configurar o protocolo do anel redundante em dois NICs diferentes e em dois interruptores, no caso de um NIC ou um interruptor falhar.

- Não use o comando **ifdown** ou o comando **service network stop** para simular uma falha de rede. Isto destrói todo o cluster e requer que você reinicie todos os nós no cluster para recuperar.
- Não use o **NetworkManager**, pois ele irá executar o comando **ifdown** se o cabo não estiver desligado.
- Quando um nó de um NIC falha, todo o anel é marcado como falho.
- Na intervenção manual, é necessário recuperar um anel falho. Para recuperar, você precisa somente reparar a razão original para a falha, tal como um NIC ou interruptor falho.

Para especificar uma segunda interface de rede para usar para protoloco e anel redundadnte, você adiciona um componente **altname** à seção **clusternode** do arquivo de configuração **cluster.conf**. Quando especificar um **altname**, especifique um atributo de **name** para indicar um segundo nome de host ou um endereço IP para o nó.

O exemplo a seguir especifca o **clusternet-node1-eth2** como um nome alternativo para o nó de cluster **clusternet-node1-eth1**.

```
<cluster name="mycluster" config_version="3" >
  <logging debug="on"/>
  <clusternodes>
    <clusternode name="clusternet-node1-eth1" votes="1" nodeid="1">
      <fence>
        <method name="single">
          <device name="xvm" domain="clusternet-node1"/>
        </method>
      </fence>
      <altname name="clusternet-node1-eth2"/>
    </clusternode>
```

A seção **altname** dentro do bloco **clusternode** não é uma posição dependente. Ele pode vir antes ou depois da seção **fence**. Não especifique mais do que um componente **altname** para um nó de cluster ou o sistema irá falhar ao iniciar.

Opcionalmente, você pode especificar manualmente um endereço multicast, uma porta e um TTL para o segundo anel, incluindo um componente **altmulticast** na seção **cman** do arquivo de configuração **cluster.conf** . O componente **altmulticast** aceita um parâmetro **addr**, um **port**, e um **ttl**.

O exemplo a seguir exibe a seção **cman** de um arquivo de configuração de cluster que define um endereço de multicast, porta, e TTL para o segundo anel.

```
<cman>
   <multicast addr="239.192.99.73" port="666" ttl="2"/>
   <altmulticast addr="239.192.99.88" port="888" ttl="3"/>
</cman>
```

7.7. CONFIGURAÇÃO DAS OPÇÕES DE DEPURAÇÃO

Você pode habilitar a depuração em todos os daemons em um cluster, ou pode desabilitar para processamento de cluster específico.

Para habilitar a depuração em todos os daemons, adicione o seguite ao arquivo **/etc/cluster/cluster.conf**. Por padrão, o logging é direcionado para o arquivo **/var/log/cluster/*daemon*.log**.

```
<cluster config_version="7" name="rh6cluster">
  <logging debug="on"/>
   ...
</cluster>
```

Para habilitar a depuração para processos de cluster individuais, adicione as seguintes linhas ao arquivo **/etc/cluster/cluster.conf**. A configuração de logging per-daemon sobrescreve as configurações globais.

```
<cluster config_version="7" name="rh6cluster">
   ...
  <logging>
       <!-- turning on per-subsystem debug logging -->
       <logging_daemon name="corosync" debug="on" />
       <logging_daemon name="fenced" debug="on" />
       <logging_daemon name="qdiskd" debug="on" />
       <logging_daemon name="rgmanager" debug="on" />
       <logging_daemon name="dlm_controld" debug="on" />
       <logging_daemon name="gfs_controld" debug="on" />
  </logging>
   ...
</cluster>
```

Para uma lista de daemons de log para os quais você pode habilitar o logging assim como as opções de logging adicionais que você pode configurar para ambos logging global e per-daemon, consulte a página man **cluster.conf**(5).

7.8. VERIFICANDO UMA CONFIGURAÇÃO

Uma vez que você criou seu arquivo de configuração de cluster, verifique que ele está rodando corretamente realizando os seguintes passos:

1. Em cada nó, reinicie o software de cluster. Esta ação garante que quaisquer adições de configuração que estão marcadas somente no momento de inicialização são incluidas da configuração de execução. Você pode reiniciar o software de cluster rodando **service cman restart**. Por exemplo:

   ```
   [root@example-01 ~]# service cman restart
   Stopping cluster:
      Leaving fence domain...                                 [  OK
   ]
      Stopping gfs_controld...                                [  OK
   ]
   ```

```
   Stopping dlm_controld...                                [  OK
]
   Stopping fenced...                                      [  OK
]
   Stopping cman...                                        [  OK
]
   Waiting for corosync to shutdown:                       [  OK  ]
   Unloading kernel modules...                             [  OK
]
   Unmounting configfs...                                  [  OK
]
Starting cluster:
   Checking Network Manager...                             [  OK
]
   Global setup...                                         [  OK
]
   Loading kernel modules...                               [  OK
]
   Mounting configfs...                                    [  OK
]
   Starting cman...                                        [  OK
]
   Waiting for quorum...                                   [  OK
]
   Starting fenced...                                      [  OK
]
   Starting dlm_controld...                                [  OK
]
   Starting gfs_controld...                                [  OK
]
   Unfencing self...                                       [  OK
]
   Joining fence domain...                                 [  OK
]
```

2. Execute o **`service clvmd start`**, so o CLVM estiver sendo usado para criar volumes clusterizados. Por exemplo:

```
[root@example-01 ~]# service clvmd start
Activating VGs:                                            [  OK
]
```

3. Rode o **`service gfs2 start`**, se você estiver configurando o Red Hat GFS2. Por exemplo:

```
[root@example-01 ~]# service gfs2 start
Mounting GFS2 filesystem (/mnt/gfsA):                      [  OK  ]
Mounting GFS2 filesystem (/mnt/gfsB):                      [  OK  ]
```

4. Rode o **`service rgmanager start`**, se você estiver usando os serviços de Alta Disponibilidade. Por exemplo:

```
[root@example-01 ~]# service rgmanager start
Starting Cluster Service Manager:                          [  OK  ]
```

5. Em qualquer nó no cluster, rode o **cman_tools nodes** para verificar que os nós estão funcionando como membros no cluster (mostrados como "M" na coluna de estado "Sts"). Por exemplo:

```
[root@example-01 ~]# cman_tool nodes
Node  Sts   Inc   Joined               Name
   1   M    548   2010-09-28 10:52:21  node-01.example.com
   2   M    548   2010-09-28 10:52:21  node-02.example.com
   3   M    544   2010-09-28 10:52:21  node-03.example.com
```

6. Em qualquer nó, usando o utilitário **clustat**, verifique que os serviços de Alta Disponibilidade estão rodando conforme esperados. Além disso, o **clustat**, exibe o estado dos nós do cluster. Por exemplo:

```
[root@example-01 ~]#clustat
Cluster Status for mycluster @ Wed Nov 17 05:40:00 2010
Member Status: Quorate

 Member Name                             ID   Status
 ------ ----                             ---- ------
 node-03.example.com                         3 Online, rgmanager
 node-02.example.com                         2 Online, rgmanager
 node-01.example.com                         1 Online, Local,
rgmanager

 Service Name                   Owner (Last)
State
 ------- ----                   ----- ------                    --
---
 service:example_apache         node-01.example.com
started
 service:example_apache2        (none)
disabled
```

7. Se o cluster estiver rodando conforme esperado, você terminou de criar um arquivo de configuração. Você pode gerenciar o cluster com as ferramentas de linha de comando descritas no Capítulo 8, *Gerenciando o Complemento de Alta Disponibilidade da Red Hat com Ferramentas da Linha de Comando*.

CAPÍTULO 8. GERENCIANDO O COMPLEMENTO DE ALTA DISPONIBILIDADE DA RED HAT COM FERRAMENTAS DA LINHA DE COMANDO.

Este capítulo descreve várias tarefas administrativas para gerenciar o Complemento de Alta Disponibilidade da Red Hat e consiste das seguintes seções:

- Seção 8.1, “Iniciar e Parar o Software de Cluster”
- Seção 8.2, “Deletando ou Adicionando um Nó”
- Seção 8.3, “Gerenciando Serviços de Alta Disponibilidade”
- Seção 8.4, “Atualizando uma Configuração”

IMPORTANTE

Certifique-se que sua implantação do Complemento de Alta Disponibilidade atenda suas necessidades e possa ser suportada. Consulte um representante autorizado Red Hat para verificar suas configurações antes da implementação. Além disso, separe um tempo para testar as configurações para testar modos de falhas.

IMPORTANTE

Este capítulo referencia elementos e atributos **`cluster.conf`** comumente usados. Para uma lista compreensiva e a descrição dos elementos e atributos do **`cluster.conf`**, consulte o esquema de cluster em **`/usr/share/cluster/cluster.rng`** e o esquema anotado em **`/usr/share/doc/cman-X.Y.ZZ/cluster_conf.html`** (por exemplo **`/usr/share/doc/cman-3.0.12/cluster_conf.html`**).

IMPORTANTE

Certos procedimentos neste capítulo pedem pelo uso do comando **`cman_tool version -r`** para propagar a configuração do cluster através de um cluster. O uso deste comando requer que o **`ricci`** esteja em execução.

NOTA

Procedimentos neste capítulo, podem incluir comandos especificos para algumas das ferramentas da linha de comando listada no Apêndice E, *Resumo das Ferramentas da Linha de Comando*. Para mais informações sobre todos os comandos e variáveis, consulte a página man para cada ferramenta da linha de comando.

8.1. INICIAR E PARAR O SOFTWARE DE CLUSTER

Você pode iniciar ou parar um software de cluster em um nó de acordo com a Seção 8.1.1, “Iniciar o Software do Cluster” e Seção 8.1.2, “Parando um Software de Cluster” . Iniciar um software de cluster em um nó faz esse nó se juntar ao cluster; parando o software de cluster em um nó, o faz deixar o cluster.

8.1.1. Iniciar o Software do Cluster

Para iniciar o software do cluster em um nó, digite os seguintes comandos nesta ordem:

1. **`service cman start`**
2. **`service clvmd start`**, Se o CLVM foi usado para criar volumes clusterizados
3. **`service gfs2 start`**, Se você estiver usando o Red Hat GFS2
4. **`service rgmanager start`**, se você estiver usando os serviços de alta disponibilidade (HA) (**`rgmanager`**).

Por exemplo:

```
[root@example-01 ~]# service cman start
Starting cluster:
   Checking Network Manager...                             [  OK  ]
   Global setup...                                         [  OK  ]
   Loading kernel modules...                               [  OK  ]
   Mounting configfs...                                    [  OK  ]
   Starting cman...                                        [  OK  ]
   Waiting for quorum...                                   [  OK  ]
   Starting fenced...                                      [  OK  ]
   Starting dlm_controld...                                [  OK  ]
   Starting gfs_controld...                                [  OK  ]
   Unfencing self...                                       [  OK  ]
   Joining fence domain...                                 [  OK  ]
[root@example-01 ~]# service clvmd start
Starting clvmd:                                            [  OK  ]
Activating VG(s):   2 logical volume(s) in volume group "vg_example" now
active
                                                           [  OK  ]
[root@example-01 ~]# service gfs2 start
Mounting GFS2 filesystem (/mnt/gfsA):                      [  OK  ]
Mounting GFS2 filesystem (/mnt/gfsB):                      [  OK  ]
[root@example-01 ~]# service rgmanager start
Starting Cluster Service Manager:                          [  OK  ]
[root@example-01 ~]#
```

8.1.2. Parando um Software de Cluster

Para parar o software de cluster em um nó, digite os seguintes comandos nesta ordem:

1. **`service rgmanager stop`**, se você estiver usando os serviços de alta disponibilidae (HA) (**`rgmanager`**).
2. **`service gfs2 stop`**, se você estiver usando o Red Hat GFS2
3. **`umount -at gfs2`**, se você estiver usando o Red Hat GFS2 em conjunto com o **`rgmanager`**, para certificar que quaisquer arquivos GFS2 montados durante a inicialização do **`rgmanager`** (mas não desmontados durante o desligamento) foram também desmontados.
4. **`service clvmd stop`**, se o CLVM foi usado para criar volumes clusterizados

5. **service cman stop**

Por exemplo:

```
[root@example-01 ~]# service rgmanager stop
Stopping Cluster Service Manager:                          [  OK  ]
[root@example-01 ~]# service gfs2 stop
Unmounting GFS2 filesystem (/mnt/gfsA):                    [  OK  ]
Unmounting GFS2 filesystem (/mnt/gfsB):                    [  OK  ]
[root@example-01 ~]# umount -at gfs2
[root@example-01 ~]# service clvmd stop
Signaling clvmd to exit                                    [  OK  ]
clvmd terminated                                           [  OK  ]
[root@example-01 ~]# service cman stop
Stopping cluster:
   Leaving fence domain...                                 [  OK  ]
   Stopping gfs_controld...                                [  OK  ]
   Stopping dlm_controld...                                [  OK  ]
   Stopping fenced...                                      [  OK  ]
   Stopping cman...                                        [  OK  ]
   Waiting for corosync to shutdown:                       [  OK  ]
   Unloading kernel modules...                             [  OK  ]
   Unmounting configfs...                                  [  OK  ]
[root@example-01 ~]#
```

NOTA

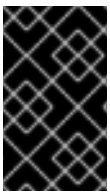
Parando o software de cluster em um nó faz que seus serviços de Alta Disponibilidade (HA) fazerem um fail over para outro nó. Como uma alternativa a isso, considere em realocar ou migrar os serviços HA para outro nó antes de parar o software de cluster. Para informações sobre gerenciar serviços HA, consulte a Seção 8.3, “Gerenciando Serviços de Alta Disponibilidade”.

8.2. DELETANDO OU ADICIONANDO UM NÓ

Esta seção descreve como excluir um nó de um cluster e adicionar um nó ao cluster. Você pode deletar um nó de um cluster de acordo com a Seção 8.2.1, “Deletar um Nó de um Cluster” ; você pode adicionar um nó a um cluster de acordo com a Seção 8.2.2, “Adicionando um Nó ao um Cluster” .

8.2.1. Deletar um Nó de um Cluster

Excluir um nó de um cluster consiste em fechar o software do cluster no nó a ser excluído e atualizar a configuração do cluster para refletir a mudança.



IMPORTANTE

Se deletar um nó do cluster faz uma mudança de um número maior de dois nós para dois nós, você deve reiniciar o software de cluster em cada nó depois de adicionar o arquivo de configuração do cluster.

Para excluir um nó de um cluster, realize os seguintes passos:

1. Em qualquer nó, use o utilitário **clusvcadm** para realocar, migrar ou parar cada serviço de Alta

Disponibilidade em execução no nó que está sendo excluído do cluster. Para informações sobre usar o **clusvcadm**, consulte a Seção 8.3, "Gerenciando Serviços de Alta Disponibilidade".

2. No nó a ser deletado do cluster, pare o software de cluster de acordo com a Seção 8.1.2, "Parando um Software de Cluster". Por exemplo:

```
[root@example-01 ~]# service rgmanager stop
Stopping Cluster Service Manager:                          [  OK  ]
[root@example-01 ~]# service gfs2 stop
Unmounting GFS2 filesystem (/mnt/gfsA):                    [  OK  ]
Unmounting GFS2 filesystem (/mnt/gfsB):                    [  OK  ]
[root@example-01 ~]# service clvmd stop
Signaling clvmd to exit                                    [  OK
]
clvmd terminated                                           [  OK
]
[root@example-01 ~]# service cman stop
Stopping cluster:
   Leaving fence domain...                                 [  OK
]
   Stopping gfs_controld...                                [  OK
]
   Stopping dlm_controld...                                [  OK
]
   Stopping fenced...                                      [  OK
]
   Stopping cman...                                        [  OK
]
   Waiting for corosync to shutdown:                       [  OK  ]
   Unloading kernel modules...                             [  OK
]
   Unmounting configfs...                                  [  OK
]
[root@example-01 ~]#
```

3. Em qualquer nó no cluster, edite o **/etc/cluster/cluster.conf** para remover a seção do nó que será deletado. Por exemplo, no Exemplo 8.1, "Configuração de Cluster de Três Nós", se o node-03.example.com é suposto a ser removido, então delete a seção **clusternode** para esse nó. Se remover um nó (ou nós) fizer que o cluster seja um cluster de dois nós, você pode adicionar a seguinte linha ao arquivo de configuração para permitir que um nó único mantenha quorum (por exemplo, se um nó falhar):

   <cman two_node="1" expected_votes="1"/>

   Consulte a Seção 8.2.3, "Exemplos de Configurações de Três e Dois Nós" para comparação entre uma configuração de três nós e uma de dois nós.

4. Atualize o atributo **config_version** incrementando seu valor (por exemplo, mudando de **config_version="2"** para **config_version="3">**).

5. Salve o **/etc/cluster/cluster.conf**.

6. **(Opcional)** Valide o arquivo atualizado contra o esquema de cluster (**cluster.rng**) rodando o comando **ccs_config_validate**. Por exemplo:

```
[root@example-01 ~]# ccs_config_validate
Configuration validates
```

7. Rode o comando **`cman_tool version -r`** para propagar a configuração ao resto dos nós no cluster.

8. Verifique que o arquivo de configuração atualizado foi propagado.

9. Se a contagem de nós foi alterada de um número maior de dois nós para dois nós, você deve reinicializar o software do cluster conforme se segue:

   1. Em cada nó, pare o software do cluster de acordo com a Seção 8.1.2, “Parando um Software de Cluster”. Por exemplo:

```
[root@example-01 ~]# service rgmanager stop
Stopping Cluster Service Manager:                          [  OK
]
[root@example-01 ~]# service gfs2 stop
Unmounting GFS2 filesystem (/mnt/gfsA):                    [  OK
]
Unmounting GFS2 filesystem (/mnt/gfsB):                    [  OK
]
[root@example-01 ~]# service clvmd stop
Signaling clvmd to exit                                    [
OK  ]
clvmd terminated                                           [
OK  ]
[root@example-01 ~]# service cman stop
Stopping cluster:
   Leaving fence domain...                                 [
OK  ]
   Stopping gfs_controld...                                [
OK  ]
   Stopping dlm_controld...                                [
OK  ]
   Stopping fenced...                                      [
OK  ]
   Stopping cman...                                        [
OK  ]
   Waiting for corosync to shutdown:                       [  OK
]
   Unloading kernel modules...                             [
OK  ]
   Unmounting configfs...                                  [
OK  ]
[root@example-01 ~]#
```

   2. Em cada nó, inicie o software de cluster de acordo com a Seção 8.1.1, “Iniciar o Software do Cluster”. Por exemplo:

```
[root@example-01 ~]# service cman start
Starting cluster:
   Checking Network Manager...                             [
OK  ]
   Global setup...                                         [
```

```
OK  ]
   Loading kernel modules...                                  [
OK  ]
   Mounting configfs...                                       [
OK  ]
   Starting cman...                                           [
OK  ]
   Waiting for quorum...                                      [
OK  ]
   Starting fenced...                                         [
OK  ]
   Starting dlm_controld...                                   [
OK  ]
   Starting gfs_controld...                                   [
OK  ]
   Unfencing self...                                          [
OK  ]
   Joining fence domain...                                    [
OK  ]
[root@example-01 ~]# service clvmd start
Starting clvmd:                                               [
OK  ]
Activating VG(s):   2 logical volume(s) in volume group
"vg_example" now active
                                                              [
OK  ]
[root@example-01 ~]# service gfs2 start
Mounting GFS2 filesystem (/mnt/gfsA):                         [  OK
]
Mounting GFS2 filesystem (/mnt/gfsB):                         [  OK
]
[root@example-01 ~]# service rgmanager start
Starting Cluster Service Manager:                             [  OK
]
[root@example-01 ~]#
```

3. Em qualquer nó no cluster, rode o **cman_tools nodes** para verificar que os nós estão funcionando como membros no cluster (mostrados como "M" na coluna de estado "Sts"). Por exemplo:

```
[root@example-01 ~]# cman_tool nodes
Node  Sts   Inc   Joined               Name
   1   M    548   2010-09-28 10:52:21  node-01.example.com
   2   M    548   2010-09-28 10:52:21  node-02.example.com
```

4. Em qualquer nó, usando o utilitário **clustat**, verifique que os serviços de Alta Disponibilidade estão rodando conforme esperados. Além disso, o **clustat**, exibe o estado dos nós do cluster. Por exemplo:

```
[root@example-01 ~]#clustat
Cluster Status for mycluster @ Wed Nov 17 05:40:00 2010
Member Status: Quorate

 Member Name                             ID   Status
 ------ ----                             ---- ------
```

```
 node-02.example.com                          2 Online, rgmanager
 node-01.example.com                          1 Online, Local,
rgmanager

 Service Name                    Owner (Last)
State
 ------- ----                    ----- ------
-----
 service:example_apache          node-01.example.com
started
 service:example_apache2         (none)
disabled
```

8.2.2. Adicionando um Nó ao um Cluster

Adicionar um nó ao um cluster consiste da atualização de configuração do cluster, propagando a configuração atualizada no nó a ser adicionado e inicializar o software do cluster neste nó. Para adicionar um nó ao cluster, realize os seguintes passos:

1. Em qualquer nó no cluster, edite o **`/etc/cluster/cluster.conf`** para adicionar uma seção **`clusternode`** para o nó a ser adicionado. Por exemplo, no Exemplo 8.2, “Configuração de Cluster de Dois Nós” se o node-03.example.com devesse ser adicionado, então adicione uma seção **`clusternode`** para o nó. Se adicionar um nó (ou nós) causa uma transição de um cluster de dois nós para um cluster com três nós ou mais, remova os seguintes atributos **`cman`** do **`/etc/cluster/cluster.conf`**:

   - **`cman two_node="1"`**
   - **`expected_votes="1"`**

   Consulte a Seção 8.2.3, “Exemplos de Configurações de Três e Dois Nós” para comparação entre uma configuração de três nós e uma de dois nós.

2. Atualize o atributo **`config_version`** incrementando seu valor (por exemplo, mudando de **`config_version="2"`** para **`config_version="3">`**).

3. Salve o **`/etc/cluster/cluster.conf`**.

4. **`(Opcional)`** Valide o arquivo atualizado contra o esquema de cluster (**`cluster.rng`**) rodando o comando **`ccs_config_validate`**. Por exemplo:

   ```
   [root@example-01 ~]# ccs_config_validate
   Configuration validates
   ```

5. Rode o comando **`cman_tool version -r`** para propagar a configuração ao resto dos nós no cluster.

6. Verifique que o arquivo de configuração atualizado foi propagado.

7. Propague o arquivo de configuração atualizado no **`/etc/cluster/`** em cada nó a ser adicionado ao cluster. Por exemplo, use o comando **`scp`** para enviar o arquivo de configuração atualizado a cada nó a ser adicionado ao cluster.

8. Se a contagem de nó do cluster mudou de dois nós para um número maior de dois nós, você deve reiniciar o software do cluster no nó existente do cluster conforme a seguir:

1. Em cada nó, pare o software do cluster de acordo com a Seção 8.1.2, “Parando um Software de Cluster”. Por exemplo:

```
[root@example-01 ~]# service rgmanager stop
Stopping Cluster Service Manager:                          [  OK
]
[root@example-01 ~]# service gfs2 stop
Unmounting GFS2 filesystem (/mnt/gfsA):                    [  OK
]
Unmounting GFS2 filesystem (/mnt/gfsB):                    [  OK
]
[root@example-01 ~]# service clvmd stop
Signaling clvmd to exit                                    [
OK  ]
clvmd terminated                                           [
OK  ]
[root@example-01 ~]# service cman stop
Stopping cluster:
   Leaving fence domain...                                 [
OK  ]
   Stopping gfs_controld...                                [
OK  ]
   Stopping dlm_controld...                                [
OK  ]
   Stopping fenced...                                      [
OK  ]
   Stopping cman...                                        [
OK  ]
   Waiting for corosync to shutdown:                       [  OK
]
   Unloading kernel modules...                             [
OK  ]
   Unmounting configfs...                                  [
OK  ]
[root@example-01 ~]#
```

2. Em cada nó, inicie o software de cluster de acordo com a Seção 8.1.1, “Iniciar o Software do Cluster”. Por exemplo:

```
[root@example-01 ~]# service cman start
Starting cluster:
   Checking Network Manager...                             [
OK  ]
   Global setup...                                         [
OK  ]
   Loading kernel modules...                               [
OK  ]
   Mounting configfs...                                    [
OK  ]
   Starting cman...                                        [
OK  ]
   Waiting for quorum...                                   [
OK  ]
   Starting fenced...                                      [
OK  ]
```

```
   Starting dlm_controld...                                   [
OK  ]
   Starting gfs_controld...                                   [
OK  ]
   Unfencing self...                                          [
OK  ]
   Joining fence domain...                                    [
OK  ]
[root@example-01 ~]# service clvmd start
Starting clvmd:                                               [
OK  ]
Activating VG(s):   2 logical volume(s) in volume group
"vg_example" now active
                                                              [
OK  ]
[root@example-01 ~]# service gfs2 start
Mounting GFS2 filesystem (/mnt/gfsA):                         [  OK
]
Mounting GFS2 filesystem (/mnt/gfsB):                         [  OK
]
[root@example-01 ~]# service rgmanager start
Starting Cluster Service Manager:                             [  OK
]
[root@example-01 ~]#
```

9. Em cada nó a ser adicionado ao cluster, inicie o software de cluster de acordo com a Seção 8.1.1, “Iniciar o Software do Cluster”. Por exemplo:

```
[root@example-01 ~]# service cman start
Starting cluster:
   Checking Network Manager...                             [  OK
]
   Global setup...                                         [  OK
]
   Loading kernel modules...                               [  OK
]
   Mounting configfs...                                    [  OK
]
   Starting cman...                                        [  OK
]
   Waiting for quorum...                                   [  OK
]
   Starting fenced...                                      [  OK
]
   Starting dlm_controld...                                [  OK
]
   Starting gfs_controld...                                [  OK
]
   Unfencing self...                                       [  OK
]
   Joining fence domain...                                 [  OK
]
[root@example-01 ~]# service clvmd start
Starting clvmd:                                            [  OK
]
Activating VG(s):   2 logical volume(s) in volume group "vg_example"
```

```
now active
                                                           [  OK
]
[root@example-01 ~]# service gfs2 start
Mounting GFS2 filesystem (/mnt/gfsA):                      [  OK  ]
Mounting GFS2 filesystem (/mnt/gfsB):                      [  OK  ]

[root@example-01 ~]# service rgmanager start
Starting Cluster Service Manager:                          [  OK  ]
[root@example-01 ~]#
```

10. Em qualquer nó, usando o utilitário **clustat**, verifique que cada nó adicionado está rodando e é parte do cluster. Por exemplo:

```
[root@example-01 ~]#clustat
Cluster Status for mycluster @ Wed Nov 17 05:40:00 2010
Member Status: Quorate

 Member Name                              ID   Status
 ------ ----                              ---- ------
 node-03.example.com                          3 Online, rgmanager
 node-02.example.com                          2 Online, rgmanager
 node-01.example.com                          1 Online, Local,
rgmanager

 Service Name                    Owner (Last)
State
 ------- ----                    ----- ------                    --
---
 service:example_apache          node-01.example.com
started
 service:example_apache2         (none)
disabled
```

Para informações sobre usar o **clustat**, consulte a Seção 8.3, “Gerenciando Serviços de Alta Disponibilidade”.

Além disso, você pode usar o **cman_tool status** para verificar os votos do nós e contagem de quorum. Por exemplo:

```
[root@example-01 ~]#cman_tool status
Version: 6.2.0
Config Version: 19
Cluster Name: mycluster
Cluster Id: 3794
Cluster Member: Yes
Cluster Generation: 548
Membership state: Cluster-Member
Nodes: 3
Expected votes: 3
Total votes: 3
Node votes: 1
Quorum: 2
Active subsystems: 9
Flags:
```

```
Ports Bound: 0 11 177
Node name: node-01.example.com
Node ID: 3
Multicast addresses: 239.192.14.224
Node addresses: 10.15.90.58
```

11. Em qualquer nó, você pode usar o utilitário **clusvcadm** para migrar ou realocar um serviço em execução ao nó recém unido. Também, você pode habilitar quaisquer serviços desabilitados. Para informações sobre usar o **clusvcadm**, consulte a Seção 8.3, “Gerenciando Serviços de Alta Disponibilidade”

8.2.3. Exemplos de Configurações de Três e Dois Nós

Consulte os exemplos que seguem para comparação entre uma configuração de Três ou Dois Nós.

Exemplo 8.1. Configuração de Cluster de Três Nós

```
<cluster name="mycluster" config_version="3">
   <cman/>
   <clusternodes>
     <clusternode name="node-01.example.com" nodeid="1">
         <fence>
            <method name="APC">
              <device name="apc" port="1"/>
             </method>
         </fence>
     </clusternode>
     <clusternode name="node-02.example.com" nodeid="2">
         <fence>
            <method name="APC">
              <device name="apc" port="2"/>
            </method>
         </fence>
     </clusternode>
     <clusternode name="node-03.example.com" nodeid="3">
         <fence>
            <method name="APC">
              <device name="apc" port="3"/>
            </method>
         </fence>
     </clusternode>
   </clusternodes>
   <fencedevices>
         <fencedevice agent="fence_apc" ipaddr="apc_ip_example"
login="login_example" name="apc" passwd="password_example"/>
   </fencedevices>
   <rm>
       <failoverdomains>
           <failoverdomain name="example_pri" nofailback="0"
ordered="1" restricted="0">
               <failoverdomainnode name="node-01.example.com"
priority="1"/>
               <failoverdomainnode name="node-02.example.com"
priority="2"/>
```

```
                <failoverdomainnode name="node-03.example.com"
priority="3"/>
            </failoverdomain>
        </failoverdomains>
        <resources>
            <fs name="web_fs" device="/dev/sdd2" mountpoint="/var/www"
fstype="ext3"/>
            <ip address="127.143.131.100" monitor_link="yes"
sleeptime="10"/>
            <apache config_file="conf/httpd.conf" name="example_server"
server_root="/etc/httpd" shutdown_wait="0"/>
        </resources>
        <service autostart="0" domain="example_pri" exclusive="0"
name="example_apache" recovery="relocate">
            <fs ref="web_fs"/>
            <ip ref="127.143.131.100"/>
            <apache ref="example_server"/>
        </service>
        <service autostart="0" domain="example_pri" exclusive="0"
name="example_apache2" recovery="relocate">
            <fs name="web_fs2" device="/dev/sdd3" mountpoint="/var/www"
fstype="ext3"/>
            <ip address="127.143.131.101" monitor_link="yes"
sleeptime="10"/>
            <apache config_file="conf/httpd.conf" name="example_server2"
server_root="/etc/httpd" shutdown_wait="0"/>
        </service>
    </rm>
</cluster>
```

Exemplo 8.2. Configuração de Cluster de Dois Nós

```
<cluster name="mycluster" config_version="3">
    <cman two_node="1" expected_votes="1"/>
    <clusternodes>
      <clusternode name="node-01.example.com" nodeid="1">
          <fence>
             <method name="APC">
               <device name="apc" port="1"/>
              </method>
          </fence>
      </clusternode>
      <clusternode name="node-02.example.com" nodeid="2">
          <fence>
             <method name="APC">
               <device name="apc" port="2"/>
             </method>
          </fence>
    </clusternodes>
    <fencedevices>
          <fencedevice agent="fence_apc" ipaddr="apc_ip_example"
login="login_example" name="apc" passwd="password_example"/>
```

```
   </fencedevices>
   <rm>
       <failoverdomains>
           <failoverdomain name="example_pri" nofailback="0"
ordered="1" restricted="0">
               <failoverdomainnode name="node-01.example.com"
priority="1"/>
               <failoverdomainnode name="node-02.example.com"
priority="2"/>
           </failoverdomain>
       </failoverdomains>
       <resources>
           <fs name="web_fs" device="/dev/sdd2" mountpoint="/var/www"
fstype="ext3"/>
           <ip address="127.143.131.100" monitor_link="yes"
sleeptime="10"/>
           <apache config_file="conf/httpd.conf" name="example_server"
server_root="/etc/httpd" shutdown_wait="0"/>
       </resources>
       <service autostart="0" domain="example_pri" exclusive="0"
name="example_apache" recovery="relocate">
           <fs ref="web_fs"/>
           <ip ref="127.143.131.100"/>
           <apache ref="example_server"/>
       </service>
       <service autostart="0" domain="example_pri" exclusive="0"
name="example_apache2" recovery="relocate">
           <fs name="web_fs2" device="/dev/sdd3" mountpoint="/var/www"
fstype="ext3"/>
           <ip address="127.143.131.101" monitor_link="yes"
sleeptime="10"/>
           <apache config_file="conf/httpd.conf" name="example_server2"
server_root="/etc/httpd" shutdown_wait="0"/>
       </service>
   </rm>
</cluster>
```

8.3. GERENCIANDO SERVIÇOS DE ALTA DISPONIBILIDADE

Você pode gerenciar serviços de alta disponibilidade usando o **Utilitário de Estado do Cluster** , **clustat** e o **Utilitário de Administração de Serviço de Usuários do Cluster** , **clusvcadm**. O **clustat** exibe o estado de um cluster e o **clusvcadm** fornece os meios para gerenciar os serviços de alta disponibilidade.

Esta seção fornece informações básicas sobre como gerenciar serviços de Alta Disponibilidade usando o **clustat** e o **clusvcadm**. Ela consiste das seguintes subseções:

- Seção 8.3.1, “Exibindo o Estado de Serviços de Alta Disponibilidade com o **clustat**.”
- Seção 8.3.2, “Gerenciando Serviços de Alta Disponibilidade com o **clusvcadm**”

8.3.1. Exibindo o Estado de Serviços de Alta Disponibilidade com o clustat.

O **clustat** exibe o estado inteiro do cluster. Ele mostra informações de afiliação, vizualização de quorum, o estado de todos os serviços de alta disponibilidade e indica qual nó o comando **clustat** está sendo rodado (local). A Tabela 8.1, "Estados dos Serviços" descreve os estados que os serviços podem estar e são exibidas quando executar o **clustat**. O Exemplo 8.3, "Exibição do **clustat**" exibe um exemplo de uma exibição do **clustat**. Para informações mais detalhadas sobre rodar o comando **clustat** consulte a página man **clustat**.

Tabela 8.1. Estados dos Serviços

Estados dos Serviços	Descrição
Started (Iniciado)	Os recursos do serviço estão configurados e disponíveis no sistema de cluster que possui o serviço.
Recovering (Recuperação)	Os serviço está pendente para iniciar em outro nó.
Disabled (Desabilitado)	O serviço foi desabilitado e não possui um proprietário atribuído. Um serviço desabilitado nunca é reinicializado automaticamente pelo cluster.
Stopped (Parado)	No estado parado, o serviço será avaliado para iniciar depois do próximo serviço ou transição de nó. Este é um estado temporário. Você pode desabilitar ou habilitar o serviço deste estado.
Failed (Falhado)	O serviço é pressuposto como morto. Um serviço é colocado neste estado toda vez que uma operação de parar recurso falha. Depois que um serviço é colocado neste estado, você deve verificar que não há recursos alocados (sistemas de arquivos montados por exemplo) antes de emitir um pedido de *desabilitação*. A única operação que pode tomar lugar quando um serviço tiver entrado neste estado é a **disable**.
Uninitialized (Não inicializado)	Este estado pode aparecer em certos casos durante a inicialização e execução do **clustat -f**.

Exemplo 8.3. Exibição do clustat

```
[root@example-01 ~]#clustat
Cluster Status for mycluster @ Wed Nov 17 05:40:15 2010
Member Status: Quorate

 Member Name                             ID   Status
 ------ ----                             ---- ------
 node-03.example.com                         3 Online, rgmanager
 node-02.example.com                         2 Online, rgmanager
 node-01.example.com                         1 Online, Local,
rgmanager

 Service Name                   Owner (Last)                   State
 ------- ----                   ----- ------                   -----
 service:example_apache         node-01.example.com            started
 service:example_apache2        (none)
disabled
```

8.3.2. Gerenciando Serviços de Alta Disponibilidade com o `clusvcadm`

Você pode gerenciar serviços HA usando o comando **`clusvcadm`**. Com ele você pode realizar as seguintes operações:

- Habilitar e iniciar um serviço.
- Desabilitar um serviço.
- Parar um serviço.
- Congelar um serviço
- Descongelar um serviço
- Migrar um serviço (somente para serviços de máquinas virtuais)
- Realocar um serviço.
- Reiniciar um serviço.

A Tabela 8.2, "Operações dos Serviços" descreve as operações em maiores detalhes. Para uma completa descrição de como realizar essas operações, consulte a página man **`clusvcadm`**.

Tabela 8.2. Operações dos Serviços

Operação de Serviço	Descrição	Sintaxe de Comando
Enable (Habilitar)	Inicie o serviço, opcionalmente em um alvo preferido e opcionalmente de acordo com regras de domínio de failover. Na falta de ambos, a máquina local onde o **`clusvcadm`** está sendo executado, inicializará o serviço. Se a inicialização original falhar, o serviço se comporta como se uma operação de *realocação* fosse solicitada (consulte *Realocar* nesta tabela). Se a operação é bem sucedida, o serviço é colocado no estado de inicializado.	**`clusvcadm -e <service_name>`** ou **`clusvcadm -e <service_name> -m <member>`** (Usando a opção -m especifica o membro alvo preferido no qual iniciará o serviço.)
Disable (Desabilitar)	Pára o serviço e coloca em um estado desabilitado. Esta é a única operação permissiva quando um serviço está em um estado de *falha*.	**`clusvcadm -d <service_name>`**

Operação de Serviço	Descrição	Sintaxe de Comando
Relocate (Realocar)	Move o serviço para outro nó. Opcionalmente, você pode especificar um nó preferido para receber o serviço, mas a inabilidade do serviço de executar neste host (por exemplo, se o serviço falha em iniciar ou o host estiver offline) não previne a realocação e um outro nó é escolhido. O **`rgmanager`** tenta iniciar o serviço em cada nó permissível no cluster. Se nenhum nó alvo permissível no cluster iniciar o serviço com sucesso, a realocação falha e o serviço é tentado a ser iniciado no proprietário original. Se o proprietário original não pode reiniciar o serviço, o serviço é colocado em um estado *parado*.	**`clusvcadm -r <service_name>`** or **`clusvcadm -r <service_name> -m <member>`** (Usando a opção -m especifica o membro alvo preferido no qual o serviço inicia.)
Stop (Parar)	Pára o serviço e o coloca no estado *parado*.	**`clusvcadm -s <service_name>`**
Freeze (Congelar)	Congela um serviço no nó onde está rodando atualmente. Isto previne a verificação de estado do serviço tanto quanto um failover no evento do nó falhar ou o rgmanager estiver parado. Isto pode ser usado para suspender um serviço para permitir manutenção de recursos subjacentes. Consulte "Considerações para Usar as Operações de Congelar (Freeze) e Descongelar (Unfreeze)" para informações importantes sobre usar as operações de *freeze* e *unfreeze*.	**`clusvcadm -Z <service_name>`**
Unfreeze (Descongelar)	Descongelar tira o serviço do estado *congelado*. Isto rehabilita a verificação do estado. Consulte "Considerações para Usar as Operações de Congelar (Freeze) e Descongelar (Unfreeze)" para informações importantes sobre o uso do *freeze* (congelar) e *unfreeze* (descongelar).	**`clusvcadm -U <service_name>`**
Migrate (Migrar)	Migrar uma máquina virtual para um outro nó. Você deve especificar um nó alvo. Dependendo da falha, a falha para migrar pode resultar na máquina virtual no estado de *falha* ou no estado iniciado no proprietário original.	**`clusvcadm -M <service_name> -m <member>`** **IMPORTANTE** Para a operação de *migrar*, você *deve especificar* um nó alvo usando a opção **`-m <member>`**.

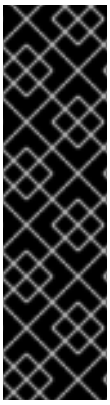
Operação de Serviço	Descrição	Sintaxe de Comando
Restart (Reiniciar)	Reiniciar um serviço no nó onde ele está rodando atualmente.	`clusvcadm -R <service_name>`

8.3.2.1. Considerações para Usar as Operações de Congelar (Freeze) e Descongelar (Unfreeze)

Usando a operação *congelar* permite manutenção de partes dos serviços `rgmanager`. Por exemplo, se você tiver um banco de dados e um servidor web em um serviço `rmanager`, você pode congelar o serviço `rgmanager`, parar o banco de dados, realizar manutenção, reiniciar o banco de dados e descongelar o serviço.

Quando um serviço está congelado, ele se comporta assim:

- Verificação do *Estado* são desabilitados.
- Operações de *Iniciar* são desabilitadas.
- Operações de *Parar* são desabilitadas.
- O Failover não ocorrerá (mesmo se você desligar o proprietário do serviço).



IMPORTANTE

O não cumprimento destas orientações podem resultar em recursos sendo alocados em hosts múltiplos:

- Você *não deve* parar todas as instâncias do rgmanager quando um serviço estiver congelado a menos que você planeje reinicializar os hosts antes de reiniciar o rgmanager.
- Você *não deve* descongelar um serviço até que o proprietário do serviço reingresse no cluster e reinicie o rgmanager.

8.4. ATUALIZANDO UMA CONFIGURAÇÃO

Atualizando a configuração do cluster consiste em editar o arquivo de configuração do cluster (`/etc/cluster/cluster.conf`) e propaga-lo a cada nó no cluster. Você pode atualizar a configuração usando quaisquer dos seguintes procedimentos:

- Seção 8.4.1, “Atualizando uma Configuração Usando o `cman_tool version -r`”
- Seção 8.4.2, “Atualizar a Configuração Usando o `scp`”

8.4.1. Atualizando uma Configuração Usando o `cman_tool version -r`

Para atualizar a configuração usando o comando `cman_tool version -r`, realize os seguintes passos:

1. Em qualquer nó no cluster, edite o arquivo `/etc/cluster/cluster.conf`.

2. Atualize o atributo **config_version** incrementando seu valor (por exemplo, mudando de **config_version="2"** para **config_version="3">**).

3. Salve o **/etc/cluster/cluster.conf**.

4. Execute o comando **cman_tool version -r** para propagar a configuração ao resto dos nós no cluster. É necessário que o **ricci** esteja rodando em cada nó no cluster para ser capaz de propagar a informação de configuração do cluster atualizada.

5. Verifique que o arquivo de configuração atualizado foi propagado.

6. Você pode pular este passo (reiniciar o software de cluster) se você fez somente as seguintes mudanças na configuração:

   - Deletar um nó de uma configuração de cluster — *exceto* onde a contagem de nós mudar para um número maior de dois para dois nós. Para informações sobre deletar um nó em um cluster e alterar de um número maior de dois nós para dois nós, consulte Seção 8.2, “Deletando ou Adicionando um Nó”.

   - Adicionar um nó às configurações do cluster — *exceto* onde a contagem do nó muda de um número maior de dois nós para dois nós. Para informações sobre como adicionar um nó a um cluster em uma transição do dois nós para um número maior que dois nós, consulte a Seção 8.2.2, “Adicionando um Nó ao um Cluster” .

   - Mudanças em como o daemons registra as informações de log.

   - Serviço HA/Manutenção VM (adicionar, editar ou deletar).

   - Manutenção de Recursos (adicionar, editar ou deletar).

   - Manutenção de Domínio de Failover (adicionar, editar ou deletar).

   De outra maneira, você deve reiniciar o software de cluster conforme a seguir:

   1. Em cada nó, pare o software do cluster de acordo com a Seção 8.1.2, “Parando um Software de Cluster”. Por exemplo:

```
[root@example-01 ~]# service rgmanager stop
Stopping Cluster Service Manager:                          [  OK
]
[root@example-01 ~]# service gfs2 stop
Unmounting GFS2 filesystem (/mnt/gfsA):                    [  OK
]
Unmounting GFS2 filesystem (/mnt/gfsB):                    [  OK
]
[root@example-01 ~]# service clvmd stop
Signaling clvmd to exit                                    [
OK  ]
clvmd terminated                                           [
OK  ]
[root@example-01 ~]# service cman stop
Stopping cluster:
   Leaving fence domain...                                 [
OK  ]
   Stopping gfs_controld...                                [
OK  ]
   Stopping dlm_controld...                                [
```

```
OK  ]
   Stopping fenced...                                      [
OK  ]
   Stopping cman...                                        [
OK  ]
   Waiting for corosync to shutdown:                       [  OK
]
   Unloading kernel modules...                             [
OK  ]
   Unmounting configfs...                                  [
OK  ]
[root@example-01 ~]#
```

2. Em cada nó, inicie o software de cluster de acordo com a Seção 8.1.1, “Iniciar o Software do Cluster”. Por exemplo:

```
[root@example-01 ~]# service cman start
Starting cluster:
   Checking Network Manager...                             [
OK  ]
   Global setup...                                         [
OK  ]
   Loading kernel modules...                               [
OK  ]
   Mounting configfs...                                    [
OK  ]
   Starting cman...                                        [
OK  ]
   Waiting for quorum...                                   [
OK  ]
   Starting fenced...                                      [
OK  ]
   Starting dlm_controld...                                [
OK  ]
   Starting gfs_controld...                                [
OK  ]
   Unfencing self...                                       [
OK  ]
   Joining fence domain...                                 [
OK  ]
[root@example-01 ~]# service clvmd start
Starting clvmd:                                            [
OK  ]
Activating VG(s):   2 logical volume(s) in volume group
"vg_example" now active
                                                           [
OK  ]
[root@example-01 ~]# service gfs2 start
Mounting GFS2 filesystem (/mnt/gfsA):                      [  OK
]
Mounting GFS2 filesystem (/mnt/gfsB):                      [  OK
]
[root@example-01 ~]# service rgmanager start
Starting Cluster Service Manager:                          [  OK
]
[root@example-01 ~]#
```

-

Parar e iniciar o software de cluster certifica que qualquer mudança de configuração que são verificadas somente na hora da inicialização são incluídas na configuração em execução.

7. Em qualquer nó no cluster, rode o **cman_tools nodes** para verificar que os nós estão funcionando como membros no cluster (mostrados como "M" na coluna de estado "Sts"). Por exemplo:

```
[root@example-01 ~]# cman_tool nodes
Node  Sts   Inc   Joined               Name
   1   M    548   2010-09-28 10:52:21  node-01.example.com
   2   M    548   2010-09-28 10:52:21  node-02.example.com
   3   M    544   2010-09-28 10:52:21  node-03.example.com
```

8. Em qualquer nó, usando o utilitário **clustat**, verifique que os serviços de Alta Disponibilidade estão rodando conforme esperados. Além disso, o **clustat**, exibe o estado dos nós do cluster. Por exemplo:

```
[root@example-01 ~]#clustat
Cluster Status for mycluster @ Wed Nov 17 05:40:00 2010
Member Status: Quorate

 Member Name                             ID   Status
 ------ ----                             ---- ------
 node-03.example.com                         3 Online, rgmanager
 node-02.example.com                         2 Online, rgmanager
 node-01.example.com                         1 Online, Local,
rgmanager

 Service Name                   Owner (Last)
State
 ------- ----                   ----- ------                    --
---
 service:example_apache         node-01.example.com
started
 service:example_apache2        (none)
disabled
```

9. Se o cluster estiver em execução conforme esperado, você terminou a atualização da configuração.

8.4.2. Atualizar a Configuração Usando o scp

Para atualizar a configuração usando o comando **scp**, realize os seguintes passos:

1. Em cada nó, pare o software do cluster de acordo com a Seção 8.1.2, “Parando um Software de Cluster”. Por exemplo:

```
[root@example-01 ~]# service rgmanager stop
Stopping Cluster Service Manager:                          [  OK  ]
[root@example-01 ~]# service gfs2 stop
Unmounting GFS2 filesystem (/mnt/gfsA):                    [  OK  ]
Unmounting GFS2 filesystem (/mnt/gfsB):                    [  OK  ]
```

```
[root@example-01 ~]# service clvmd stop
Signaling clvmd to exit                                    [  OK
]
clvmd terminated                                           [  OK
]
[root@example-01 ~]# service cman stop
Stopping cluster:
   Leaving fence domain...                                 [  OK
]
   Stopping gfs_controld...                                [  OK
]
   Stopping dlm_controld...                                [  OK
]
   Stopping fenced...                                      [  OK
]
   Stopping cman...                                        [  OK
]
   Waiting for corosync to shutdown:                       [  OK  ]
   Unloading kernel modules...                             [  OK
]
   Unmounting configfs...                                  [  OK
]
[root@example-01 ~]#
```

2. Em qualquer nó no cluster, edite o arquivo **`/etc/cluster/cluster.conf`**.

3. Atualize o atributo **`config_version`** incrementando seu valor (por exemplo, mudando de **`config_version="2"`** para **`config_version="3">`**).

4. Salve o **`/etc/cluster/cluster.conf`**.

5. Valide o arquivo atualizado no esquema de cluster (**`cluster.rng`**) rodando o comando **`ccs_config_validate`**. Por exemplo:

```
[root@example-01 ~]# ccs_config_validate
Configuration validates
```

6. Se o arquivo atualizado é válido, use o comando **`scp`** para propaga-lo no **`/etc/cluster/`** em cada nó no cluster.

7. Verifique que o arquivo de configuração atualizado foi propagado.

8. Em cada nó, inicie o software de cluster de acordo com a Seção 8.1.1, “Iniciar o Software do Cluster”. Por exemplo:

```
[root@example-01 ~]# service cman start
Starting cluster:
   Checking Network Manager...                             [  OK
]
   Global setup...                                         [  OK
]
   Loading kernel modules...                               [  OK
]
   Mounting configfs...                                    [  OK
]
```

```
   Starting cman...                                           [  OK
]
   Waiting for quorum...                                      [  OK
]
   Starting fenced...                                         [  OK
]
   Starting dlm_controld...                                   [  OK
]
   Starting gfs_controld...                                   [  OK
]
   Unfencing self...                                          [  OK
]
   Joining fence domain...                                    [  OK
]
[root@example-01 ~]# service clvmd start
Starting clvmd:                                               [  OK
]
Activating VG(s):   2 logical volume(s) in volume group "vg_example"
now active
                                                              [  OK
]
[root@example-01 ~]# service gfs2 start
Mounting GFS2 filesystem (/mnt/gfsA):                         [  OK  ]
Mounting GFS2 filesystem (/mnt/gfsB):                         [  OK  ]
[root@example-01 ~]# service rgmanager start
Starting Cluster Service Manager:                             [  OK  ]
[root@example-01 ~]#
```

9. Em qualquer nó no cluster, rode o **cman_tools nodes** para verificar que os nós estão funcionando como membros no cluster (mostrados como "M" na coluna de estado "Sts"). Por exemplo:

```
[root@example-01 ~]# cman_tool nodes
Node  Sts   Inc   Joined               Name
   1   M    548   2010-09-28 10:52:21  node-01.example.com
   2   M    548   2010-09-28 10:52:21  node-02.example.com
   3   M    544   2010-09-28 10:52:21  node-03.example.com
```

10. Em qualquer nó, usando o utilitário **clustat**, verifique que os serviços de Alta Disponibilidade estão rodando conforme esperados. Além disso, o **clustat**, exibe o estado dos nós do cluster. Por exemplo:

```
[root@example-01 ~]#clustat
Cluster Status for mycluster @ Wed Nov 17 05:40:00 2010
Member Status: Quorate

 Member Name                             ID   Status
 ------ ----                             ---- ------
 node-03.example.com                         3 Online, rgmanager
 node-02.example.com                         2 Online, rgmanager
 node-01.example.com                         1 Online, Local,
rgmanager

 Service Name                   Owner (Last)
State
```

```
 ------- ----                    ----- ------                    --
---
 service:example_apache          node-01.example.com
started
 service:example_apache2         (none)
disabled
```

11. Se o cluster estiver em execução conforme esperado, você terminou a atualização da configuração.

CAPÍTULO 9. DIAGNOSTICANDO E CORRIGINDO PROBLEMAS EM UM CLUSTER

Problemas em clusters, por natureza, podem ser difíceis de solucionar. Isto é devido à grande complexidade que um cluster de sistemas apresenta diferente de diagnosticar problemas em sistemas únicos. Entretanto, há problemas comuns que administradores de sistemas estão mais propensos a encontrar quando implementando ou administrando um cluster. Entendendo como lidar com estes problemas comuns podem fazer a implementação e administração de um cluster muito mais facilmente.

Este capítulo fornece informações sobre alguns problemas de cluster comuns e como soluciona-los. Ajuda adicional pode ser encontrada em nossa base de conhecimento e contactando um representante de suporte autorizado Red Hat. Se seu problema é relacionado ao sistema de arquivos GFS2 especificamente, você pode encontrar informações sobre resolução de problemas comuns do GFS2 no documento *Global File System 2*(Sistema de Arquivos Global 2).

9.1. MUDANÇA DE CONFIGURAÇÃO NÃO É EFETUADA

Quando você faz mudanças em um cluster, você precisa propagar estas mudanças à todos os nós no cluster.

- Quando você configurar um cluster utilizando o **Conga**, **Conga** propaga as mudanças automaticamente quando você aplica as mudanças.
- Para informações sobre como propagar mudanças na configuração do cluster com o comando **ccs**, veja Seção 5.15, “Propagar o Arquivo de Configuração aos Nós do Cluster” .
- Para informações sobre como propagar mudanças na configuração do cluster com as ferramentas de linha de comando, consulte a Seção 8.4, “Atualizando uma Configuração”.

Se você fizer qualquer uma destas mudanças de configuração em seu cluster, não será necessário reiniciar o cluster após propagar estas mudanças para que as mesmas sejam efetuadas.

- Deletar um nó de uma configuração de cluster — *exceto* onde a contagem de nós mudar para um número maior de dois para dois nós.
- Adicionar um nó às configurações do cluster — *exceto* onde a contagem do nó muda de um número maior de dois nós para dois nós.
- Mudança de configuração de autenticação
- Adicionando, editando ou removendo os serviços de HA ou componentes VM.
- Adicionando, editando, ou removendo recursos de cluster
- Adicionando, editando ou removendo domínios de failover.

Se você fizer qualquer uma destas mudanças de configuração em seu cluster, você precisará reiniciar o cluster para implementar tais mudanças. As mudanças de configuração de cluster a seguir requerem que você reinicie um cluster para que estas sejam efetuadas.

- Adicionando ou removendo a opção **two_node** do arquivo de configuração do cluster.
- Renomeando o Cluster.
- Modificando qualquer **corosync** ou timers de **openais**.

- Adicionando, mudando ou removendo heurísticos para disco de quorum, modificando qualquer timers de disco de quorum, ou modificando o dispositivo de disco de quorum. Para que estas mudanças sejam efetuadas, é necessário um reinício global do daemon do **qdiskd**.

- Mudando do modo **central_processing** para **rgmanager**.Para que esta mudança tome efeito, é necessário um reinício global do **rgmanager**.

- Modificando o endereço multicast

- Mudando o modo de transporte de UDP multicast para UDP unicast, ou mudando de UDP unicast para UDP multicast.

Você pode reiniciar o cluster utilizando o **Conga**, o comando **ccs** ou as ferramentas de linha de comando.

- Para informações sobre como reiniciar um cluster com o usar o **Conga**, consulte o Seção 4.4, "Iniciando, Parando, Reinicializando e Deletando Clusters".

- Para informações sobre como reiniciar um cluster com o comando **ccs** command, refer to Seção 6.2, "Iniciando e Parando um Cluster" .

- Para informações sobre como reiniciar um cluster com as ferramentas de linha de comando, consulte a Seção 8.1, "Iniciar e Parar o Software de Cluster" .

9.2. O CLUSTER NÃO SE FORMA

Se você achar que está tendo problemas em formar um novo cluster, cheque os seguintes itens:

- Certifique-se que você possui uma resolução de nomes configurados corretamente. O nó do cluster no arquivo **cluster.conf** deve corresponder ao nome usado para resolver esse endereço de cluster na rede em que o cluster estará usando para se comunicar. Por exemplo, se os nomes dos nós de seu cluster são **nodea** e **nodeb** certifique-se que ambos os nós tenham entradas nos arquivos **/etc/cluster/cluster.conf** e **/etc/hosts** que correspondam a esses nomes.

- Se o cluster usa o multicast para comunicação entre os nós, assegure-se de que o tráfego do multicast não está sendo bloqueado, atrasado ou de outra maneira interferido na rede em que o cluster está usando para se comunicar. Note que alguns switches Cisco possuem recursos que podem causar atrasos em tráfego multicast.

- Use **telnet** ou **SSH** para verificar que você pode alcançar nós remotos.

- Execute o comando **ethtool eth1 | grep link** para checar se o link ethernet está ativo.

- Use o comando **tcpdump** em cada nó para checar o tráfego de rede.

- Certifique-se que você não possui regras de firewall bloqueando a comunicação entre seus nós.

- Certifique-se de que as interfaces que o cluster utiliza para uma comunicação entre nós, não está utilizando qualquer modo de vínculo a não ser o 0, 1 e 2. (Os modos de vínculo 0 e 2 são suportados desde o Red Hat Enterprise Linux 6.4).

9.3. OS NÓS ESTÃO INCAPAZES DE SE JUNTAR AO CLUSTER DEPOIS DE UM FENCE OU REINICIALIZAÇÃO

Se seus nós não se religam ao cluster depois de um fence ou reinicialização, cheque pelos seguintes itens:

- Clusters que estão passando seu tráfego por um switch Cisco Catalyst podem ter este problema.
- Certifique-se que todos os nós do cluster possuem a mesma versão do arquivo **cluster.conf**. Se o arquivo **cluster.conf** for diferente em qualquer um dos nós, então os nós podem ser incapazes de se ligar ao cluster após um fence.

  A partir do lançamento do Red Hat Enterprise 6.1, você pode usar o seguinte comando para verificar que todos os nós especificados no arquivo de configuração de cluster no host possuem arquivos de configurações idênticos:

  ```
  ccs -h host --checkconf
  ```

  Para informações sobre o comando **ccs**, veja o Capítulo 5, *Configurando o Complemento de Alta Disponibilidade da Red Hat com o comando* **ccs** e o Capítulo 6, *Gerenciando o Complemento de Alta Disponibilidade da Red Hat com o* **ccs**.
- Certifique-se que você tenha configurado o **chkconfig on** para serviços de cluster no nó que está tentando se ligar ao cluster.
- Certifique-se que não há regras de firewall bloqueando o nó de se comunicar com outros nós no cluster.

9.4. O DAEMON DO CLUSTER TRAVA

O RGManager possui processo de watchdog que reinicializa a máquina se o processo principal **rgmanager** falha inesperadamente. Isto faz com que o nó de cluster seja preso em um fence e que o **rgmanager** recupere o serviço em outra máquina. Quando o daemon do watchdog detecta que o processo principal do **rgmanager** travou, ele então reinicializa o nó de cluster, e os nós de cluster ativos irão detectar que o nó de cluster saiu e irá retirá-lo do cluster.

O número mais baixo do *process ID* (PID) é o processo do watchdog que leva a ação se seu filho (o processo com o número PID mais alto) trava. Capturar o centro do processo com um número PID mais alto utilizando o **gcore** pode ajudar na solução de problemas de um daemon travado.

Instale os pacotes que são necessários para capturar e visualizar o núcleo e certifique-se de que ambos o **rgmanager** e **rgmanager-debuginfo** são da mesma versão ou o núcleo do aplicativo capturado pode estar em desuso.

```
$ yum -y --enablerepo=rhel-debuginfo install gdb rgmanager-debuginfo
```

9.4.1. Capturar o Núcleo rgmanager durante o tempo de execução.

Existem dois processos **rgmanager** que estão em execução desde o início. Você precisa capturar o núcleo para que o processo do **rgmanager** com o PID mais alto.

O exemplo a seguir é um resultado do comando **ps** mostrando dois processos para **rgmanager**.

```
$ ps aux | grep rgmanager | grep -v grep
```

```
root     22482  0.0  0.5  23544  5136 ?        S<Ls Dec01   0:00 rgmanager
root     22483  0.0  0.2  78372  2060 ?        S<l  Dec01   0:47 rgmanager
```

No exemplo a seguir, o programa **pidof** é usado para determinar automaticamente o pid com maior número, o qual é o pid apropriado para criar um núcleo. O comando completo capta o núcleo de aplicativo para o processo 22483 o qual possui o número de pid maior.

```
$ gcore -o /tmp/rgmanager-$(date '+%F_%s').core $(pidof -s rgmanager)
```

9.4.2. Capturando o Núcleo Quando o Daemon Travar

Por padrão, o script **/etc/init.d/functions** bloqueia os arquivos núcleo dos daemons chamados de **/etc/init.d/rgmanager**. Para que o daemon crie os núcleos de aplicativos, você precisa habilitar aquela opção. Este procedimento deve ser feito em todos os nós de cluster que precisarem de um núcleo de aplicativo capturado.

Para criar um arquivo central para quando daemon rgmanager trava, edite o arquivo **/etc/sysconfig/cluster**. O parâmetro ***DAEMONCOREFILELIMIT*** permite que o daemon crie os arquivos centrais se o processo travar. Existe uma opção **-w** que previne que o processo do watchdog seja executado. O daemon do watchdog é responsável por reinicializar o nó de cluster se o **rgmanager** travar e em alguns casos, se o daemon do watchdog estiver em execução o arquivo central não será gerado, portanto deve ser desabilitado para capturar arquivos centrais.

```
DAEMONCOREFILELIMIT="unlimited"
RGMGR_OPTS="-w"
```

Reiniciar o rgmanager para ativar as novas opções de configuração:

```
service rgmanager restart
```

NOTA

Se os serviços de cluster estiverem sendo executados neste nó de cluster, ele pode então deixar os serviços em execução em mau estado.

O arquivo núcleo será gravado quando ele for gerado a partir de um travamento do processo do **rgmanager**.

```
ls /core*
```

O resultado deve ser semelhante a este:

```
/core.11926
```

Mude ou remova qualquer arquivo de núcleo antigo sob o diretório / antes de reiniciar o **rgmanager** para capturar o núcleo do aplicativo. O nó de cluster que experienciava o travamento do **rgmanager** deve ser reinicializado ou em fence após o núcleo ser capturado para certificar de que o processo do watchdog não estava em execução.

9.4.3. Gravando um gdb Backtrace Session

Depois que você capturou o arquivo núcleo, você poderá visualizar seu conteúdo usando o **gdb**, o Depurador do GNU. Para gravar uma sessão do script do **gdb** no arquivo núcleo a partir do sistema afetado, execute o seguinte:

```
$ script /tmp/gdb-rgmanager.txt
$ gdb /usr/sbin/rgmanager /tmp/rgmanager-.core.
```

Isto irá iniciar uma sessão **gdb**, enquanto o **script** a grava no arquivo texto adequado. Enquanto estiver no **gdb**, execute os seguintes comandos:

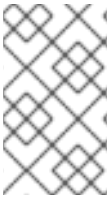
```
(gdb) thread apply all bt full
(gdb) quit
```

Pressione o **ctrl-D** para interromper a sessão do script e salvá-la no arquivo texto.

9.5. SUSPENSÃO DE SERVIÇOS DE CLUSTER

Quando o serviço de cluster tenta fazer um fence em um nó, os serviços de cluster param até que a operação fence tenha sido completada com sucesso. Portanto se seu armazenamento controlado de cluster ou serviços são suspensos e os nós de cluster mostram diferentes visões de afiliação do cluster ou se seu cluster está suspenso quando você tenta fazer um fence em um nó e você precisa reinicializar um nó para a recuperação, verifique pelas seguintes condições:

- O cluster pode ter tentado fazer um fence em um nó e a operação de fence pode ter falhado.
- Veja no arquivo **/var/log/messages** em todos os nós e veja se há qualquer mensagem de falha de fence. Se sim, então reinicialize os nós no cluster e configure o fence corretamente.
- Verifique que a partição da rede não ocorreu, conforme descrito na Seção 9.8, “Cada Nó em um Cluster de Dois Nós Reporta que o Segundo Nó está Desativado”, e verifique que a comunicação entre os nós é possível e que a rede está ativa.
- Se os nós deixam um cluster, os nós restantes podem estar fora de quorum. O cluster necessita estar em quorum para operar. Se os nós são removidos de maneira que o cluster não está mais em quorum, então os serviços e armazenamento estarão suspensos. Tanto ajuste os votos esperados ou retorne a quantidade requerida de nós ao cluster.



NOTA

Você pode fazer um fence em um nó manualmente com o comando **fence_node** ou com o **Conga**. Para informações, veja a página man **fence_node** e a Seção 4.3.2, “Faz um nó sair ou se juntar a um Cluster”.

9.6. O SERVIÇO DE CLUSTER NÃO INICIA

Se um serviço de cluster controlado não iniciar, cheque pelas seguintes condições.

- Pode haver um erro de sintaxe na configuração do serviço no arquivo **cluster.conf**. Você pode usar o comando **rg_test** para validar a sintaxe em sua configuração. Se existir qualquer falha na configuração ou na sintaxe, o **rg_test** informará qual é o problema.

  ```
  $ rg_test test /etc/cluster/cluster.conf start service servicename
  ```

Para informações sobre o comando `rg_test` veja a Seção C.5, “Depurando e Testando Serviços e Ordenação de Recursos”.

Se a configuração estiver válida, então aumente o recurso de log do gerenciador de grupo e então leia as mensagens de log para determinar o que está causando a falha do início de serviço. Você pode aumentar o nível de log adicionando o parametro `loglevel="7"` à tag `rm` no arquivo `cluster.conf`. Você então terá uma verbosidade aumentada em seus logs de mensagens em relação aos serviços de início, parada e migração em clusters.

9.7. SERVIÇOS DE CONTROLE DO CLUSTER FALHAM NA MIGRAÇÃO

Se um serviço de controle do cluster falha na migração para outro nó mas o serviço inicia em algum nó especifico, cheque pelas seguintes condições.

- Certifique-se que os recursos requeridos para rodar um serviço determinado estão presentes em todos os nós no cluster que podem ser requeridos para rodar tal serviço. Por exemplo, se seu serviço de cluster pressupõe que um arquivo de script está em uma determinada locação ou sistema de arquivo está montado em um determinado ponto de montagem, então você deve se certificar que aqueles recursos estão disponíveis nos respectivos lugares em todos os nós no cluster.
- Certifique-se que domínios failover, dependencia de serviços e exclusividade de serviços não estão configurados de tal maneira que você fique incapaz de migrar serviços à nós da maneira que você espera.
- Se o serviço em questão é um recurso de máquina virtual, cheque a documentação para assegurar que toda a configuração esteja correta e completa.
- Aumente o recurso de log do gerenciador de grupo, conforme descrito na Seção 9.6, “O Serviço de Cluster não inicia”, e então leia os logs de mensagens para determinar o que está causando a falha de migração do início de serviço.

9.8. CADA NÓ EM UM CLUSTER DE DOIS NÓS REPORTA QUE O SEGUNDO NÓ ESTÁ DESATIVADO

Se seu cluster é um cluster de dois nós e cada nó reporta que está ativo mas o outro nós está inativo, isto indica que seus nós do cluster estão incapazes de se comunicar via multicast pela rede de pulsação do cluster. Isto é conhecido como "split brain" ou "partição de rede". Para entender isso, cheque as condições delineadas na Seção 9.2, “O Cluster não se forma”.

9.9. NÓS ESTÃO EM FENCE NA FALHA DE CAMINHO DO LUN

Se um nó ou nós em seu cluster recebem fence toda vez que você tem uma falha de LUN path, isto pode ser um resultado do uso de um disco em quorum sob um armazenamento multi path. Se você estiver usando um disco de quorum sob armazenamento multipath. Se você estiver usando um disco de quorum e este está sob armazenamento multipath, assegure-se que você possui a cronometração correta definida para tolerar uma falha de caminho.

9.10. O DISCO DE QUORUM NÃO APARECE COMO MEMBRO DO CLUSTER

Se você configurou seu sistema para usar um disco de quorum mas o disco de quorum não aparece como membro de seu cluster, cheque pelas seguintes condições.

- Certifique-se que você definiu o **`chkconfig on`** para o serviço **`qdisk`**.
- Certifique-se que você iniciou o serviço **`qdisk`**.
- Note que isso pode levar vários minutos para o disco de quorum se registrar com o cluster. Isto é normal e esperado.

9.11. COMPORTAMENTO INCOMUM DE FAILOVER

Um problema comum com servidores de clusters é o comportamento incomum de failover. Os serviços param quando outros serviços iniciam ou os serviços se recusam a iniciar sob failover. Isto pode ser devido a possui complexos sistemas de failover consistindo em domínios de failover, dependências de serviços e exclusividade de serviços. Tente voltar para um serviço ou configuração de domínio mais simples e veja se o problema persiste. Evite recursos como exclusividade de serviços e dependências a menos que você entenda totalmente como esses recursos podem afetar o failover sob todas as condições.

9.12. FENCING OCORRE ALEATÓRIAMENTE

Se você acha que um nó está recebendo fence aleatóriamente, verifique pelas seguintes condições.

- O causador principal de fences é *sempre* um nó perdendo um token, significando que perdeu comunicação com o resto do cluster e parou de retornar pulsações.
- Qualquer situação que resulta em um sistema não retornar pulsações dentro de um intervalo específico de token pode resultar em um fence. Por padrão o intervalo de token é de 10 segundos. Isso pode ser especificado adicionando o valor desejado (em milisegundos) ao parametro de token no rótulo totem no arquivo **`cluster.conf`** (por exemplo, configurando **`totem token="30000"`** para 30 segundos).
- Certifique-se que a rede está rodando perfeitamente conforme esperado.
- Certifique-se de que as interfaces que o cluster utiliza para uma comunicação entre nós, não está utilizando qualquer modo de vínculo a não ser o 0, 1 e 2. (Os modos de vínculo 0 e 2 são suportados desde o Red Hat Enterprise Linux 6.4).
- Tome medidas para determinar se um sistema está "congelado" ou o kernel em pânico. Configure o utilitário **`kdump`** e veja se você recebe um núcleo durante um destes fences.
- Certifique-se que você não está atribuindo o problema a um fence erroneamente, por exemplo o disco de quorum expulsando um nó devido a uma falha de armazenamento ou um produto de terceitos como Oracle RAC reinicializando um nó devido a uma condição externa. As mensagens de log são muitas vezes úteis para determinar tais problemas. Toda vez que uma reinicialização de fence ou nó ocorrer, isso deve ser uma prática padrão verificar as mensagens de log de todos os nós no cluster a partir do momento que o reboot/fence tiver ocorrido.
- Verifique completamente o sistema por falhas em hardware que podem levar sistemas a não responder às pulsações quando esperados.

9.13. AUTENTICAÇÃO DE DEPUG PARA O GERENCIADOR DE BLOQUEIO DISTRIBUÍDO (DLM) PRECISA SER HABILITADA

Existem duas opções de depuração para o Distributed Lock Manager (DLM) que você pode habilitar, se necessário: O DLM kernel debugging, e a depuração de bloqueio POSIX.

Para habilitar a depuração do DLM, edite o arquivo **/etc/cluster/cluster.conf** para adicionar as opções de configuração a marcação de **dlm**. A opção **log_debug** habilita as mensagens de depuração do kernel do DLM, e a opção **plock_debug** habilita as mensagens de depuração de bloqueio do POSIX.

As seguintes seções de exemplo de um arquivo **/etc/cluster/cluster.conf** exibe a marcação **dlm** que permite ambas opções de depuração do DLM:

```
<cluster config_version="42" name="cluster1">
  ...
  <dlm log_debug="1" plock_debug="1"/>
  ...
</cluster>
```

Após a edição do arquivo **/etc/cluster/cluster.conf**, execute o comando **cman_tool version -r** para propagar a configuração ao resto dos nós no cluster.

CAPÍTULO 10. CONFIGURAÇÃO DO SNMP COM COMPLEMENTO DE ALTA DISPONIBILIDADE DA RED HAT

A partir do lançamento do Red Hat Enterprise Linux 6.1 e posteriores o Complemento de Alta Disponibilidade da Red Hat fornece suporte para SNMP traps (sinais). Este capítulos descreve como configurar seus sistema para SNMP seguido por um resumo das traps que o Complemento de Alta Disponibilidade da Red Hat emite para eventos específicos do cluster.

10.1. O SNMP E O COMPLEMENTO DE ALTA DISPONIBILIDADE DA RED HAT

O sub agente SNMP do Complemento de Alta Disponibilidade da Red Hat é **foghorn**, que emite traps SNMP. O sub agente **foghorn** conversa com o daemon **snmpd** por meios do protocolo AgentX. O sub agente **foghorn** somente cria traps SNMP. Ele não suporta outras operações SNMP tais como **get** ou **set**.

Atualmente não existem opções **config** para o sub agente **foghorn**. Ele não pode ser configurado para usar soquetes específicos; somente o soquete padrão AgentX é atualmente suportado.

10.2. CONFIGURANDO O SNMP COM O COMPLEMENTO DE ALTA DISPONIBILIDADE DA RED HAT

Para configurar o SNMP com o Complemento de Alta Disponibilidade da Red Hat, realize os seguintes passos em cada nó no cluster para garantir que os serviços necessários estão habilitados e rodando.

1. Para usar o SNMP traps com o Complemento de Alta Disponibilidade da Red Hat, o serviço **snmp** é requerido e age como um agente master. Já que o serviço **foghorn** é o sub agente e usa o protocolo AgentX, você deve adicionar a seguinte linha ao arquivo **/etc/snmp/snmpd.conf** para habilitar o suporte AgentX:

   ```
   master agentx
   ```

2. Para especificar o host onde as notificações do SNMP trap devem ser enviadas, adicione a seguinte linha ao arquivo **/etc/snmp/snmpd.conf**:

   ```
   trap2sink host
   ```

   Para mais informações sobre manuseio de notificações, veja a página man **snmpd.conf**.

3. Certifique-se que o daemon **snmpd** está habilitado e rodando executando os seguintes comandos:

   ```
   # chkconfig snmpd on
   # service snmpd start
   ```

4. Se o daemon **messagebus** não estiver já habilitado e rodando, execute os seguintes comandos:

   ```
   # chkconfig messagebus on
   # service messagebus start
   ```

5. Certfique-se que o daemon **foghorn** está habilitado e rodando executando os seguintes comandos:

```
# chkconfig foghorn on
# service foghorn start
```

6. Execute o seguinte comando para configurar seu sistema para que então o **COROSYNC-MIB** gere SNMP traps (sinais) e garanta que o daemon **corosync-notifyd** está habilitado e rodando:

```
# echo "OPTIONS=\"-d\" " > /etc/sysconfig/corosync-notifyd
# chkconfig corosync-notifyd on
# service corosync-notifyd start
```

Depois de você ter configurado cada nó no cluster para o SNMP e ter certificado que todos os serviços necessários estão rodando, sinais D-Bus serão recebidos pelo serviço **foghorn** e traduzidos em SNMPv2 traps. Estas traps (sinais) são então passados para o host que você definiu com a entrada **trapsink** para receber SNMPv2 traps (sinais).

10.3. ENCAMINHANDO SNMP TRAPS

É possível encaminhar SNMP traps para uma máquina que não é parte do cluster onde você pode usar o daemon **snmptrapd** na máquina externa e personalizar como responder às notificações.

Realize os seguintes passos para encaminhar SNMP traps em um cluster para uma máquina que não é uma dos nós do cluster:

1. Para cada nó no cluster, siga o procedimento descrito na Seção 10.2, “Configurando o SNMP com o Complemento de Alta Disponibilidade da Red Hat”, definindo a entrada **trap2sink *host*** no arquivo **/etc/snmp/snmpd.conf** para especificar o host externo que estará rodando o daemon **snmptrapd**.

2. No host externo que receberá os traps, edite o arquivo de configuração **/etc/snmp/snmptrapd.conf** para especificar suas strings da comunidade. Por exemplo, você pode usar a seguinte entrada para permitir que o daemon **snmptrapd** para processar notificações usando a string da comunidade **public**.

```
authCommunity log,execute,net public
```

3. No host externo que receberá as traps, certifique-se que o daemon **snmptrapd** está habilitado e rodando executando os seguintes comandos:

```
# chkconfig snmptrapd on
# service snmptrapd start
```

Para mais informações sobre processar notificações SNMP, veja a página man **snmptrapd.conf**.

10.4. SNMP TRAPS PRODUZIDAS PELO COMPLEMENTO DE ALTA DISPONIBILIDADE DA RED HAT

O daemon **foghorn** gera as seguintes traps:

- **fenceNotifyFenceNode**

  Esta trap (sinal) ocorre quando um nó em fence tenta fazer um fence em outro nó. Note que esta trap é somente gerada em um nó -- o nó que tentou realizar a operação de fence. A notificação inclui os seguintes campos:

  - **fenceNodeName** - nome do nó em fence
  - **fenceNodeID** - id do nó em fence
  - **fenceResult** - o resultado da operação fence (0 para com sucesso, -1 para se algo deu errado, -2 para nenhum método de fence definido)

- **rgmanagerServiceStateChange**

  Esta trap ocorre quando o estado de um serviço de cluster muda. A notificação inclui os seguintes campos:

  - **rgmanagerServiceName** - o nome do serviço, que inclui o tipo do serviço (por exemplo, **service:foo** ou **vm:foo**).
  - **rgmanagerServiceState** - o estado do serviço. Isto exclui estados transitórios como **starting** e **stopping** para reduzir desordem nas traps.
  - **rgmanagerServiceFlags** - as flags (bandeiras) do serviço: **frozen**, indicando que um serviço foi congelado usando **clusvcadm -Z**, e **partial**, indicando que um serviço no qual um recurso com falha recebeu uma flag como **non-critical** para que então o recurso possa falhar e seus componentes reiniciados manualmente sem que o serviço inteiro seja afetado.
  - **rgmanagerServiceCurrentOwner** - o proprietário do serviço. Se o serviço não estiver rodando, ele será **(none)** (nenhum).
  - **rgmanagerServicePreviousOwner** - o último dono do serviço, se conhecido. Se o último não é conhecido, poderá indicar **(none)** (nenhum).

O daemon **corosync-nodifyd** gera as seguintes traps:

- **corosyncNoticesNodeStatus**

  Esta trap (sinal) ocorre quando um nó se junta ou deixa um cluster. A notificação inclui os seguintes campos:

  - **corosyncObjectsNodeName** - nome do nó
  - **corosyncObjectsNodeID** - id do nó
  - **corosyncObjectsNodeAddress** - node endereço IP
  - **corosyncObjectsNodeStatus** - estado do nó (**joined** (se juntou) ou **left** (saiu))

- **corosyncNoticesQuorumStatus**

  Esta trap ocorre quando um estado quorum muda. A notificação inclui os seguintes campos:

  - **corosyncObjectsNodeName** - nome do nó

- **`corosyncObjectsNodeID`** - id do nó
- **`corosyncObjectsQuorumStatus`** - novo estado do quorum (**`quorate`** (em quorum) ou **`NOT quorate`** (sem quorum))

- **`corosyncNoticesAppStatus`**

  Esta trap ocorre quando uma aplicação cliente se conecta ou desconecta do Corosync.

  - **`corosyncObjectsNodeName`** - nome do nó
  - **`corosyncObjectsNodeID`** - id do nó
  - **`corosyncObjectsAppName`** - nome da aplicação
  - **`corosyncObjectsAppStatus`** - novo estado da aplicação (**`connected`** (conectado) ou **`disconnected`** (desconectado))

CAPÍTULO 11. CONFIGURAÇÕES DO SAMBA EM CLUSTER

Desde o lançamento do Red Hat Enterprise Linux 6.2, o Red Hat High Availability Add-On fornece suporte para o Samba em Cluster em execução em uma configuração ativa/ativa. Isto requer que você instale e configure um CTDB em todos os nós em um cluster, o qual você utiliza junto aos sistemas de arquivo em cluster do GFS2.

NOTA

O Red Hat Enterprise Linux 6 suporta um máximo de quatro nós em execução no Samba em cluster.

Este capítulo descreve os procedimentos para configurar o CTDB, configurando um sistema de exemplo. Para mais informações sobre como configurar o sistema de arquivo GFS2, consulte o *Sistemas de Arquivo Global 2* Para mais informações sobre como configurar volumes lógicos, consulte o *Administração do Gerenciador de Volume Lógica*

11.1. VISÃO GERAL DO CTDB

O CTDB é uma implementação de cluster do banco de dados TDB usado pelo Samba. Para usar o CTDB, um sistema de arquivo em cluster deve estar disponível e compartilhado em todos os nós em um cluster. O CTDB fornce recursos em clsuter acima deste sistema de arquivo em cluster. Desde o lançamento do Red Hat Enterprise Linux 6.2, o CTDB também executa uma pilha de cluster em paralelo para um fornecido pelo Red Hat Enterprise Linux clustering. O CTDB gerencia a associação de nós, recuperação/falha, realocação de IP e serviços do Samba.

11.2. PACOTES REQUERIDOS

Além dos pacotes padrão requeridos ao rodar o Complemento de Alta Disponibilidade da Red Hat e Complemento de Armazenamento Resiliente da Red Hat, para executa o Samba com os clusters do Red Hat Enterprise Linux, é necessário os seguintes pacotes:

- `ctdb`
- `samba`
- `samba-common`
- `samba-winbind-clients`

11.3. CONFIGURAÇÃO DE GFS2

Configurar o Samba com o Red Hat Enterprise Linux em cluster requer dois sistemas de arquivo do GFS2: Um sistema de arquivos pequeno para o CTDB e um segundo sistema de arquivo para o compartilhamento do Samba. Este exemplo mostra como criar os dois sistemas de arquivo GFS2.

Antes de criar os sistemas de arquivo GFS2, crie primeiro um volume lógico LVM para cada um dos sistemas de arquivo. Para informações sobre como criar os volumes lógicos LVM, consulte o *Logical Volume Manager Administration*. Este exemplo usa os seguintes volumes lógicos:

- **`/dev/csmb_vg/csmb_lv`**, que irá manter os dados de usuário que serão exportados via compartilhamento do Samba e devem ser do tamanho ideal. Este exemplo cria um volume lógico de 100GB de tamanho.

- **/dev/csmb_vg/ctdb_lv**, que irão armazenar as informações do estado do CTDB compartilhado e precisa ter 1GB.

Você cria gruposd e volume em cluster e volumes lógicos em somente um nó de cluster.

Para criar um sistema de arquivo GFS2 em um volume lógico, execute o comando **mkfs.gfs2**. Você executa este comando em somente um nó de cluster.

Para criar o sistema de arquivo para acomodar o compartilhamento do Samba em um volume lógico **/dev/csmb_vg/csmb_lv**, execute o seguinte comando:

```
[root@clusmb-01 ~]# mkfs.gfs2 -j3 -p lock_dlm -t csmb:gfs2
/dev/csmb_vg/csmb_lv
```

O significado dos parâmetros é o que se segue:

-j

Especifica o número de diários a criar no sistema de arquivo. Este exemplo usa um cluster com três nós, para que possamos criar um diário por nó.

-p

Especifica o protocolo de bloqueio. O **lock_dlm** é o protocolo de bloqueio que o GFS2 usa para a comunicação entre nós.

-t

Especifica o nome da tabela de bloqueio e é de formato *cluster_name:fs_name*. Neste exemplo, o nome do cluster como especificado no arquivo **cluster.conf** é **csmb**, e usamos o **gfs2** como o nome para o sistema de arquivo.

O resultado deste comando se assemelha a este abaixo:

```
This will destroy any data on /dev/csmb_vg/csmb_lv.
  It appears to contain a gfs2 filesystem.

Are you sure you want to proceed? [y/n] y

Device:
/dev/csmb_vg/csmb_lv
Blocksize:  4096
Device Size  100.00 GB (26214400 blocks)
Filesystem Size: 100.00 GB (26214398 blocks)
Journals:  3
Resource Groups:  400
Locking Protocol:   "lock_dlm"
Lock Table:   "csmb:gfs2"
UUID:
  94297529-ABG3-7285-4B19-182F4F2DF2D7
```

Neste exemplo, o sistema de arquivo **/dev/csmb_vg/csmb_lv** será montado em **/mnt/gfs2** em todos os nós. Este ponto de montagem deve coincidir o valor que você especifica como o local do diretório de **share** com a opção do **path =** no arquivo **/etc/samba/smb.conf** como descrito em Seção 11.5, “Configuração do Samba”.

Para criar o sistema de arquivo para acomodar as informações de estado do CTDB no volume lógico **/dev/csmb_vg/ctdb_lv**, execute o seguinte comando:

```
[root@clusmb-01 ~]# mkfs.gfs2 -j3 -p lock_dlm -t csmb:ctdb_state
/dev/csmb_vg/ctdb_lv
```

Observe que este comando especifica uma tabela de bloqueio diferente do que a tabela de bloqueio no exemplo que criou o sistema de arquivo no **/dev/csmb_vg/csmb_lv**. Isto difere dos nomes de tabela de bloqueio para dispositivos diferentes usados para o sistema de arquivo.

O resultado de **mkfs.gfs2** se assemelha a este abaixo:

```
This will destroy any data on /dev/csmb_vg/ctdb_lv.
  It appears to contain a gfs2 filesystem.

Are you sure you want to proceed? [y/n] y

Device:
/dev/csmb_vg/ctdb_lv
Blocksize:    4096
Device Size   1.00 GB (262144 blocks)
Filesystem Size:  1.00 GB (262142 blocks)
Journals:  3
Resource Groups:  4
Locking Protocol:  "lock_dlm"
Lock Table:   "csmb:ctdb_state"
UUID:
  BCDA8025-CAF3-85BB-B062-CC0AB8849A03
```

Neste exemplo, o sistema de arquivo **/dev/csmb_vg/ctdb_lv** será montado em **/mnt/ctdb** em todos os nós. Este ponto de montagem deve coincidir o valor que você especifica como o local do diretório de **.ctdb.lock** com a opção do **CTDB_RECOVERY_LOCK** no arquivo **/etc/sysconfig/ctdb** como descrito em Seção 11.4, “Configurações CTDB”.

11.4. CONFIGURAÇÕES CTDB

O arquivo de configuração do CTDB está localizado em **/etc/sysconfig/ctdb**. Os campos obrigatórios que precisam ser configurados para a operação do CTDB são estas a seguir:

- **CTDB_NODES**
- **CTDB_PUBLIC_ADDRESSES**
- **CTDB_RECOVERY_LOCK**
- **CTDB_MANAGES_SAMBA** (deve ser habilitado)
- **CTDB_MANAGES_WINBIND** (deve ser habilitado se estiver sendo executado em um sevidor de membro)

Os seguintes exemplos mostram um arquivo de configuração com os campos obrigatórios para o conjunto de operações do CTDB com parâmetros de exemplo:

```
CTDB_NODES=/etc/ctdb/nodes
```

```
CTDB_PUBLIC_ADDRESSES=/etc/ctdb/public_addresses
CTDB_RECOVERY_LOCK="/mnt/ctdb/.ctdb.lock"
CTDB_MANAGES_SAMBA=yes
CTDB_MANAGES_WINBIND=yes
```

O significado dos parâmetros é o que se segue:

CTDB_NODES

Especifica o local do arquivo que contém a lista de nó de cluster.

O arquivo **/etc/ctdb/nodes** onde o **CTDB_NODES** referencia, simplesmente lista os endereços IP dos nós de cluster como se segue no exemplo:

```
192.168.1.151
192.168.1.152
192.168.1.153
```

Neste exemplo, existe somente uma interface/IP em cada nó que seja usada para ambos clientes de comunicação e serviços do cluster/CTDB. Entretanto, é altamente recomendado que cada nó de cluster possua duas interfaces de rede para que um conjunto de interfaces possa ser dedicado a comunicação de cluster/CTDB e outro conjunto de interfaces possa ser dedicado ao acesso de cliente público. Use os endereços iP apropriados da rede de cluster aqui e certifique-se que os endereços de hostname/IP no arquivo **cluster.conf** sejam o mesmo. Da mesma forma, use as interfaces apropriadas de rede pública para acesso ao cliente no arquivo **public_addresses**.

É crucial que o arquivo **/etc/ctdb/nodes** seja idêntico em todos os nós porque a ordem é mais importante e o CTDB irá falhar se encontrar informações diferentes em nós diferentes.

CTDB_PUBLIC_ADDRESSES

Especifica o local do arquivo que lista os endereços IP que podem ser usados para acessar os compartilhamentos do Samba exportados pelo cluster. Estes são endereços IP que você de configurar no DNS para o nome do servidor Samba em cluster e são endereços que os clientes CIFS se conectarão. Configure o nome do servidor Samba em cluster como uma gravação tipo A de DNS com endereços IP múltiplos e deixe os DNS de repetição alternada distribuídos entre clientes nos nós de cluster.

Para este exemplo, configuramos a entrada do DNS de repetição alternada **csmb-server** com todos os endereços listados no arquivo **/etc/ctdb/public_addresses**. O DNS irá distribuir os clientes que utilizam esta entrda entre os clusters como repetição alternada.

O conteúdo do arquivo **/etc/ctdb/public_addresses** em cada nó segue abaixo:

```
192.168.1.201/0 eth0
192.168.1.202/0 eth0
192.168.1.203/0 eth0
```

Este exemplo utiliza três endereços que estão em desuso atualmente na rede. Em sua própria configuração, escolha os endereços que podem ser acessados pelos clientes pretendidos.

Como forma alternativa, este exemplo demonstra que o conteúdo dos arquivo **/etc/ctdb/public_addresses** em um cluster no qual existem três nós mas um total de quatro endereços públicos.Neste exemplo o endereço IP 198.162.2.1 pode ser acomodado pelo nó 0 ou nó 1 e estará disponível aos clientes desde que ao menos um destes nós estejam disponíveis. Somente

se ambos os nós 0 e 1 falharem, o endereço público se tornará indisponível aos clientes. Todos os outros endereços públicos podem ser servidos somente por um único nó respectivamente e estará disponível se o nó respectivo também estiver disponível.

O arquivo **/etc/ctdb/public_addresses** no nó 0 inclui o seguinte conteúdo:

```
198.162.1.1/24 eth0
198.162.2.1/24 eth1
```

O arquivo **/etc/ctdb/public_addresses** no nó 1 inclui o seguinte conteúdo:

```
198.162.2.1/24 eth1
198.162.3.1/24 eth2
```

O arquivo **/etc/ctdb/public_addresses** no nó 2 inclui o seguinte conteúdo:

```
198.162.3.2/24 eth2
```

CTDB_RECOVERY_LOCK

Especifica um arquivo de bloqueio que o CTDB utiliza internamente para a recuperação. Este arquivo deve residir em armazenamento compartilhado de tal forma que todos os nós de cluster possuam acesso à ele. O exemplo nesta seção utiliza o sistema de arquivo GFS2 que será montado em **/mnt/ctdb** em todos os nós. Isto é diferente do sistema de arquivo GFS2 que acomodará o compartilhamento do Samba que será exportado. Este arquivo de bloqueio de recuperação é utilizado para prevenir cenários de quebra de memória (split-brain). Com versões mais recentes do CTDB (1.0.112 e posteriores), especificar este arquivo é opcional desde que seja substituído por outro mecanismo de prevenção split-brain.

CTDB_MANAGES_SAMBA

Quando você habilitar utilizando a definição **yes**, especifique se o CTDB pode iniciar e interromper o serviço do Samba como deve para fornecer serviço de migração/failover.

Quando o **CTDB_MANAGES_SAMBA** estiver habilitado, você precisará desabilitar a inicialização automática **init** dos daemons **smb** e **nmb** executando os seguintes comandos:

```
[root@clusmb-01 ~]# chkconfig snb off
[root@clusmb-01 ~]# chkconfig nmb off
```

CTDB_MANAGES_WINBIND

Quando você habilitar utilizando a definição **yes**,especifique que o CTDB pode iniciar e interromper o daemon **winbind** como requerido. Isto deve ser habilitado quando você estiver utilizando CTDB em um domínio de Windows ou em um modo de segurança de diretório ativo.

Quando o **CTDB_MANAGES_SAMBA** estiver habilitado, você precisará desabilitar a inicialização automática **init** dos daemons **winbind** executando o seguinte comando:

```
[root@clusmb-01 ~]# chkconfig windinbd off
```

11.5. CONFIGURAÇÃO DO SAMBA

O arquivo de configuração do Samba **smb.conf** está localizado em **/etc/samba/smb.conf** neste exemplo. Ele contém os parâmetros a seguir:

```
[global]
 guest ok = yes
 clustering = yes
 netbios name = csmb-server
[csmb]
 comment = Clustered Samba
  public = yes
 path = /mnt/gfs2/share
 writeable = yes
 ea support = yes
```

Este exemplo exporta um compartilhamento com o nome **csmb** localizado em **/mnt/gfs2/share**. Isto é diferente do sistema de arquivo compartilhado GFS2 em **/mnt/ctdb/.ctdb.lock** que especificamos como o parâmetro **CTDB_RECOVERY_LOCK** no arquivo de configuração do CTDB **/etc/sysconfig/ctdb**.

Neste exemplo criaremos um diretório **share** em **/mnt/gfs2** quando montamos ele pela primeira vez. A entrada **clustering = yes** instrui o Samba para utilizar o CTDB. A entrada **netbios name = csmb-server** define explicitamente todos os nós para terem o nome do NetBIOS em comum. O parâmetro **ea support** é necessário se você planejar utilizar atributos estendidos.

O arquivo de configuração do **smb.conf** deve ser idêntico a todos os nós de cluster.

O Samba também oferece configuração baseada em registro utilizando o comando **net conf** para manter a configuração automaticamente em sincronização entre os membros do cluster sem precisar copiar os arquivos de configuração entre os nós de cluster. Para informações sobre o comando **net conf** consulte a página man do **net**(8)

11.6. INICIANDO O CTDB E OS SERVIÇOS DO SAMBA

Após iniciar o cluster, você precisa montar os sistemas de arquivo GFS2 que você criou, como descrito em Seção 11.3, “Configuração de GFS2”. As permissões no diretório do Samba **share** e contas de usuário nos nós de cluster deve ser definido para acesso de clientes.

Execute o seguinte comando em todos os nós para iniciar o daemon **ctdbd**. Como este exemplo configurou o CTDB com **CTDB_MANAGES_SAMBA=yes**, CTDB também iniciará o serviço Samba em todos os nós e exportará todas as partes do Samba configuradas.

```
[root@clusmb-01 ~]# service ctdb start
```

Pode levar alguns minutos para o CTDB iniciar o Samba, exporte as partes e estabilize. Executar o **ctdb status** exibe o status do CTDB, como no exemplo a seguir:

```
[root@clusmb-01 ~]# ctdb status
Number of nodes:3
pnn:0 192.168.1.151     OK (THIS NODE)
pnn:1 192.168.1.152     OK
pnn:2 192.168.1.153     OK
Generation:1410259202
Size:3
hash:0 lmaster:0
```

```
hash:1 lmaster:1
hash:2 lmaster:2
Recovery mode:NORMAL (0)
Recovery master:0
```

Quando você vir que todos os nós estão "OK", é o momento seguro para mudar e utilizar o servidor Samba em cluster, como descrito em Seção 11.7, “Usando o Servidor Samba em Cluster” .

11.7. USANDO O SERVIDOR SAMBA EM CLUSTER

Os Clientes podem se conectar a parte do Samba que foi exportada ao se conectar em um dos endereços IP especificados no arquivo **/etc/ctdb/public_addresses** ou utilizando a entrada do DNS **csmb-server** que configuramos anteriormente, como exibido abaixo:

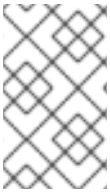
```
[root@clusmb-01 ~]# mount -t cifs //csmb-server/csmb /mnt/sambashare -o
user=testmonkey
```

ou

```
[user@clusmb-01 ~]$ smbclient //csmb-server/csmb
```

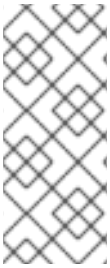
APÊNDICE A. PARÂMETROS DE DISPOSITOS FENCE

Este apêndice fornece tabelas com descrições dos parâmetros dos dispositivos fence. Você pode configurar os parâmetros com o **luci**, usando o comando `ccs` ou editando o `etc/cluster/cluster.conf`. Para uma lista compreensiva e descrição dos parâmetros de dispositivo do fence para cada agente fence, consulte a página man para aquele agente.



NOTA

O parâmetro **Name** para um dispositivo fence especifica um nome arbitrário pra o dispositivo que será usado pelo Complemento de Alta Disponibilidade da Red Hat. Isto não é a mesma coisa que um nome DNS para o dispositivo.



NOTA

Certos dispositivos fence possuem parâmetros opcionais **Password Script**. O parâmetro **Password Script** permite que você especifique que uma senha de dispositivo fence é fornecido de um script em vez de um parâmetro de **Password**. Usando o parâmetro **Password Script** substitui o parâmetro **Senha**, permitindo senhas não serem vísiveis no arquivo de configuração do cluster (`/etc/cluster/cluster.conf`).

Tabela A.1, “Sumário de Dispositos Fence” lista os dispositivos de fence, os agentes de dispositivos de fence associados com os dispositivo de fence, e fornece uma referência tabela documentando os parâmetros para os dispositivos de fence.

Tabela A.1. Sumário de Dispositos Fence

Dispositivo Fence	Agente Fence	Descrição de Parâmetro de Referência
Switch de Energia APC (telnet/SSH)	fence_apc	Tabela A.2, “Switch de Energia APC (telnet/SSH)”
Brocade Fabric Switch	fence_brocade	Tabela A.4, “Brocade Fabric Switch”
Cisco MDS	fence_cisco_mds	Tabela A.5, “Cisco MDS”
Cisco UCS	fence_cisco_ucs	Tabela A.6, “Cisco UCS”
Dell DRAC 5	fence_drac5	Tabela A.7, “Dell DRAC 5”
Eaton Network Power Switch (SNMP Interface)	fence_eaton_snmp	Tabela A.8, “Eaton Network Power Controller (SNMP Interface) (Red Hat Enterprise Linux 6.4 e posteriores)”
Egenera SAN Controller	fence_egenera	Tabela A.9, “Egenera SAN Controller”

Dispositivo Fence	Agente Fence	Descrição de Parâmetro de Referência
ePowerSwitch	fence_eps	Tabela A.10, “ePowerSwitch”
Fence virt	fence_virt	Tabela A.11, “Fence virt”
Fujitsu Siemens Remoteview Service Board (RSB)	fence_rsb	Tabela A.12, “Fujitsu Siemens Remoteview Service Board (RSB)”
HP BladeSystem	fence_hpblade	Tabela A.13, “HP BladeSystem (Red Hat Enterprise Linux 6.4 and later)”
HP iLO/iLO2 (Integrated Lights Out)	fence_ilo	Tabela A.14, “HP iLO/iLO2 (Integrated Lights Out)”
HP iLO (Integrated Lights Out) MP	fence_ilo_mp	Tabela A.15, “HP iLO (Integrated Lights Out) MP”
IBM BladeCenter	fence_bladecenter	Tabela A.16, “IBM BladeCenter”
IBM BladeCenter SNMP	fence_ibmblade	Tabela A.17, “IBM BladeCenter SNMP”
IBM iPDU	fence_ipdu	Tabela A.18, “IBM iPDU (Red Hat Enterprise Linux 6.4 e posteriores)”
IF MIB	fence_ifmib	Tabela A.19, “IF MIB”
Intel Modular	fence_intelmodular	Tabela A.20, “Intel Modular”
IPMI (Intelligent Platform Management Interface) LAN	fence_ipmilan	Tabela A.21, “IPMI (Intelligent Platform Management Interface) LAN”
RHEV-M REST API	fence_rhevm	Tabela A.22, “RHEV-M REST API (RHEL 6.2 e posteriores em RHEV 3.0 e posteriores)”

Dispositivo Fence	Agente Fence	Descrição de Parâmetro de Referência
SCSI Fencing	fence_scsi	Tabela A.23, “SCSI Fencing”
VMware Fencing (SOAP Interface)	fence_vmware_soap	Tabela A.24, “Fencing do VMware (Interface SOAP) (Red Hat Enterprise Linux 6.2 e posterior)”
WTI Power Switch	fence_wti	Tabela A.25, “WTI Power Switch”

Tabela A.2, “Switch de Energia APC (telnet/SSH)” lista os parâmetros de dispositivo de fence usado pelo **fence_apc**, o agente do fence para APC sob telnet/SSH.

Tabela A.2. Switch de Energia APC (telnet/SSH)

Campo luci	Recursos `cluster.conf`	Descrição
Name	**name**	Um nome para o dispositivo APC conectado ao cluster no qual o daemon faz um log via telnet/ssh.
Endereço IP ou Hostname	**ipaddr**	O endereço IP ou hostname atribuído ao dispositivo.
IP Port (opcional)	**ipport**	A porta TCP a ser usada para se conectar ao dispositivo.
Login	**login**	O nome de login usado para acessar o dispositivo.
Password	**passwd**	A senha usada para autenticar a conexão ao dispositivo.
Password Script (opcional)	**passwd_script**	O script que fornece uma senha para acesso ao dispositivo de fence. Usando isto se substitui o parâmetro de **Password**.
Power wait	**power_wait**	Número de segundos para esperar depois de emitir um comando de ligar ou desligar.
Port	**port**	TCP port
Switch (opcional)	**switch**	O número do switch para o switch APC que se conecta ao nó quando você tiver múltiplos switches daisy-chained.
Use SSH	**secure**	Indica que o sistema usará o SSH para acessar o dispositivo.

Campo luci	Recursos `cluster.conf`	Descrição
Caminho para identificar o arquivo SSH	`identity_file`	O arquivo de identificação para o SSH.

Tabela A.3, "APC Power Switch pelo SNMP" lista os parâmetros de dispositivos de fence usados pelo `fence_apc_snmp`, o agente de fence para o APC que autentica no dispositivo SNP via protocolo SNMP.

Tabela A.3. APC Power Switch pelo SNMP

Campo luci	Recursos `cluster.conf`	Descrição
Name	`name`	O nome do dispositivo APC conectado ao cluster no qual o daemon fence faz logs via protocolo SNMP.
Endereço IP ou Hostname	`ipaddr`	O endereço IP ou hostname atribuído ao dispositivo.
UDP/TCP port	`udpport`	A porta UDP/TCP a ser usada para conexão com o dispositivo; o valor padrão é 161.
Login	`login`	O nome de login usado para acessar o dispositivo.
Password	`passwd`	A senha usada para autenticar a conexão ao dispositivo.
Password Script (opcional)	`passwd_script`	O script que fornece uma senha para acesso ao dispositivo de fence. Usando isto se substitui o parâmetro de **Password**.
Versão SNMP	`snmp_version`	A versão SNMP a ser usada (1, 2c, 3); o valor padrão é 1.
Comunidade SNMP	`comunidade`	A série da comunidade SNMP; o valor padrão é `private`.
Nível de segurança SNMP	`snmp_sec_level`	O nível de segurança SNMP (noAuthNoPriv, authNoPriv, authPriv).
Protocolo de autenticação do SNMP	`snmp_auth_prot`	O protocolo de autenticação SNMP (MD5, SHA).

Campo luci	Recursos `cluster.conf`	Descrição
Protocolo de privacidade do SNMP	`snmp_priv_prot`	O protocolo de privacidade SNMP (DES, AES).
Senha do Protocolo de privacidade do SNMP	`snmp_priv_passwd`	A senha do protocolo de privacidade SNMP.
Script do Protocolo de Privacidade SNMP	`snmp_priv_passwd_script`	O script que fornece uma senha para o protocolo de privacidade SNMP. Usando isto se substitui o parâmetro **SNMP privacy protocol password**.
Espera de Energia	`power_wait`	Número de segundos para esperar depois de emitir um comando de ligar ou desligar.
Número de Porta (Outlet)	`port`	TCP port

Tabela A.4, “Brocade Fabric Switch” lista os parâmetros de dispositivo de fence usados pelo `fence_brocade`, o agente de fence para os interruptores Brocade FC.

Tabela A.4. Brocade Fabric Switch

Campo luci	Recursos `cluster.conf`	Descrição
Name	`name`	O nome para o dispositivo Brocade conectado ao cluster.
Endereço IP ou Hostname	`ipaddr`	O endereço IP atribuído ao dispositivo.
Login	`login`	O nome de login usado para acessar o dispositivo.
Password	`passwd`	A senha usada para autenticar a conexão ao dispositivo.
Password Script (opcional)	`passwd_script`	O script que fornece uma senha para acesso ao dispositivo de fence. Usando isto se substitui o parâmetro de **Password**.
Port	`port`	O número da saída do switch.

Tabela A.5, “Cisco MDS” lista os parâmetros de dispositivo de fence usados pelo `fence_cisco_mds`, o agente de fence para o Cisco MDS.

Tabela A.5. Cisco MDS

Campo luci	Recursos `cluster.conf`	Descrição
Name	`name`	Um nome para o dispositivo Cisco MDS 9000 series com o SNMP habilitado.
Endereço IP ou Hostname	`ipaddr`	O endereço IP ou hostname atribuído ao dispositivo.
UDP/TCP port	`udpport`	A porta UDP/TCP a ser usada para conexão com o dispositivo; o valor padrão é 161.
Login	`login`	O nome de login usado para acessar o dispositivo.
Password	`passwd`	A senha usada para autenticar a conexão ao dispositivo.
Password Script (opcional)	`passwd_script`	O script que fornece uma senha para acesso ao dispositivo de fence. Usando isto se substitui o parâmetro de **Password**.
Número de Porta (Outlet)	`port`	TCP port
Versão SNMP	`snmp_version`	A versão SNMP para usar (1, 2c, 3).
Comunidade SNMP	`comunidade`	A séria da comunidade SNMP.
Nível de segurança SNMP	`snmp_sec_level`	O nível de segurança SNMP (noAuthNoPriv, authNoPriv, authPriv).
Protocolo de autenticação do SNMP	`snmp_auth_prot`	O protocolo de autenticação SNMP (MD5, SHA).
Protocolo de privacidade do SNMP	`snmp_priv_prot`	O protocolo de privacidade SNMP (DES, AES).
Senha do Protocolo de privacidade do SNMP	`snmp_priv_passwd`	A senha do protocolo de privacidade SNMP.

Campo luci	Recursos `cluster.conf`	Descrição
Script do Protocolo de Privacidade SNMP	`snmp_priv_passwd_script`	O script que fornece uma senha para o protocolo de privacidade SNMP. Usando isto se substitui o parâmetro **SNMP privacy protocol password**.
Espera de Energia	`power_wait`	Número de segundos para esperar depois de emitir um comando de ligar ou desligar.

Tabela A.6, “Cisco UCS” lista os parâmetros de dispositivo de fence usado pelo `fence_cisco_ucs`, o agente de fence para Cisco UCS.

Tabela A.6. Cisco UCS

Campo luci	Recursos `cluster.conf`	Descrição
Name	`name`	O nome do dispositivo Cisco UCS.
Endereço IP ou Hostname	`ipaddr`	O endereço IP ou hostname atribuído ao dispositivo.
IP port (opcional)	`ipport`	A porta TCP a ser usada para se conectar ao dispositivo.
Login	`login`	O nome de login usado para acessar o dispositivo.
Password	`passwd`	A senha usada para autenticar a conexão ao dispositivo.
Password Script (opcional)	`passwd_script`	O script que fornece uma senha para acesso ao dispositivo de fence. Usando isto se substitui o parâmetro de **Password**.
Use SSH	`ssl`	A porta TCP a ser usada para uma conexão com o dispositivo.
Sub-Organização	`suborg`	Caminho adicional necessário para acessar a suborganização.
Número de Porta (Outlet)	`port`	Máquina Virtual
Espera de Energia	`power_wait`	Número de segundos para esperar depois de emitir um comando de ligar ou desligar.

Tabela A.7, “Dell DRAC 5” lista os parâmetros de dispositivo de fence usados para `fence_drac5`, o agente de fence para o Dell DRAC 5.

Tabela A.7. Dell DRAC 5

Campo luci	Recursos `cluster.conf`	Descrição
Name	`name`	O nome atribuído para o DRAC.
Endereço IP ou Hostname	`ipaddr`	O endereço IP ou hostname atribuído ao DRAC.
IP Port (opcional)	`ipport`	A porta TCP a ser usada para se conectar ao dispositivo.
Login	`login`	O nome de login usado para acessar o DRAC.
Password	`passwd`	A senha usada para autenticar a conexão ao DRAC.
Password Script (opcional)	`passwd_script`	O script que fornece uma senha para acesso ao dispositivo de fence. Usando isto se substitui o parâmetro de **Password**.
Use SSH	`secure`	Indica que um sistema usará o SSH para acessar o dispositivo.
Caminho para identificar o arquivo SSH	`identity_file`	O arquivo de identificação para o SSH.
Nome do Módulo	`module_name`	(opcional) O nome do módulo para o DRAC quando você tiver múltiplos módulos DRAC.
Solicitação de comando de força	`cmd_prompt`	O prompt de comando a ser usado. O valor padrão é ’\$’.
Espera de Energia	`power_wait`	Número de segundos para esperar depois de emitir um comando de ligar ou desligar.

Tabela A.8, “Eaton Network Power Controller (SNMP Interface) (Red Hat Enterprise Linux 6.4 e posteriores)” lista os parâmetros do dispositivo de fence utilizados pelo `fence_eaton_snmp`, o agente do fence para o Eaton sob o interruptor de força da rede SNMP.

Tabela A.8. Eaton Network Power Controller (SNMP Interface) (Red Hat Enterprise Linux 6.4 e posteriores)

Campo luci	Recursos `cluster.conf`	Descrição
Name	`name`	Um nome para o interruptor de força da rede Eaton conectado ao cluster.
Endereço IP ou Hostname	`ipaddr`	O endereço IP ou hostname atribuído ao dispositivo.
Porta UDP/TCP (opcional)	`udpport`	A porta UDP/TCP a ser usada para conexão com o dispositivo; o valor padrão é 161.
Login	`login`	O nome de login usado para acessar o dispositivo.
Password	`passwd`	A senha usada para autenticar a conexão ao dispositivo.
Password Script (opcional)	`passwd_script`	O script que fornece uma senha para acesso ao dispositivo de fence. Usando isto se substitui o parâmetro de **Password**.
Versão SNMP	`snmp_version`	A versão SNMP a ser usada (1, 2c, 3); o valor padrão é 1.
Comunidade SNMP	`comunidade`	A série da comunidade SNMP; o valor padrão é `private`.
Nível de segurança SNMP	`snmp_sec_level`	O nível de segurança SNMP (noAuthNoPriv, authNoPriv, authPriv).
Protocolo de autenticação do SNMP	`snmp_auth_prot`	O protocolo de autenticação SNMP (MD5, SHA).
Protocolo de privacidade do SNMP	`snmp_priv_prot`	O protocolo de privacidade SNMP (DES, AES).
Senha do Protocolo de privacidade do SNMP	`snmp_priv_passwd`	A senha do protocolo de privacidade SNMP.
Script do Protocolo de Privacidade SNMP	`snmp_priv_passwd_script`	O script que fornece uma senha para o protocolo de privacidade SNMP. Usando isto se substitui o parâmetro **SNMP privacy protocol password**.

Campo luci	Recursos `cluster.conf`	Descrição
Espera de energia (segundos)	`power_wait`	Número de segundos para esperar depois de emitir um comando de ligar ou desligar.
Número de Porta (Outlet)	`port`	Número do plug físico ou nome da máquina virtual. Este parâmetro é sempre necessário.

Tabela A.9, "Egenera SAN Controller" lista os parâmetros de dispositivo de fence usados pelo **`fence_egenera`**, agente de fence para o Egenera BladeFrame.

Tabela A.9. Egenera SAN Controller

Campo luci	Recursos `cluster.conf`	Descrição
Name	`name`	O nome para o dispositivo EGenera BladeFrame conectado ao cluster.
CServer	`cserver`	O hostname (e opcionalmente o nome de usuário no formato **`username@hostname`**) atribuído ao dispositivo. Consulte a página man fence_egenera(8) para maiores informações.
ESH Path (opcional)	`esh`	O caminho do comando esh no cserver (padrão é /opt/pan-mgr/bin/esh)
Nome de Usuário	`usuário`	O nome de login. O valor padrão é **`root`**.
lpan	`lpan`	A rede de área de processo lógico (LPAN) do dispositivo.
pserver	`pserver`	O nome do processo blade (pserver) do dispositivo.

Tabela A.10, "ePowerSwitch" lista os parâmetros de dispositivo de fence usados pelo **`fence_eps`**, o agente de fence para ePowerSwitch.

Tabela A.10. ePowerSwitch

Campo luci	Recursos `cluster.conf`	Descrição
Name	`name`	Um nome para o dispositivo ePowerSwitch conectado ao cluster.
Endereço IP ou Hostname	`ipaddr`	O endereço IP ou hostname atribuído ao dispositivo.

Campo luci	Recursos `cluster.conf`	Descrição
Login	`login`	O nome de login usado para acessar o dispositivo.
Password	`passwd`	A senha usada para autenticar a conexão ao dispositivo.
Password Script (opcional)	`passwd_script`	O script que fornece uma senha para acesso ao dispositivo de fence. Usando isto se substitui o parâmetro de **Password**.
Nome da página Escondida	`hidden_page`	O nome da página escondida para o dispositivo.
Número de Porta (Outlet)	`port`	Número de plug físico ou nome de uma máquina virtual.

Tabela A.11, "Fence virt" lista os parâmetros de dispositivo de fence usados pelo `fence_virt`, o agente de fence para um dispositivo fence Fence virt.

Tabela A.11. Fence virt

Campo luci	Recursos `cluster.conf`	Descrição
Name	`name`	Um nome para o dispositivo Fence virt fence
Dispositivo em Série	`serial_device`	No host, o dispositivo serial deve ser mapeado em cada arquivo de configuração de domínio. Para mais informações, veja a página man `fence_virt.conf`. Se este campo estiver especificado, ele faz que o agente fence `fence_virt` opere em modo serial. Sem especificar um valor fará o agente fence `fence_virt` operar no modo de canal VM.
Parâmetros em Série	`serial_params`	Os parâmetros seriais. O padrão é 115200, 8N1.
Endereço IP do Canal VM	`channel_address`	O canal IP. O valor padrão é 10.0.2.179.
Porta ou Domínio (obsoleto)	`port`	Máquina Virtual (domínio UUID ou nome) para fazer fence.
	`ipport`	A porta do canal. O valor padrão é 1229, o qual é o valor usado ao configurar este dispositivo de fence com o **luci**.

Tabela A.12, "Fujitsu Siemens Remoteview Service Board (RSB)" lista os parâmetros de dispositivo de fence usado pelo **fence_rsb**, o agente do fence para Fujitsu-Siemens RSB.

Tabela A.12. Fujitsu Siemens Remoteview Service Board (RSB)

Campo luci	Recursos cluster.conf	Descrição
Name	**name**	O nome para o RSB usar como um dispositivo fence.
Endereço IP ou Hostname	**ipaddr**	O hostname atribuído ao dispositivo.
Login	**login**	O nome de login usado para acessar o dispositivo.
Password	**passwd**	A senha usada para autenticar a conexão ao dispositivo.
Password Script (opcional)	**passwd_script**	O script que fornece uma senha para acesso ao dispositivo de fence. Usando isto se substitui o parâmetro de **Password**.
Porta TCP	**ipport**	O número de porta no qual o serviço de telnet escuta. O valor padrão é 3172

Tabela A.13, "HP BladeSystem (Red Hat Enterprise Linux 6.4 and later)" lista os parâmetros de dispositivo de fence usados pelo **fence_hpblade**, o agente de fence para os dispositivos HP BladeSystem.

Tabela A.13. HP BladeSystem (Red Hat Enterprise Linux 6.4 and later)

Campo luci	Recursos cluster.conf	Descrição
Name	**name**	O nome atribuído ao dispositivo HP Bladesystem conectado ao cluster.
Endereço IP ou Hostname	**ipaddr**	O endereço IP ou hostname atribuído ao dispositivo HP BladeSystem.
IP Port (opcional)	**ipport**	A porta TCP a ser usada para se conectar ao dispositivo.
Login	**login**	O nome de login utilizado para acessar o dispositivo HP BladeSystem. Este parâmetro é requerido.
Password	**passwd**	A senha utilizada para autenticar a conexão ao dispositivo do fence.

Campo luci	Recursos `cluster.conf`	Descrição
Password Script (opcional)	`passwd_script`	O script que fornece uma senha para acesso ao dispositivo de fence. Usando isto se substitui o parâmetro de **Password**.
Solicitação de comando de força	`cmd_prompt`	O prompt de comando a ser usado. O valor padrão é '\$'.
Porta que está faltando retorna como OFF ao invés de falha.	`missing_as_off`	Porta que está faltando retorna como OFF ao invés de falha.
Espera de Energia (segundos)	`power_wait`	Número de segundos para esperar depois de emitir um comando de ligar ou desligar.
Use SSH	`secure`	Indica que um sistema usará o SSH para acessar o dispositivo.
Caminho para identificar o arquivo SSH	`identity_file`	O arquivo de identificação para o SSH.

Tabela A.14, "HP iLO/iLO2 (Integrated Lights Out)" lista os parâmetros de dispositivo de fence usados pelo `fence_ilo`, o agente de fence para os dispositivos HP iLO.

Tabela A.14. HP iLO/iLO2 (Integrated Lights Out)

Campo luci	Recursos `cluster.conf`	Descrição
Name	`name`	Um nome para o servidor com suporte HP iLO.
Endereço IP ou Hostname	`ipaddr`	O endereço IP ou hostname atribuído ao dispositivo.
IP Port (opcional)	`ipport`	A porta TCP a ser usada para uma conexão com o dispositivo.
Login	`login`	O nome de login usado para acessar o dispositivo.
Password	`passwd`	A senha usada para autenticar a conexão ao dispositivo.

Campo luci	Recursos `cluster.conf`	Descrição
Password Script (opcional)	`passwd_script`	O script que fornece uma senha para acesso ao dispositivo de fence. Usando isto se substitui o parâmetro de **Password**.
Espera de Energia	`power_wait`	Número de segundos para esperar depois de emitir um comando de ligar ou desligar.

Tabela A.15, “HP iLO (Integrated Lights Out) MP” lista os parâmetros de dispositivo de fence usados pelo `fence_ilo_mp`, o agente de fence para os dispositivos HP iLO MP.

Tabela A.15. HP iLO (Integrated Lights Out) MP

Campo luci	Recursos `cluster.conf`	Descrição
Name	`name`	Um nome para o servidor com suporte HP iLO.
Endereço IP ou Hostname	`ipaddr`	O endereço IP ou hostname atribuído ao dispositivo.
IP Port (opcional)	`ipport`	A porta TCP a ser usada para uma conexão com o dispositivo.
Login	`login`	O nome de login usado para acessar o dispositivo.
Password	`passwd`	A senha usada para autenticar a conexão ao dispositivo.
Password Script (opcional)	`passwd_script`	O script que fornece uma senha para acesso ao dispositivo de fence. Usando isto se substitui o parâmetro de **Password**.
Use SSH	`secure`	Indica que um sistema usará o SSH para acessar o dispositivo.
Caminho para identificar o arquivo SSH	`identity_file`	O arquivo de identificação para o SSH.
Solicitação de comando de força	`cmd_prompt`	O prompt de comando a ser usado. O valor padrão é ’MP>’, ’hpiLO->’.
Espera de Energia	`power_wait`	Número de segundos para esperar depois de emitir um comando de ligar ou desligar.

Tabela A.16, “IBM BladeCenter” lista os parâmetros de dispositivo de fence usados para `fence_bladecenter`, o agente de fence para o IBM BladeCenter.

Tabela A.16. IBM BladeCenter

Campo luci	Recursos `cluster.conf`	Descrição
Name	`name`	Um nome para o dispositivo IBM BladeCenter conectado ao cluster.
Endereço IP ou Hostname	`ipaddr`	O endereço IP ou hostname atribuído ao dispositivo.
IP port (opcional)	`ipport`	A porta TCP a ser usada para uma conexão com o dispositivo.
Login	`login`	O nome de login usado para acessar o dispositivo.
Password	`passwd`	A senha usada para autenticar a conexão ao dispositivo.
Password Script (opcional)	`passwd_script`	O script que fornece uma senha para acesso ao dispositivo de fence. Usando isto se substitui o parâmetro de **Password**.
Espera de Energia	`power_wait`	Número de segundos para esperar depois de emitir um comando de ligar ou desligar.
Use SSH	`secure`	Indica que o sistema usará o SSH para acessar o dispositivo.
Caminho para identificar o arquivo SSH	`identity_file`	O arquivo de identificação para o SSH.

Tabela A.17, “IBM BladeCenter SNMP” lista os parâmetros de dispositivo de fence usados para `fence_ibmblade`, o agente de fence para o IBM BladeCenter sob SNMP.

Tabela A.17. IBM BladeCenter SNMP

Campo luci	Recursos `cluster.conf`	Descrição
Name	`name`	Um nome para o dispositivo IBM BladeCenter SNMP conectado ao cluster.
Endereço IP ou Hostname	`ipaddr`	O endereço IP ou hostname atribuído ao dispositivo.

Campo luci	Recursos `cluster.conf`	Descrição
Porta UDP/TCP (opcional)	`udpport`	A porta UDP/TCP a ser usada para conexões com o dispositivo; o valor padrão é 161.
Login	`login`	O nome de login usado para acessar o dispositivo.
Password	`passwd`	A senha usada para autenticar a conexão ao dispositivo.
Password Script (opcional)	`passwd_script`	O script que fornece uma senha para acesso ao dispositivo de fence. Usando isto se substitui o parâmetro de **Password**.
Versão SNMP	`snmp_version`	A versão SNMP a ser usada (1, 2c, 3); o valor padrão é 1.
Comunidade SNMP	`comunidade`	A séria da comunidade SNMP.
Nível de segurança SNMP	`snmp_sec_level`	O nível de segurança SNMP (noAuthNoPriv, authNoPriv, authPriv).
Protocolo de autenticação do SNMP	`snmp_auth_prot`	O protocolo de autenticação SNMP (MD5, SHA).
Protocolo de privacidade do SNMP	`snmp_priv_prot`	O protocolo de privacidade SNMP (DES, AES).
SNMP privacy protocol password	`snmp_priv_passwd`	A senha do protocolo de privacidade SNMP.
Script do Protocolo de Privacidade SNMP	`snmp_priv_passwd_script`	O script que fornece uma senha para o protocolo de privacidade SNMP. Usando isto se substitui o parâmetro **SNMP privacy protocol password**.
Espera de Energia	`power_wait`	Número de segundos para esperar depois de emitir um comando de ligar ou desligar.
Port	`port`	Número de plug físico ou nome de uma máquina virtual.

Tabela A.18, “IBM iPDU (Red Hat Enterprise Linux 6.4 e posteriores)” lista os parâmetros do dispositivo fence utilizado pelo `fence_ipdu`, o agente fence para o iPDU sobre dispositivos SNMP.

Tabela A.18. IBM iPDU (Red Hat Enterprise Linux 6.4 e posteriores)

Campo luci	Recursos `cluster.conf`	Descrição
Name	**`name`**	O nome para o dispositivo IBM iPDU conectado ao cluster no qual o daemon do fence se autentica via protocolo SNMP.
Endereço IP ou Hostname	**`ipaddr`**	O endereço IP ou hostname atribuído ao dispositivo.
UDP/TCP Port	**`udpport`**	A porta UDP/TCP a ser usada para conexão com o dispositivo; o valor padrão é 161.
Login	**`login`**	O nome de login usado para acessar o dispositivo.
Password	**`passwd`**	A senha usada para autenticar a conexão ao dispositivo.
Password Script (opcional)	**`passwd_script`**	O script que fornece uma senha para acesso ao dispositivo de fence. Usando isto se substitui o parâmetro de **Password**.
Versão SNMP	**`snmp_version`**	A versão SNMP a ser usada (1, 2c, 3); o valor padrão é 1.
Comunidade SNMP	**`comunidade`**	A série da comunidade SNMP; o valor padrão é **`private`**.
Nível de segurança SNMP	**`snmp_sec_level`**	O nível de segurança SNMP (noAuthNoPriv, authNoPriv, authPriv).
Protocolo de autenticação do SNMP	**`snmp_auth_prot`**	O Protocolo de Autenticação SNMP (MD5, SHA).
Protocolo de privacidade do SNMP	**`snmp_priv_prot`**	O protocolo de privacidade SNMP (DES, AES).
Senha do Protocolo de privacidade do SNMP	**`snmp_priv_passwd`**	A senha do protocolo de privacidade SNMP.
Script do Protocolo de Privacidade SNMP	**`snmp_priv_passwd_script`**	O script que fornece uma senha para o protocolo de privacidade SNMP. Usando isto se substitui o parâmetro **SNMP privacy protocol password**.

Campo luci	Recursos cluster.conf	Descrição
Espera de Energia	power_wait	Número de segundos para esperar depois de emitir um comando de ligar ou desligar.
Port	port	TCP port

Tabela A.19, "IF MIB" lista os parâmetros de dispositivo de fence usados pelo fence_ifmib, o agente de fence para os dispositivos IF-MIB.

Tabela A.19. IF MIB

Campo luci	Recursos cluster.conf	Descrição
Name	name	Um nome para o dispositivo IF MIB conectado ao cluster.
Endereço IP ou Hostname	ipaddr	O endereço IP ou hostname atribuído ao dispositivo.
Porta UDP/TCP (opcional)	udpport	A porta UDP/TCP a ser usada para conexão com o dispositivo; o valor padrão é 161.
Login	login	O nome de login usado para acessar o dispositivo.
Password	passwd	A senha usada para autenticar a conexão ao dispositivo.
Password Script (opcional)	passwd_script	O script que fornece uma senha para acesso ao dispositivo de fence. Usando isto se substitui o parâmetro de **Password**.
Versão SNMP	snmp_version	A versão SNMP a ser usada (1, 2c, 3); o valor padrão é 1.
Comunidade SNMP	comunidade	A séria da comunidade SNMP.
Nível de segurança SNMP	snmp_sec_level	O nível de segurança SNMP (noAuthNoPriv, authNoPriv, authPriv).
Protocolo de autenticação do SNMP	snmp_auth_prot	O protocolo de autenticação SNMP (MD5, SHA).

Campo luci	Recursos `cluster.conf`	Descrição
Protocolo de privacidade do SNMP	`snmp_priv_prot`	O protocolo de privacidade SNMP (DES, AES).
Senha do Protocolo de privacidade do SNMP	`snmp_priv_passwd`	A senha do protocolo de privacidade SNMP.
Script do Protocolo de Privacidade SNMP	`snmp_priv_passwd_script`	O script que fornece uma senha para o protocolo de privacidade SNMP. Usando isto se substitui o parâmetro **SNMP privacy protocol password**.
Espera de Energia	`power_wait`	Número de segundos para esperar depois de emitir um comando de ligar ou desligar.
Port	`port`	Número de plug físico ou nome de uma máquina virtual.

Tabela A.20, “Intel Modular” lista os parâmetros de dispositivo de fence usados pelo `fence_intelmodular`, o agente de fence para o Intel Modular.

Tabela A.20. Intel Modular

Campo luci	Recursos `cluster.conf`	Descrição
Name	`name`	Um nome para o dispositivo Intel Modular conectado ao cluster.
Endereço IP ou Hostname	`ipaddr`	O endereço IP ou hostname atribuído ao dispositivo.
Login	`login`	O nome de login usado para acessar o dispositivo.
Password	`passwd`	A senha usada para autenticar a conexão ao dispositivo.
Password Script (opcional)	`passwd_script`	O script que fornece uma senha para acesso ao dispositivo de fence. Usando isto se substitui o parâmetro de **Password**.
Versão SNMP	`snmp_version`	A versão SNMP a ser usada (1, 2c, 3); o valor padrão é 1.

Campo luci	Recursos `cluster.conf`	Descrição
Comunidade SNMP	`comunidade`	A série da comunidade SNMP; o valor padrão é `private`.
Nível de segurança SNMP	`snmp_sec_level`	O nível de segurança SNMP (noAuthNoPriv, authNoPriv, authPriv).
Protocolo de autenticação do SNMP	`snmp_auth_prot`	O protocolo de autenticação SNMP (MD5, SHA).
Protocolo de privacidade do SNMP	`snmp_priv_prot`	O protocolo de privacidade SNMP (DES, AES).
Senha do Protocolo de privacidade do SNMP	`snmp_priv_passwd`	A senha do protocolo de privacidade SNMP.
Script do Protocolo de Privacidade SNMP	`snmp_priv_passwd_script`	O script que fornece uma senha para o protocolo de privacidade SNMP. Usando isto se substitui o parâmetro **SNMP privacy protocol password**.
Espera de Energia	`power_wait`	Número de segundos para esperar depois de emitir um comando de ligar ou desligar.
Port	`port`	Número de plug físico ou nome de uma máquina virtual.

Tabela A.21, “IPMI (Intelligent Platform Management Interface) LAN” lista os parâmetros de dispositivo de fence usados pelo `fence_ipmilan`, o agente de fence para os dispositivos IPMI sob LAN.

Tabela A.21. IPMI (Intelligent Platform Management Interface) LAN

Campo luci	Recursos `cluster.conf`	Descrição
Name	`name`	Um nome para o dispositivo IPMI LAN conectado ao cluster.
Endereço IP ou Hostname	`ipaddr`	O endereço IP ou hostname atribuído ao dispositivo.
Login	`login`	O nome de login para um usuário ser capaz de emitir comando de ligar e desligar à porta IPMI dada.

Campo luci	Recursos `cluster.conf`	Descrição
Password	`passwd`	A senha a ser usada para autenticar a conexão à porta IPMI.
Password Script (opcional)	`passwd_script`	O script que fornece uma senha para acesso ao dispositivo de fence. Usando isto se substitui o parâmetro de **Password**.
Authentication Type	`auth`	IPMI LAN authentication type: `none`, `password`, or `md5`.
Use Lanplus	`lanplus`	`True` ou `1`. Se em branco, então o valor é `False`.
Ciphersuite to use	`cipher`	A autenticação remota do servidor, intergridade e algorítmos de encriptação para usado para conexões IPMIv2 lanplus.
Nível de Privilégio	`privlvl`	O nível de privilégio no dispositivo IPMI,

Tabela A.22, "RHEV-M REST API (RHEL 6.2 e posteriores em RHEV 3.0 e posteriores)" lista os parâmetros de dispositivo de fence usados para `fence_rhevm`, o agente de fence para o RHEV-M REST API.

Tabela A.22. RHEV-M REST API (RHEL 6.2 e posteriores em RHEV 3.0 e posteriores)

Campo luci	Recursos `cluster.conf`	Descrição
Name	`name`	Um nome para o dispositivo de fence RHEV-M REST API
Endereço IP ou Hostname	`ipaddr`	O endereço IP ou hostname atribuído ao dispositivo.
IP Port (opcional)	`ipport`	A porta TCP a ser usada para uma conexão com o dispositivo.
Login	`login`	O nome de login usado para acessar o dispositivo.
Password	`passwd`	A senha usada para autenticar a conexão ao dispositivo.
Password Script (opcional)	`passwd_script`	O script que fornece uma senha para acesso ao dispositivo de fence. Usando isto se substitui o parâmetro de **Password**.
Use SSH	`ssl`	A porta TCP a ser usada para uma conexão com o dispositivo.

Campo luci	Recursos cluster.conf	Descrição
Espera de Energia	power_wait	Número de segundos para esperar depois de emitir um comando de ligar ou desligar.
Port	port	Número de plug físico ou nome de uma máquina virtual.

Tabela A.23, "SCSI Fencing" lista os parâmetros de dispositivo de fence usados pelo `fence_scsi`, o agente de fence para as reservas persistentes do SCSI.

NOTA

Uso de reservas persistentes SCSI como um método fence é suportado com as seguintes limitações:

- Quando usar um fence SCSI, todos os nós no cluster devem se registrar com o os mesmos dispositivos para que cada nó possa remover a chave de registro de outros nós a partir de todos os dispositivos em que se está registrado.
- Dispositivos usados para os volumes de cluster devem ser um LUN completo, não partições. Reservas persistentes SCSI trabalham somente em um LUN inteiro, significando que o acesso é controlado para cada LUN, não partições individuais.

Tabela A.23. SCSI Fencing

Campo luci	Recursos cluster.conf	Descrição
Name	name	Um nome para o dispositivo SCSI fence
Node name		
Chave para ação atual		(sobrescreve o nome do nó)

Tabela A.24, "Fencing do VMware (Interface SOAP) (Red Hat Enterprise Linux 6.2 e posterior)" lista os parâmetros de dispositivo de fence usados para `fence_vmware_soap`, o agente de fence para o VMWARE sob SOAP API.

Tabela A.24. Fencing do VMware (Interface SOAP) (Red Hat Enterprise Linux 6.2 e posterior)

Campo luci	Recursos cluster.conf	Descrição
Name	name	Um nome para o dispositivo Fence virt fence

Campo luci	Recursos `cluster.conf`	Descrição
Endereço IP ou Hostname	**ipaddr**	O endereço IP ou hostname atribuído ao dispositivo.
IP Port (opcional)	**ipport**	A porta TCP a ser usada para uma conexão com o dispositivo.
Login	**login**	O nome de login usado para acessar o dispositivo.
Password	**passwd**	A senha usada para autenticar a conexão ao dispositivo.
Password Script (opcional)	**passwd_script**	O script que fornece uma senha para acesso ao dispositivo de fence. Usando isto se substitui o parâmetro de **Password**.
Separador	**separador**	Separador para o CSV criado pela lista de operação. O valor padrão é uma vírgula (,).
Espera de Energia	**power_wait**	Número de segundos para esperar depois de emitir um comando de ligar ou desligar.
Nome VM	**port**	Nome da máquina virtual no formato de caminho do inventário (ex.: /datacenter/vm/Discovered_virtual_machine/myMachine).
VM UUID	**uuid**	O UUID da máquina virtual para fazer o fence.
Use SSH	**ssl**	A porta TCP a ser usada para uma conexão com o dispositivo.

Tabela A.10, “ePowerSwitch” lista os parâmetros de dispositivo de fence usados pelo **fence_wti**, o agente de fence para o interruptor de energia de rede WTI .

Tabela A.25. WTI Power Switch

Campo luci	Recursos `cluster.conf`	Descrição
Name	**name**	Um nome para o WTI power switch conectado ao cluster.
Endereço IP ou Hostname	**ipaddr**	O endereço IP ou hostname atribuídos ao dispositivo.
IP Port (opcional)	**ipport**	A porta TCP a ser usada para se conectar ao dispositivo.

Campo luci	Recursos `cluster.conf`	Descrição
Login	`login`	O nome de login usado para acessar o dispositivo.
Password	`passwd`	A senha usada para autenticar a conexão ao dispositivo.
Password Script (opcional)	`passwd_script`	O script que fornece uma senha para acesso ao dispositivo de fence. Usando isto se substitui o parâmetro de **Password**.
Port	`port`	Número de plug físico ou nome de uma máquina virtual.
Force command prompt	`cmd_prompt`	O prompt de comando a ser usado. O valor padrão é ['RSM>', '>MPC', 'IPS>', 'TPS>', 'NBB>', 'NPS>', 'VMR>']
Espera de Energia	`power_wait`	Número de segundos para esperar depois de emitir um comando de ligar ou desligar.
Use SSH	`secure`	Indica que o sistema usará o SSH para acessar o dispositivo.
Caminho para identificar o arquivo SSH	`identity_file`	O arquivo de identificação para o SSH.

APÊNDICE B. PARÂMETROS DOS RECURSOS DE ALTA DISPONIBILIDADE

Este apêndice fornece descrições dos parâmetros de recursos de HA (Alta Disponibilidade). Você pode configurar os parâmetros com o **luci**, usando o comando `ccs` ou editando o `etc/cluster/cluster.conf`. O Tabela B.1, "Resumo de Recursos HA (Alta Disponibilidade)" lista os recursos, seus agentes de recursos correspondentes e referências para outras tabelas contendo descriços de parâmetros. Para entender os agentes de recursos com mais detalhes você pode vizualiza-los no `/usr/share/cluster` em qualquer nó no cluster.

Além dos agentes de recursos descritos neste apêndice, o diretório `/usr/share/cluster` inclui um script do OCF dummy para um grupo de recursos, `service.sh`. Para mais informações sobre os parâmetros incluídos neste script, consulte o próprio `service.sh`.

Para uma lista compreensiva e descrição dos elementos e atributos do `cluster.conf`, consulte o esquema de cluster em `/usr/share/cluster/cluster.rng`, e o esquema anotado em `/usr/share/doc/cman-X.Y.ZZ/cluster_conf.html` (por exemplo `/usr/share/doc/cman-3.0.12/cluster_conf.html`).

Tabela B.1. Resumo de Recursos HA (Alta Disponibilidade)

Recurso	Agente de Recursos	Descrição de Parâmetro de Referência
Apache	apache.sh	Tabela B.2, "Servidor Apache"
Instância de Condor	condor.sh	Tabela B.3, "Instância de Condor"
Sistema de Arquivo	fs.sh	Tabela B.4, "Sistema de Arquivo"
Sistema de Arquivo GFS2	clusterfs.sh	Tabela B.5, "GFS2"
Endereço IP	ip.sh	Tabela B.6, "Endereço IP"
HA LVM	lvm.sh	Tabela B.7, "HA LVM"
MySQL	mysql.sh	Tabela B.8, "MySQL"
NFS Client	nfsclient.sh	Tabela B.9, "NFS Client"
Exportar NFS	nfsexport.sh	Tabela B.10, "Exportar NFS"
NFS Server	nfsserver.sh	Tabela B.11, "NFS Server"
NFS/CIFS Mount	netfs.sh	Tabela B.12, "NFS/CIFS Mount"
Open LDAP	openldap.sh	Tabela B.13, "Open LDAP"

Recurso	Agente de Recursos	Descrição de Parâmetro de Referência
Oracle 10g/11g Instância de Failover	oracledb.sh	Tabela B.14, “Oracle 10g/11G Instância do Failover”
Oracle 10g Instância de Failover	orainstance.sh	Tabela B.15, “Oracle 10g Instância de Failover”
Oracle 10g Listener	oralistener.sh	Tabela B.16, “Oracle 10g Listener”
PostgreSQL 8	postgres-8.sh	Tabela B.17, “PostgreSQL 8”
SAP Database	SAPDatabase	Tabela B.18, “SAP Database”
Instância SAP	SAPInstance	Tabela B.19, “Instância SAP”
Samba	samba.sh	Tabela B.20, “Servidor Samba”
Script	script.sh	Tabela B.21, “Script”
Sybase ASE	ASEHAagent.sh	Tabela B.22, “Instância do Failover do ASE Sybase”
Tomcat 6	tomcat-6.sh	Tabela B.23, “Tomcat 6”
Máquina Virtual	vm.sh	Tabela B.24, “Máquina Virtual” Nota: o **luci** exibe isso como uma máquina virtual se o cluster do host pode suportar máquinas virtuais.

Tabela B.2. Servidor Apache

Campo luci	Atributo `cluster.conf`	Descrição
Nome	`nome`	O nome do Serviço Apache
Servidor Root	`server_root`	O valor padrão é `/etc/httpd`.
Arquivo Config	`config_file`	Especifica o arquivo de configuração do Apache. O valor padrão é `/etc/httpd/conf`.
Opções de httpd	`httpd_options`	Outras opções de linha de comando para o `httpd`.
Espera de fechamento (segundos)	`shutdown_wait`	Especifica o número de segundos para esperar pelo término correto de desligamento do serviço.

Tabela B.3. Instância de Condor

Campo	Campo luci	Atributo `cluster.conf`
Nome da Instância	**`nome`**	Especifica um nome único para a instância Condor.
Tipo de Subsistema do Condor	**`tipo`**	Especifica o tipo de subsistema do Condor para esta instância: **`schedd`**, **`job_server`**, ou **`query_server`**.

Tabela B.4. Sistema de Arquivo

Campo luci	Atributo `cluster.conf`	Descrição
Nome	**`nome`**	Especifica um nome para o recurso do sistema de arquivo.
Tipo de Sistema de Arquivo	**`fstype`**	Se não especificado, o **`mount`** tenta determinar o tipo de sistema de arquivo.
Mount Point	**`mountpoint`**	O caminho na hierarquia do sistema de arquivo para montar este sistema de arquivo.
Dispositivo, Rótulo FS, ou UUID	**`dispositivo`**	Especifica o dispositivo associado com o recurso de sistema de arquivo. Este pode ser um dispositivo de bloco, rótulo do sistema de arquivo ou o UUID de um sistema de arquivo.
Opções de montagem.	**`opções`**	Opções de montagem; que são opções usadas quando um sistema de arquivo é montado. Este pode ser um especifico sistema de arquivo. Consulte a página man ***`mount`*** *(8)* para opções de montagem suportadas.
File System ID (opcional)	**`fsid`**	 **NOTA** ***`ID do Sistema de Arquivo`*** é usado somente pelos serviços NFS. Quando criar um novo recurso de sistema de arquivo, você pode deixar este campo em branco. Deixando o campo em branco faz o ID do sistema de arquivos ser atribuido automaticamente depois de você enviar o parâmetro durante a configuração. Se você precisar atribuir um ID de sistema de arquivo explicitamente, especifique-o neste campo.

Campo luci	Atributo cluster.conf	Descrição
Forçar Desmontagem (Force Unmount)	**force_unmount**	Se habilitado, força o sistema de arquivos a desmontar. A configuração padrão é ***disabled*** (desabilitado). O ***Force Unmount*** (forçar desmontagem) termina todos os processos usando o ponto de montagem para liberar a montagem quando tentar desmontar.
Forçar fsck	**force_fsck**	Se habilitado, faz que o **fsck** ser executado no sistema de arquivo antes de monta-lo. A definição padrão é ***disabled*** (desabilitado).
Habilita o daemon do NFS e reparo bloqueado (Red Hat Enterprise Linux 6.4 e posteriores)	**nfsrestart**	Se seu sistema de arquivo é exportado via NFS e raramente falha em desmontar (tanto ao fechar quanto na realocação do serviço), configurar esta opção irá apagar todas as referências do sistema de arquivo antes de desmontar a operação. Configurar esta opção requer que você habilite a opção **Force unmount** e não deve ser utilizada junto com o recurso **NFS Server**. Você deve configurar esta opção como último recurso, pois é uma tentatica muito difíciç de descmontar um sistema de arquivo.
Usar as Verificações do Quick Status	**quick_status**	Caso esteja habilitado, realizar verificações do quick status.
Reinicializar o Nó da Máquina Caso a Desmontagem Falhar	**self_fence**	Se habilitado, renicializa o nó caso a desmontagem deste sistema de arquivo falhar. O agente de recurso **filesystem** aceita o valor 1, **yes**, **on**, ou **true** para habilitar este parâmetro e um valorf 0,**no**, **off**, ou **false** para desabilitá-lo. A configuração padrão é ***disabled***.

Tabela B.5. GFS2

Campo luci	Atributo cluster.conf	Descrição
Nome	**nome**	O nome do recurso do sistema de arquivo.
Mount Point	**mountpoint**	O caminho o qual o recursos do sistema de arquivo está montado.
Dispositivo, Rótulo FS, ou UUID	**dispositivo**	O arquivo de dispositivo associado com o recurso do sistema de arquivo.
Tipo de Sistema de Arquivo	**fstype**	Definir para GFS2 no **luci**

Campo luci	Atributo `cluster.conf`	Descrição
Opções de montagem.	**`opções`**	Opções de montagem.
File System ID (opcional)	**`fsid`**	 **NOTA** ***`ID do Sistema de Arquivo`*** é usado somente pelos serviços NFS. Quando criar um novo recurso GFS2, você pode deixar este campo em branco. Deixando este campo em branco faz com que o ID do sistema de arquivo ser atribuído automaticamente depois de você enviar o parâmetro durante a configuração. Se você precisar atribuir um ID do sistema de arquivo explicitamente, especifique neste campo.
Forçar Desmontagem (Force Unmount)	**`force_unmount`**	Se habilitado, força o sistema de arquivos desmontar. A definição padrão é ***`disabled`*** (desabilitado). O ***`Force Unmount`*** termina todos os processos usando o ponto de montagem para liberar a montagem quando ele tentar a desmontagem. Com recursos GFS2, o ponto de montagem *não é* desmontado no desmantelamento do serviço a menos que ***`Force Unmount`*** esteja *desabilitado* (habilitado).
Habilita o daemon do NFS e reparo bloqueado (Red Hat Enterprise Linux 6.4 e posteriores)	**`nfsrestart`**	Se seu sistema de arquivo é exportado via NFS e raramente falha em desmontar (tanto ao fechar quanto na realocação do serviço), configurar esta opção irá apagar todas as referências do sistema de arquivo antes de desmontar a operação. Configurar esta opção requer que você habilite a opção **`Force unmount`** e não deve ser utilizada junto com o recurso **`NFS Server`**. Você deve configurar esta opção como último recurso, pois é uma tentatica muito difíciç de descmontar um sistema de arquivo.
Reinicializar o Nó da Máquina Caso a Desmontagem Falhar	**`self_fence`**	Se habilitado, e a desmontagem deste sistema de arquivo falhar, o nó irá reinicializar imediatamente. Geralmente, isto é usado com o suporte do forçar demontagem, mas não é necessário. O agente de recurso **GFS2** aceita o valor 1, **`yes`**, **on**, ou **`true`** para habilitar este parâmetro e um valorf 0, **no**, **`off`**, ou **`false`** para desabilitá-lo.

Tabela B.6. Endereço IP

Campo luci	Atributo `cluster.conf`	Descrição

Campo luci	Atributo cluster.conf	Descrição
Endereço IP, Netmask Bits	**endereço**	O endereço IP (e, opcionalmente os bits de netmask) para o recurso. Os bits de netmask, ou comprimento de prefixo de rede, podem vir depois do endereço com uma barra como um separador, condizente com a anotação do CIDR (por exemplo, 10.1.1.1/8). Este é um endereço IP virtual. Os endereços IPv4 são suportados, assim como acontece no monitoramento de link do NIC para cada endereço IP.
Monitor Link	**monitor_link**	Habilitando este faz com que a verificação do estado falhe se o link no NIC no qual o endereço de IP estiver ligado não está presente.
Desabilitar Atualizações para Rotas Estáticas	**disable_rdisc**	Desabilitar atualização de roteamento usando o protocolo RDISC
Número de Segundos para Esperar Após Remoção do Endereço IP	**sleeptime**	Especifica quanto tempo esperar (em segundos)

Tabela B.7. HA LVM

Campo luci	Atributo cluster.conf	Descrição
Nome	**nome**	Um nome único para este recurso LVM.
Volume Group Name	**vg_name**	Um nome descritivo do grupo de volume sendo gerenciado.
Logical Volume Name (opcional)	**lv_name**	O nome do volume lógico sendo gerenciado. Este parâmetro é opcional se não há mais que um volume lógico no grupo de volume sendo gerenciado.
Fazer um fence do Nó caso seja incapaz de limpar as marcações UP LVM.	**self_fence**	Faça um fence no nó caso não consiga limpar as marcações do LV. O agente de recurso do LVM aceita um valor de 1 ou **yes** para habilitar este parâmetro, e um valor de 0 ou **no** para desabilitá-lo.

Tabela B.8. MySQL

Campo luci	Atributo `cluster.conf`	Descrição
Nome	**`nome`**	Especifica um nome do recurso do servidor MySQL.
Arquivo Config	**`config_file`**	Especifica o arquivo de configuração. O valor padrão é `/etc/my.cnf`.
Listen Address	**`listen_address`**	Especifica o endereço de IP para o servidor MySQL. Se um endereço de IP não é fornecido, o primeiro endereço IP para o serviço é tomado.
mysqld Options	**`mysqld_options`**	Outras opções de linha de comando para o **`httpd`**.
Iniciar Espera (segundos)	**`startup_wait`**	Especifica o número de segundos para esperar pelo término correto da inicialização do serviço.
Espera de fechamento (segundos)	**`shutdown_wait`**	Especifica o número de segundos para esperar pelo término correto de desligamento do serviço.

Tabela B.9. NFS Client

Campo luci	Atributo `cluster.conf`	Descrição
Nome	**`nome`**	Este é um nome simbólico de um cliente usado para referenciar-lo na árvore de recursos. Este *não é* a mesma coisa que a opção***`Target`***.
Target Hostname, Wildcard, ou Netgroup	**`target`**	Este é o servidor no qual você está montando. Ele pode ser especificado usando um nome de host, um curinga (baseado em endereço IP e nome de host), ou um netgroup definindo um host ou hosts para serem exportados.
Permitir Recuperação deste Cliente NFS	**`allow_recover`**	Permitir Recuperação.
Options	**`opções`**	Define uma lista de opções para este cliente — por exemplo, direitos de acesso do cliente adicional. Para mais informações, consulte a página man ***`exports`*** *(5), Opções Gerais.*

Tabela B.10. Exportar NFS

Campo luci	Atributo cluster.conf	Descrição
Nome	nome	Nome do descritivo do recurso. O recurso de exportação NFS certifica que os daemons NFS estejam rodando. Ele é totalmente reusável, tipicamente, somente um recurso de Exportação NFS é necessário.  **NOTA** Nomeie o recurso de Exportação NFS para que seja claramente distinguível de outros recursos NFS.

Tabela B.11. NFS Server

Campo luci	Atributo cluster.conf	Descrição
Nome	nome	O nome descriptivo do recurso de servidor do NFS. O recurso de servidor NFS é útil para exportar o sistema de arquivo NFSv4 para clientes. Por causa da forma que o NFSv4 funciona, somente um recurso NFSv4 pode existir em um servidor por vez. Além disso, não é possível utilizar o recurso do servidor NFS quando estiver utilizando também instâncias locais do NFS em cada nó de cluster.

Tabela B.12. NFS/CIFS Mount

Campo luci	Atributo cluster.conf	Descrição
Nome	nome	Nome simbólico para a montagem NFS ou CIFS.  **NOTA** Este recurso é requerido quando um serviço de cluster estiver configurado para ser um cliente NFS.
Mount Point	mountpoint	O caminho no qual o recurso do sistema de arquivo é montado
Host	host	Endereço de IP do servidor NFS/CIFS ou hostname.
Nome do Diretório de Exportação de NFS ou compartilhar CIFS	export	Nome do diretório de exportação NFS ou nome de compartilhamento CIFS.

Campo luci	Atributo cluster.conf	Descrição
Tipo do sistema de arquivo	**fstype**	Tipo do sistema de arquivo: • ***NFS*** – Especifica o uso da versão NFS padrão. Este é a configuração padrão. • ***NFS v4*** – Especifica o uso do protocolo NFSv4. • ***CIFS*** – Especifica o uso do protocolo CIFS.
Forçar Desmontagem (Force Unmount)	**force_unmount**	Se o ***Force Unmount*** estiver habilitado, o cluster termina todos os processos usando o sistema de arquivo quando o serviço é parado. Terminando todos processos usando o sistema de arquivo libera o sistema de arquivo. De outra maneira, a desmontagem falhará e o serviço será reinicializado.
Não desmonte o sistema de arquivo durante a interrupção da operação de relocação.	**no_unmount**	Se habilitado, especifica que o sistema de arquivo não deve ser desmontado durante uma parada ou operação de realocação.
Options	**opções**	Opções de montagem. Especifica uma lista de opções de montagem. Se nenhuma é especificada, o sistema de arquivo é montado **-o sync**.

Tabela B.13. Open LDAP

Campo luci	Atributo cluster.conf	Descrição
Nome	**nome**	Especifica um nome de serviço para o log e outros propósitos.
Arquivo Config	**config_file**	Especifica o caminho absoluto do arquivo de configuração. O valor padrão é **/etc/openldap/slapd.conf**.
URL List	**url_list**	O valor padrão é **ldap:///**.
slapd Options	**slapd_options**	Outras opções da linha de comando para o **slapd**.
Espera de fechamento (segundos)	**shutdown_wait**	Especifica o número de segundos para esperar pelo término correto de desligamento do serviço.

Tabela B.14. Oracle 10g/11G Instância do Failover

Campo luci	Atributo `cluster.conf`	Descrição
Instance name (SID) of Oracle instance	**`nome`**	Nome da Instância.
Nome do Usuário da Oracle	**`user`**	Este é o nome de usuário do usuário Oracle de que como a instância Oracle AS roda.
Diretório Home do Aplicativo Oracle	**`home`**	Este é o diretório home do Oracle (aplicativo, não usuário). Ele é configurado quando você instala o Oracle.
Tipo de Instalação da Oracle	**`tipo`**	O tipo de instalação da Oracle. Padrão: **`10g`**, Instância de Database e Listener somente. **`base`**, Database, Listener, Gerenciador de Empresas e ISQL*Plus: **`base-em`** (ou **`10g`**), ou Servidor de Aplicativo de Internet (Infraestrutura): **`ias`** (ou **`10g-ias`**).
Virtual hostname (opcional)	**`vhost`**	O Hostname Virtual corresponde com a instalação hostname do Oracle 10g. Note que durante o início/parada de um recurso oracledb, seu hostname é alterado temporariamente para este hostname. Portanto, você deve configurar um recurso oracledb como parte de um serviço exclusivo somente.

Tabela B.15. Oracle 10g Instância de Failover

Campo luci	Atributo `cluster.conf`	Descrição
Instance name (SID) of Oracle instance	**`nome`**	Nome da Instância.
Nome do Usuário da Oracle	**`user`**	Este é o nome do usuário da Oracle com o qual a instância Oracle AS é executada.
Diretório Home do Aplicativo Oracle	**`home`**	Este é o diretório home do Oracle (aplicativo, não usuário). Ele é configurado quando você instala o Oracle.

Campo luci	Atributo cluster.conf	Descrição
Lista de Ouvintes (Listeners) da Oracle (opcional, separado por espaços)	listeners	Lista de ouvintes (listeners) da Oracle que será iniciado com a instância do banco de dados. Os nomes dos listeners são separados por um espaço em branco. Padrão definido para vazio que desabilita os listeners.
Caminho para Bloquear Arquivo (opcional)	lockfile	Localização do lockfile que será usado para verificar se a Oracle deve estar sendo executada ou não. Padrão definido para local sob o /tmp.

Tabela B.16. Oracle 10g Listener

Campo luci	Atributo cluster.conf	Descrição
Nome do Listener	nome	Nome do Listener.
Nome do Usuário da Oracle	user	Este é o nome do usuário da Oracle com o qual a instância Oracle AS é executada.
Diretório Home do Aplicativo Oracle	home	Este é o diretório home do Oracle (aplicativo, não usuário). Ele é configurado quando você instala o Oracle.

Tabela B.17. PostgreSQL 8

Campo luci	Atributo cluster.conf	Descrição
Nome	nome	Especifica um nome de serviço para o log e outros propósitos.
Arquivo Config	config_file	Defina um caminho absoluto para o arquivo de configuração. O valor padrão é /var/lib/pgsql/data/postgresql.conf.
Postmaster User	postmaster_user	O usuário que roda o servidor de banco de dados porque ele não pode ser rodado pelo root. O valor padrão é postgres.

Campo luci	Atributo cluster.conf	Descrição
Postmaster Options	**postmaster_options**	Outras opções da linha de comando para o postmaster.
Espera de fechamento (segundos)	**shutdown_wait**	Especifica o número de segundos para esperar pelo término correto de desligamento do serviço.

Tabela B.18. SAP Database

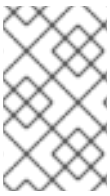
Campo luci	Atributo cluster.conf	Descrição
SAP Database Name	**SID**	Especifica um identificador de sistema SAP único. Por exemplo, P01.
Diretório de executáveis do SAP	**DIR_EXECUTABLE**	Especifica o caminho totalmente qualificado para o **sapstartsrv** e **sapcontrol**.
Tipo de Banco de Dados	**DBTYPE**	Especifica um dos seguintes tipos de banco de dados: Oracle, DB6 ou ADA.
Nome do Listener da Oracle	**NETSERVICENAME**	Especifica o ouvinte do Oracle TNS.
A pilha ABAP não está instalada, Somente a Pilha do Java está instalada.	**DBJ2EE_ONLY**	Se você não tem um stack ABAP instalado no banco de dados SAP, habilite este parâmetro.
Monitoramento do Nível de Aplicativo.	**STRICT_MONITORING**	Ativa o monitoramento de nível do aplicativo.
Inicia Recuperação Automaticamente	**AUTOMATIC_RECOVER**	Habilitar ou desabilitar inicialização de recuperação automática.
Caminho para o Java SDK	**JAVE_HOME**	Caminho para o Java SDK.

Campo luci	Atributo cluster.conf	Descrição
Nome do Arquivo do JDBC Driver.	**DB_JARS**	O nome do arquivo do driver JDBC
Caminho para Pré-Iniciar o Script.	**PRE_START_USEREXIT**	Caminho para pré-iniciar script.
Caminho para pós-início do Script.	**POST_START_USEREXIT**	Caminho para um pós-início script.
Caminho para um pré-interromper Script.	**PRE_STOP_USEREXIT**	Caminho para um script de pré-interrupção
Caminho para um Script de pós-interrupção	**POST_STOP_USEREXIT**	Caminho para um script de pós-interrupção
Diretório de bootstrap de Instância J2EE	**DIR_BOOTSTRAP**	O caminho totalmente qualificado para o diretório bootstrap da instância J2EE. Por exemplo, **/usr/sap/P01/J00/j2ee/cluster/bootstrap**.
Caminho de armazenamento de segurança J2EE	**DIR_SECSTORE**	O caminho totalmente qualificado do diretório de armazenamento de segurança J2EE. Por exemplo, **/usr/sap/P01/SYS/global/security/lib/tools**.

Tabela B.19. Instância SAP

Campo luci	Atributo cluster.conf	Descrição
SAP Instance Name	**InstanceName**	O nome totalmente qualificado da instância SAP. Por exemplo, P01_DVEBMGS00_sapp01ci.
Diretório de executáveis do SAP	**DIR_EXECUTABLE**	O caminho totalmente qualificado para o **sapstartsrv** e **sapcontrol**.

Campo luci	Atributo cluster.conf	Descrição
Directory containing the SAP START profile	**DIR_PROFILE**	O caminho totalmente qualificado para o perfil SAP START.
Nome do perfil SAP START	**START_PROFILE**	Especifica o nome do perfil SAP START.
Número de Segundos para Esperar Antes de Verificar o Status de Inicialização	**START_WAITTIME**	Especifica o número de segundos para esperar antes do status de inicialização (não esperar pelo J2EE-Admin).
Habilitar Recuperação de Inicialização Automático	**AUTOMATIC_RECOVER**	Habilitar ou desabilitar inicialização de recuperação automática.
Caminho para Pré-Iniciar o Script.	**PRE_START_USEREXIT**	Caminho para pré-iniciar script.
Caminho para pós-início do Script.	**POST_START_USEREXIT**	Caminho para um pós-início script.
Caminho para um pré-interromper Script.	**PRE_STOP_USEREXIT**	Caminho para um script de pré-interrupção
Caminho para um Script de pós-interrupção	**POST_STOP_USEREXIT**	Caminho para um script de pós-interrupção



NOTA

Em relação á Tabela B.20, “Servidor Samba”, quando criar ou editar um serviço de cluster, conecte um recurso de serviço Samba diretamente ao serviço, *não* a um recurso dentro de um serviço.

Tabela B.20. Servidor Samba

Campo luci	Atributo `cluster.conf`	Descrição
Nome	**`nome`**	Especifica o nome do servidor Samba.
Arquivo Config	**`config_file`**	O arquivo de configuração do Samba
Outras opções de Linha de Comando para o smbd	**`smbd_options`**	Outras opções da linha de comando para smbd.
Outras opções da linha de comando para nmbd.	**`nmbd_options`**	Outras opções da linha de comando para o nmbd.
Espera de fechamento (segundos)	**`shutdown_wait`**	Especifica o número de segundos para esperar pelo término correto de desligamento do serviço.

Tabela B.21. Script

Campo luci	Atributo `cluster.conf`	Descrição
Nome	**`nome`**	Especifica o nome para o script de usuário personalizado. O recurso de script permite um script init compatível LSB padrão para ser usado para iniciar um serviço clusterizado.
Caminho completa para o Arquivo do Script	**`file`**	Digite o caminho onde este script personalizado está localizado (por exemplo, **`/etc/init.d/`*`userscript`***).

Tabela B.22. Instância do Failover do ASE Sybase

Campo luci	Atributo `cluster.conf`	Descrição
Nome da Instância	**`nome`**	Especifica o nome da instância do recurso Sybase ASE
Nome do Servidor ASE	**`server_name`**	O nome do servidor ASE que é configurado para o serviço HA.

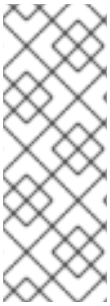
Campo luci	Atributo cluster.conf	Descrição
Diretório do Home SYBASE	sybase_home	O diretório home dos produtos Sybase.
Arquivo de Login	login_file	O caminho completo do arquivo de login que contém o par de senha do login.
Arquivos de Interface	interfaces_file	O caminho completo do arquivo de interface que é usado para iniciar/acessar o servidor ASE.
Nome de Diretório SYBASE_ASE	sybase_ase	O nome do diretório sob o sybase_home onde os produtos ASE estão instalados.
Nome do diretório SYBASE_OCS	sybase_ocs	O nome do diretório sob o sybase_home onde os produtos OCS estão instalados. Por exemplo, ASE-15_0.
Usuário Sybase	sybase_user	O usuário que pode rodar o servidor ASE.
Iniciar Timeout (segundos)	start_timeout	O valor do O valor de expiração de sessão da afiliação CMAN
Fechamento do Timeout (segundos)	shutdown_timeout	O valor de fechamento do timeout
Timeout de análise profunda	deep_probe_timeout	O número máximo de segundos para esperar pela resposta do servidor ASE antes de determinar que o servidor não tinha resposta enquanto rodando a deep probe.

Tabela B.23. Tomcat 6

Campo luci	Atributo cluster.conf	Descrição
Nome	nome	Especifica um nome de serviço para o log e outros propósitos.
Arquivo Config	config_file	Especifica o caminho absoluto para a configuração do arquivo. O valor padrão é `/etc/tomcat6/tomcat6.conf`.
Espera de fechamento (segundos)	shutdown_wait	Especifica o número de segundos para esperar pelo término correto do desligamento do serviço. O padrão é 30.

IMPORTANTE

A respeito do Tabela B.24, “Máquina Virtual” , quando você configurar seu cluster com os recursos da máquina virtual, você deve usar as ferramentas do **rgmanager** para iniciar e interromper as máquinas virtuais. Usar o **virsh** para iniciar a máquina pode resultar na execução da máquina virtual em mais de um local, o qual pode causar a danos de dados na máquina virtual. Para informações sobre configuração de seu sistema para reduzir as chances de administradores acidentalmente "duplo-início" máquinas virtuais por usar ambos cluster e ferramentas sem cluster, consulte o Seção 2.14, “Configurando as Máquinas Virtuais em um Ambiente Cluster” .



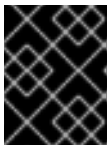
NOTA

Recursos de máquina virtual são configurados de forma diferente do que outros recursos de cluster. Para configurar recursos de máquina virtual com o **luci**, adicione um grupo de serviço para o cluster e depois adicione um recurso no serviço, selecionando **Virtual Machine** como tipo de recurso e inserindo os parâmetros de recurso de máquina virtual. Para informação sobre configurar uma máquina virtual com o **ccs**, consulte o Seção 5.12, “Recursos de Máquina Virtual” .

Tabela B.24. Máquina Virtual

Campo luci	Atributo cluster.conf	Descrição
Nome do Serviço	**nome**	Especifica o nome da máquina virtual. Quando usar a interface **luci**, você o especifica como um nome de serviço.
Iniciar este serviço automaticamente	**autostart**	Se habilitado, esta máquina virtual é iniciada automaticamente depois do cluster formar um quorum. Se este parâmetro estiver *disabled*, esta máquina virtual *não* é iniciada automaticamente depois do cluster formar um quorum; a máquina virtual é colocada no estado ***disabled***.
Executar exclusivo	**exclusive**	Se habilitado, esta máquina virtual pode somente ser realocada para rodar em um outro nó exclusivamente; que é, rodar em um nó que não possui outras máquinas virtuais rodando nela. Se não houverem nós disponíveis para uma máquina virtual rodar exclusivamente, a máquina virtual não é reiniciada depois da falha. Além disso, outras máquinas virtuais não realocam automaticamente para um nó rodando esta máquina virtual como ***Run exclusive***. Você pode sobrescrever esta opção para inicial manual ou realocar operações.
Failover Domain	**domain**	Define listas de membros do cluster para tentar no evento que uma máquina virtual falha.

Campo luci	Atributo `cluster.conf`	Descrição
Política de recuperação	`recovery`	***Política de Recuperação*** fornece as seguintes opções: • ***Disable*** — Desabilita a máquina virtual se ela falhar. • ***Relocate*** — Tenta reiniciar a máquina virtual em outro nó; onde nao tenta reiniciar no nó atual. • ***Restart*** (Reiniciar) — Tenta reiniciar a máquina virtual localmente (no nó atual) antes de tentar realocar (padrão) a máquina virtual para um outro nó. • ***Restart-Disable*** (Reiniciar-Desabilitado) — O serviço será reiniciado no lugar se ele falhar. Entretanto, se reiniciar o serviço falhar, o serviço será desabilitado em vez de ser movido para um outro host no cluster.
Opções de reinício	`max_restarts`, `restart_expire_time`	Com o **Restart** ou **Restart-Disable** selecionados como a política de recuperação para um serviço, se especifica o número máximo de falhas de reinicialização antes de realocar ou desabilitar o serviço e especifica o período de tempo em segundos após para parar uma reinicialização.
Tipo de Migração	`migrate`	Especifica o tipo de migração de ***live*** (viva) or ***pause*** (pausa). A definição padrão é ***live*** (viva).
Mapeamento de migração	`migration_mapping`	Especifica uma interface alternada para migração. Você pode especificar isto quando, por exemplo, o endereço de rede usado para a migração de máquina virtual em um nó difere do endereço do nó usado para comunicação do cluster. Especificando o seguinte indica que quando você migrar uma máquina virtual do `member` para o `member2`, você pode na verdade migrar para o `target2`. Igualmente, quando você migrar do`member2` para o `member`, você pode migrar usando o `target`. `member:target,member2:target2`
Status Program	`status_program`	O estado do programa para rodar além da verificação padrão de presença de uma máquina virtual. Se especificado, o estado do programa é executado uma vez por minuto. Isto lhe permite determinar o estado de serviços críticos dentro de uma máquina virtual. Por exemplo, se uma máquina virtual roda um servidor web, seu programa de estado pode checar se um servidor web está ativo e em execução; se a verificação de estado falhar (mostrado com o retorno de um valor diferente de zero), a máquina virtual é recuperada. Depois que uma máquina virtual é iniciada, o agente de recurso de máquina virtual chamará periodicamente o programa de estado e espera pelo retorno de um código de sucesso (zero). Ele expira depois de cinco minutos.

Campo luci	Atributo `cluster.conf`	Descrição
Caminho para o arquivo XML usado para criar o VM	`xmlfile`	O caminho completo para o arquivo XML `libvirt` que contém a definição do domínio `libvirt`.
Caminho de arquivo de configuração VM	`path`	Uma especificação de caminho de dois pontos que o Agente de Recurso de Máquina Virtual (`vm.sh`) busca para o arquivo de configuração da máquina virtual. Por exemplo: `/mnt/guests/config:/etc/libvirt/qemu`.  **IMPORTANTE** O caminho *nunca* deve apontar diretamente para o arquivo de configuração da máquina virtual.
Caminho para o diretório de snapshot VM	`snapshot`	Caminho para o diretório snapshot onde a imagem da máquina virtual será armazenada.
Hypervisor URI	`hypervisor_uri`	Hypervisor URI (normalmente automático).
Migration URI	`migration_uri`	Migration URI (normalmente automático).
Dados de túnel sob ssh durante a migração	`tunnelled`	Dados de túnel sob ssh durante a migração

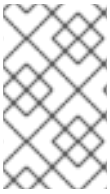
APÊNDICE C. COMPORTAMENTO DO RECURSO DE ALTA DISPONIBILIDADE

Este apêndice descreve os comportamentos comuns dos recursos de Alta Disponibilidade (HA). Ele se destina a fornecer informações auxiliares que podem ser úteis em configurar os serviços HA. Você pode configurar os parâmetros com o **luci** ou editando `etc/cluster/cluster.conf`. Para descrições dos parâmetros de recursos HA, consulte o Apêndice B, *Parâmetros dos Recursos de Alta Disponibilidade*. Para entender agentes de recursos em mais detalhes você pode vizualiza-los em `/usr/share/cluster` de qualque nó no cluster.

NOTA

Para entender completamente as informações neste apêndice, você pode precisar de um entendimento detalhado dos agentes de recursos e do arquivo de configuração de cluster, `/etc/cluster/cluster.conf`.

Um serviço HA é um grupo de recursos de cluster configurados em uma entidade coerente que fornece serviços especializados à clientes. Um serviço HA é representado como uma árvore de recursos em um arquivo de configuração de cluster, `/etc/cluster/cluster.conf` (em cada nó no cluster). No arquivo de configuração do cluster, cada árvore de recursos é uma representação XML que especifica cada recurso, seus atributos e seu relacionamento entre outros recursos na árvore de recursos (relacionamentos pai, filhos e irmãos).



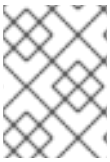
NOTA

Por causa que um serviço HA consiste de recursos organizados em uma árvore hierárquica, um serviço é as vezes referido como um *recurso de árvore* ou *grupo de recursos*. Ambos são sinônimos em um *serviço HA*.

Na raíz de cada árvore de recurso há um especial tipo de recurso – um *recurso de serviço*. Outros tipos de recursos compreendem do resto de um serviço, determinando suas características. Configurando em serviço HA consiste em criar um recurso de serviço, criando recursos de cluster subordinados e organiza-los em uma entidade coerente que tem conformidade com às restrições hierárquicas do serviço.

Este apêndice consiste das seguintes seções:

- Seção C.1, “Relacionamentos de níveis Pai, Filho e Irmãos entre Recursos”
- Seção C.2, “Ordenação de Início de Irmãos e Ordenação de Recursos Filhos”
- Seção C.3, “Herança, os Blocos de <recursos< e Reusando Recursos”
- Seção C.4, “Recuperação de Falhas e Sub Árvores Independentes”
- Seção C.5, “Depurando e Testando Serviços e Ordenação de Recursos”



NOTA

As seções seguintes apresentam exemplos do arquivo de configuração do cluster, `/etc/cluster/cluster.conf`, para propósitos ilustrativos somente.

C.1. RELACIONAMENTOS DE NÍVEIS PAI, FILHO E IRMÃOS ENTRE RECURSOS

Um serviço de cluster é uma entidade integrada que roda sob o controle do **rgmanager**. Todos os recursos em um serviço rodam no mesmo nó. Da perspectiva do **rgmanager**, um serviço de cluster é somente uma entidade que pode ser iniciada, parada ou realocada. Dentro de um serviço de cluster, entretanto, a hierarquia é iniciada e parada. Os níveis hierarquicos consistem em pai, filho e irmãos.

O Exemplo C.1, “Hierarquia de Recursos do Serviço foo ” exibe uma amostra de árvores de recursos do serviço *foo*. Neste exemplo, o relacionamento entre os recursos são como a seguir:

- **fs:myfs** (<fs name="myfs" ...>) e **ip:10.1.1.2** (<ip address="10.1.1.2 .../>) são irmãos.
- O **fs:myfs** (<fs name="myfs" ...>) é o pai do **script:script_child** (<script name="script_child"/>).
- O **script:script_child** (<script name="script_child"/>) é o filho do **fs:myfs** (<fs name="myfs" ...>).

Exemplo C.1. Hierarquia de Recursos do Serviço foo

```
<service name="foo" ...>
    <fs name="myfs" ...>
        <script name="script_child"/>
    </fs>
    <ip address="10.1.1.2" .../>
</service>
```

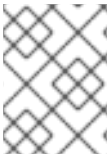
As seguintes regras se aplicam à relacionamentos pai/filho em uma árvore de recursos:

- Pais são iniciados antes dos filhos.
- Filhos devem todos serem parados antes de um pai ser parado.
- Para um recurso ser considerado em boa saúde, todos os filhos devem estar em boa saúde.

C.2. ORDENAÇÃO DE INÍCIO DE IRMÃOS E ORDENAÇÃO DE RECURSOS FILHOS

O recurso de serviços determina a ordem de início e a ordem de parada de um recurso filho de acordo se ele designa um atributo de tipo filho para um recurso filho conforme a seguir:

- Designa o atributo tipo filho (recurso filho *tipificado* — Se o recurso de serviço designa um atributo tipo filho para um recurso filho, o recurso filho é *tipificado*. O atributo tipo filho determina explicitamente a ordem de início e parada do recurso filho.
- *Não designa* o atributo tipo filho (recurso filho *não tipificado*) — Se o recurso de Serviço *não designa* um atributo tipo filho para um recurso filho, o recurso filho é *não tipificado*. O recurso de serviço não controla explicitamente a ordem de início e ordem de parada de um recurso filho não tipificado. Entretanto, um recurso filho não tipificado é iniciado e parado de acordo com sua ordem no **/etc/cluster.cluster.conf**. Além disso, recursos filhos não especificados são iniciados depois de todos recursos filhos tipificados terem iniciado e são parados antes de qualquer recursos filhos tiverem parado.



NOTA

O único recurso para implementar a ordem do *tipo de recurso filho*definido é o recurso de Serviço.

Para mais informações sobre ordenação de início e parada recursos filhos tipificados consulte a Seção C.2.1, “Ordenação de Início e Parada de Recursos Filhos Tipificados” . Para mais informações sobre ordenação de início e parada recursos filhos não tipificados, consulte a Seção C.2.2, “Ordenação de Início e Parada de Recurso Filho Não Tipificado”.

C.2.1. Ordenação de Início e Parada de Recursos Filhos Tipificados

Para um recurso filho tipificado, o tipo de atributo para o recurso filho define a ordem de início e a ordem de parada para cada tipo de recurso com um número de 1 e 100; um valor para início e outro para parada. Menor o número, mais cedo um tipo de recurso inicia ou pára. Por exemplo, a Tabela C.1, “Ordem de Início e Parada de Tipo de Recurso Filho” mostra os valores de início e parada para cada tipo de recurso; o Exemplo C.2, “Valores de Início e Parada de Recursos: Resumo do Agente de Recurso de Serviço `service.sh`” mostra os valores de início e parada conforme eles aparecem no agente de recurso de Serviço, o `service.sh`. Para o recurso de Serviço, todos os filhos LVM são iniciados primeiro, seguidos por todos os filhos Sistema de Arquivo, seguidos por todos filhos Script e assim por diante.

Tabela C.1. Ordem de Início e Parada de Tipo de Recurso Filho

Recurso	Tipo Filho	Valor ordem-início	Valor ordem-parada
LVM	lvm	1	9
Sistema de Arquivo	fs	2	8
Sistema de Arquivo GFS2	clusterfs	3	7
Montagem NFS	netfs	4	6
Exportar NFS	nfsexport	5	5
NFS Client	nfsclient	6	4
Endereço IP	ip	7	2
Samba	smb	8	3
Script	script	9	1

Exemplo C.2. Valores de Início e Parada de Recursos: Resumo do Agente de Recurso de Serviço `service.sh`

```
<special tag="rgmanager">
    <attributes root="1" maxinstances="1"/>
```

```
    <child type="lvm" start="1" stop="9"/>
    <child type="fs" start="2" stop="8"/>
    <child type="clusterfs" start="3" stop="7"/>
    <child type="netfs" start="4" stop="6"/>
    <child type="nfsexport" start="5" stop="5"/>
    <child type="nfsclient" start="6" stop="4"/>
    <child type="ip" start="7" stop="2"/>
    <child type="smb" start="8" stop="3"/>
    <child type="script" start="9" stop="1"/>
</special>
```

A ordenação dentro de um tipo de recurso é preservada conforme ela existe no arquivo de configuração do cluster, **/etc/cluster/cluster.conf**. Por exemplo, considere a ordem de início e ordem de parada dos recursos filhos tipificados no Exemplo C.3, “Ordenação Dentro de um Tipo de Recurso”.

Exemplo C.3. Ordenação Dentro de um Tipo de Recurso

```
<service name="foo">
  <script name="1" .../>
  <lvm name="1" .../>
  <ip address="10.1.1.1" .../>
  <fs name="1" .../>
  <lvm name="2" .../>
</service>
```

C.2.1.1. Ordem de Início do Recurso Filho Tipificado

No Exemplo C.3, “Ordenação Dentro de um Tipo de Recurso” , os recursos são iniciados na seguinte ordem:

1. **lvm:1** — Este é um recurso LVM. Todos os recursos LVM são iniciados primeiro. O **lvm:1** (**<lvm name="1" .../>**) é o primeiro recurso LVM iniciado entre os recursos LVM porque ele é o primeiro recurso LVM listado na porção do Serviço *foo* do **/etc/cluster/cluster.conf**.

2. O **lvm:2** — Este é um recurso LVM. Todos os recursos LVM são iniciados primeiro. O **lvm:2** (**<lvm name="2" .../>**) é iniciado depois do **lvm:1** porque ele é listado depois do **lvm:1** na porção do Serviço *foo* do **/etc/cluster/cluster.conf**.

3. O **fs:1** — Este é um recurso do Sistema de Arquivo. Se existissem outros recursos do Sistema de Arquivo no Serviço *foo*, eles iniciariam na ordem listada na porção do Serviço *foo* do **/etc/cluster/cluster.conf**.

4. **ip:10.1.1.1** — Este é um recurso de Endereço IP. Se houvessem outros recursos de endereço IP no Serviço *foo*, eles iniciariam na ordem listada na porção do Serviço *foo* do **/etc/cluster/cluster.conf**.

5. **script:1** — Este é um recurso de Script. Se houvessem outros recursos de Script no Serviço *foo*, eles iniciariam na ordem listada na porção do Serviço *foo* do **/etc/cluster/cluster.conf**.

C.2.1.2. Ordem de Parada do Recurso Filho Tipificado

No Exemplo C.3, "Ordenação Dentro de um Tipo de Recurso" , os recursos são parados na seguinte ordem:

1. **`script:1`** — Este é um recurso de Script. Se houvessem outros recursos Scripts no Serviço *foo*, eles parariam pela ordem reversa listada na porção do Serviço *foo* do **`/etc/cluster/cluster.conf`**.

2. **`ip:10.1.1.1`** — Este é um recurso de Endereço IP. Se houvessem outros recursos de endereço IP no Serviço *foo*, eles parariam pela ordem reversa listada na porção do Serviço *foo* do **`/etc/cluster/cluster.conf`**.

3. **`fs:1`** — Este é um recurso de Sistema de Arquivo. Se houvessem outros recursos de Sistema de Arquivo no Serviço *foo*, eles parariam pela ordem reversa listada na porção do Serviço *foo* do **`/etc/cluster/cluster.conf`**.

4. **`lvm:2`** — Este é um recurso LVM. Todos recursos LVM são parados por último. O **`lvm:2`** (**`<lvm name="2" .../>`**) é parado antes do **`lvm:1`**; recursos dentro de um grupo de um tipo de recurso são parados pela ordem reversa listada na porção do Serviço *foo* do **`/etc/cluster/cluster.conf`**.

5. **`lvm:1`** — Este é um recurso LVM. Todos recursos LVM são parados por último. O **`lvm:1`** (**`<lvm name="1" .../>`**) é parado depois dos recursos **`lvm:2`**; dentro de um grupo de um tipo de recurso são parados pela ordem reversa listada na porção do Serviço *foo* do **`/etc/cluster/cluster.conf`**.

C.2.2. Ordenação de Início e Parada de Recurso Filho Não Tipificado

Considerações adicionais são necessárias para recursos filhos não especificados. Para tais, o início do pedido ou final do pedido não são explícitamente especificados pelo recurso do Serviço. Ao invés disso, o início do pedido ou final do pedido é determinado de acordo com o recurso filho em **`/etc/cluster/cluster.conf`**. Além disso, os recursos filho não especificados são iniciados depois de todos os recursos filho especificados e parados antes de qualquer recurso filho.

Por exemplo, considere a ordem de início e parada dos recursos filho não tipificados no Exemplo C.4, "Recursos Filhos Não tipificados e Tipificados em um Serviço".

Exemplo C.4. Recursos Filhos Não tipificados e Tipificados em um Serviço

```
<service name="foo">
  <script name="1" .../>
  <nontypedresource name="foo"/>
  <lvm name="1" .../>
  <nontypedresourcetwo name="bar"/>
  <ip address="10.1.1.1" .../>
  <fs name="1" .../>
  <lvm name="2" .../>
</service>
```

C.2.2.1. Ordem de Início do Recurso Filho Não tipificado

No Exemplo C.4, “Recursos Filhos Não tipificados e Tipificados em um Serviço” , os recursos filhos são iniciados na seguintes ordem:

1. **lvm:1** — Este é um recurso LVM. Todos os recursos LVM são iniciados primeiro. O **lvm:1** (**<lvm name="1" .../>**) é o primeiro recurso LVM iniciado entre os recursos LVM porque ele é o primeiro recurso LVM listado na porção do Serviço *foo* do **/etc/cluster/cluster.conf**.

2. O **lvm:2** — Este é um recurso LVM. Todos os recursos LVM são iniciados primeiro. O **lvm:2** (**<lvm name="2" .../>**) é iniciado depois do **lvm:1** porque ele é listado depois do **lvm:1** na porção do Serviço *foo* do **/etc/cluster/cluster.conf**.

3. O **fs:1** — Este é um recurso do Sistema de Arquivo. Se existissem outros recursos do Sistema de Arquivo no Serviço *foo*, eles iniciariam na ordem listada na porção do Serviço *foo* do **/etc/cluster/cluster.conf**.

4. **ip:10.1.1.1** — Este é um recurso de Endereço IP. Se houvessem outros recursos de endereço IP no Serviço *foo*, eles iniciariam na ordem listada na porção do Serviço *foo* do **/etc/cluster/cluster.conf**.

5. **script:1** — Este é um recurso de Script. Se houvessem outros recursos de Script no Serviço *foo*, eles iniciariam na ordem listada na porção do Serviço *foo* do **/etc/cluster/cluster.conf**.

6. **nontypedresource:foo** — Este é recurso não tipificado. Pelo Motivo este é um recurso não tipificado, ele é iniciado depois depois que os recursos tipificados iniciam. Além disso, sua ordem no recurso do Serviço é antes do outro recurso não tipificado, **nontypedresourcetwo:bar**; portanto, ele é iniciado antes do **nontypedresourcetwo:bar**. (Recursos não tipificados são iniciados na ordem que eles aparecem no recurso do Serviço).

7. **nontypedresourcetwo:bar** — Este é um recurso não tipificado. Por causa que é um recurso não tipificado, ele é iniciado depois dos recursos tipificados iniciarem. Além disso, sua ordem no recurso de Serviço é depois do outro recurso não tipificado, o **nontypedresource:foo**; portanto é iniciado depois do **nontypedresource:foo**. (Recursos não tipificados são iniciados na ordem que eles aparecem no recurso do Serviço).

C.2.2.2. Ordem de Parada do Recurso Filho Não tipificado

No Exemplo C.4, “Recursos Filhos Não tipificados e Tipificados em um Serviço” , os recursos são parados na seguinte ordem:

1. **nontypedresourcetwo:bar** — Este é um recurso não tipificado. Por causa que é um recurso não tipificado, ele é parado antes que os recursos tipificados são parados. Além disso, sua ordem no recurso de Serviço é depois dos outros recursos não tipificados, o **nontypedresource:foo**; portanto é parado antes do **nontypedresource:foo**. (Recursos não tipificados são parados pela ordem reversa que eles aparecem no recurso do Serviço).

2. **nontypedresource:foo** — Este é um recurso não tipificado. Por causa que é um recurso não tipificado, ele é parado antes que os recursos tipificados são parados. Além disso, sua ordem no recurso do Serviço é antes do outro recurso não tipificado, **nontypedresourcetwo:bar**; portanto, ele é parado depois do **nontypedresourcetwo:bar**. (Recursos Não tipificados são parados pela ordem reversa que eles aparecem no recurso do Serviço).

3. **`script:1`** – Este é um recurso de Script. Se houvessem outros recursos Scripts no Serviço *foo*, eles parariam pela ordem reversa listada na porção do Serviço *foo* do **`/etc/cluster/cluster.conf`**.

4. **`ip:10.1.1.1`** – Este é um recurso de Endereço IP. Se houvessem outros recursos de endereço IP no Serviço *foo*, eles parariam pela ordem reversa listada na porção do Serviço *foo* do **`/etc/cluster/cluster.conf`**.

5. **`fs:1`** – Este é um recurso de Sistema de Arquivo. Se houvessem outros recursos de Sistema de Arquivo no Serviço *foo*, eles parariam pela ordem reversa listada na porção do Serviço *foo* do **`/etc/cluster/cluster.conf`**.

6. **`lvm:2`** – Este é um recurso LVM. Todos recursos LVM são parados por último. O **`lvm:2`** (**`<lvm name="2" .../>`**) é parado antes do **`lvm:1`**; recursos dentro de um grupo de um tipo de recurso são parados pela ordem reversa listada na porção do Serviço *foo* do **`/etc/cluster/cluster.conf`**.

7. **`lvm:1`** – Este é um recurso LVM. Todos recursos LVM são parados por último. O **`lvm:1`** (**`<lvm name="1" .../>`**) é parado depois dos recursos **`lvm:2`**; dentro de um grupo de um tipo de recurso são parados pela ordem reversa listada na porção do Serviço *foo* do **`/etc/cluster/cluster.conf`**.

C.3. HERANÇA, OS BLOCOS DE <RECURSOS< E REUSANDO RECURSOS

Alguns recursos se beneficiam herdando valores de um recurso pai; que é comumente o caso de um serviço NFS. O Exemplo C.5, “Configuração do Serviço NFS para Reuso e Herança do Recurso” mostra uma configuração de serviço NFS típica, configure para herança e reuso do recurso.

Exemplo C.5. Configuração do Serviço NFS para Reuso e Herança do Recurso

```
    <resources>
        <nfsclient name="bob" target="bob.example.com"
options="rw,no_root_squash"/>
        <nfsclient name="jim" target="jim.example.com"
options="rw,no_root_squash"/>
        <nfsexport name="exports"/>
    </resources>
    <service name="foo">
        <fs name="1" mountpoint="/mnt/foo" device="/dev/sdb1"
fsid="12344">
            <nfsexport ref="exports">  <!-- nfsexport's path and fsid
attributes
                                       are inherited from the
mountpoint &
                                       fsid attribute of the
parent fs
                                       resource -->
                <nfsclient ref="bob"/> <!-- nfsclient's path is
inherited from the
                                       mountpoint and the fsid
is added to the
                                       options string during
```

```
export -->
                <nfsclient ref="jim"/>
            </nfsexport>
        </fs>
        <fs name="2" mountpoint="/mnt/bar" device="/dev/sdb2"
fsid="12345">
            <nfsexport ref="exports">
                <nfsclient ref="bob"/> <!-- Because all of the critical
data for this
                                            resource is either
defined in the
                                            resources block or
inherited, we can
                                            reference it again! -->
                <nfsclient ref="jim"/>
            </nfsexport>
        </fs>
        <ip address="10.2.13.20"/>
    </service>
```

Se o serviço fosse horizontal (que é, sem relacionamentos pai/filho), ele precisaria ser configurado como a seguir:

- O serviço precisaria de quatro recursos nfsclient — um por sistema de arquivo (um total de dois por sistema de arquivo) e um por máquina alvo (um total de dois por máquina alvo).
- O serviço precisaria especificar o caminho de exportação e o ID do sistema de arquivo para cada nfsclient, que apresenta chances de erros na configuração.

No Exemplo C.5, "Configuração do Serviço NFS para Reuso e Herança do Recurso" entretanto, os recursos de cliente NFS *nfsclient:bob* e *nfsclient:jim* são definidos uma vez; da mesma maneira, o recurso de exportação do NFS *nfsexport:exports* é definido uma vez. Todos os atributos necessários pelos recursos são herdados dos recursos pais. Por causa que os atributos herdados são dinâmicos (e não entram em conflito entre si), é possível reusar aqueles recursos — que é porque eles são definidos no bloco de recursos. Pode não ser prático para configurar alguns recursos em múltiplos lugares. Por exemplo, configurar um recurso de sistema de arquivo em múltiplos lugares pode resultar em montar um sistema de arquivos em dois nós causando problemas.

C.4. RECUPERAÇÃO DE FALHAS E SUB ÁRVORES INDEPENDENTES

Na maioria dos ambientes corporativos, o curso normal de ação para recuperação de falha de um serviço é reiniciar o serviço inteiro se qualquer componente no serviço falhar. Por exemplo, no Exemplo C.6, "Recuperação de Falha Normal do Serviço *foo*", se qualquer dos scripts definidos neste serviço falhar, o curso normal de ação é reiniciar (ou realocar ou desabilitar, de acordo com a política de recuperação do serviço) o serviço. Entretanto, em algumas circunstâncias certas partes de um serviço podem ser consideradas não críticas; podendo ser necessário reiniciar somente parte do serviço em questão antes de tentar uma recuperação normal. Para fazer isso, você pode usar o atributo ***__independent_subtree***. Por exemplo, no Exemplo C.7, "A Recuperação de Falha do Serviço *foo* com o Atributo ***__independent_subtree***.", o atributo ***__independent_subtree*** é usado para fazer as seguintes ações:

- Se o script:script_one falhar, reinicie script:script_one, script:script_two, e script:script_three.

- Se o script:script_two falhar, reinicie apenas o script:script_two.
- Se o script:script_three falhar, reinicie o script:script_one, script:script_two, e script:script_three.
- Se o script:script_four falhar, reinicie o serviço inteiro.

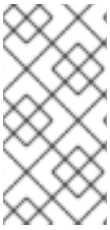
Exemplo C.6. Recuperação de Falha Normal do Serviço *foo*

```
<service name="foo">
      <script name="script_one" ...>
          <script name="script_two" .../>
      </script>
      <script name="script_three" .../>
</service>
```

Exemplo C.7. A Recuperação de Falha do Serviço *foo* com o Atributo *__independent_subtree*.

```
<service name="foo">
      <script name="script_one" __independent_subtree="1" ...>
          <script name="script_two" __independent_subtree="1" .../>
          <script name="script_three" .../>
      </script>
      <script name="script_four" .../>
</service>
```

Em algumas circunstâncias, se um componente de um serviço falhar você pode querer desabilitar somente esse componente sem desabilitar o serviço inteiro, para evitar que afete outros serviços que usam outros componentes deste serviço. A partir do lançamento do Red Hat Enterprise Linux 6.1, você pode fazer isso usando o atributo ***__independent_subtree="2"***, que designa a subárvore independente como não crítica.



NOTA

Você pode usar somente a bandeira não crítica em recursos isoladamente referenciados. A bandeira não crítica trabalha com todos os recursos em todos os níveis da árvore de recursos mas não deve ser usada no nível superior quando definir serviços ou máquinas virtuais.

A partir do lançamento do Red Hat Enterprise Linux 6.1, você pode definir reinicializações máximas e expirações de reinicializações em uma base por nó na árvore de recursos para subárvores independentes. Para definir estes limites, você pode usar os seguintes atributos:

- O **__max_restarts** configura o número máximo de reinicializações toleradas antes de desistir.
- **__restart_expire_time** configura o período de tempo, em segundos, depois que uma reinicialização não é mais tentada.

C.5. DEPURANDO E TESTANDO SERVIÇOS E ORDENAÇÃO DE RECURSOS

Você pode depurar e testar serviços e ordenação de recursos com o utilitário **rg_test**. O **rg_test** é um utilitário de linha de comando fornecido pelo pacote **rgmanager** que é executado a partir do shell ou um terminal (não disponível no **Conga**). A Tabela C.2, "Resumo do Utilitário **rg_test**" resume as ações e sintaxes do utilitário **rg_test**.

Tabela C.2. Resumo do Utilitário rg_test

Ação	Sintaxe
Exibe as regras do recurso que o **rg_test** entende.	**rg_test rules**
Testa a configuração (e o /usr/share/cluster) por erros ou agentes de recursos redundantes.	**rg_test test /etc/cluster/cluster.conf**
Exibe a ordem de início e parada de um serviço.	Exibe a ordem de início: **rg_test noop /etc/cluster/cluster.conf start service *servicename*** Exibe a ordem de parada: **rg_test noop /etc/cluster/cluster.conf stop service *servicename***
Explicitamente inicia ou para um serviço.	 **IMPORTANTE** Somente faça isso em um nó e sempre desabilite o serviço no rgmanager primeiro. Inicie um serviço: **rg_test test /etc/cluster/cluster.conf start service *servicename*** Pára um serviço: **rg_test test /etc/cluster/cluster.conf stop service *servicename***

Ação	Sintaxe
Calcular e exibir a árvore delta de recurso entre dois arquivos cluster.conf .	**rg_test delta** ***cluster.conf file 1*** ***cluster.conf file 2*** Por exemplo: **rg_test delta /etc/cluster/cluster.conf.bak /etc/cluster/cluster.conf**

APÊNDICE D. CHECAGEM DE RECURSOS DE SERVIÇO DE CLUSTER E EXPIRAÇÃO DE FAILOVER

Este apêndice descreve como o **rgmanager** monitora o estado dos recursos do cluster, e como modificar o estado do intervalo de verificação. O apêndice também descreve o parâmetro do serviço **__enforce_timeouts**, o qual indica que um timeout para uma operação deve causar falha no serviço.

NOTA

Para entender todas as informações neste apêndice, você precisa entender sobre os agentes de recursos e arquivo de configuração de cluster, **/etc/cluster/cluster.conf**. Para uma lista compreensiva e a descrição dos elementos e atributos do **cluster.conf**, consulte o esquema de cluster em **/usr/share/cluster/cluster.rng** e o esquema anotado em **/usr/share/doc/cman-X.Y.ZZ/cluster_conf.html** (por exemplo **/usr/share/doc/cman-3.0.12/cluster_conf.html**).

D.1. MODIFICANDO O INTERVALO DE CHECAGEM DE ESTADO DO RECURSO

O **rgmanager** checa o estado de recursos individuais, não os serviços inteiros. A cada 10 segundos, o rgmanager escaneia o árvore de recursos, buscando por recursos que tiveram seus intervalos "de verficação do estado" passados.

Cada agente de recurso especifica a quantidade de tempo entre verificações de estado periódicos. Cada recurso utiliza estes valores de expirações a menos que sejam explicitamente sobrescritos no arquivo **cluster.conf** usando a tag especial **<action>**:

```
<action name="status" depth="*" interval="10" />
```

Esta tag é um filho especial do próprio recurso no arquivo **cluster.conf**. Por exemplo, se você tiver um recurso de sistema de arquivos para o qual você quer sobrescrever o intervalo de verificação de estado você pode especificar o recurso de sistema de arquivos no arquivo **cluster.conf** como se segue:

```
<fs name="test" device="/dev/sdb3">
  <action name="status" depth="*" interval="10" />
  <nfsexport...>
  </nfsexport>
</fs>
```

Alguns agentes fornecem múltiplas "profundidades" de checagem. Por exemplo, uma checagem normal de sistema de arquivos (profundidade 0) verifica se o sistema de arquivos está montado no local correto. Uma verificação mais profunda é de valor 10, que checa se você pode ler um arquivo do sistema de arquivos. Uma verificação ainda mais profunda é valor 20, que checa se você pode escrever no sistema de arquivos. No exemplo dado aqui, a profundidade **depth** está configurada para *****, que indica que estes valores devem ser usados para todas as profundidades. O resultado é que o sistema de arquivos **test** é checado no nível mais alto de profundidade fornecido pelo agente de recurso (no caso, 20) a cada 10 segundos.

D.2. FORÇANDO EXPIRAÇÕES DE RECURSOS

Não há expiração de tempo para iniciar, parar, ou causar um fail over de recursos. Alguns recursos levam uma quantidade indeterminada de tempo para iniciar ou parar. Infelizmente, uma falha para parar (incluindo expirações) leva à inoperação do serviço (estado de falha). Você pode, se desejado, ativar o cumprimento de expiração em cada recurso em um serviço individualmente adicionando **__enforce_timeouts="1"** à referência no arquivo **cluster.conf**.

O seguinte exemplo mostra um serviço de cluster que foi configurado com o atributo **_enforce_timeouts** ajustado para o recurso **netfs**. Com este atributo ajustado, então se durante uma recuperação demorar mais de 30 segundos para desmontar o sistema de arquivos NFS, a operação expirará, fazendo o serviço entrar no estado de falha.

```
</screen>
<rm>
  <failoverdomains/>
  <resources>
    <netfs export="/nfstest" force_unmount="1" fstype="nfs"
host="10.65.48.65"
           mountpoint="/data/nfstest" name="nfstest_data"
options="rw,sync,soft"/>
  </resources>
  <service autostart="1" exclusive="0" name="nfs_client_test"
recovery="relocate">
    <netfs ref="nfstest_data" __enforce_timeouts="1"/>
  </service>
</rm>
```

APÊNDICE E. RESUMO DAS FERRAMENTAS DA LINHA DE COMANDO

Tabela E.1, “Resumo das Ferramentas da Linha de Comando” resume as ferramentas preferidas da linha de comando para configurar e gerenciar a Complemento de Alta Disponibilidade. Para mais informações sobre os comandos e variáveis, consulte a página man para cada ferramenta da linha de comando.

Tabela E.1. Resumo das Ferramentas da Linha de Comando

Ferramentas da Linha de Comando	Usadas com	Propósito
ccs_config_dump – Ferramenta de Dump de Configuração de Cluster	Infraestrutura do cluster	O **ccs_config_dump** gera um resultado XML para configuração de execução. A configuração de execução é, as vezes, diferente da configuração armazenada no arquivo porque alguns subsistemas armazenam ou definem algumas informações padrões na configuração. Aqueles valores geralmente não estão presentes na versão do disco de configuração mas são requeridos no tempo de execução para o cluster trabalhar apropriadamente. Para mais informações sobre esta ferramenta, consulte a página man ccs_config_dump(8).
ccs_config_validate – Ferramenta de Validação de Configuração de Cluster	Infraestrutura do cluster	A **ccs_config_validate** valida o **cluster.conf** contra o esquema, **cluster.rng** (localizado em **/usr/share/cluster/cluster.rng** em cada nó. Para mais informações sobre esta ferramenta, consulte a página man ccs_config_validate(8).
clustat – Utilitário de Estado do Cluster	Componentes de Gerenciamento de Serviços de Alta Disponibilidade	O comando **clustat** mostra o estado do cluster. Ele exibe informações de afiliação, vizualização do quorum e o estado de todos os serviços de usuários configurados. Para mais informações sobre esta ferramenta, consulte a página man clustat(8).
clusvcadm – Utilitário de Serviço de Usuários do Cluster	Componentes de Gerenciamento de Serviços de Alta Disponibilidade	O comando **clusvcadm** lhe permite habilitar, desabilitar, realocar e reiniciar os serviços de alta disponibilidade em um cluster. Para mais informações sobre esta ferramenta, consulte a página man clusvcadm(8).

Ferramentas da Linha de Comando	Usadas com	Propósito
`cman_tool` — Ferramenta de Gerenciamento do Cluster	Infraestrutura do cluster	O **`cman_tool`** é um programa que gerencia o gerenciador de cluster CMAN. Ele fornece a capacidade de se unir a um cluster, deixar um cluster, matar um nó ou mudar os votos quorum esperados de um nó em um cluster. Para mais informações sobre esta ferramenta, consulte a página man cman_tool(8).
`fence_tool` — Ferramenta Fence	Infraestrutura do cluster	A **`fence_tool`** é um programa usado para se unir e deixar um domínio fence. Para mais informações sobre esta ferramenta, consulte a página man fence_tool(8).

APÊNDICE F. ALTA DISPONIBILIDADE LVM (HA-LVM)

O Complemento de Alta Disponibilidade da Red Hat fornece suporte para volumes LVM de alta disponibilidade (HA-LVM) em uma configuração de failover. Esta configuração é diferente das configurações ativa/ativa pelo Clustered Logical Volume Manager (CLVM), o qual é um conjunto de extensões em cluster para LVM que permitem que um cluster de computadores gerencie armazenamento compartilhado.

A decisão de quando utilizar ou o CLVM or HA-LVM deve ser baseado nas necessidades dos aplicativos ou serviços sendo implementados.

- Caso os aplicativos sejam conscientes do cluster e foram ajustados para rodar simultaneamente em máquinas múltiplas de uma só vez, depois o CLVM deve ser usado. Especialmente se mais de um nó de seu cluster precisar acessar seu armazenamento que é então compartilhado entre nós ativos, aí então você precisará usar o CLVM. O CLVM permite que um usuário configure volumes lógicos em armazenamento compartilhado bloqueando acesso em armazenamento físico enquanto um volume lógico estiver sendo configurado, e utiliza serviços de bloqueio em cluster para gerenciar o armazenamento compartilhado. Para mais informações sobre o CLVM, e sobre a configuração do LVM em geral, consulte o *Logical Volume Manager Administration*.

- Se os aplicativos rodarem de forma ideal em configurações ativa/passiva (failover), onde somente um único nó que acessa o armazenamento está ativa em qualquer momento, você deve utilizar agentes do Gerenciamento de Volume Lógico de Disponibilidade (HA-LVM).

A maioria dos aplicativos serão executados de melhor maneira na configuração ativa/passiva, pois estes não foram criados ou optimizados para serem executados ao mesmo tempo que outras instâncias. Escolher executar um aplicativo que não possui a consciência do cluster em volumes lógicos de cluster, pode resultar em queda de desempenho se o volume lógico for espelhado. Isto ocorre porque existe cabeçalho de comunicação de cluster para os próprios volumes lógicos nestas instâncias. Um aplicativo consciente do cluster deve ser capaz de alcançar um desempenho de gamjp acima do desempenho de perda, apresentado pelos sistemas de arquivo de cluster e volumes lógicos de cluster consciente. Isto é possível mais para alguns aplicativos e as cargas de trabalho do que para outras. Determinar quais os requerimentos do cluster e se o esforço extra em optimizar para um cluster ativo/ativo pagará dividendos é a forma de escolher entre as duas variantes LVM. A maioria dos usuários irá alcançar melhores resultados com HA oposto a utilizar HA-LVM.

O HA-LVM e CLVM são semelhantes no fato de que ambos evitam danos dos metadados do LVM e seus volumes lógicos, que poderia ocorrer caso as máquinas múltiplas pudessem fazer mudanças sobrepostas. O HA-LVM impóe a restrição de que um volume lógico só pode se ativado exclusivamente, ou seja, ativo em somente uma máquina por vez. Isto significa que somente implementações (em não cluster) locais dos drivers de armazenamento são usadas. Evitar o cabeçalho da coordenação do cluster desta forma, aumenta o desempenho. O CLVM não impõe estas restrições, um usuário é livre para ativar um volume lógico em todas as máquinas em um cluster. Isto força o uso dos drivers de armazenamento de consciência de cluster, o qual permite que sistemas de arquivo de cluster conscientes e aplicativos sejam colocados em primeiro lugar.

O HA-LVM pode ser configurado para usar um dos dois métodos para alcançar seus mandados de ativação de volume lógico exclusivo.

- O método preferido usa o CLVM, mas ele irá ativar os volumes lógicos somente de forma exclusiva. Isto tem a vantagem de uma configuração mais fácil e melhor prevenção de erros administrativos (como remover um volume lógico que esteja em uso). Para usar o CLVM, o Complemento de Alta Disponibilidade e Software de complemento de armazenamento resiliente, incluindo o daemon **`clvmd`**, devem estar em execução.

O procedimento para configuração do HA-LVM utilizando este método é descrito no Seção F.1, “Configurando um Failover HA-LVM com o CLVM (preferido)”.

- O segundo método usa bloqueio de máquina local e "marcações" de LVM. Este método possui uma vantagem de não precisar de qualquer pacote de cluster LVM; no entanto, existem mais passos envolvidos na configuração do mesmo e não previne um administrador de remover um volume lógico por engano de um nó no cluster, onde ele não é ativo. O procedimento para a configuração do HA-LVM usando este método está descrito em Seção F.2, “Configurando um Failover de HA-LVM com a Marcação”.

F.1. CONFIGURANDO UM FAILOVER HA-LVM COM O CLVM (PREFERIDO)

Para configurar um failover de HA-LVM (utilizando a variante CLVM preferida), realize os seguintes passos:

1. Certifique-se de que seu sistema está configurado para suportar o CLVM, o qual requer o seguinte:
   - Os Complementos de Alta Disponibilidade e Armazenamento Resiliente estão instalados, incluindo o pacote **cmirror** se os volumes lógicos CLVM estiverem espelhados.
   - O parâmetro **locking_type** na seção global do arquivo **/etc/lvm/lvm.conf** está definido para o valor '3'.
   - Os Complementos de Alta Disponibilidade e Software de Armazenamento Resiliente, incluindo o daemon do **clvmd** devem estar rodando. Para o espelhamento do CLVM, o serviço **cmirror** também deve ser iniciado.
2. Crie um volume lógico e sistema de arquivo utilizando o LVM padrão e comandos de sistema de arquivo, como no exemplo a seguir.

```
# pvcreate /dev/sd[cde]1

# vgcreate -cy shared_vg /dev/sd[cde]1

# lvcreate -L 10G -n ha_lv shared_vg

# mkfs.ext4 /dev/shared_vg/ha_lv

# lvchange -an shared_vg/ha_lv
```

   Para informações sobre como criar os volumes lógicos LVM, consulte o *Logical Volume Manager Administration*.
3. Edite o arquivo **/etc/cluster/cluster.conf** para incluir um volume lógico recente como um recurso em um dos seus serviços. Como forma alternativa, você pode utilizar o **Conga** ou o **ccs** para configurar o LVM e recursos de sistema de arquivo para o cluster. Segue um exemplo de seção de gerenciador de recurso do arquivo **/etc/cluster/cluster.conf** que configura um volume lógico CLVM como um recurso de cluster:

```
<rm>
   <failoverdomains>
       <failoverdomain name="FD" ordered="1" restricted="0">
```

```
            <failoverdomainnode name="neo-01" priority="1"/>
            <failoverdomainnode name="neo-02" priority="2"/>
        </failoverdomain>
    </failoverdomains>
    <resources>
        <lvm name="lvm" vg_name="shared_vg" lv_name="ha-lv"/>
        <fs name="FS" device="/dev/shared_vg/ha-lv" force_fsck="0"
force_unmount="1" fsid="64050" fstype="ext4" mountpoint="/mnt"
options="" self_fence="0"/>
    </resources>
    <service autostart="1" domain="FD" name="serv"
recovery="relocate">
        <lvm ref="lvm"/>
        <fs ref="FS"/>
    </service>
</rm>
```

F.2. CONFIGURANDO UM FAILOVER DE HA-LVM COM A MARCAÇÃO

Para configurar um failover HA-LVM usando marcações no arquivo **/etc/lvm/lvm.conf**, realize os seguintes passos:

1. Certifique-se de que o parâmetro **locking_type** na seção global do arquivo **/etc/lvm/lvm.conf** é definida para valor '1'.

2. Crie um volume lógico e sistema de arquivo utilizando o LVM padrão e comandos de sistema de arquivo, como no exemplo a seguir.

```
# pvcreate /dev/sd[cde]1

# vgcreate shared_vg /dev/sd[cde]1

# lvcreate -L 10G -n ha_lv shared_vg

# mkfs.ext4 /dev/shared_vg/ha_lv
```

   Para informações sobre como criar os volumes lógicos LVM, consulte o *Logical Volume Manager Administration*.

3. Edite o arquivo **/etc/cluster/cluster.conf** para incluir um volume lógico recente como um recurso em um dos seus serviços. Como forma alternativa, você pode utilizar o **Conga** ou o **ccs** para configurar o LVM e recursos de sistema de arquivo para o cluster. Segue um exemplo de seção de gerenciador de recurso do arquivo **/etc/cluster/cluster.conf** que configura um volume lógico CLVM como um recurso de cluster:

```
<rm>
    <failoverdomains>
        <failoverdomain name="FD" ordered="1" restricted="0">
            <failoverdomainnode name="neo-01" priority="1"/>
            <failoverdomainnode name="neo-02" priority="2"/>
        </failoverdomain>
    </failoverdomains>
    <resources>
```

```
        <lvm name="lvm" vg_name="shared_vg" lv_name="ha_lv"/>
        <fs name="FS" device="/dev/shared_vg/ha_lv" force_fsck="0"
force_unmount="1" fsid="64050" fstype="ext4" mountpoint="/mnt"
options="" self_fence="0"/>
   </resources>
   <service autostart="1" domain="FD" name="serv"
recovery="relocate">
        <lvm ref="lvm"/>
        <fs ref="FS"/>
   </service>
</rm>
```

NOTA

Se houver volumes lógicos múltiplos no grupo de volume, então o nome do volume lógico (**lv_name**) no recurso **lvm** deve ser deixado em branco ou não especificado. Também observe que em uma configuração HA-LVM, um grupo de volume pode ser usado por somente um único serviço.

4. Edite o campo **volume_list** no arquivo **/etc/lvm/lvm.conf**. Inclua o nome de seu grupo de volume root e seu hostname como listado no arquivo **/etc/cluster/cluster.conf** precedido por um @. O hostname a ser incluído aqui, é a máquina na qual você está editando o arquivo **lvm.conf**, e não qualquer hostname remoto. Observe, que esta faixa *DEVE* coincidir com o nome do nó dado no arquivo **cluster.conf**. Segue abaixo uma entrada de exemplo do arquivo **/etc/lvm/lvm.conf**:

   ```
   volume_list = [ "VolGroup00", "@neo-01" ]
   ```

   Esta marcação será usada para ativar VGs ou LVs compartilhados. *Não* inclua os nomes de qualquer grupo de volume que forem compartilhados utilizando o HA-LVM.

5. Atualize o dispositivo **initrd** em todos os nós de cluster:

   ```
   # dracut -H -f /boot/initramfs-$(uname -r).img $(uname -r)
   ```

6. Reinicialize todos os nós para certificar-se de que o dispositivo correto **initrd** está sendo usado:

APÊNDICE G. HISTÓRICO DE REVISÕES

Revisão 5.0-25.3.400 **2013-10-31** **Rüdiger Landmann**
Rebuild with publican 4.0.0

Revisão 5.0-25.3 **Mon Apr 29 2013** **Glaucia Cintra**
pt-BR translation completed

Revisão 5.0-25.2 **Mon Apr 29 2013** **Glaucia Cintra**
pt-BR translation completed

Revisão 5.0-25.1 **Thu Apr 18 2013** **Chester Cheng**
Tradução de arquivos sincronizados com a versão 5.0-25 de fontes do XML

Revisão 5.0-25 **Mon Feb 18 2013** **Steven Levine**
Versão para o lançamento do 6.4 GA

Revisão 5.0-23 **Wed Jan 30 2013** **Steven Levine**
Resolve: 901641
Corrige e explica regras do iptables.

Revisão 5.0-22 **Tue Jan 29 2013** **Steven Levine**
Resolve: 788636
Documenta a configuração do RRP através do comando **`ccs`**.

Resolve: 789010
Documenta a configuração do RRP no arquivo **`cluster.conf`**.

Revisão 5.0-20 **Fri Jan 18 2013** **Steven Levine**
Resolve: 894097
Remove a sugestão de você se assegurar que não está usando a marcação do VLAN.

Resolve: 845365
Indica que os modos de vínculo 0 e 2 são agora suportados.

Revisão 5.0-19 **Thu Jan 17 2013** **Steven Levine**
Resolve: 896234
Explica a terminologia das referências do nó de cluster.

Revisão 5.0-16 **Mon Nov 26 2012** **Steven Levine**
Lançamento da versão 6.4 Beta

Revisão 5.0-15 **Wed Nov 20 2012** **Steven Levine**

Resolve: 838988
Documenta o atributo nfsrestart para os agentes de recurso do sistema de arquivo.

Resolve: 843169
Documenta o agente fence do IBM iPDU.

Resolve: 846121
Documenta o agente do fence do Eaton Network Power Controller (SNMP Interface).

Resolve: 856834
Documenta o agente do fence do HP Bladesystem.

Resolve: 865313
Documenta o agente de recurso do NFS Server.

Resolve: 862281
Explica quais comandos **`ccs`** sobrescrevem as configurações anteriores.

Resolve: 846205
Documenta o filtro do firewall do **`iptables`** para o componente de **`igmp`** component.

Resolve: 857172
Documenta a habilidade de remover os usuários do luci.

Resolve: 857165
Documenta o parâmetro do nível de privilégio do agente fence do IPMI.

Resolve: 840912
LImpa o problema de formatação com a tabela de parâmetro de recurso.

Resolve: 849240, 870292
Explica o procedimento de instalação.

Resolve: 871165
Explica a descrição do parâmetro do endereço IP na descrição do agente de recurso do endereço IP.

Resolve: 845333, 869039, 856681
Repara erros de digitação e explica ambiguidades de técnicas pequenas.

Revisão 5.0-12 **Thu Nov 1 2012** **Steven Levine**
Foram adicionados os agentes de fence recentemente suportados.

Revisão 5.0-7 **Thu Oct 25 2012** **Steven Levine**
Foi adicionada a seção sobre semântica de sobreposição.

Revisão 5.0-6 **Tue Oct 23 2012** **Steven Levine**
Foi reparado o valor padrão do Post Join Delay.

Revisão 5.0-4 **Tue Oct 16 2012** **Steven Levine**
Foi adicionada a descrição do recurso de servidor NFS.

Revisão 5.0-2 **Thu Oct 11 2012** **Steven Levine**
Atualizações nas descrições do Conga.

Revisão 5.0-1 **Mon Oct 8 2012** **Steven Levine**
Explicação das semânticas do ccs.

Revisão 4.0-5 **Fri Jun 15 2012** **Steven Levine**

Versão para lançamento do 6.3 GA

Revisão 4.0-4 **Tue Jun 12 2012** **Steven Levine**

Resolve: 830148

Garante consistência de exemplos de número de portas para luci.

Revisão 4.0-3 **Tue May 21 2012** **Steven Levine**

Resolve: 696897

Adiciona informações sobre o parâmetro cluster.conf nas tabelas de parâmetors de dispositivos fence e parâmetros de recursos.

Resolve: 811643

Adiciona procedimento para restaurar um banco de dados **luci** em uma máquina separada.

Revisão 4.0-2 **Wed Apr 25 2012** **Steven Levine**

Resolve: 815619

Remove aviso sobre o uso do UDP Unicast com os sistemas de arquivo GFS2.

Revisão 4.0-1 **Fri Mar 30 2012** **Steven Levine**

Resolve: 771447, 800069, 800061

Atualiza a documentação de **luci** para ser consistente com a versão do Red Hat Enterprise Linux 6.3.

Resolves: 712393

Adiciona informações sobre como capturar um núcleo de aplicativo para o RGManager.

Resolve: 800074

Documenta o agente de recurso do `condor`.

Resolves: 757904

Documenta o backup de configuração do **luci** e os restaura.

Resolves: 772374

Adiciona a seção sobre gerenciamento de máquinas virtuais em um cluster.

Resolves: 712378

Adiciona documenta para a configuração do HA-LVM.

Resolves: 712400

Documenta opções de depuração.

Resolve: 751156

Documenta novos parâmetros de `fence_ipmilan`.

Resolve: 721373

Documenta quais mudanças de configuração requerem uma reiniciação de cluster.

Revisão 3.0-5 **Thu Dec 1 2011** **Steven Levine**

Lançamento para o GA do Red Hat Enterprise Linux 6.2

Resolve: 755849

Corrige o exemplo de parâmetro do monitor_link.

Revisão 3.0-4 **Mon Nov 7 2011** **Steven Levine**

Resolve: 749857

Adiciona a documentação para o dispositivo RHEV-M REST API fence.

Revisão 3.0-3 **Fri Oct 21 2011** **Steven Levine**

Resolve: #747181, #747182, #747184, #747185, #747186, #747187, #747188, #747189, #747190, #747192
Corrige os erros de digitação e ambiguidades encontrados durante a revisão de QE da documentação para o Red Hat Enterprise Linux 6.2.

Revisão 3.0-2 **Fri Oct 7 2011** **Steven Levine**

Resolve: #743757
Corrige a referência para modo de vínculo suportado na seção de troubleshooting.

Revisão 3.0-1 **Wed Sep 28 2011** **Steven Levine**

Revisão inicial para o lançamento Red Hat Enterprise Linux 6.2 Beta.

Resolve: #739613
Suporte de documentos para novas opções do **ccs** para exibir os dispositivos do fence disponíveis e serviços disponíveis.

Resolve: #707740
Atualizações dos documentos para a interface do Conga e suporte de documentos para configurar permissões de usuário para administrar o Conga.

Resolve: #731856
Suportes de documentos para configurar o **luci** através do arquivo `/etc/sysconfig/luci`.

Resolve: #736134
Documentação de Suporte para transporte UDPU.

Resolve: #736143
Documentação de Suporte para Samba em cluster.

Resolve: #617634
Documenta como configurar o único endereço IP do **luci** no qual é servido.

Resolve: #713259
Documentação de suporte para o agente `fence_vmware_soap`.

Resolve: #721009
Fornece um link para artigo de Suporte de Essenciais.

Resolve: #717006
Fornece informações sobre como permitir o tráfego de multicast através do firewall `iptables`.

Resolve: #717008
Fornece informações sobre o estado do serviço cluster e failover timeout.

Resolve: #711868
Explica descrição do autostart.

Resolve: #728337
Documenta procedimento para adicionar os recursos do **vm** com o **ccs**.

Resolve: #725315, #733011, #733074, #733689
Corrige erros pequenos de digitação.

Revisão 2.0-1 **Thu May 19 2011** **Steven Levine**

Revisão inicial do Red Hat Enterprise Linux 6.1

Resolve: #671250
Documentação de Suporte para SNMP traps.

Resolves: #659753
Documenta comando **ccs**

Resolves: #665055
Atualiza documnetação Conga para refletir exibição atualizada e suporte de recurso.

Resolves: #680294
Documenta a necessidade de senha de acesso para o agente **ricci**.

Resolves: #687871
Adicionado capítulo na resolução de problemas

Resolves: #673217
Consertado erros de digitação

Resolves: #675805
Adiciona referência ao esquema **cluster.conf** às tabelas de parâmetros de recursos HA.

Resolves: #672697
Atualiza tabelas de dispositivos fence para incluir todos dispositivos fencing atualmente suportados.

Resolves: #677994
Corrige informações para parâmetros de agente fence **fence_ilo**.

Resolves: #629471
Adiciona nota técnica sobre configurar valor consensus em um cluster de dois nós.

Resolves: #579585
Atualiza seção de atualização do Complemento de Alta Disponibilidade Red Hat.

Resolves: #643216
Clarifica pequenos problemas no documento.

Resolves: #643191
Fornece aprimoramentos e correções para a documentação **luci**.

Resolves: #704539
Atualiza a tabela de parâmetros de recursos de Máquina Virtual.

Revisão 1.0-1 **Wed Nov 10 2010** **Paul Kennedy**
Lançamento inicial do Red Hat Enterprise Linux 6

ÍNDICE REMISSIVO

A

fence_ilo, Parâmetros de Dispositos Fence
fence_ilo_mp, Parâmetros de Dispositos Fence
fence_intelmodular, Parâmetros de Dispositos Fence
fence_ipdu, Parâmetros de Dispositos Fence
fence_ipmilan, Parâmetros de Dispositos Fence
fence_rhevm, Parâmetros de Dispositos Fence
fence_rsb, Parâmetros de Dispositos Fence
fence_scsi, Parâmetros de Dispositos Fence
fence_virt, Parâmetros de Dispositos Fence
fence_vmware_soap, Parâmetros de Dispositos Fence
fence_wti, Parâmetros de Dispositos Fence

fence device
APC power switch over SNMP, Parâmetros de Dispositos Fence
Brocade fabric switch, Parâmetros de Dispositos Fence
Cisco MDS, Parâmetros de Dispositos Fence
Cisco UCS, Parâmetros de Dispositos Fence
Dell DRAC 5, Parâmetros de Dispositos Fence
Eaton network power switch, Parâmetros de Dispositos Fence
Egenera SAN controller, Parâmetros de Dispositos Fence
ePowerSwitch, Parâmetros de Dispositos Fence
Fence virt, Parâmetros de Dispositos Fence
Fujitsu Siemens Remoteview Service Board (RSB), Parâmetros de Dispositos Fence
HP BladeSystem, Parâmetros de Dispositos Fence
HP iLO MP, Parâmetros de Dispositos Fence
HP iLO/iLO2, Parâmetros de Dispositos Fence
IBM BladeCenter, Parâmetros de Dispositos Fence
IBM iPDU, Parâmetros de Dispositos Fence
IF MIB, Parâmetros de Dispositos Fence
Intel Modular, Parâmetros de Dispositos Fence
interruptor de energia APC sob telnet/SSH, Parâmetros de Dispositos Fence
IPMI LAN, Parâmetros de Dispositos Fence
RHEV-M REST API, Parâmetros de Dispositos Fence
SCSI fencing, Parâmetros de Dispositos Fence
VMware (SOAP interface), Parâmetros de Dispositos Fence
WTI power switch, Parâmetros de Dispositos Fence

fence_apc fence agent, Parâmetros de Dispositos Fence
fence_apc_snmp fence agent, Parâmetros de Dispositos Fence
fence_bladecenter fence agent, Parâmetros de Dispositos Fence
fence_brocade fence agent, Parâmetros de Dispositos Fence
fence_cisco_mds fence agent, Parâmetros de Dispositos Fence

G

H

I

recursos novos e modificados, Recursos Novos e Modificados

VMware (SOAP interface) fence device , Parâmetros de Dispositos Fence