Red Hat Enterprise Linux 8

Configuração das configurações básicas do sistema

Um guia para configurar as configurações básicas do sistema no Red Hat Enterprise Linux 8

Last Updated: 2021-02-24

Red Hat Enterprise Linux 8 Configuração das configurações básicas do sistema

Um guia para configurar as configurações básicas do sistema no Red Hat Enterprise Linux 8

Enter your first name here. Enter your surname here.
Enter your organisation's name here. Enter your organisational division here.
Enter your email address here.

Nota Legal

Resumo

Este documento descreve o básico da administração de sistemas no Red Hat Enterprise Linux 8. O título enfoca: tarefas básicas que um administrador de sistemas precisa fazer logo após o sistema operacional ter sido instalado com sucesso, instalar software com yum, usar systemd para gerenciamento de serviços, gerenciar usuários, grupos e permissões de arquivos, usar chrony para configurar o NTP, trabalhar com Python 3 e outros.

Índice

TORNANDO O CÓDIGO ABERTO MAIS INCLUSIVO

A Red Hat tem o compromisso de substituir a linguagem problemática em nosso código, documentação e propriedades da web. Estamos começando com estes quatro termos: master, slave, blacklist e whitelist. Por causa da enormidade deste esforço, estas mudanças serão implementadas gradualmente ao longo de vários lançamentos futuros. Para mais detalhes, veja a mensagem de nosso CTO Chris Wright.

FORNECENDO FEEDBACK SOBRE A DOCUMENTAÇÃO DA RED HAT

Agradecemos sua contribuição em nossa documentação. Por favor, diga-nos como podemos melhorá-la. Para fazer isso:

- Para comentários simples sobre passagens específicas:
  1. Certifique-se de que você está visualizando a documentação no formato *Multi-page HTML*. Além disso, certifique-se de ver o botão **Feedback** no canto superior direito do documento.
  2. Use o cursor do mouse para destacar a parte do texto que você deseja comentar.
  3. Clique no pop-up **Add Feedback** que aparece abaixo do texto destacado.
  4. Siga as instruções apresentadas.
- Para enviar comentários mais complexos, crie um bilhete Bugzilla:
  1. Ir para o site da Bugzilla.
  2. Como Componente, use **Documentation**.
  3. Preencha o campo **Description** com sua sugestão de melhoria. Inclua um link para a(s) parte(s) relevante(s) da documentação.
  4. Clique em **Submit Bug**.

CAPÍTULO 1. COMEÇANDO COM A ADMINISTRAÇÃO DO SISTEMA

As seções seguintes fornecem uma visão geral das tarefas básicas de administração do sistema instalado.

NOTA

As seguintes tarefas administrativas básicas podem incluir itens que normalmente já são feitos durante o processo de instalação, mas não precisam ser feitos necessariamente, tais como o registro do sistema. As seções que tratam de tais tarefas fornecem um resumo de como se pode alcançar os mesmos objetivos durante a instalação.

Para informações sobre a instalação do Red Hat Enterprise Linux, consulte Execução de uma instalação padrão da RHEL.

Embora você possa realizar todas as tarefas pós-instalação através da linha de comando, você também pode usar o console web RHEL 8 para realizar algumas delas.

1.1. COMEÇANDO A USAR O CONSOLE WEB RHEL

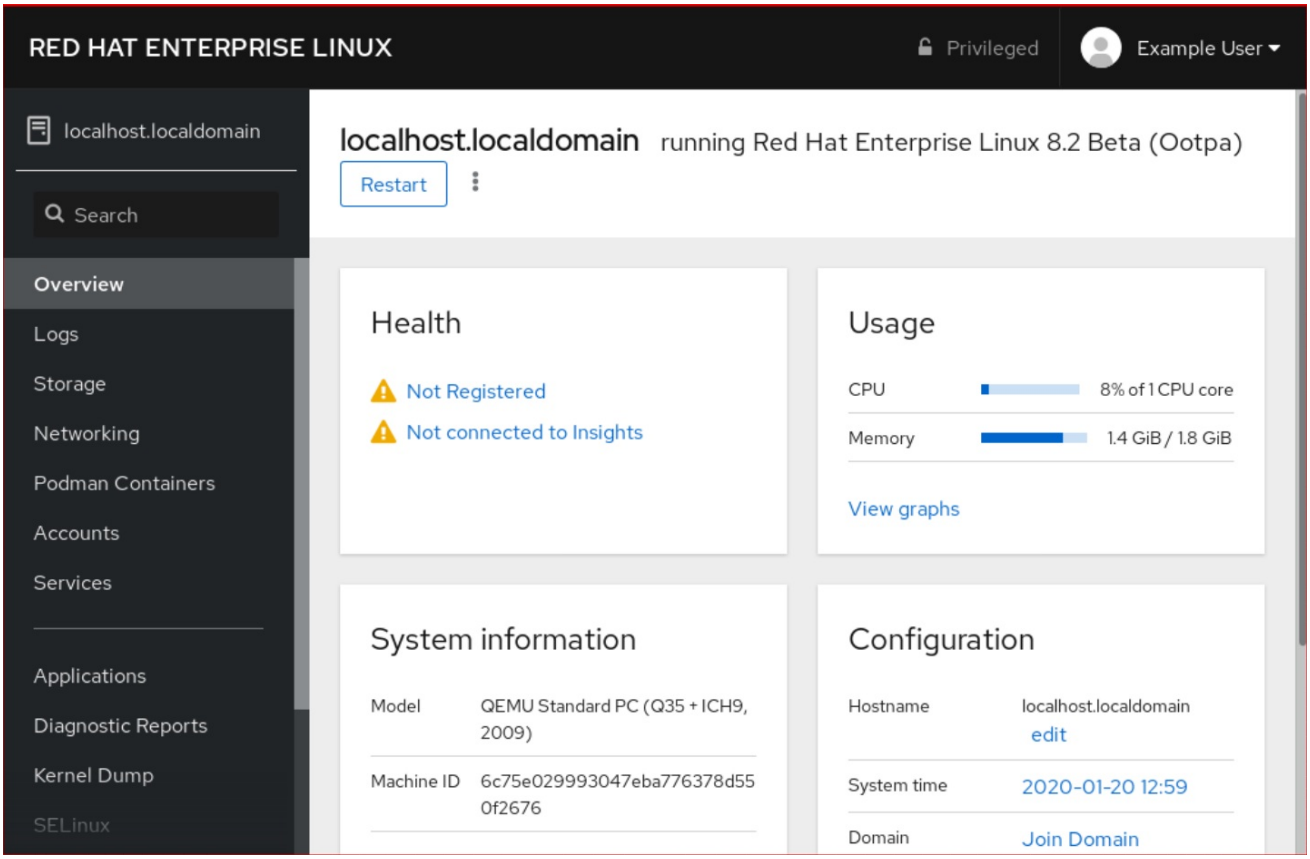
Instale o console web no Red Hat Enterprise Linux 8 e aprenda como adicionar máquinas remotas e monitorá-las no console web RHEL 8.

Pré-requisitos

- Instalado o Red Hat Enterprise Linux 8.
- Rede de contatos capacitada.
- Sistema registrado com assinatura apropriada anexada.
  Para obter uma assinatura, consulte Gerenciando assinaturas no console web .

1.1.1. O que é o console web RHEL

O console web RHEL é uma interface baseada na web do Red Hat Enterprise Linux 8 projetada para gerenciar e monitorar seu sistema local, assim como servidores Linux localizados em seu ambiente de rede.



O console web RHEL lhe permite uma ampla gama de tarefas administrativas, inclusive:

- Gerenciamento de serviços
- Gerenciando contas de usuários
- Gerenciamento e monitoramento de serviços do sistema
- Configuração de interfaces de rede e firewall
- Revisão dos logs do sistema
- Gerenciamento de máquinas virtuais
- Criação de relatórios de diagnóstico
- Configuração do kernel dump
- Configurando a SELinux
- Atualização de software
- Gestão de assinaturas de sistemas

O console web RHEL usa as mesmas APIs do sistema que você usaria em um terminal, e as ações realizadas em um terminal são imediatamente refletidas no console web RHEL.

Você pode monitorar os logs dos sistemas no ambiente da rede, assim como seu desempenho, exibidos como gráficos. Além disso, você pode alterar as configurações diretamente no console web ou através do terminal.

1.1.2. Instalando e habilitando o console web

Para acessar o console web RHEL 8, primeiro habilite o serviço **cockpit.socket**.

O Red Hat Enterprise Linux 8 inclui o console web RHEL 8 instalado por default em muitas variantes de instalação. Se este não for o caso em seu sistema, instale o pacote **cockpit** antes de habilitar o serviço **cockpit.socket**.

Procedimento

1. Se o console web não for instalado por padrão em sua variante de instalação, instale manualmente o pacote **cockpit**:

```
# yum install cockpit
```

2. Habilitar e iniciar o serviço **cockpit.socket**, que roda um servidor web:

```
# systemctl enable --now cockpit.socket
```

3. Se o console web não foi instalado por padrão em sua variante de instalação e você estiver usando um perfil de firewall personalizado, adicione o serviço **cockpit** a **firewalld** para abrir a porta 9090 no firewall:

```
# firewall-cmd --add-service=cockpit --permanent
# firewall-cmd --reload
```

Etapas de verificação

1. Para verificar a instalação e configuração anteriores, abra o console web.

1.1.3. Login no console web

Use as etapas deste procedimento para o primeiro login no console web RHEL usando um nome de usuário e senha do sistema.

Pré-requisitos

- Use um dos seguintes navegadores para abrir o console web:
  - Mozilla Firefox 52 e posteriores
  - Google Chrome 57 e posteriores
  - Microsoft Edge 16 e posteriores
- Credenciais de conta de usuário do sistema
  O console web RHEL utiliza uma pilha PAM específica, localizada em **/etc/pam.d/cockpit**. A autenticação com PAM permite o login com o nome de usuário e senha de qualquer conta local no sistema.

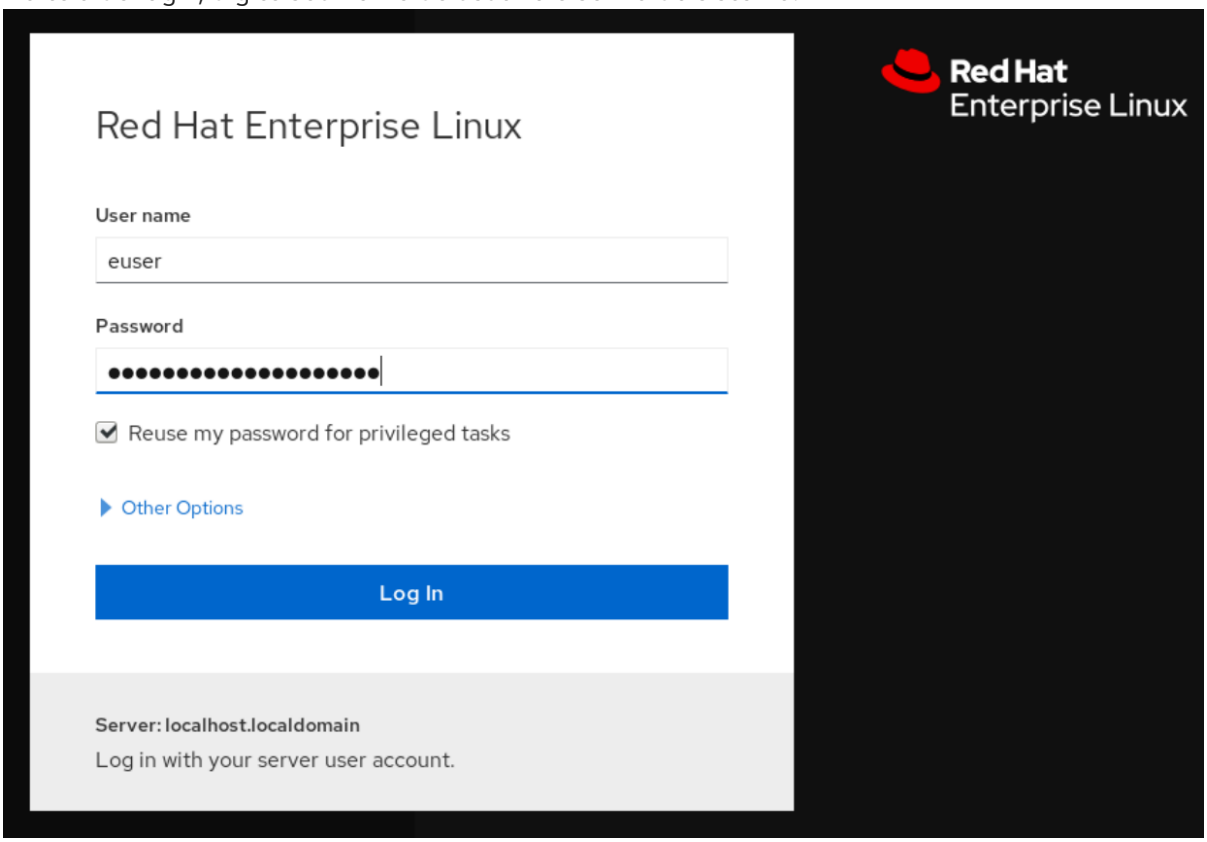
Procedimento

1. Abra o console web em seu navegador web:
   - Localmente **https://localhost:9090**

- Remotamente com o hostname do servidor **https://example.com:9090**
- Remotamente com o endereço IP do servidor **https://192.0.2.2:9090**
  Se você usar um certificado autoassinado, o navegador emite um aviso. Verifique o certificado e aceite a exceção de segurança para proceder com o login.

  O console carrega um certificado do diretório **/etc/cockpit/ws-certs.d** e usa o último arquivo com uma extensão **.cert** em ordem alfabética. Para evitar ter que conceder exceções de segurança, instale um certificado assinado por uma autoridade certificadora (CA).

2. Na tela de login, digite seu nome de usuário e senha do sistema.



3. Opcionalmente, clique na opção **Reuse my password for privileged tasks**
   Se a conta de usuário que você está usando para fazer login tem privilégios sudo, isto torna possível executar tarefas privilegiadas no console web, tais como instalar software ou configurar o SELinux.

4. Clique em **Log In**.

Após a autenticação bem sucedida, a interface do console web RHEL se abre.

1.1.4. Conexão ao console web a partir de uma máquina remota

É possível conectar-se à interface de seu console web a partir de qualquer sistema operacional do cliente e também a partir de telefones celulares ou tablets.

Pré-requisitos

- Dispositivo com um navegador de internet suportado, como por exemplo:
  - Mozilla Firefox 52 e posteriores

- Google Chrome 57 e posteriores
- Microsoft Edge 16 e posteriores

- O servidor RHEL 8 que você deseja acessar com um console web instalado e acessível. Para mais informações sobre a instalação do console web, consulte Instalando o console web.

Procedimento

1. Abra seu navegador web.
2. Digite o endereço do servidor remoto em um dos seguintes formatos:
   a. Com o nome do anfitrião do servidor **server.hostname.example.com:port_number**
   b. Com o endereço IP do servidor **server.IP_address:port_number**
3. Após a abertura da interface de login, faça o login com suas credenciais da máquina RHEL.

1.1.5. Login no console web usando uma senha única

Se seu sistema faz parte de um domínio de Gerenciamento de Identidade (IdM) com configuração de senha única (OTP) habilitada, você pode usar um OTP para fazer login no console web RHEL.

IMPORTANTE

É possível entrar usando uma senha única somente se seu sistema for parte de um domínio de Gerenciamento de Identidade (IdM) com configuração OTP habilitada. Para mais informações sobre OTP em IdM, consulte Senha única em Gerenciamento de Identidade.

Pré-requisitos

- O console web RHEL foi instalado.
  Para detalhes, consulte Instalando o console web.
- Um servidor de Gerenciamento de Identidade com configuração OTP habilitada.
  Para obter detalhes, consulte Senha única em Gerenciamento de Identidade .
- Um dispositivo de hardware ou software configurado que gera fichas OTP.

Procedimento

1. Abra o console web RHEL em seu navegador:
   - Localmente **https://localhost:PORT_NUMBER**
   - Remotamente com o hostname do servidor **https://example.com:PORT_NUMBER**
   - Remotamente com o endereço IP do servidor
     https://EXAMPLE.SERVER.IP.ADDR:PORT_NUMBER
     Se você usar um certificado autoassinado, o navegador emite um aviso. Verifique o certificado e aceite a exceção de segurança para proceder com o login.

     O console carrega um certificado do diretório **/etc/cockpit/ws-certs.d** e usa o último arquivo com uma extensão **.cert** em ordem alfabética. Para evitar ter que conceder

exceções de segurança, instale um certificado assinado por uma autoridade certificadora (CA).

2. A janela de Login se abre. Na janela de Login, digite seu nome de usuário e senha do sistema.
3. Gerar uma senha única em seu dispositivo.
4. Digite a senha única em um novo campo que aparece na interface do console web depois de confirmar sua senha.
5. Clique em **Log in**.
6. O login bem sucedido o leva à página **Overview** da interface do console web.

1.1.6. Reiniciar o sistema usando o console web

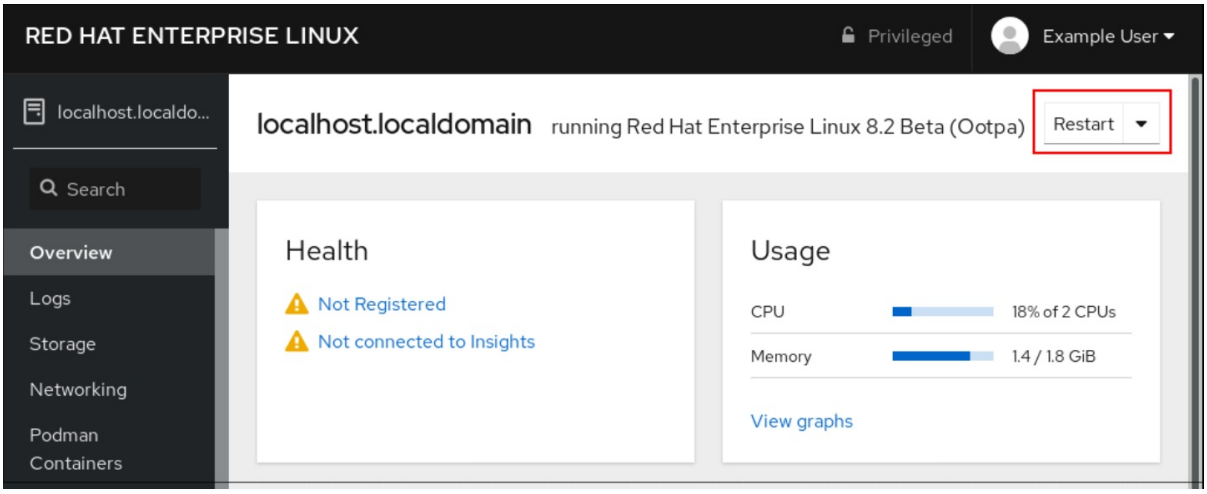
Você pode usar o console web para reiniciar um sistema RHEL ao qual o console web está anexado.

Pré-requisitos

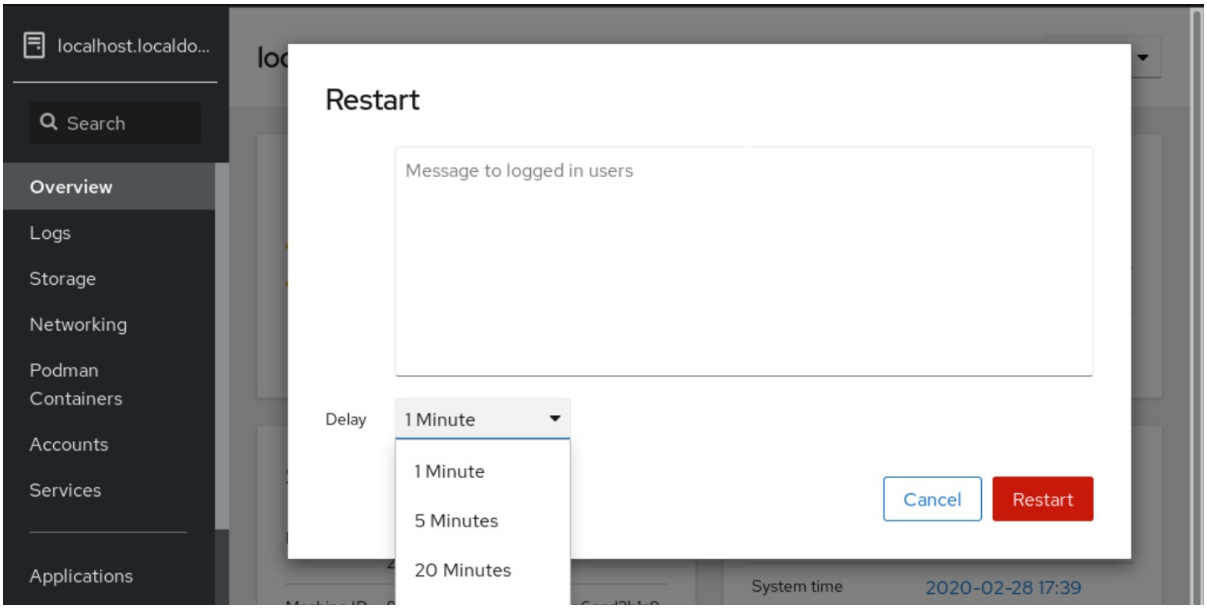
- O console web é instalado e acessível.
  Para detalhes, consulte Instalando o console web.

Procedimento

1. Acesse o console web RHEL 8.
   Para obter detalhes, consulte Login no console web.
2. Clique em **Overview**.
3. Clique no botão reiniciar do **Restart**.



4. Se algum usuário estiver logado no sistema, escreva um motivo para o reinício na caixa de diálogo **Restart**.
5. Opcional: Na lista suspensa **Delay**, selecione um intervalo de tempo.



6. Clique em **Restart**.

1.1.7. Desligamento do sistema usando o console web

Você pode usar o console web para desligar um sistema RHEL ao qual o console web está anexado.

Pré-requisitos

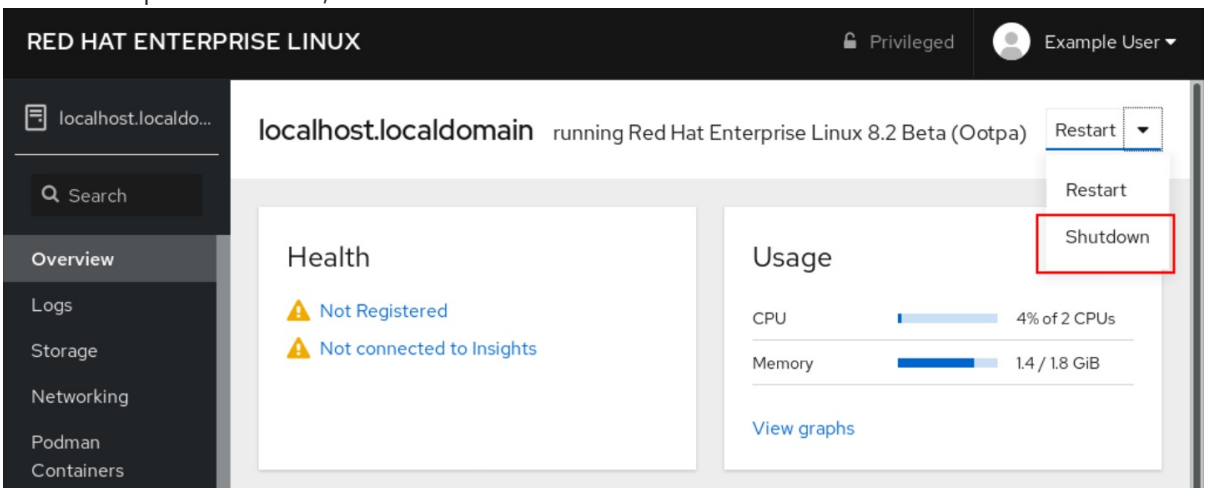
- O console web é instalado e acessível.
  Para detalhes, consulte Instalando o console web.

Procedimento

1. Acesse o console web RHEL 8.
   Para obter detalhes, consulte Login no console web.

2. Clique em **Overview**.

3. Na lista suspensa **Restart**, selecione **Shut Down**.



4. Se algum usuário estiver logado no sistema, escreva um motivo para o desligamento na caixa de diálogo **Shut Down**.

5. Opcional: Na lista suspensa **Delay**, selecione um intervalo de tempo.

6. Clique em **Shut Down**.

1.1.8. Configuração de tempo usando o console web

Você pode definir um fuso horário e sincronizar a hora do sistema com um servidor Network Time Protocol (NTP).

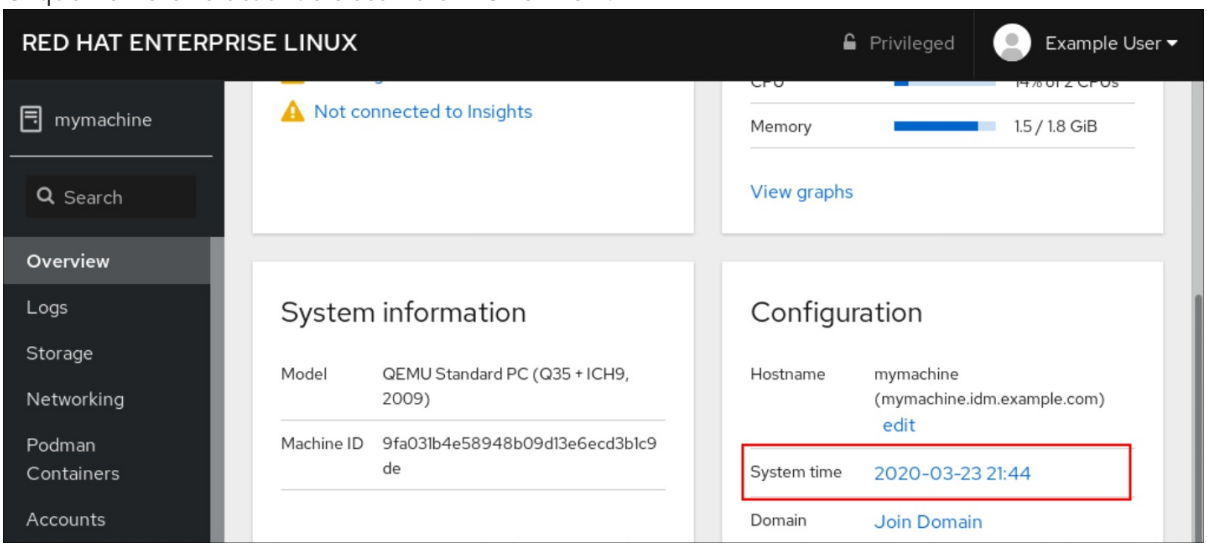
Pré-requisitos

- O console web é instalado e acessível.
  Para detalhes, consulte Instalando o console web.

Procedimento

1. Acesse o console web RHEL 8.
   Para obter detalhes, consulte Login no console web.

2. Clique no horário atual do sistema em **Overview**.



3. Na caixa de diálogo **Change System Time**, altere o fuso horário, se necessário.

4. No menu suspenso **Set Time**, selecione uma das seguintes opções:

   Manualmente

   Use esta opção se você precisar definir o tempo manualmente, sem um servidor NTP.

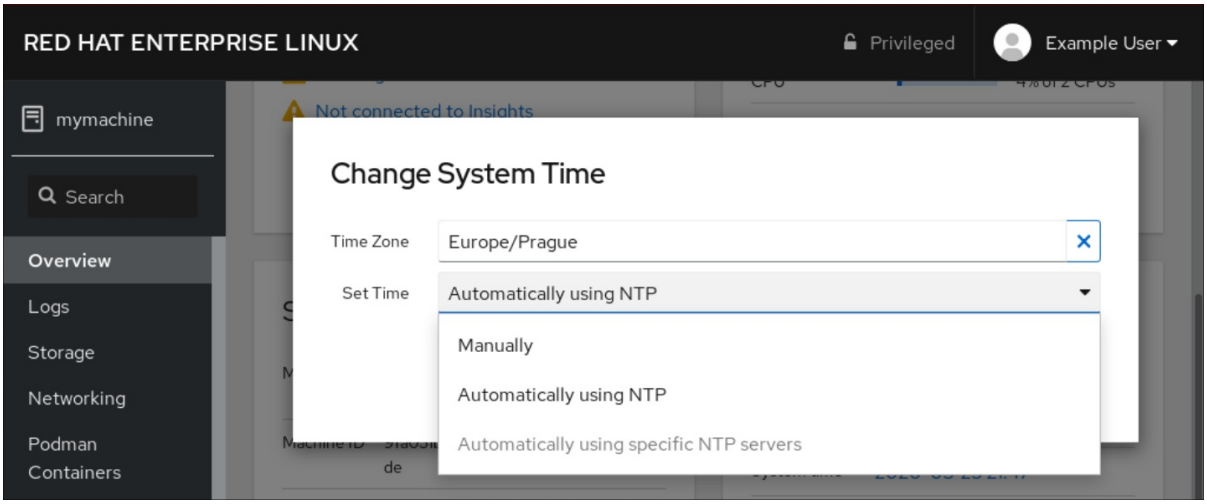
   Usando automaticamente o servidor NTP

   Esta é uma opção padrão, que sincroniza o tempo automaticamente com os servidores NTP pré-definidos.

   Usando automaticamente servidores NTP específicos

   Use esta opção somente se você precisar sincronizar o sistema com um servidor NTP específico. Especifique o nome DNS ou o endereço IP do servidor.

5. Clique em **Change**.



Etapas de verificação

- Verifique o tempo do sistema exibido na guia **System**.

Recursos adicionais

- Usando a suíte Chrony para configurar o NTP .

1.1.9. Juntando um sistema RHEL 8 a um domínio IdM usando o console web

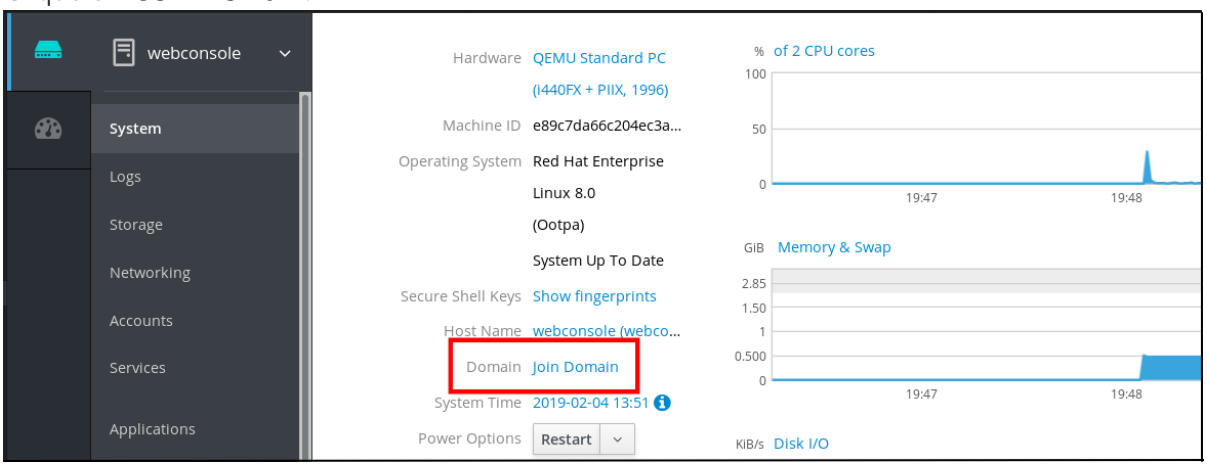
Você pode usar o console web para unir o sistema Red Hat Enterprise Linux 8 ao domínio de Gerenciamento de Identidade (IdM).

Pré-requisitos

- O domínio IdM está funcionando e é acessível a partir do cliente ao qual você deseja aderir.
- Você tem as credenciais de administrador do domínio IdM.

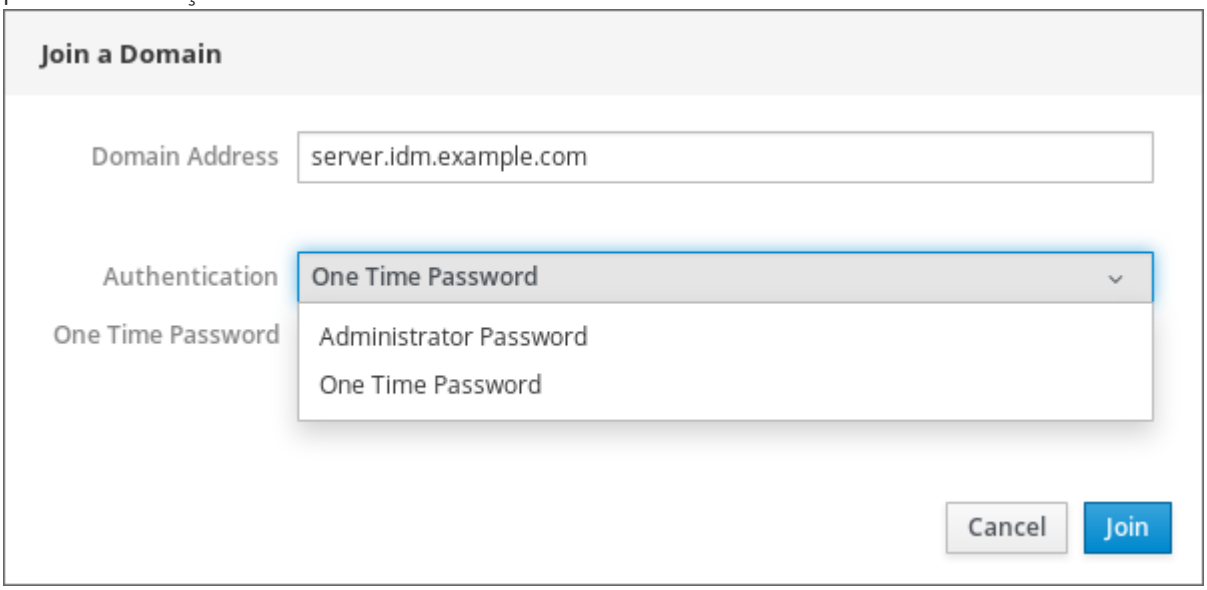
Procedimento

1. Acesse o console web RHEL.
   Para obter detalhes, consulte Login no console web.
2. Abra a aba **System**.
3. Clique em **Join Domain**.



4. Na caixa de diálogo **Join a Domain**, digite o nome do host do servidor IdM no campo **Domain Address**.

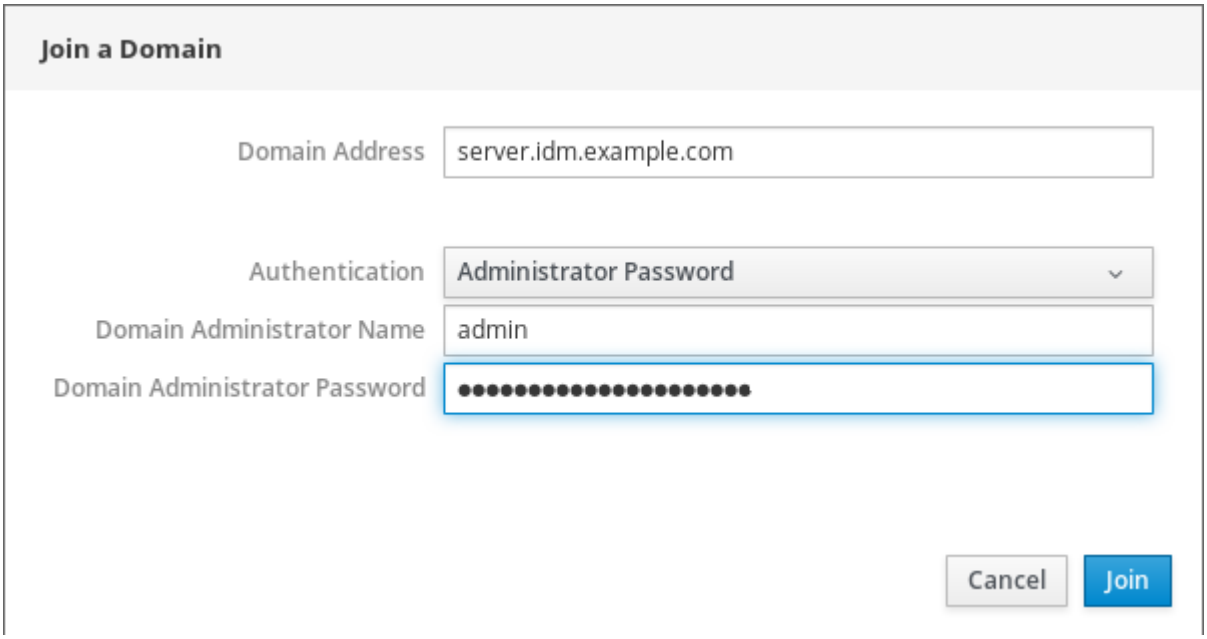
5. Na lista suspensa **Authentication**, selecione se você deseja usar uma senha ou uma senha única para autenticação.



6. No campo **Domain Administrator Name**, digite o nome do usuário da conta de administração da IdM.

7. No campo senha, adicione a senha ou senha única de acordo com o que você selecionou na lista suspensa **Authentication** anteriormente.

8. Clique em **Join**.



Etapas de verificação

1. Se o console web RHEL 8 não exibiu um erro, o sistema foi anexado ao domínio IdM e você pode ver o nome do domínio na tela **System**.

2. Para verificar se o usuário é um membro do domínio, clique na página Terminal e digite o comando **id**:

```
$ id
euid=548800004(example_user) gid=548800004(example_user)
groups=548800004(example_user) context=unconfined_u:unconfined_r:unconfined_t:s0-
s0:c0.c1023
```

Recursos adicionais

- Gerenciamento de Identidade de Planejamento
- Instalando o Gerenciamento de Identidade
- Configuração e gerenciamento da Gestão de Identidade

1.1.10. Desabilitando o SMT para evitar problemas de segurança da CPU usando o console web

Desativar SMT (Simultaneous Multi Threading) em caso de ataques que utilizem indevidamente a CPU SMT. Desabilitar o SMT pode mitigar as vulnerabilidades de segurança, tais como L1TF ou MDS.

IMPORTANTE

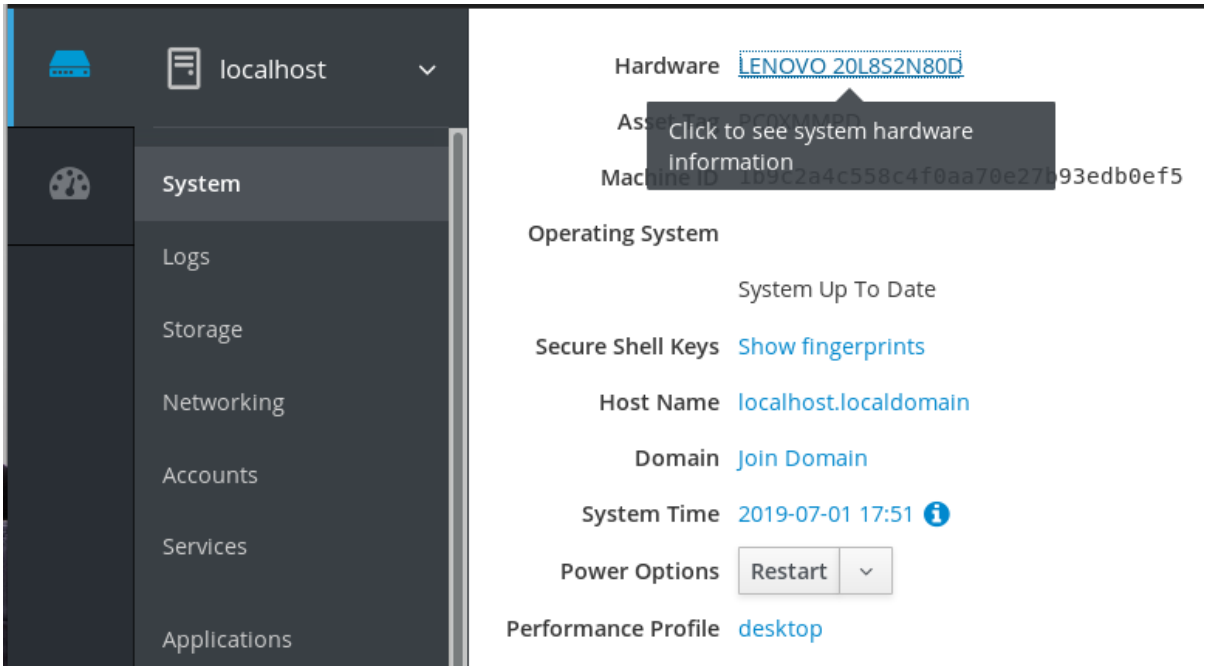
A desativação do SMT pode reduzir o desempenho do sistema.

Pré-requisitos

- O console web deve ser instalado e acessível.
  Para detalhes, consulte Instalando o console web.

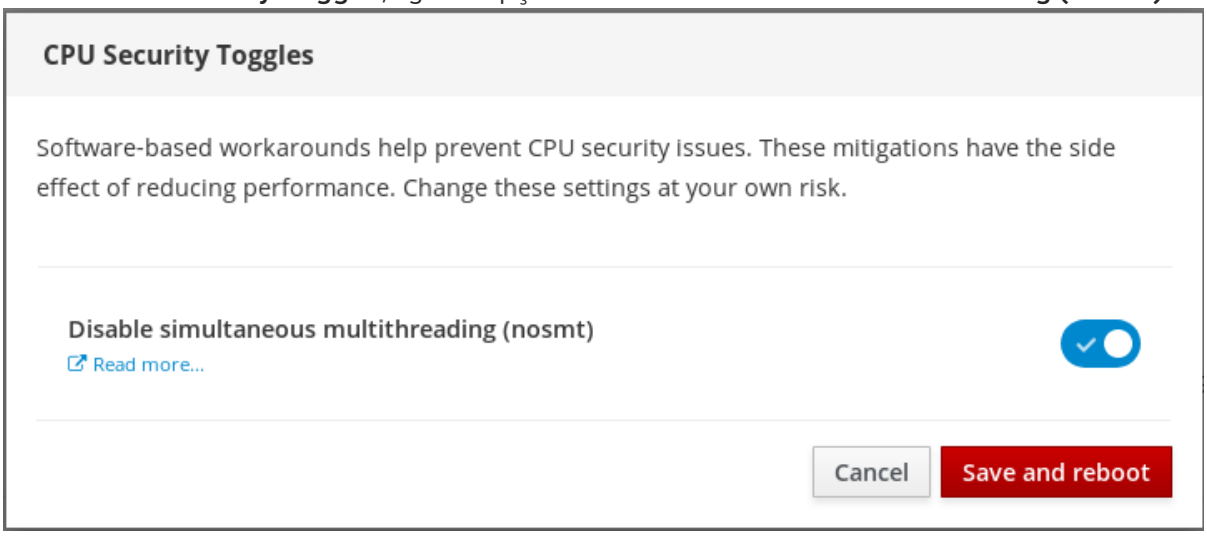
Procedimento

1. Acesse o console web RHEL 8.
   Para obter detalhes, consulte Login no console web.
2. Clique em **System**.
3. No item **Hardware**, clique nas informações de hardware.



4. No item **CPU Security**, clique em **Mitigations**.
   Se este link não estiver presente, significa que seu sistema não suporta SMT e, portanto, não é vulnerável.

5. No site **CPU Security Toggles**, ligue a opção **Disable simultaneous multithreading (nosmt)**



6. Clique no botão **Save and reboot**

Após o reinício do sistema, a CPU não usa mais SMT.

Recursos adicionais

- L1TF - L1 Terminal Fault Attack - CVE-2018-3620 & CVE-2018-3646
- MDS - Amostragem de dados microarquitetônicos - CVE-2018-12130, CVE-2018-12126, CVE-2018-12127, e CVE-2019-11091

1.1.11. Adicionando um banner à página de login

Empresas ou agências às vezes precisam mostrar um aviso de que o uso do computador é para fins legais, o usuário está sujeito a vigilância e qualquer invasão de propriedade será processada. O aviso

deve ser visível antes do login. Assim como o SSH, o console web pode opcionalmente mostrar o conteúdo de um arquivo de banner na tela de login. Para ativar os banners em suas sessões do console web, você precisa modificar o arquivo **/etc/cockpit/cockpit.conf**. Note que o arquivo não é necessário e você pode precisar criá-lo manualmente.

Pré-requisitos

- O console web é instalado e acessível. Para detalhes, consulte Instalando o console web.
- Você deve ter privilégios de sudo.

Procedimento

1. Crie o arquivo **/etc/issue.cockpit** em um editor de texto de sua preferência se você ainda não o tiver. Adicione o conteúdo que você deseja exibir como banner ao arquivo.
   Não inclua nenhuma macros no arquivo, pois não há nenhuma re-formatação feita entre o conteúdo do arquivo e o conteúdo exibido. Use as quebras de linha pretendidas. É possível usar a arte ASCII.
2. Salvar o arquivo.
3. Abra ou crie o arquivo **cockpit.conf** no diretório **/etc/cockpit/** em um editor de texto de sua preferência.

```
$ sudo vi cockpit.conf
```

4. Acrescente o seguinte texto ao arquivo:

```
[Session]
Banner=/etc/issue.cockpit
```

5. Salvar o arquivo.
6. Reinicie o console web para que as mudanças entrem em vigor.

```
# systemctl try-restart cockpit
```

Etapas de verificação

- Abra novamente a tela de login do console web para verificar se o banner está agora visível.

Exemplo 1.1. Adicionando um banner de exemplo à página de login

1. Criar um arquivo **/etc/issue.cockpit** com um texto desejado usando um editor de texto:

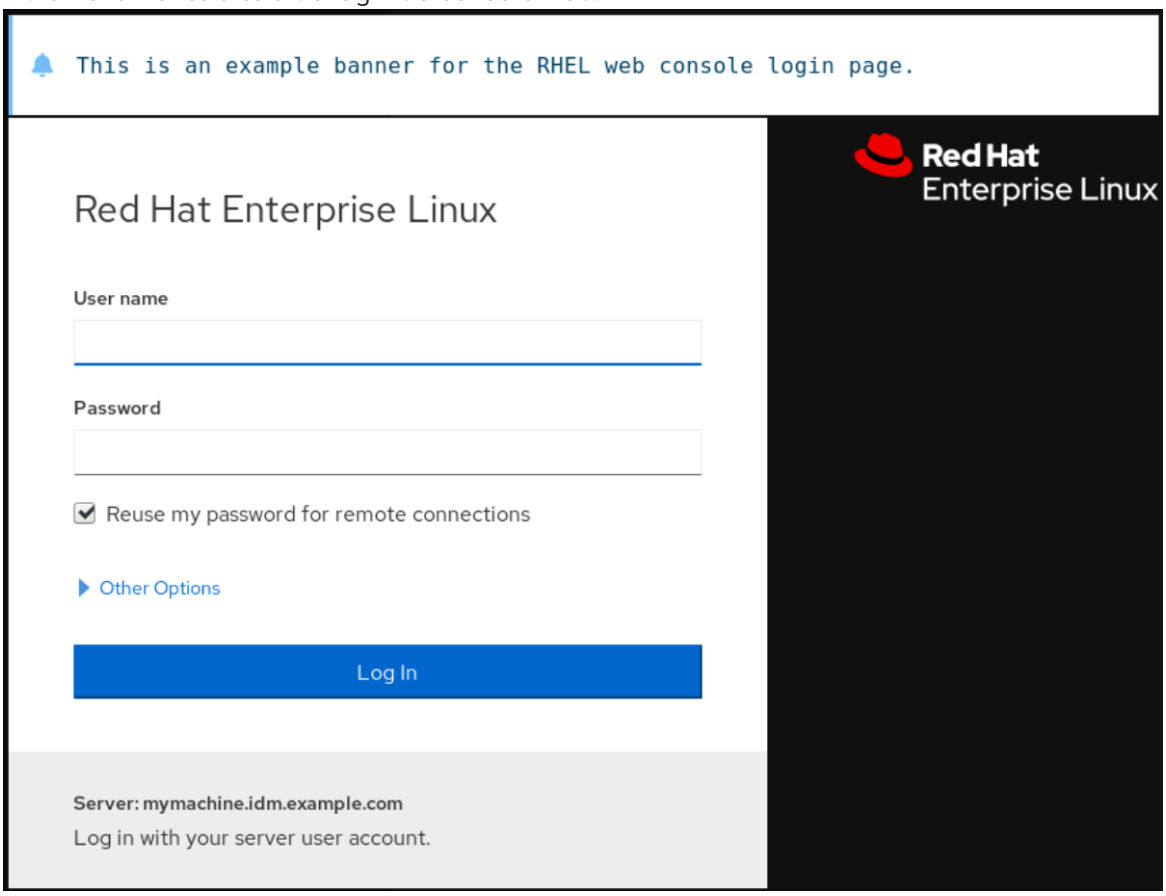
```
Este é um banner de exemplo para a página de login do console web RHEL.
```

2. Abra ou crie o arquivo **/etc/cockpit/cockpit.conf** e adicione o seguinte texto:

```
[Session]
Banner=/etc/issue.cockpit
```

3. Reinicie o console web.

4. Abra novamente a tela de login do console web.



1.1.12. Configuração do bloqueio automático de ociosidade no console web

Por padrão, não há tempo limite de inatividade definido na interface do console web. Se você desejar ativar um timeout ocioso em seu sistema, você pode fazê-lo modificando o arquivo de configuração **/etc/cockpit/cockpit.conf**. Note que o arquivo não é necessário e você pode precisar criá-lo manualmente.

Pré-requisitos

- O console web deve ser instalado e acessível.
  Para detalhes, consulte Instalando o console web.
- Você deve ter privilégios de sudo.

Procedimento

1. Abra ou crie o arquivo **cockpit.conf** no diretório **/etc/cockpit/** em um editor de texto de sua preferência.

```
$ sudo vi cockpit.conf
```

2. Acrescente o seguinte texto ao arquivo:

```
[Session]
IdleTimeout=X
```

Substitua **X** por um número por um período de tempo à sua escolha em minutos.

3. Salvar o arquivo.
4. Reinicie o console web para que as mudanças entrem em vigor.

```
# systemctl try-restart cockpit
```

Etapas de verificação

- Verifique se a sessão o registra após um determinado período de tempo.

1.2. CONFIGURANDO O NOME DO HOST NO CONSOLE WEB

Saiba como usar o console web RHEL 8 para configurar diferentes formas do nome do host no sistema ao qual o console web está anexado.

1.2.1. Nome do anfitrião

O nome do host identifica o sistema. Por padrão, o nome do host está definido para **localhost**, mas você pode alterá-lo.

Um nome de anfitrião consiste em duas partes:

Nome do anfitrião

É um nome único que identifica um sistema.

Domínio

Adicione o domínio como um sufixo por trás do nome do host ao usar um sistema em uma rede e ao usar nomes em vez de apenas endereços IP.

Um nome de host com um nome de domínio anexado é chamado de nome de domínio totalmente qualificado (FQDN). Por exemplo: **mymachine.example.com**.

Os nomes dos anfitriões são armazenados no arquivo **/etc/hostname**.

1.2.2. Bonito nome do host no console web

Você pode configurar um bonito nome de host no console web RHEL. O bonito nome de host é um nome de host com letras maiúsculas, espaços e assim por diante.

O bonito nome do host aparece no console web, mas não precisa corresponder com o nome do host.

Exemplo 1.2. Formatos do nome do host no console web

Bonito nome do anfitrião

My Machine

Nome do anfitrião

mymachine

Nome do verdadeiro host - nome de domínio totalmente qualificado (FQDN)

mymachine.idm.company.com

1.2.3. Definição do nome do host usando o console web

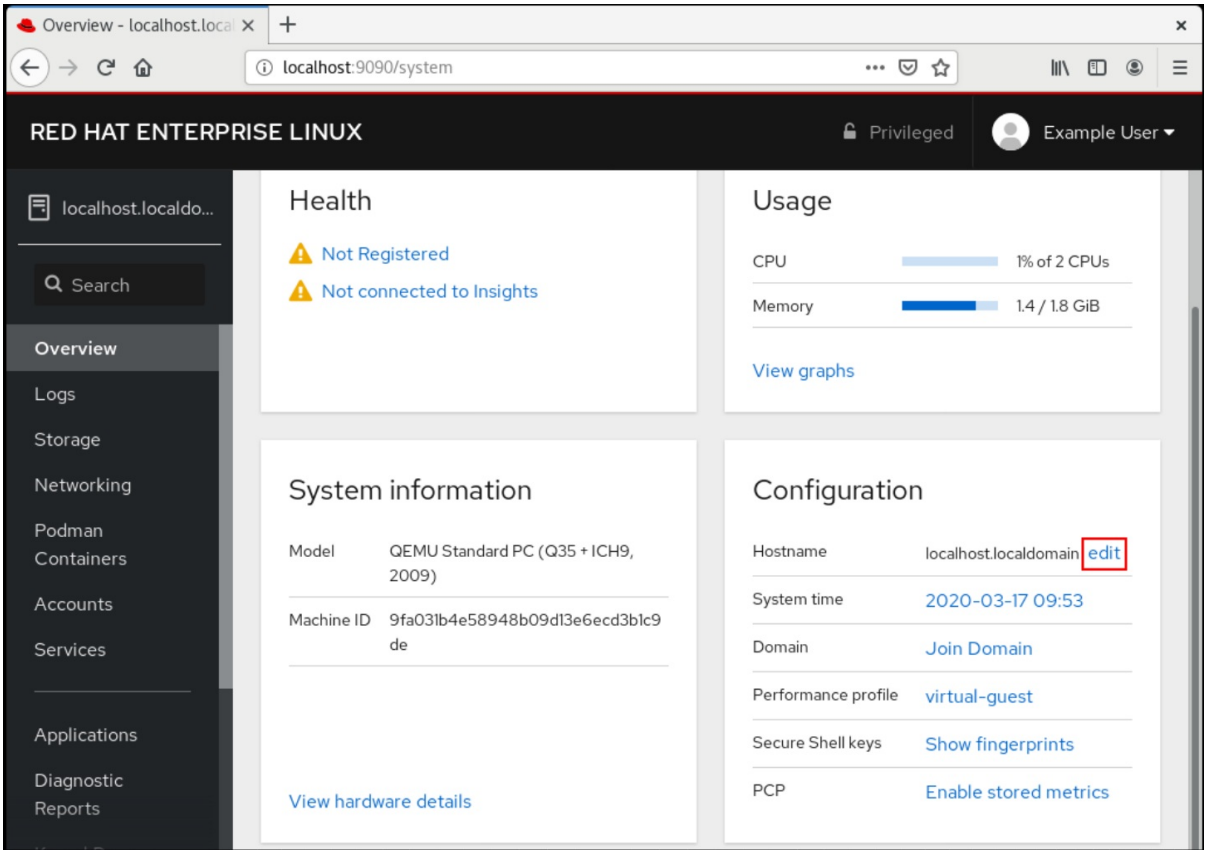
Este procedimento define o verdadeiro nome do host ou o bonito nome do host no console web.

Pré-requisitos

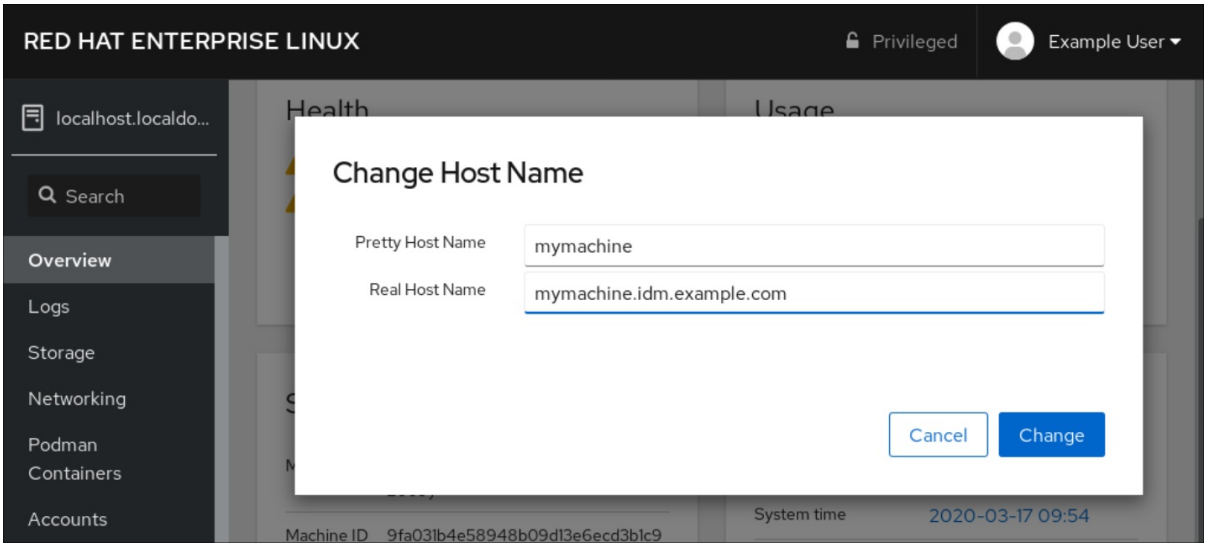
- O console web é instalado e acessível.
  Para detalhes, consulte Instalando o console web.

Procedimento

1. Acesse o console web RHEL 8.
   Para obter detalhes, consulte Login no console web.
2. Clique em **Visão Geral**.
3. Clique em **editar** ao lado do nome do host atual.

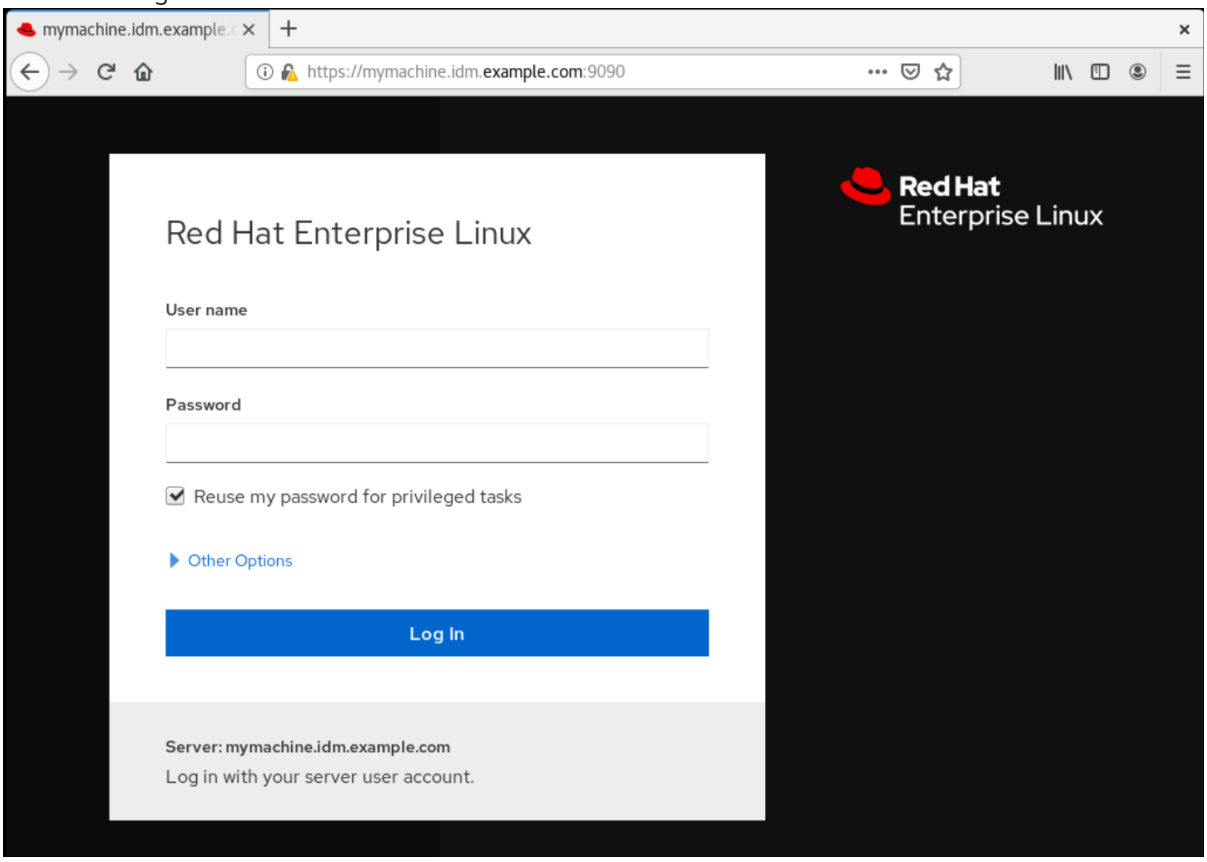


4. Na caixa de diálogo **Change Host Name**, digite o nome do anfitrião no campo **Pretty Host Name**.
5. O campo **Real Host Name** anexa um nome de domínio ao bonito nome.
   Você pode mudar o verdadeiro nome do anfitrião manualmente se ele não corresponder com o bonito nome do anfitrião.
6. Clique em **Change**.



Etapas de verificação

1. Sair do console web.
2. Reabra o console web inserindo um endereço com o novo nome do host na barra de endereços do seu navegador.



1.3. COMPLEMENTOS DO CONSOLE WEB RED HAT

Instale add-ons no console web RHEL 8 e saiba quais aplicações adicionais estão disponíveis para você.

1.3.1. Instalação de add-ons

O pacote **cockpit** é uma parte do Red Hat Enterprise Linux 8 por default. Para poder usar aplicativos adicionais, você deve instalá-los separadamente.

Pré-requisitos

- Instalado e habilitado o pacote **cockpit**. Se você precisar instalar primeiro o console web, verifique a seção de instalação.

Procedimento

- Instale um suplemento.

```
# yum install <add-on>
```

1.3.2. Complementos para o console web RHEL 8

A tabela a seguir lista as aplicações adicionais disponíveis para o console web RHEL 8.

Nome da característica	Nome do pacote	Utilização
Compositor	cockpit-compositor	Construindo imagens de SO personalizadas
Painel de controle	cockpit-dashboard	Gerenciamento de múltiplos servidores em uma IU
Máquinas	cockpit-machines	Gerenciando máquinas virtuais libvirt
PacoteKit	cockpit-packagekit	Atualizações de software e instalação de aplicativos (geralmente instalados por padrão)
PCP	cockpit-pcp	Dados de desempenho persistentes e de granulação mais fina (instalados sob demanda da IU)
podman	cockpit-podman	Gestão de containers de podman (disponível na RHEL 8.1)
Gravação da sessão	gravação da sessão de cockpit	Gravação e gerenciamento das sessões dos usuários

1.4. OTIMIZANDO O DESEMPENHO DO SISTEMA USANDO O CONSOLE WEB

Saiba como definir um perfil de desempenho no console web RHEL 8 para otimizar o desempenho do sistema para uma tarefa selecionada.

1.4.1. Opções de ajuste de desempenho no console web

O Red Hat Enterprise Linux 8 fornece vários perfis de desempenho que otimizam o sistema para as seguintes tarefas:

- Sistemas que utilizam a área de trabalho
- Desempenho de produção
- Desempenho na latência
- Desempenho da rede
- Baixo consumo de energia
- Máquinas virtuais

O serviço **tuned** otimiza as opções do sistema para corresponder ao perfil selecionado.

No console web, você pode definir o perfil de desempenho que seu sistema utiliza.

Recursos adicionais

- Para obter detalhes sobre o serviço **tuned**, consulte Monitoramento e gerenciamento do status e desempenho do sistema.

1.4.2. Definição de um perfil de desempenho no console web

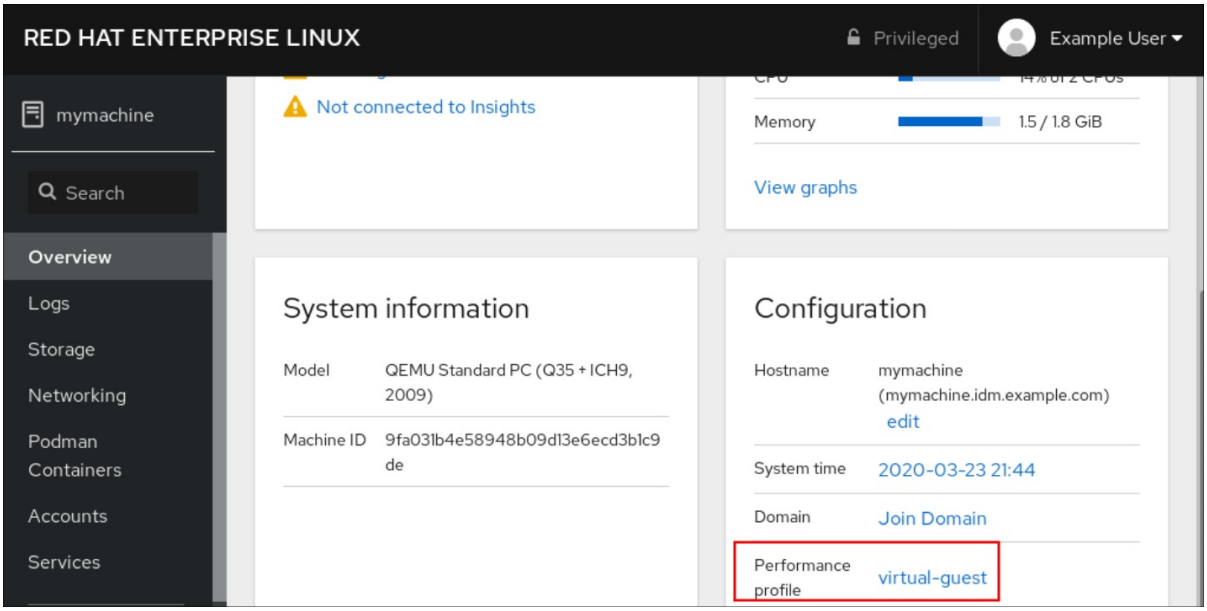
Este procedimento utiliza o console web para otimizar o desempenho do sistema para uma tarefa selecionada.

Pré-requisitos

- O console web é instalado e acessível.
  Para detalhes, consulte Instalando o console web.

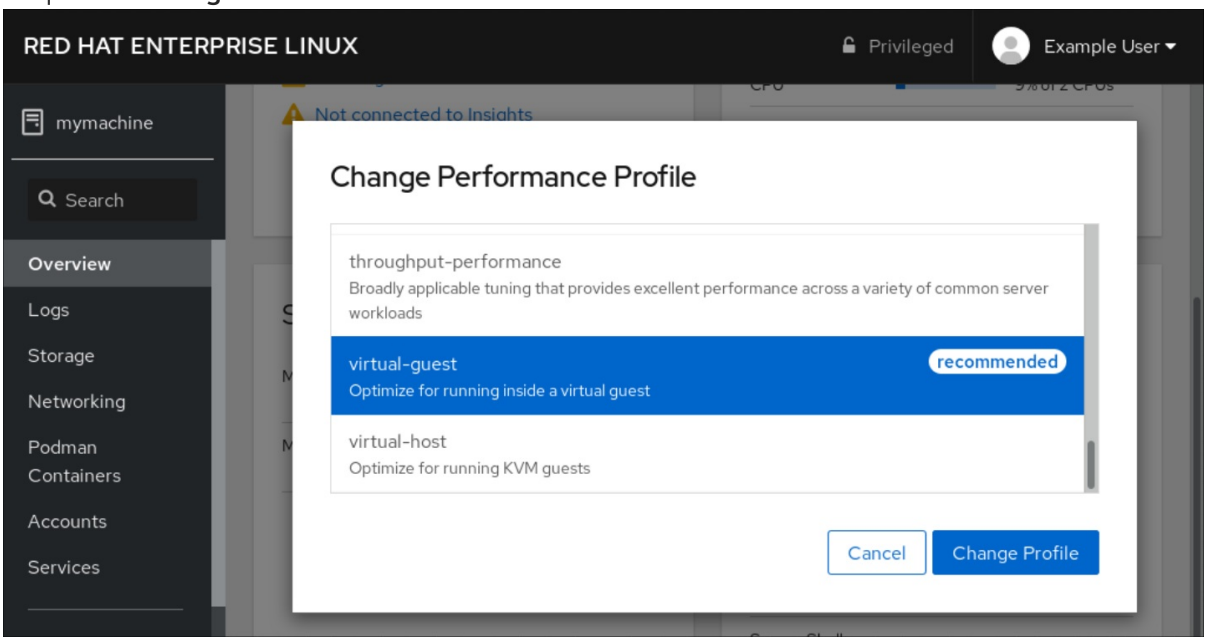
Procedimento

1. Acesse o console web RHEL 8.
   Para obter detalhes, consulte Login no console web.
2. Clique em **Overview**.
3. No campo **Performance Profile**, clique no perfil de desempenho atual.



4. Na caixa de diálogo **Change Performance Profile**, altere o perfil, se necessário.

5. Clique em **Change Profile**.



Etapas de verificação

- A aba **Overview** agora mostra o perfil de desempenho selecionado.

1.5. COMEÇANDO COM OS PAPÉIS DO SISTEMA RHEL

Esta seção explica quais são os papéis do Sistema RHEL. Além disso, descreve como aplicar um determinado papel através de um caderno de atividades para realizar várias tarefas de administração do sistema.

1.5.1. Introdução aos papéis do sistema RHEL

O Sistema RHEL Roles é um conjunto de funções e módulos possíveis. As funções do sistema RHEL fornecem uma interface de configuração para gerenciar remotamente vários sistemas RHEL. A interface permite gerenciar configurações de sistema através de múltiplas versões do RHEL, bem como

adotar novas versões principais.

No Red Hat Enterprise Linux 8, a interface atualmente consiste nas seguintes funções:

- kdump
- rede
- selinux
- armazenagem
- certificado
- kernel_settings
- madeireiro
- métricas
- nbde_client e nbde_server
- timesync
- tlog

Todas essas funções são fornecidas pelo pacote **rhel-system-roles** disponível no repositório **AppStream**.

Recursos adicionais

- Para uma visão geral dos Papéis do Sistema RHEL, veja o artigo do Red Hat Enterprise Linux (RHEL) System Roles Red Hat Knowledgebase.
- Para informações sobre uma função específica, consulte a documentação sob o diretório **/usr/share/doc/rhel-system-roles**. Esta documentação é instalada automaticamente com o pacote **rhel-system-roles**.
- Introdução ao papel do sistema SELinux
- Introdução à função de armazenamento

1.5.2. Terminologia dos papéis do Sistema RHEL

Você pode encontrar os seguintes termos ao longo desta documentação:

Terminologia dos papéis do sistema

Livro de jogo possível

Os playbooks são a linguagem de configuração, implantação e orquestração do Ansible. Eles podem descrever uma política que você quer que seus sistemas remotos apliquem, ou um conjunto de passos em um processo geral de TI.

Nó de controle

Qualquer máquina com Ansible instalado. Você pode executar comandos e playbooks, invocando /usr/bin/ansible ou /usr/bin/ansible-playbook, a partir de qualquer nó de controle. Você pode usar

qualquer computador que tenha o Python instalado nele como um nó de controle - laptops, desktops compartilhados e servidores podem todos rodar o Ansible. Entretanto, você não pode usar uma máquina Windows como nó de controle. Você pode ter múltiplos nós de controle.

Inventário

Uma lista de nós administrados. Um arquivo de inventário também é às vezes chamado de "arquivo hospedeiro". Seu inventário pode especificar informações como endereço IP para cada nó gerenciado. Um inventário também pode organizar nós administrados, criando e aninhando grupos para facilitar o escalonamento. Para saber mais sobre o inventário, consulte a seção Trabalhando com o inventário.

Nós administrados

Os dispositivos de rede, servidores, ou ambos que você administra com Ansible. Os nós gerenciados também são às vezes chamados de "hosts". O Ansible não é instalado em nós gerenciados.

1.5.3. Aplicando um papel

O procedimento a seguir descreve como aplicar uma função específica.

Pré-requisitos

- O pacote **rhel-system-roles** está instalado no sistema que você deseja usar como um nó de controle:

```
# yum install rhel-system-roles
```

- O repositório Ansible Engine está habilitado, e o pacote **ansible** está instalado no sistema que você deseja usar como um nó de controle. Você precisa do pacote **ansible** para executar playbooks que usam os papéis do sistema RHEL.
  - Se você não tiver uma assinatura do Red Hat Ansible Engine, você pode usar uma versão suportada limitada do Red Hat Ansible Engine fornecida com sua assinatura do Red Hat Enterprise Linux. Neste caso, siga estes passos:
    1. Habilitar o repositório RHEL Ansible Engine:

```
# subscription-manager refresh
# subscription-manager repos --enable ansible-2-for-rhel-8-x86_64-rpms
```

    2. Instalar Motor Possível:

```
# yum install ansible
```

  - Se você tem uma assinatura de Red Hat Ansible Engine, siga o procedimento descrito em Como faço para baixar e instalar o Red Hat Ansible Engine?

- Você é capaz de criar um livro de brincadeiras possível.
  Os playbooks representam a linguagem de configuração, implantação e orquestração do Ansible. Usando playbooks, você pode declarar e gerenciar configurações de máquinas remotas, implantar várias máquinas remotas ou etapas de orquestração de qualquer processo encomendado manualmente.

  Um playbook é uma lista de um ou mais **plays**. Cada **play** pode incluir variáveis, tarefas ou papéis possíveis.

Os livros didáticos são legíveis por humanos e são expressos no formato **YAML**.

Para mais informações sobre livros didáticos, consulte Documentação possível.

Procedimento

1. Crie um caderno de atividades, incluindo o papel necessário.
   O exemplo a seguir mostra como usar os papéis através da opção **roles:** para um determinado **play**:

```
---
- hosts: webservers
  roles:
    - rhel-system-roles.network
    - rhel-system-roles.timesync
```

   Para mais informações sobre o uso de papéis em livros didáticos, consulte Documentação possível.

   Veja Exemplos possíveis, por exemplo, livros didáticos.

NOTA

Cada função inclui um arquivo README, que documenta como utilizar a função e os valores dos parâmetros suportados. Você também pode encontrar um exemplo de playbook para um determinado papel sob o diretório de documentação do papel. Tal diretório de documentação é fornecido por padrão com o pacote **rhel-system-roles**, e pode ser encontrado no local a seguir:

```
/usr/share/doc/rhel-system-roles/SUBSYSTEM/
```

Substituir *SUBSYSTEM* pelo nome da função requerida, como **selinux**, **kdump**, **network**, **timesync**, ou **storage**.

2. Verificar a sintaxe do playbook:

```
# ansible-playbook --syntax-check name.of.the.playbook
```

   O comando **ansible-playbook** oferece uma opção **--syntax-check** que você pode usar para verificar a sintaxe de um playbook.

3. Executar o playbook nos anfitriões-alvo executando o comando **ansible-playbook**:

```
# ansible-playbook -i name.of.the.inventory name.of.the.playbook
```

   Um inventário é uma lista de sistemas contra os quais o Ansible funciona. Para mais informações sobre como criar e inventariar, e como trabalhar com ele, consulte a documentação do Ansible .

   Se você não tiver um inventário, você pode criá-lo no momento da execução **ansible-playbook**:

   Se você tiver apenas um anfitrião específico contra o qual você deseja executar o playbook, use:

```
# ansible-playbook -i host1, name.of.the.playbook
```

Se você tiver vários anfitriões alvo contra os quais você deseja executar o livro de jogo, use:

```
# ansible-playbook -i host1,host2,....,hostn name.of.the.playbook
```

Recursos adicionais

- Para obter informações mais detalhadas sobre o uso do comando **ansible-playbook**, consulte a página de manual **ansible-playbook**.

1.5.4. Recursos adicionais

- Para uma visão geral dos Papéis do Sistema RHEL, veja o artigo do Red Hat Enterprise Linux (RHEL) System Roles Red Hat Knowledgebase.
- Gerenciamento do armazenamento local usando as funções do sistema RHEL
- Implantando a mesma configuração SELinux em vários sistemas usando as funções do sistema RHEL

1.6. MUDANÇA DAS CONFIGURAÇÕES BÁSICAS DO AMBIENTE

A configuração das configurações básicas do ambiente é uma parte do processo de instalação. As seções seguintes orientam quando você as altera posteriormente. A configuração básica do ambiente inclui:

- Data e hora
- Localização do sistema
- Layout do teclado
- Idioma

1.6.1. Configurando a data e a hora

A precisão do tempo é importante por uma série de razões. No Red Hat Enterprise Linux, o timekeeping é assegurado pelo protocolo **NTP**, que é implementado por um daemon rodando no espaço do usuário. O daemon de espaço do usuário atualiza o relógio do sistema rodando no kernel. O relógio do sistema pode manter o tempo usando várias fontes de relógio.

O Red Hat Enterprise Linux 8 usa o daemon **chronyd** para implementar **NTP**. **chronyd** está disponível no site **chrony** pacote. Para mais informações, consulte Utilizando a suíte chrony para configurar o NTP .

1.6.1.1. Exibição da data e hora atual

Para exibir a data e a hora atual, use uma destas etapas.

Procedimento

1. Digite o comando **date**:

```
$ date
Mon Mar 30 16:02:59 CEST 2020
```

2. Para ver mais detalhes, use o comando **timedatectl**:

```
$ timedatectl
Local time: Mon 2020-03-30 16:04:42 CEST
Universal time: Mon 2020-03-30 14:04:42 UTC
  RTC time: Mon 2020-03-30 14:04:41
 Time zone: Europe/Prague (CEST, +0200)
System clock synchronized: yes
NTP service: active
RTC in local TZ: no
```

Recursos adicionais

- Para mais informações, consulte as páginas de manual **date(1)** e **timedatectl(1)**.

1.6.1.2. Recursos adicionais

- Para mais informações sobre as configurações de tempo no console web, consulte Utilizando o console web para configurar as configurações de tempo.

1.6.2. Configuração do sistema locale

As configurações locais de todo o sistema são armazenadas no arquivo /**etc**/**locale.conf**, que é lido no início da inicialização pelo daemon **systemd**. Cada serviço ou usuário herda as configurações de locale configuradas em /**etc**/**locale.conf**, a menos que programas individuais ou usuários individuais os substituam.

Esta seção descreve como gerenciar o sistema locale.

Procedimento

1. Para listar as configurações disponíveis do sistema locale:

```
$ localectl list-locales
C.utf8
aa_DJ
aa_DJ.iso88591
aa_DJ.utf8
...
```

2. Para exibir o status atual das configurações do sistema locales:

```
$ localectl status
```

3. Para definir ou alterar as configurações padrão do sistema locale, use um sub-comando **localectl set-locale** como o usuário **root**. Por exemplo:

```
# localectl set-locale LANG=en-US
```

Recursos adicionais

- Para mais informações, consulte as páginas de manual **localectl(1)**, **locale(7)** e **locale.conf(5)**.

1.6.3. Configurando o layout do teclado

As configurações de layout do teclado controlam o layout usado no console de texto e nas interfaces gráficas de usuário.

Procedimento

1. Para listar os keymaps disponíveis:

```
$ localectl list-keymaps
ANSI-dvorak
al
al-plisi
amiga-de
amiga-us
...
```

2. Para exibir o status atual das configurações de keymaps:

```
$ localectl status
...
VC Keymap: us
...
```

3. Para definir ou alterar o keymap padrão do sistema, use um sub-comando **localectl set-keymap** como o usuário **root**. Por exemplo:

```
# localectl set-keymap us
```

Recursos adicionais

- Para mais informações, consulte as páginas de manual **localectl(1)**, **locale(7)** e **locale.conf(5)**.

1.6.4. Mudando o idioma usando a GUI de mesa

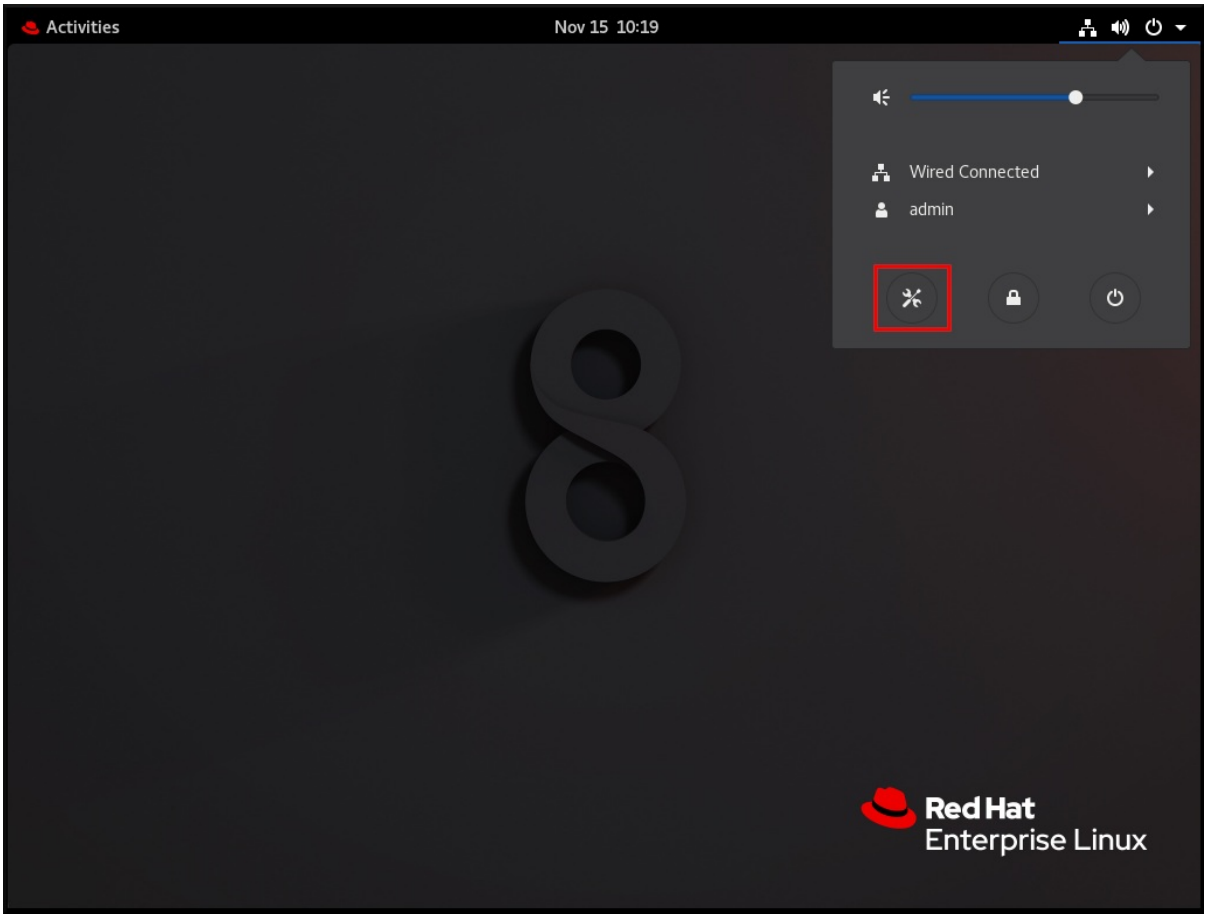
Esta seção descreve como mudar a linguagem do sistema usando a GUI da área de trabalho.

Pré-requisitos

- Os pacotes de idiomas necessários estão instalados em seu sistema

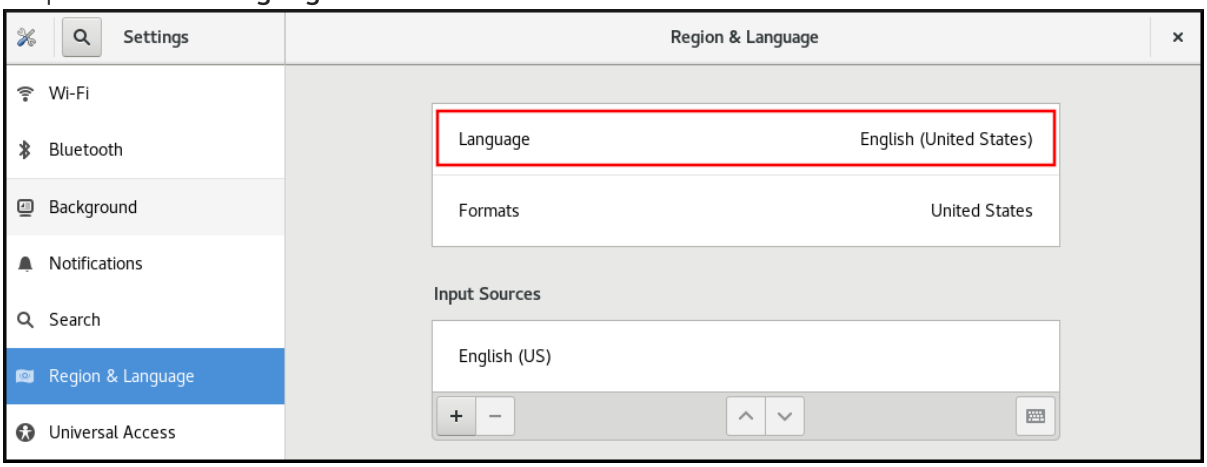
Procedimento

1. Abra o **GNOME Control Center** do site **System menu** clicando em seu ícone.



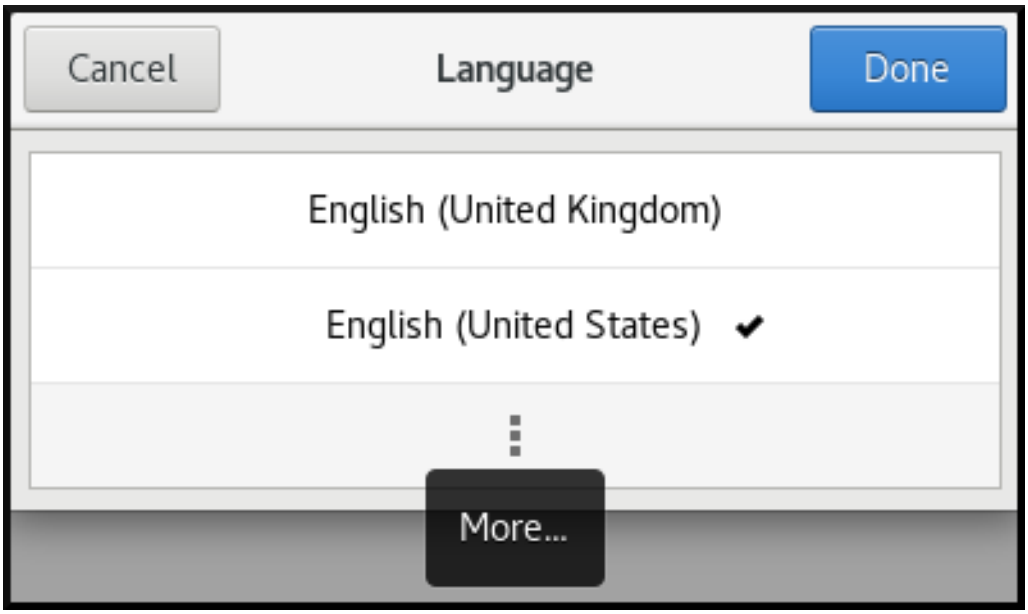
2. No site **GNOME Control Center**, escolha **Region & Language** na barra vertical esquerda.

3. Clique no menu **Language**.



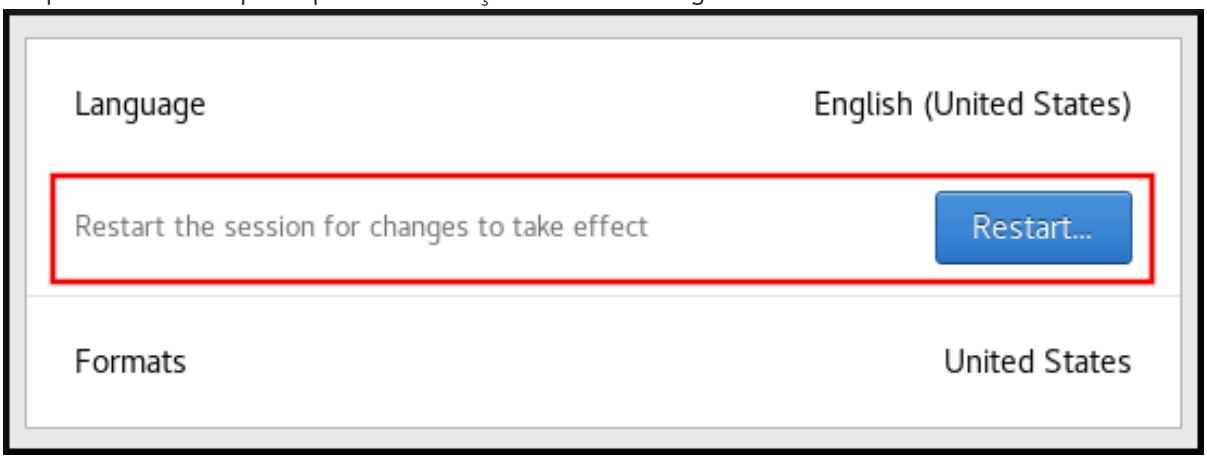
4. Selecione a região e o idioma desejado no menu.

Se sua região e idioma não estiverem listados, role para baixo e clique em **More** para selecionar entre as regiões e idiomas disponíveis.



5. Clique em **Done**.

6. Clique em **Restart** para que as mudanças entrem em vigor.



NOTA

Algumas aplicações não suportam certos idiomas. O texto de uma aplicação que não pode ser traduzido para o idioma selecionado permanece em inglês americano.

Recursos adicionais

- Para mais informações sobre como lançar o **GNOME Control Center**, veja as abordagens descritas em Aplicações de lançamento

1.6.5. Recursos adicionais

- Para mais informações sobre a configuração das configurações básicas do ambiente, consulte Execução de uma instalação padrão RHEL .

1.7. CONFIGURAÇÃO E GERENCIAMENTO DO ACESSO À REDE

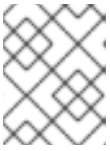
Esta seção descreve diferentes opções sobre como adicionar conexões Ethernet no Red Hat Enterprise Linux.

1.7.1. Configurando a rede e o nome do host no modo de instalação gráfica

Siga as etapas deste procedimento para configurar sua rede e nome do host.

Procedimento

1. Na janela **Installation Summary**, clique em **Network and Host Name***.
2. Da lista no painel da esquerda, selecione uma interface. Os detalhes são exibidos no painel da direita.
3. Alternar a chave **ON**/**OFF** para ativar ou desativar a interface selecionada.



NOTA

O programa de instalação detecta automaticamente as interfaces acessíveis localmente, e você não pode adicioná-las ou removê-las manualmente.

4. Clique em para adicionar uma interface de rede virtual, que pode ser qualquer uma das duas: Equipe, Bond, Bridge, ou VLAN.
5. Clique **-** para remover uma interface virtual.
6. Clique em **Configurar** para alterar configurações tais como endereços IP, servidores DNS ou configuração de roteamento para uma interface existente (tanto virtual quanto física).
7. Digite um nome de host para seu sistema no campo **Host Name**.

NOTA

- Há vários tipos de padrões de nomenclatura de dispositivos de rede utilizados para identificar dispositivos de rede com nomes persistentes, por exemplo, **em1** e **wl3sp0**. Para obter informações sobre essas normas, consulte o *Configuring and managing networking* documento.

- O nome do host pode ser um nome de domínio totalmente qualificado (FQDN) no formato *hostname.domainname*, ou um nome de host curto sem nome de domínio. Muitas redes possuem um serviço DHCP (Dynamic Host Configuration Protocol) que fornece automaticamente sistemas conectados com um nome de domínio. Para permitir que o serviço DHCP atribua o nome de domínio a esta máquina, especifique apenas o nome de host abreviado. O valor **localhost.localdomain** significa que nenhum nome de host estático específico para o sistema alvo é configurado, e o nome de host real do sistema instalado é configurado durante o processamento da configuração da rede, por exemplo, por **NetworkManager** usando DHCP ou DNS.

8. Clique em **Aplicar** para aplicar o nome do anfitrião ao ambiente.

Recursos e informações adicionais

- Para obter detalhes sobre a configuração das configurações de rede e do nome do host ao usar um arquivo Kickstart, consulte o apêndice correspondente em Execução de uma instalação RHEL avançada.

- Se você instalar o Red Hat Enterprise Linux usando o modo texto do programa de instalação **Anaconda**, use a opção **Network settings** para configurar a rede.

1.7.2. Configuração de uma conexão Ethernet estática usando nmcli

Este procedimento descreve a adição de uma conexão Ethernet com as seguintes configurações usando o utilitário **nmcli**:

- Um endereço IPv4 estático - **192.0.2.1** com uma máscara de sub-rede **/24**
- Um endereço IPv6 estático - **2001:db8:1::1** com uma máscara de sub-rede **/64**
- Um gateway padrão IPv4 - **192.0.2.254**
- Um gateway padrão IPv6 - **2001:db8:1::fffe**
- Um servidor DNS IPv4 - **192.0.2.200**
- Um servidor DNS IPv6 - **2001:db8:1::ffbb**
- Um domínio de busca DNS - **example.com**

Procedimento

1. Adicione um novo perfil de conexão NetworkManager para a conexão Ethernet:

```
# nmcli connection add con-name Example-Connection ifname enp7s0 type ethernet
```

Os próximos passos modificam o perfil de conexão **Example-Connection** que você criou.

2. Defina o endereço IPv4:

```
# nmcli connection modify Example-Connection ipv4.addresses 192.0.2.1/24
```

3. Defina o endereço IPv6:

```
# nmcli connection modify Example-Connection ipv6.addresses 2001:db8:1::1/64
```

4. Configure o método de conexão IPv4 e IPv6 para **manual**:

```
# nmcli connection modify Example-Connection ipv4.method manual
# nmcli connection modify Example-Connection ipv6.method manual
```

5. Defina os gateways padrão IPv4 e IPv6:

```
# nmcli connection modify Example-Connection ipv4.gateway 192.0.2.254
# nmcli connection modify Example-Connection ipv6.gateway 2001:db8:1::fffe
```

6. Configure os endereços dos servidores DNS IPv4 e IPv6:

```
# nmcli connection modify Example-Connection ipv4.dns "192.0.2.200"
# nmcli connection modify Example-Connection ipv6.dns "2001:db8:1::ffbb"
```

Para definir vários servidores DNS, especifique-os separados por espaço e entre aspas.

7. Definir o domínio de busca DNS para a conexão IPv4 e IPv6:

```
# nmcli connection modify Example-Connection ipv4.dns-search example.com
# nmcli connection modify Example-Connection ipv6.dns-search example.com
```

8. Ativar o perfil de conexão:

```
# nmcli connection up Example-Connection
Connection successfully activated (D-Bus active path:
/org/freedesktop/NetworkManager/ActiveConnection/13)
```

Etapas de verificação

1. Mostrar o status dos dispositivos e conexões:

```
# nmcli device status
DEVICE     TYPE      STATE      CONNECTION
enp7s0     ethernet  connected  Example-Connection
```

2. Para exibir todas as configurações do perfil de conexão:

```
# nmcli connection show Example-Connection
connection.id:                Example-Connection
connection.uuid:              b6cdfa1c-e4ad-46e5-af8b-a75f06b79f76
connection.stable-id:         --
```

```
connection.type:          802-3-ethernet
connection.interface-name:  enp7s0
...
```

3. Use o utilitário **ping** para verificar se este host pode enviar pacotes para outros hosts.

   - Pingar um endereço IP na mesma sub-rede.
     Para IPv4:

     ```
     # ping 192.0.2.3
     ```

     Para IPv6:

     ```
     # ping 2001:db8:2::1
     ```

     Se o comando falhar, verificar as configurações de IP e subrede.

   - Pingar um endereço IP em uma sub-rede remota.
     Para IPv4:

     ```
     # ping 198.162.3.1
     ```

     Para IPv6:

     ```
     # ping 2001:db8:2::1
     ```

     - Se o comando falhar, pingar o gateway padrão para verificar as configurações.
       Para IPv4:

       ```
       # ping 192.0.2.254
       ```

       Para IPv6:

       ```
       # ping 2001:db8:1::fffe
       ```

4. Use o utilitário **host** para verificar se a resolução do nome funciona. Por exemplo:

   ```
   # host client.example.com
   ```

   Se o comando retornar algum erro, como **connection timed out** ou **no servers could be reached**, verifique suas configurações de DNS.

Passos para a solução de problemas

1. Se a conexão falhar ou se a interface de rede comutar entre um estado para cima e para baixo:

   - Certifique-se de que o cabo de rede esteja conectado ao host e a um switch.
   - Verifique se a falha do link só existe neste host ou também em outros hosts conectados ao mesmo switch ao qual o servidor está conectado.

- Verificar se o cabo de rede e a interface de rede estão funcionando como esperado. Executar as etapas de diagnóstico do hardware e substituir os cabos de defeito e as placas de interface de rede.

Recursos adicionais

- Consulte a página de manual **nm-settings(5)** para mais informações sobre as propriedades do perfil de conexão e suas configurações.
- Para mais detalhes sobre a utilidade **nmcli**, consulte a página de manual **nmcli(1)**.
- Se a configuração no disco não corresponder à configuração no dispositivo, iniciar ou reiniciar o NetworkManager cria uma conexão in-memory que reflete a configuração do dispositivo. Para maiores detalhes e como evitar este problema, veja NetworkManager duplica uma conexão após o reinício do serviço NetworkManager.
- Se a conexão não tiver um gateway padrão, consulte Configurando o NetworkManager para evitar o uso de um perfil específico para fornecer um gateway padrão na documentação **Configuring and managing networking**.

1.7.3. Adicionando um perfil de conexão usando nmtui

O aplicativo **nmtui** fornece uma interface de usuário de texto para o NetworkManager. Este procedimento descreve como adicionar um novo perfil de conexão.

Pré-requisitos

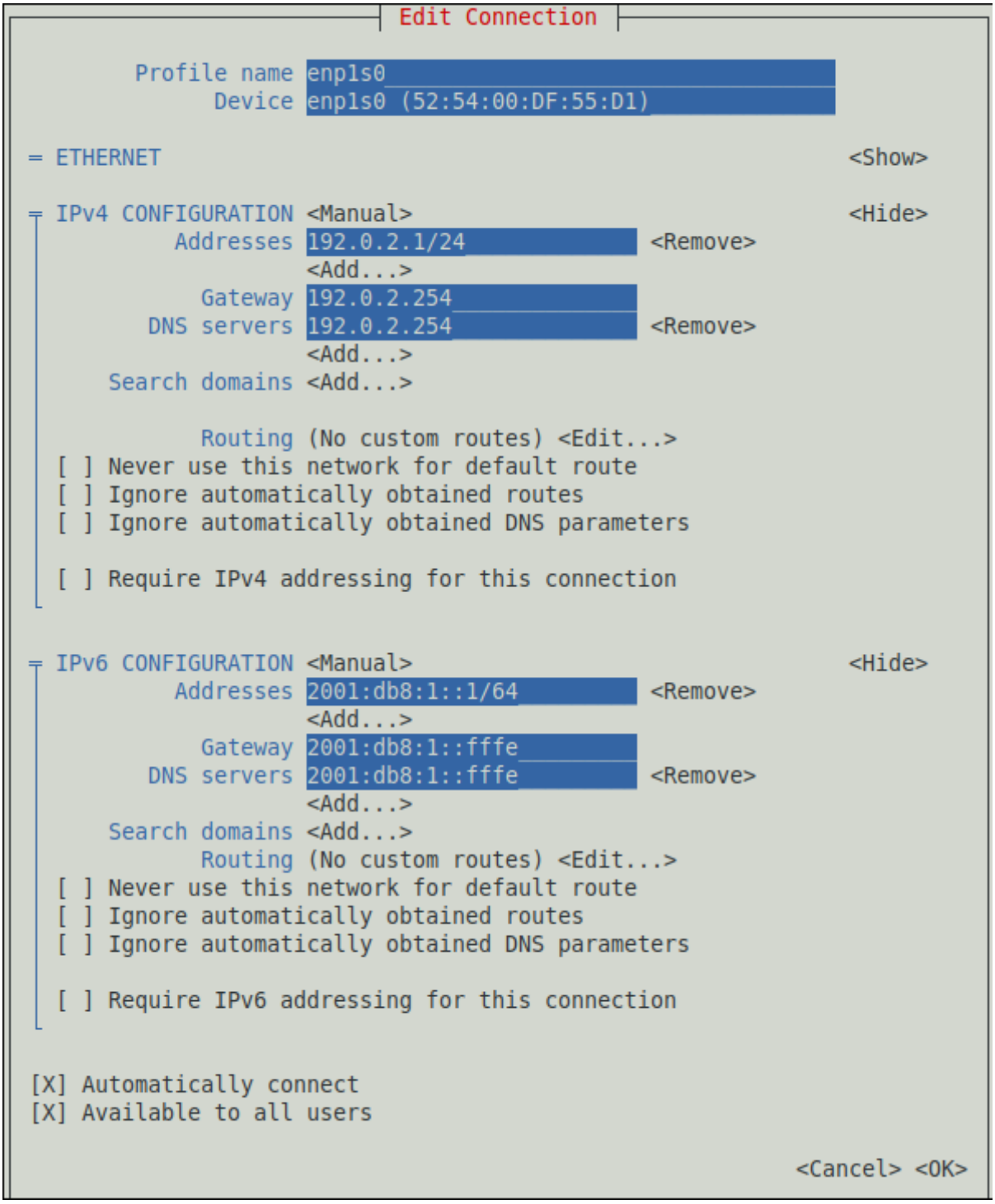
- O pacote **NetworkManager-tui** está instalado.

Procedimento

1. Inicie o utilitário de interface de usuário de texto NetworkManager:

```
# nmtui
```

2. Selecione a entrada do menu **Edit a connection**, e pressione **Enter**.
3. Selecione o botão **Adicionar**, e pressione **Enter**.
4. Selecione **Ethernet**, e pressione **Enter**.
5. Preencha os campos com os detalhes da conexão.



6. Selecione **OK** para salvar as mudanças.
7. Selecione **Back** para retornar ao menu principal.
8. Selecione **Activate a connection**, e pressione **Enter**.
9. Selecione a nova entrada de conexão, e pressione **Enter** para ativar a conexão.
10. Selecione **Voltar** para retornar ao menu principal.
11. Selecione **Quit**.

Etapas de verificação

1. Mostrar o status dos dispositivos e conexões:

```
# nmcli device status
DEVICE      TYPE      STATE      CONNECTION
enp1s0      ethernet  connected  Example-Connection
```

2. Para exibir todas as configurações do perfil de conexão:

```
# nmcli connection show Example-Connection
connection.id:                  Example-Connection
connection.uuid:                b6cdfa1c-e4ad-46e5-af8b-a75f06b79f76
connection.stable-id:           --
connection.type:                802-3-ethernet
connection.interface-name:      enp1s0
...
```

Recursos adicionais

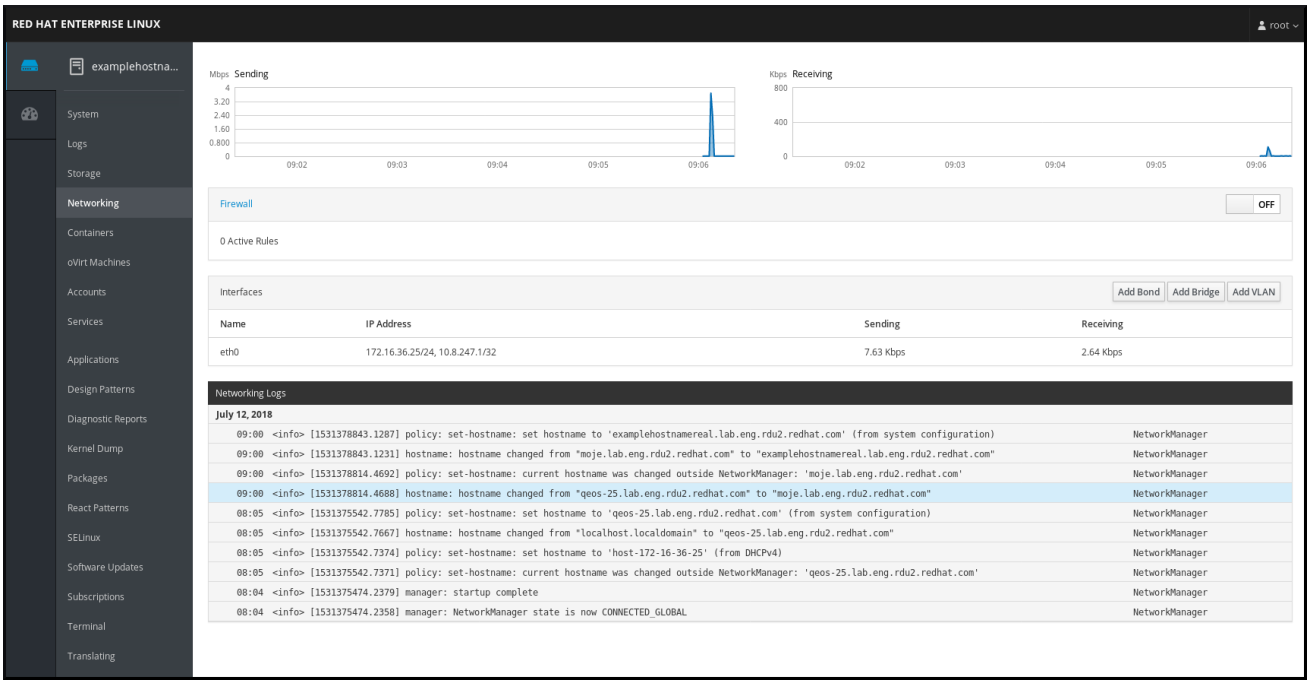
- Para obter mais informações sobre como testar conexões, consulte Testar configurações básicas de rede em **Configuring and managing networking**.
- Para mais detalhes sobre a aplicação **nmtui**, consulte a página de manual **nmtui(1)**.
- Se a configuração no disco não corresponder à configuração no dispositivo, iniciar ou reiniciar o NetworkManager cria uma conexão in-memory que reflete a configuração do dispositivo. Para maiores detalhes e como evitar este problema, veja NetworkManager duplica uma conexão após o reinício do serviço NetworkManager.

1.7.4. Gerenciamento de rede no console web RHEL 8

No console web, o menu **Networking** permite a você:

- Para exibir os pacotes recebidos e enviados atualmente
- Para exibir as características mais importantes das interfaces de rede disponíveis
- Para exibir o conteúdo dos logs da rede.
- Para adicionar vários tipos de interfaces de rede (bond, equipe, bridge, VLAN)

Figura 1.1. Gerenciamento da rede no console web RHEL 8



1.7.5. Gerenciando redes usando as funções do sistema RHEL

Você pode configurar as conexões de rede em várias máquinas alvo usando a função **network**.

A função **network** permite a configuração dos seguintes tipos de interfaces:

- Ethernet
- Ponte
- Ligado
- VLAN
- MacVLAN
- Infiniband

As conexões de rede necessárias para cada host são fornecidas como uma lista dentro da variável **network_connections**.

ATENÇÃO

A função **network** atualiza ou cria todos os perfis de conexão no sistema de destino exatamente como especificado na variável **network_connections**. Portanto, a função **network** remove opções dos perfis especificados se as opções estiverem presentes apenas no sistema, mas não na variável **network_connections**.

O exemplo a seguir mostra como aplicar a função **network** para garantir que exista uma conexão Ethernet com os parâmetros necessários:

Exemplo 1.3. Um exemplo de playbook aplicando a função de rede para estabelecer uma conexão Ethernet com os parâmetros necessários

```
# SPDX-License-Identifier: BSD-3-Clause
---
- hosts: network-test
  vars:
    network_connections:

      # Create one ethernet profile and activate it.
      # The profile uses automatic IP addressing
      # and is tied to the interface by MAC address.
      - name: prod1
        state: up
        type: ethernet
        autoconnect: yes
        mac: "00:00:5e:00:53:00"
        mtu: 1450

  roles:
    - rhel-system-roles.network
```

Para mais informações sobre a aplicação de uma função do sistema, consulte Introdução às funções do Sistema RHEL.

1.7.6. Recursos adicionais

- Para mais detalhes sobre a configuração da rede, tais como a configuração da ligação à rede e a formação de equipes, consulte o título Configurando e gerenciando a rede .

1.8. REGISTRO DO SISTEMA E GESTÃO DAS ASSINATURAS

As assinaturas cobrem os produtos instalados no Red Hat Enterprise Linux, incluindo o próprio sistema operacional.

Você pode usar uma assinatura da Red Hat Content Delivery Network para rastrear:

- Sistemas registrados
- Produtos instalados em seus sistemas
- Assinaturas anexadas aos produtos instalados

1.8.1. Registro do sistema após a instalação

Use o seguinte procedimento para registrar seu sistema caso ainda não o tenha registrado durante o processo de instalação.

Pré-requisitos

- Uma conta de usuário válida no Portal do Cliente da Red Hat.
- Veja a página Criar um Login para o Red Hat .

- Uma assinatura ativa para o sistema RHEL.
- Para mais informações sobre o processo de instalação, consulte Execução de uma instalação padrão RHEL.

Procedimento

1. Cadastre-se e assine automaticamente seu sistema em uma única etapa:

```
# subscription-manager register --username <username> --password <password> --auto-attach
Registering to: subscription.rhsm.redhat.com:443/subscription
The system has been registered with ID: 37to907c-ece6-49ea-9174-20b87ajk9ee7
The registered system name is: client1.idm.example.com
Installed Product Current Status:
Product Name: Red Hat Enterprise Linux for x86_64
Status:       Subscribed
```

   O comando solicita que você digite seu nome de usuário e senha do Portal do Cliente Red Hat.

   Se o processo de registro falhar, você pode registrar seu sistema com um pool específico. Para orientação sobre como fazer isso, proceda com as seguintes etapas:

   a. Determine o ID do pool de uma assinatura que você precisa:

```
# subscription-manager list --available
```

      Este comando exibe todas as assinaturas disponíveis para sua conta Red Hat. Para cada assinatura, várias características são exibidas, incluindo o ID do pool.

   b. Anexe a assinatura apropriada ao seu sistema, substituindo *pool_id* pelo ID do pool determinado na etapa anterior:

```
# subscription-manager attach --pool=pool_id
```

Recursos adicionais

- Para mais detalhes sobre o registro de sistemas RHEL usando a opção **--auto-attach**, veja Entendendo assinaturas de autoatendimento na seção Portal do Cliente.
- Para mais detalhes sobre o registro manual dos sistemas RHEL, consulte Entendendo o registro manual e a assinatura na seção Portal do Cliente.

1.8.2. Registro de assinaturas com credenciais no console web

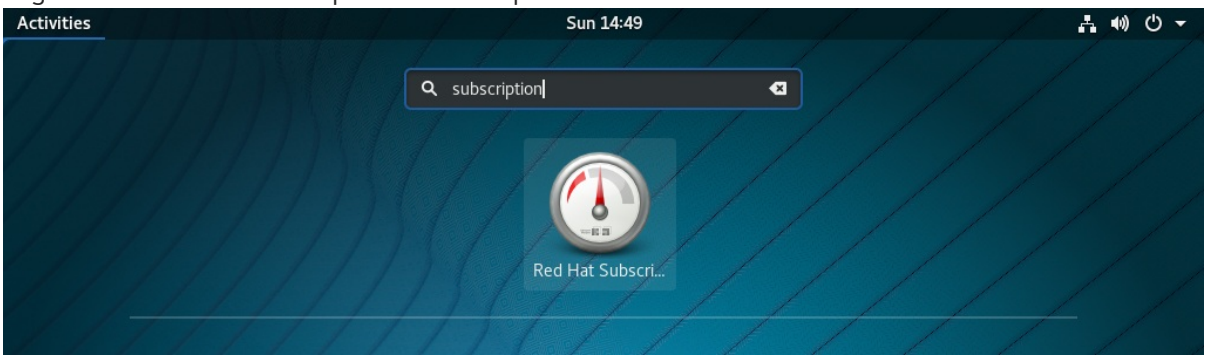
Use os seguintes passos para registrar um Red Hat Enterprise Linux recém-instalado usando o console web RHEL 8.

Pré-requisitos

- Uma conta de usuário válida no Portal do Cliente da Red Hat.
  Veja a página Criar um Login para o Red Hat .
- Assinatura ativa para seu sistema RHEL.

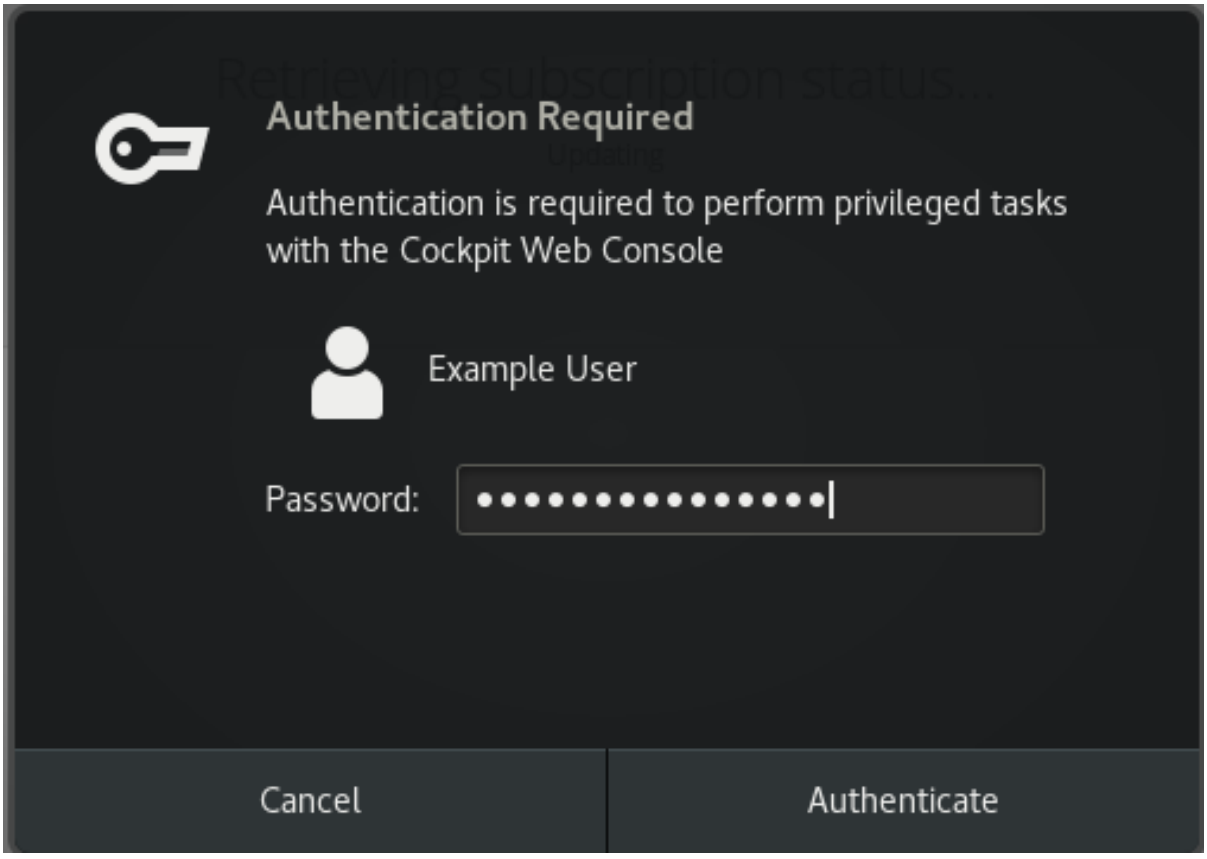
Procedimento

1. Digite a assinatura no campo de busca e pressione a tecla **Enter**.



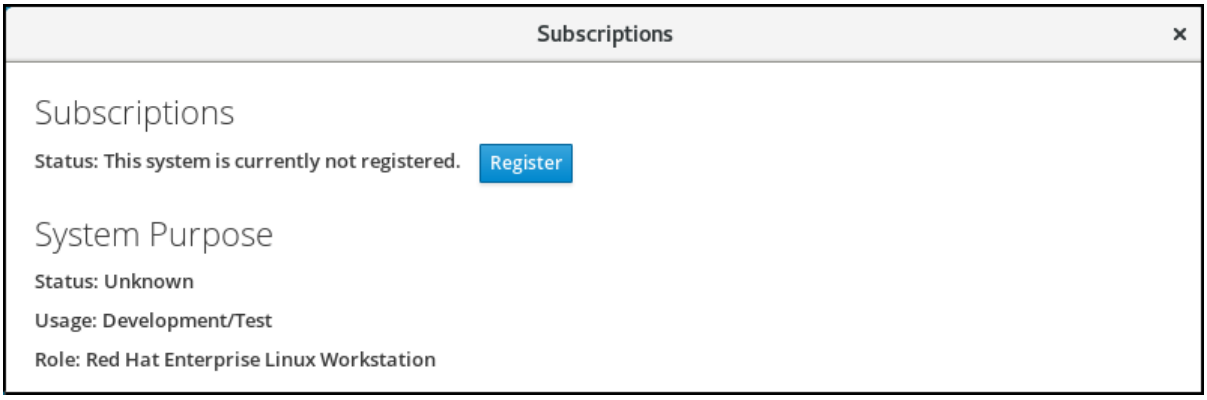
   Alternativamente, você pode entrar no console web RHEL 8. Para obter detalhes, consulte Login no console web.

2. No diálogo de autenticação **polkit** para tarefas privilegiadas, adicione a senha pertencente ao nome do usuário exibido no diálogo.

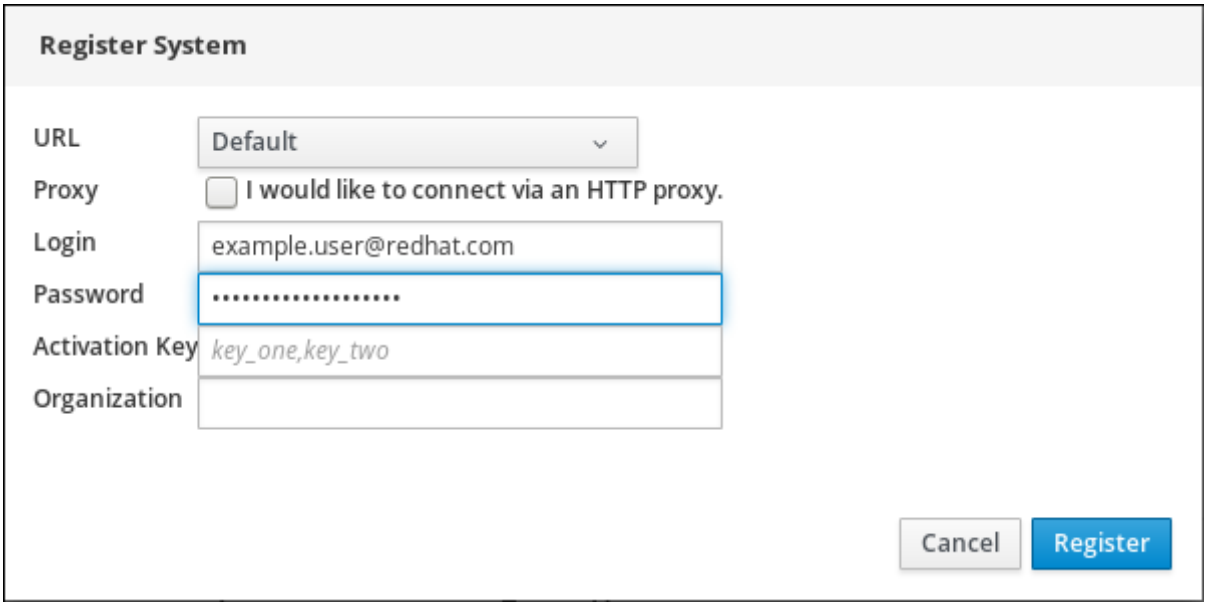


3. Clique em **Autenticar**.

4. Na caixa de diálogo **Subscriptions**, clique em **Register**.



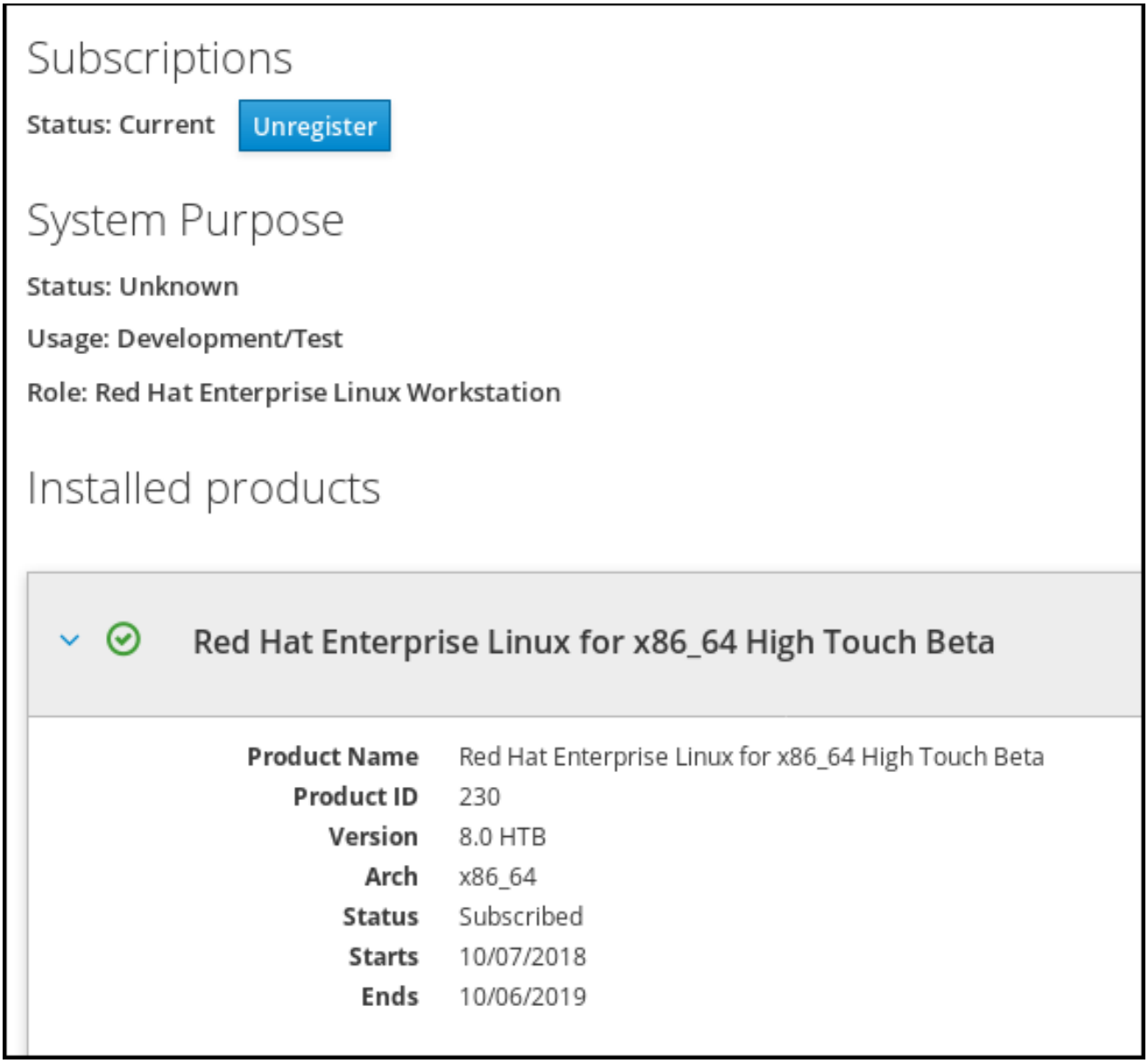
5. Digite suas credenciais no Portal do Cliente.



6. Digite o nome de sua organização.
   Se você tem mais de uma conta no Portal do Cliente da Red Hat, você tem que adicionar o nome ou ID da organização. Para obter o ID da organização, vá até seu ponto de contato da Red Hat.

7. Clique no botão **Registrar**.

Neste momento, seu sistema Red Hat Enterprise Linux 8 foi registrado com sucesso.



1.8.3. Registrando um sistema usando a conta da Red Hat no GNOME

Siga as etapas deste procedimento para cadastrar seu sistema com sua conta Red Hat.

Pré-requisitos

- Uma conta válida no portal do cliente da Red Hat.
  Veja a página Criar um Login Red Hat para registro de novo usuário.

Procedimento

1. Vá para **system menu**, que pode ser acessado no canto superior direito da tela e clique no ícone **Settings**.
2. No **Detalhes → Sobre** clique em **Register**.
3. Selecione **Registration Server**.
4. Se você não estiver usando o servidor da Red Hat, digite o endereço do servidor no campo **URL**.
5. No menu **Registration Type**, selecione **Red Hat Account**

6. Em **Registration Details**:
   - Digite o nome de usuário de sua conta de chapéu vermelho no campo **Login**,
   - Digite a senha de sua conta Red hat no campo **Password**.
   - Digite o nome de sua organização no campo **Organization**.
7. Clique em **Registrar**.

1.8.4. Registro de um sistema usando uma chave de ativação no GNOME

Siga as etapas deste procedimento para registrar seu sistema com uma chave de ativação. Você pode obter a chave de ativação do administrador de sua organização.

Pré-requisitos

- Chave ou chaves de ativação.
  Consulte a página Chaves de ativação para criar novas chaves de ativação.

Procedimento

1. Vá para **system menu**, que pode ser acessado no canto superior direito da tela e clique no ícone **Settings**.
2. No **Detalhes → Sobre** clique em **Register**.
3. Selecione **Registration Server**.
4. Digite **URL** para o servidor personalizado, se você não estiver usando o servidor da Red Hat.
5. No menu **Registration Type**, selecione **Activation Keys**.
6. Em **Registration Details**:
   - Entre **Activation Keys**.
     Separar várias chaves por vírgula (,).
   - Digite o nome ou ID de sua organização no campo **Organization**.
7. Clique em **Registrar**

1.9. FAZER COM QUE OS SERVIÇOS DE SISTEMA COMECEM NO MOMENTO DA INICIALIZAÇÃO

Systemd é um gerenciador de sistemas e serviços para sistemas operacionais Linux que introduz o conceito de unidades systemd.

Esta seção fornece informações sobre como garantir que um serviço seja habilitado ou desativado no momento da inicialização. Ela também explica como gerenciar os serviços através do console web.

1.9.1. Habilitação ou desabilitação dos serviços utilizando o CLI

Você pode determinar quais serviços estão habilitados ou desabilitados no momento da inicialização já durante o processo de instalação. Você também pode ativar ou desativar um serviço em um sistema operacional instalado.

Esta seção descreve os passos para habilitar ou desabilitar esses serviços em um sistema operacional já instalado:

Pré-requisitos

- Você deve ter acesso root ao sistema.

Procedimento

1. Para habilitar um serviço, use a opção **enable**:

```
# systemctl enable service_name
```

   Substitua *service_name* pelo serviço que você deseja habilitar.

   Você também pode habilitar e iniciar um serviço em um único comando:

```
# systemctl enable --now service_name
```

2. Para desativar um serviço, use a opção **disable**:

```
# systemctl disable service_name
```

   Substitua *service_name* pelo serviço que você deseja desativar.

ATENÇÃO

Você não pode habilitar um serviço que tenha sido previamente mascarado. Você tem que desmascará-lo primeiro:

```
# systemctl unmask service_name
```

1.9.2. Serviços de gerenciamento no console web RHEL 8

Esta seção descreve como você também pode ativar ou desativar um serviço usando o console web. Você pode gerenciar alvos, serviços, soquetes, temporizadores e caminhos do sistema. Você também pode verificar o status do serviço, iniciar ou parar serviços, habilitá-los ou desabilitá-los.

Pré-requisitos

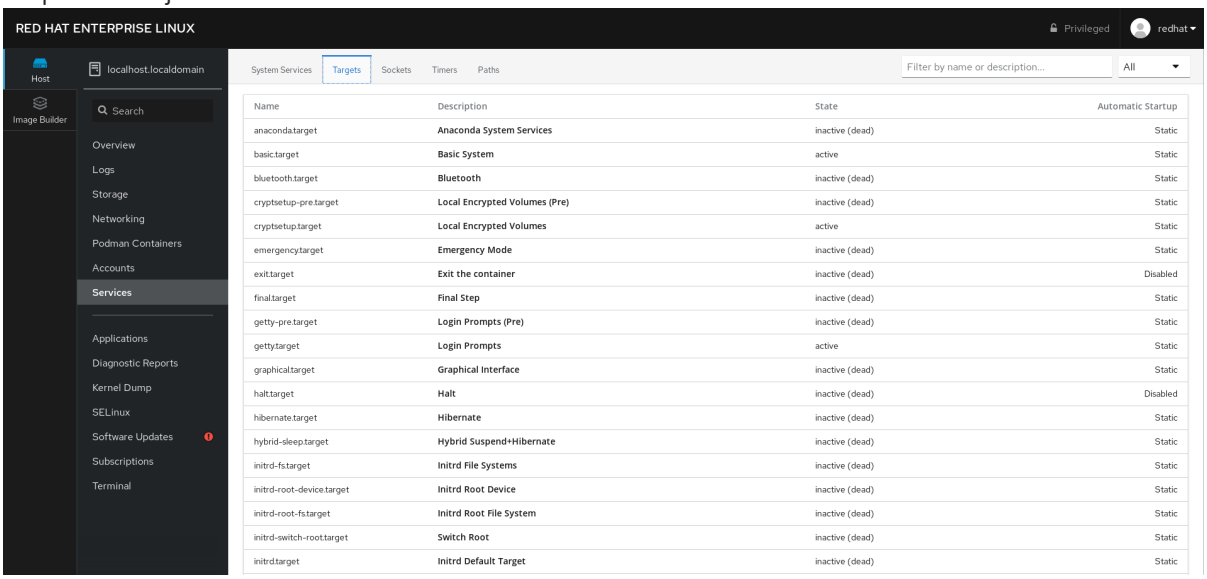
- Você deve ter acesso root ao sistema.

Procedimento

1. Aberto **https://localhost:9090/** em um navegador web de sua preferência.

2. Acesse o console web com suas credenciais de raiz no sistema.

3. Para exibir o painel do console web, clique no ícone **Host**, que se encontra no canto superior esquerdo da janela.



4. No menu, clique em **Serviços**.
   Você pode gerenciar alvos, serviços, soquetes, temporizadores e caminhos do sistema.

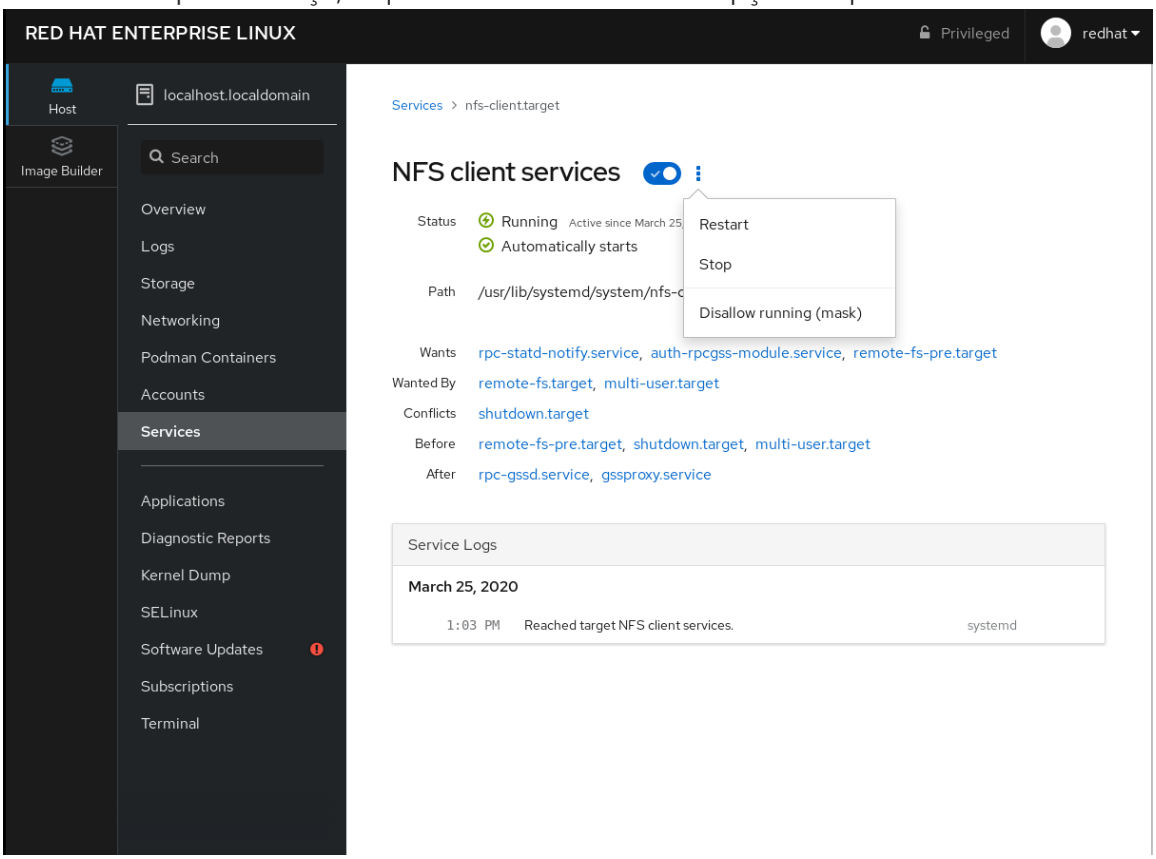
5. Por exemplo, para administrar o serviço **NFS client services**:

   a. Clique em **Alvos**.

   b. Selecione o serviço **NFS client services**.

   c. Para ativar ou desativar o serviço, clique no botão **Toogle**.

   d. Para interromper o serviço, clique no botão e escolha a opção 'Stop'.



1.10. CONFIGURANDO A SEGURANÇA DO SISTEMA

A segurança informática é a proteção dos sistemas de computador e seu hardware, software, informações e serviços contra roubo, dano, interrupção e má direção. Garantir a segurança de computadores é uma tarefa essencial, em particular em empresas que processam dados sensíveis e lidam com transações comerciais.

Esta seção cobre apenas as características básicas de segurança que podem ser configuradas após a instalação do sistema operacional. Para informações detalhadas sobre a segurança do Red Hat Enterprise Linux, veja a seção **Security** na Documentação do Produto para o Red Hat Enterprise Linux 8 .

1.10.1. Aumentar a segurança do sistema com um firewall

Um firewall é um sistema de segurança de rede que monitora e controla o tráfego de entrada e saída da rede, de acordo com as regras de segurança configuradas. Um firewall normalmente estabelece uma barreira entre uma rede interna segura e confiável e outra rede externa.

O serviço **firewalld**, que fornece um firewall no Red Hat Enterprise Linux, é automaticamente habilitado durante a instalação.

1.10.1.1. Possibilitando o serviço firewalld

Para habilitar o serviço **firewalld**, siga este procedimento.

Procedimento

1. Mostrar o status atual do site **firewalld**:

```
$ systemctl status firewalld
● firewalld.service - firewalld - dynamic firewall daemon
   Loaded: loaded (/usr/lib/systemd/system/firewalld.service; disabled; vendor preset:
enabled)
   Active: inactive (dead)
...
```

2. Se **firewalld** não estiver ativado e em funcionamento, mude para o usuário **root**, e inicie o serviço **firewalld** e habilite-o para iniciar automaticamente após o reinício do sistema:

```
# systemctl enable --now firewalld
```

Etapas de verificação

1. Verifique se **firewalld** está funcionando e habilitado:

```
$ systemctl status firewalld
● firewalld.service - firewalld - dynamic firewall daemon
   Loaded: loaded (/usr/lib/systemd/system/firewalld.service; enabled; vendor preset:
enabled)
   Active: active (running)
...
```

Recursos adicionais

- Para mais informações, consulte a página de manual **firewalld(1)**.

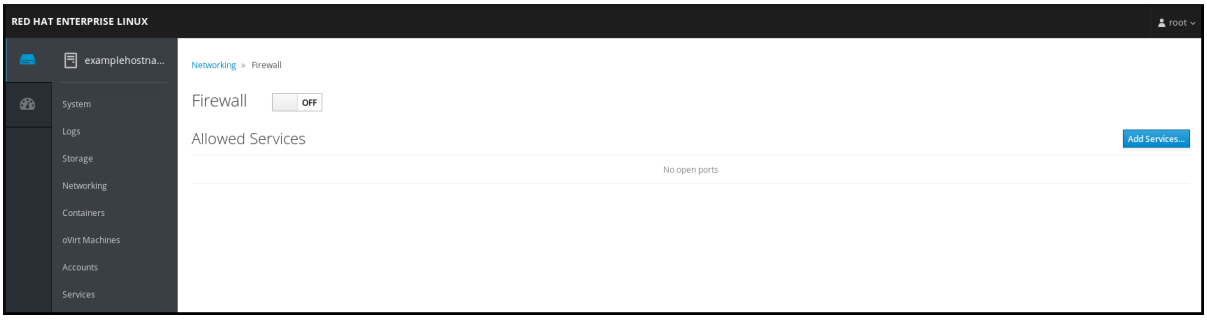
1.10.1.2. Gerenciamento do firewall no console web RHEL 8

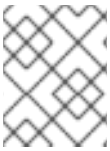
Para configurar o serviço **firewalld** no console web, navegue para **Rede → Firewall**.

Por padrão, o serviço **firewalld** está habilitado.

Procedimento

1. Para ativar ou desativar o **firewalld** no console web, troque o botão de alternância do **Firewall**.





NOTA

Além disso, você pode definir um acesso mais fino através do firewall a um serviço usando o botão **Adicionar serviços**...

1.10.1.3. Recursos adicionais

- Para informações detalhadas sobre a configuração e uso de um firewall, consulte Utilização e configuração de firewalls.

1.10.2. Gerenciando configurações básicas do SELinux

Security-Enhanced Linux (SELinux) é uma camada adicional de segurança do sistema que determina quais processos podem acessar quais arquivos, diretórios e portas. Estas permissões são definidas nas políticas do SELinux. Uma política é um conjunto de regras que guiam o mecanismo de segurança do SELinux.

1.10.2.1. SELinux estados e modos

A SELinux tem dois estados possíveis:

- Deficiente
- Habilitado

Quando a SELinux é ativada, ela funciona em uma das seguintes modalidades:

- Habilitado
  - Aplicação da lei
  - Permissiva

Em **enforcing mode**, a SELinux aplica as políticas carregadas. A SELinux nega o acesso com base nas regras da política SELinux e permite somente as interações que são explicitamente permitidas. O modo de aplicação é o modo SELinux mais seguro e é o modo padrão após a instalação.

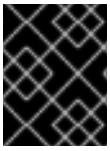
Em **permissive mode**, a SELinux não aplica as políticas carregadas. A SELinux não nega o acesso, mas relata ações que quebram as regras para o log **/var/log/audit/audit.log**. O modo permissivo é o modo padrão durante a instalação. O modo permissivo também é útil em alguns casos específicos, por exemplo, quando há problemas de solução de problemas.

Recursos adicionais

- Para mais informações sobre a SELinux, consulte Utilizando a SELinux.

1.10.2.2. Garantindo o estado necessário da SELinux

Por padrão, a SELinux opera em modo de aplicação. Entretanto, em cenários específicos, você pode configurar o SELinux para o modo permissivo ou mesmo desativá-lo.



IMPORTANTE

A Red Hat recomenda manter seu sistema no modo de aplicação. Para fins de depuração, você pode configurar o SELinux para o modo permissivo.

Siga este procedimento para mudar o estado e o modo da SELinux em seu sistema.

Procedimento

1. Exibir o modo SELinux atual:

```
$ getenforce
```

2. Para definir temporariamente a SELinux:
   a. Para o modo Enforcing:

```
# setenforce Enforcing
```

   b. Para o modo permissivo:

```
# setenforce Permissive
```

NOTA

Após o reinício, o modo SELinux é ajustado para o valor especificado no arquivo de configuração **/etc/selinux/config**.

3. Para definir o modo SELinux para persistir nas reinicializações, modifique a variável **SELINUX** no arquivo de configuração **/etc/selinux/config**.
   Por exemplo, para mudar o SELinux para o modo de aplicação:

```
# This file controls the state of SELinux on the system.
# SELINUX= can take one of these three values:
```

```
#     enforcing - SELinux security policy is enforced.
#     permissive - SELinux prints warnings instead of enforcing.
#     disabled - No SELinux policy is loaded.
SELINUX=enforcing
...
```

ATENÇÃO

A desativação do SELinux reduz a segurança de seu sistema. Evite desativar o SELinux usando a opção **SELINUX=disabled** no arquivo **/etc/selinux/config** porque isto pode resultar em vazamentos de memória e condições de corrida causando pânico no kernel. Ao invés disso, desative o SELinux adicionando o parâmetro **selinux=0** à linha de comando do kernel, conforme descrito em Mudando os modos do SELinux no momento da inicialização.

Recursos adicionais

- Para mais informações sobre as mudanças permanentes dos modos SELinux, consulte Mudança dos estados e modos SELinux.

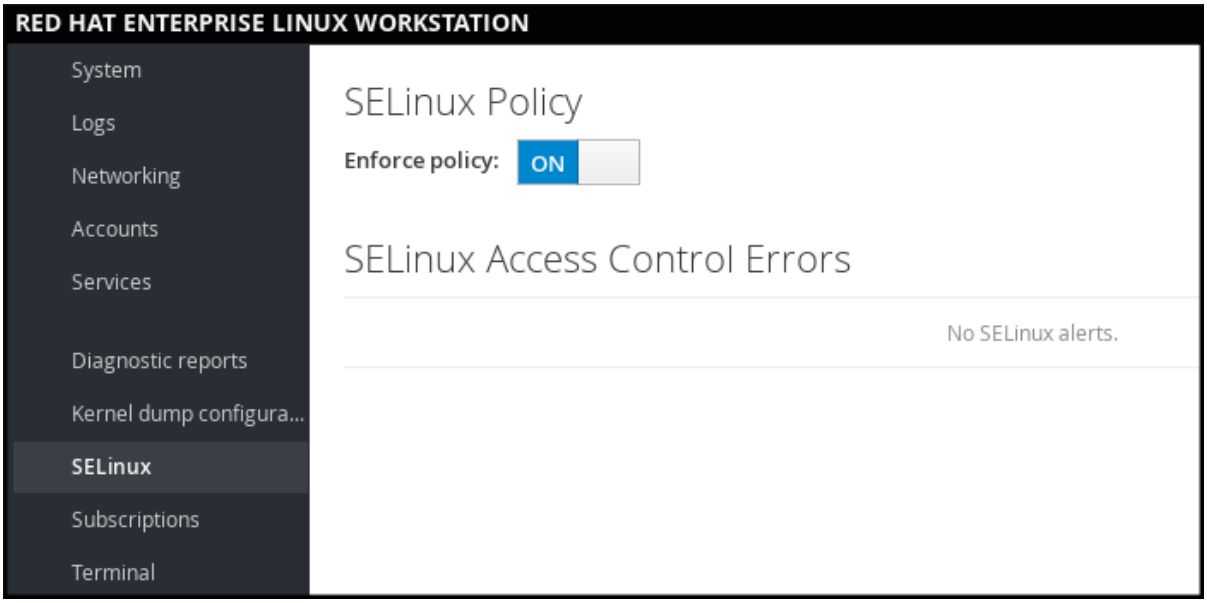
1.10.2.3. Mudança dos modos SELinux no console web RHEL 8

Você pode configurar o modo SELinux através do console web RHEL 8 no item de menu **SELinux**.

Por padrão, a SELinux aplica a política no console web, e a SELinux opera em modo de aplicação. Ao desligá-lo, você muda a SELinux para o modo permissivo. Note que esta seleção é revertida automaticamente na próxima inicialização para a configuração definida no arquivo **/etc/sysconfig/selinux**.

Procedimento

1. No console web, use o botão Alternar **política de aplicação** no item do menu SELinux para ativar ou desativar a política de aplicação da SELinux.



1.10.2.4. Próximos passos

- Você pode gerenciar várias personalizações locais SELinux em vários sistemas-alvo usando a função do sistema **selinux**. Para mais informações, consulte a seção Implementando a mesma configuração SELinux em múltiplos sistemas.

1.10.3. Próximos passos

- Usando pares de chaves ao invés de senhas para autenticação SSH
- Endurecimento da segurança
- Usando SELinux
- Redes de segurança

1.11. INTRODUÇÃO AO GERENCIAMENTO DE CONTAS DE USUÁRIOS

O Red Hat Enterprise Linux é um sistema operacional para múltiplos usuários, que permite a múltiplos usuários em diferentes computadores acessar um único sistema instalado em uma máquina.

Cada usuário opera sob sua própria conta, e o gerenciamento de contas de usuário representa assim um elemento central da administração do sistema Red Hat Enterprise Linux.

1.11.1. Visão geral das contas e grupos de usuários

Esta seção fornece uma visão geral das contas e grupos de usuários. A seguir são apresentados os diferentes tipos de contas de usuário:

- Contas de usuários normais:
  Contas normais são criadas para usuários de um determinado sistema. Tais contas podem ser adicionadas, removidas e modificadas durante a administração normal do sistema.
- Contas de usuário do sistema
  As contas de usuário do sistema representam um identificador de aplicações específicas em um sistema. Tais contas são geralmente adicionadas ou manipuladas somente no momento da instalação do software, e não são modificadas posteriormente.

ATENÇÃO

Presume-se que as contas do sistema estejam disponíveis localmente em um sistema. Se essas contas forem configuradas e fornecidas remotamente, como no caso de uma configuração LDAP, pode ocorrer uma quebra do sistema e falhas no início do serviço.

Para contas de sistema, os IDs de usuário abaixo de 1000 são reservados. Para contas normais, você pode usar IDs a partir de 1000. Entretanto, a prática recomendada é atribuir IDs a partir de 5000.

- Grupo
  Um grupo em uma entidade que une várias contas de usuários para um propósito comum, como a concessão de acesso a determinados arquivos.

- Para mais informações, veja

- Para atribuição de IDs, consulte o arquivo /**etc/login.defs**.

1.11.2. Gerenciamento de contas e grupos usando ferramentas de linha de comando

Esta seção descreve as ferramentas básicas de linha de comando para gerenciar contas e grupos de usuários.

- Para exibir as identificações do usuário e do grupo:

```
$ id
uid=1000(example.user) gid=1000(example.user) groups=1000(example.user),10(wheel)
context=unconfined_u:unconfined_r:unconfined_t:s0-s0:c0.c1023
```

- Para criar uma nova conta de usuário:

```
# useradd example.user
```

- Para atribuir uma nova senha a uma conta de usuário pertencente a *example.user*:

```
# passwd example.user
```

- Para adicionar um usuário a um grupo:

```
# usermod -a -G example.group example.user
```

Recursos adicionais

- As páginas de manual **useradd(8)**, **passwd(1)**, e **usermod(8)**.

1.11.3. Contas de usuário do sistema gerenciadas no console web

Com as contas de usuário exibidas no console web RHEL, você pode:

- Autenticar os usuários ao acessar o sistema.
- Defina os direitos de acesso ao sistema.

O console web RHEL exibe todas as contas de usuário localizadas no sistema. Portanto, você pode ver pelo menos uma conta de usuário logo após o primeiro login no console web.

Após o login no console web RHEL, você pode realizar as seguintes operações:

- Criar novas contas de usuários.
- Alterar seus parâmetros.
- Bloquear contas.
- Encerrar sessões de usuários.

1.11.4. Adicionando novas contas usando o console web

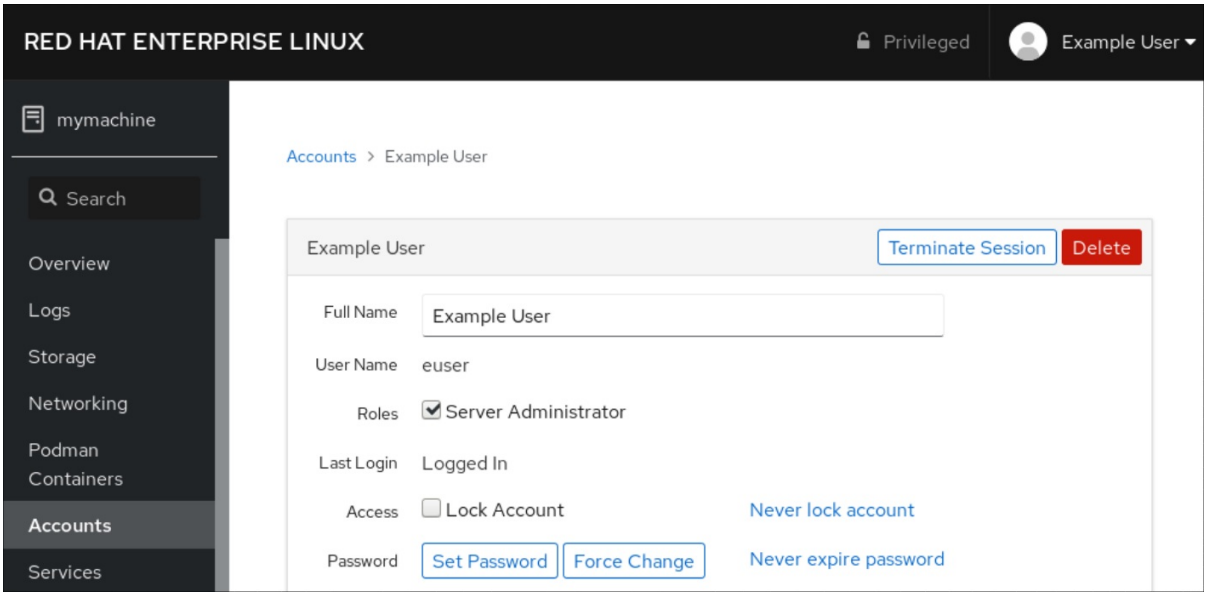
Use os seguintes passos para adicionar contas de usuário ao sistema e definir os direitos de administração das contas através do console web RHEL.

Pré-requisitos

- O console web RHEL deve ser instalado e acessível. Para detalhes, consulte Instalando o console web.

Procedimento

1. Acesse o console web RHEL.
2. Clique em **Contas**.
3. Clique em **Criar nova conta**.
1. No campo **Full Name**, digite o nome completo do usuário.
   O console web RHEL sugere automaticamente um nome de usuário a partir do nome completo e o preenche no campo **User Name**. Se você não quiser usar a convenção de nomenclatura original que consiste na primeira letra do nome e no sobrenome completo, atualize a sugestão.
2. Nos campos **Password/Confirm**, digite a senha e digite-a novamente para verificar se sua senha está correta.
   A barra colorida colocada abaixo dos campos mostra o nível de segurança da senha digitada, o que não permite criar um usuário com uma senha fraca.
1. Clique em **Criar** para salvar as configurações e fechar a caixa de diálogo.
2. Selecione a conta recém-criada.
3. Selecione **Server Administrator** no item **Roles**.



Agora você pode ver a nova conta nas configurações do **Accounts** e pode usar as credenciais para se conectar ao sistema.

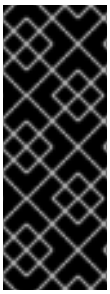
1.12. DESCARGA DE UM GRÃO QUEBRADO PARA ANÁLISE POSTERIOR

Para analisar por que um sistema falhou, você pode usar o serviço **kdump** para salvar o conteúdo da memória do sistema para análise posterior.

Esta seção fornece uma breve introdução a **kdump**, e informações sobre como configurar **kdump** usando o console web RHEL ou usando a função correspondente do sistema RHEL.

1.12.1. O que é kdump

kdump é um serviço que fornece um mecanismo de despejo de emergência. O serviço permite salvar o conteúdo da memória do sistema para análise posterior. **kdump** usa a chamada do sistema **kexec** para inicializar no segundo kernel (a *capture kernel*) sem reinicializar; e então captura o conteúdo da memória do kernel travado (a *crash dump* ou a *vmcore*) e o salva. O segundo kernel reside em uma parte reservada da memória do sistema.



IMPORTANTE

Um despejo de kernel pode ser a única informação disponível no caso de uma falha no sistema (um bug crítico). Portanto, garantir que **kdump** esteja operacional é importante em ambientes de missão crítica. A Red Hat recomenda que os administradores de sistema atualizem e testem regularmente **kexec-tools** em seu ciclo normal de atualização do kernel. Isto é especialmente importante quando novos recursos do kernel são implementados.

1.12.2. Configuração do uso da memória kdump e localização do alvo no console web

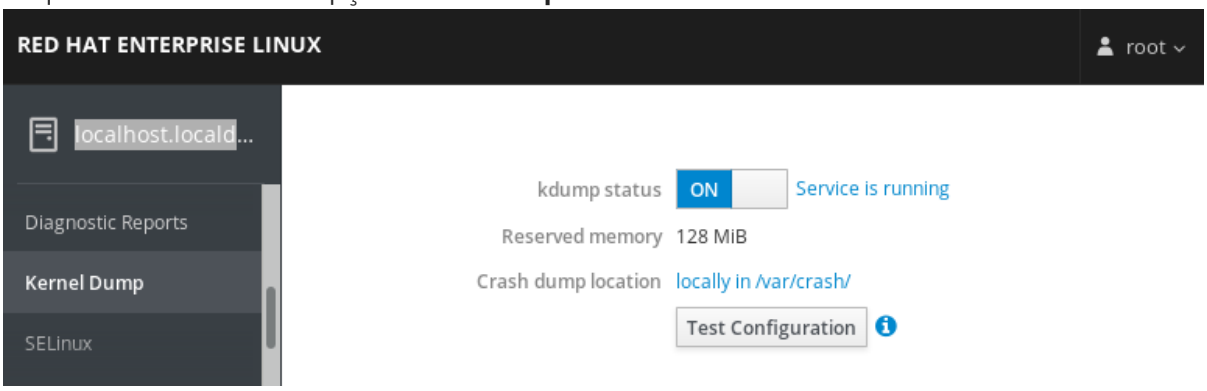
O procedimento abaixo mostra como usar a aba **Kernel Dump** na interface do console web do Red Hat Enterprise Linux para configurar a quantidade de memória que é reservada para o kdump kernel. O procedimento também descreve como especificar a localização alvo do arquivo vmcore dump e como testar sua configuração.

Pré-requisitos

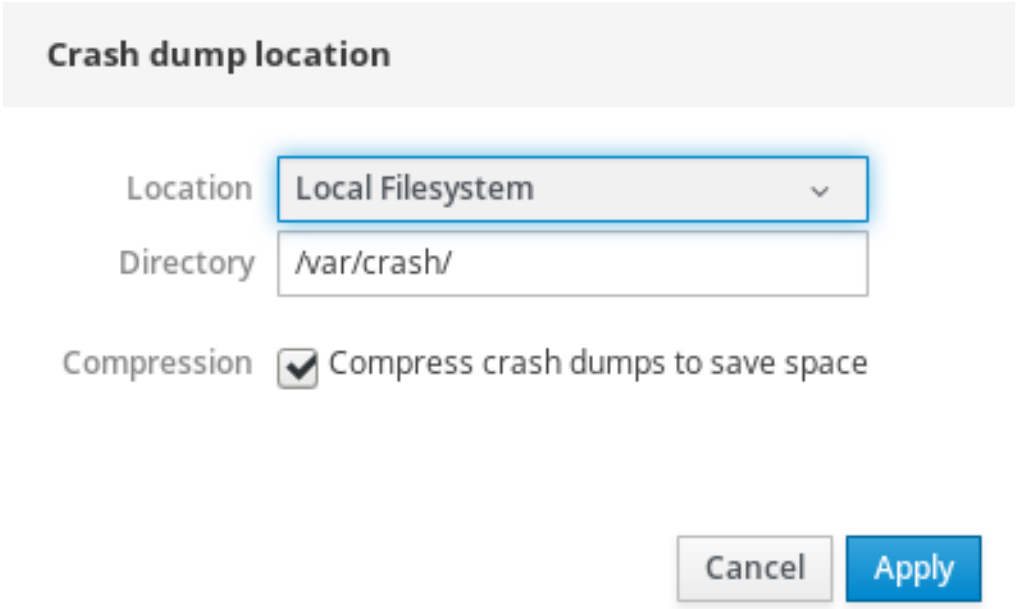
- Introdução à operação do *web console*

Procedimento

1. Abra a aba **Kernel Dump** e inicie o serviço **kdump**.
2. Configure o uso da memória **kdump** através do *command line*.
3. Clique no link ao lado da opção **Crash dump location**.



4. Selecione a opção **Local Filesystem** no menu suspenso e especifique o diretório no qual você deseja salvar o despejo.

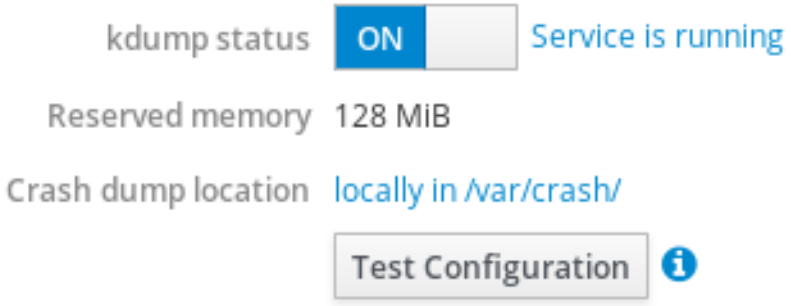


- Alternativamente, selecione a opção **Remote over SSH** no menu suspenso para enviar o vmcore para uma máquina remota usando o protocolo SSH.
  Preencha os campos **Server**, **ssh key**, e **Directory** com o endereço da máquina remota, localização da chave ssh e um diretório de destino.
- Outra opção é selecionar a opção **Remote over NFS** no menu suspenso e preencher o campo **Mount** para enviar o vmcore para uma máquina remota usando o protocolo NFS.

NOTA

Marque a caixa de seleção **Compression** para reduzir o tamanho do arquivo vmcore.

5. Teste sua configuração quebrando o kernel.



ATENÇÃO

Esta etapa interrompe a execução do núcleo e resulta em um colapso do sistema e perda de dados.

Recursos adicionais

- Para uma lista completa das metas atualmente suportadas para **kdump**, veja *Supported kdump targets*.
- Para obter informações sobre como configurar um servidor SSH e configurar uma autenticação baseada em chaves, consulte *Using secure communications between two systems with OpenSSH*.

1.12.3. Configuração do kdump usando as funções do sistema RHEL

As funções do sistema RHEL são uma coleção de funções e módulos possíveis que fornecem uma interface de configuração consistente para gerenciar remotamente vários sistemas RHEL. A função **kdump** permite definir parâmetros básicos de despejo do kernel em múltiplos sistemas.

ATENÇÃO

A função **kdump** substitui inteiramente a configuração **kdump** dos anfitriões gerenciados, substituindo o arquivo **/etc/kdump.conf**. Além disso, se a função **kdump** for aplicada, todas as configurações anteriores **kdump** também serão substituídas, mesmo que não estejam especificadas pelas variáveis da função, pela substituição do arquivo **/etc/sysconfig/kdump**.

O exemplo a seguir mostra como aplicar o papel do sistema **kdump** para definir a localização dos arquivos de despejo do acidente:

```
---
- hosts: kdump-test
  vars:
    kdump_path: /var/crash
  roles:
    - rhel-system-roles.kdump
```

Recursos adicionais

- Para uma referência detalhada sobre as variáveis de função **kdump**, instale o pacote **rhel-system-roles**, e veja os arquivos **README.md** ou **README.html** no diretório **/usr/share/doc/rhel-system-roles/kdump**.
- Para mais informações sobre os papéis do Sistema RHEL, veja Introdução aos papéis do Sistema RHEL

1.12.4. Recursos adicionais

- Para informações mais detalhadas sobre **kdump**, consulte Instalando e configurando o kdump.

1.13. RECUPERANDO E RESTAURANDO UM SISTEMA

Para recuperar e restaurar um sistema usando um backup existente, o Red Hat Enterprise Linux fornece o utilitário Relax-and-Recover (ReaR).

Você pode usar o utilitário como uma solução de recuperação de desastres e também para a migração do sistema.

O utilitário permite realizar as seguintes tarefas:

- Produzir uma imagem bootável e restaurar o sistema a partir de um backup existente, usando a imagem.
- Replicar o layout original de armazenamento.
- Restaurar os arquivos do usuário e do sistema.
- Restaurar o sistema para um hardware diferente.

Além disso, para a recuperação de desastres, você também pode integrar certo software de backup com o ReaR.

A criação do ReaR envolve as seguintes etapas de alto nível:

1. Instale o ReaR.
2. Criar sistema de resgate.
3. Modificar o arquivo de configuração ReaR, para adicionar detalhes do método de backup.
4. Gerar arquivos de backup.

1.13.1. Criando o ReaR

Use os seguintes passos para instalar os pacotes para usar o utilitário Relax-and-Recover (ReaR), criar um sistema de resgate, configurar e gerar um backup.

Pré-requisitos

- As configurações necessárias de acordo com o plano de restauração de reserva estão prontas. Note que você pode usar o método de backup **NETFS**, um método totalmente integrado e embutido com o ReaR.

Procedimento

1. Instale o ReaR, o programa de pré-masterização **genisomage**, e o pacote **syslinux** fornecendo um conjunto de carregadores de inicialização:

```
# yum install rear genisoimage syslinux
```

2. Criar um sistema de resgate:

```
# rear mkrescue
```

3. Modifique o arquivo de configuração ReaR em um editor de sua escolha, por exemplo:

```
# vi /etc/rear/local.conf
```

4. Adicione os detalhes da configuração de backup a **/etc/rear/local.conf**. Por exemplo, no caso do método de backup **NETFS**, adicione as seguintes linhas:

```
BACKUP=NETFS
BACKUP_URL=backup.location
```

   Substitua *backup.location* pela URL de seu local de backup.

5. Para configurar o ReaR para manter os arquivos de backup anteriores quando os novos forem criados, adicione também a seguinte linha ao arquivo de configuração:

```
NETFS_KEEP_OLD_BACKUP_COPY=y
```

6. Para tornar os backups incrementais, o que significa que somente os arquivos alterados são copiados em cada execução, acrescente a seguinte linha:

```
BACKUP_TYPE=incremental
```

7. Faça um backup de acordo com o plano de restauração.

1.14. SOLUÇÃO DE PROBLEMAS USANDO ARQUIVOS DE LOG

Os arquivos de log contêm mensagens sobre o sistema, incluindo o kernel, serviços e aplicações rodando nele. Estes contêm informações que ajudam a solucionar problemas ou a monitorar as funções do sistema. O sistema de registro no Red Hat Enterprise Linux é baseado no protocolo **syslog** embutido. Programas particulares usam este sistema para registrar eventos e organizá-los em arquivos de registro, que são úteis ao auditar o sistema operacional e solucionar vários problemas.

1.14.1. Serviços que lidam com mensagens do syslog

Os dois serviços a seguir tratam das mensagens **syslog**:

- O daemon **systemd-journald**
- O serviço **Rsyslog**

O daemon **systemd-journald** coleta mensagens de várias fontes e as encaminha para **Rsyslog** para processamento posterior. O daemon **systemd-journald** coleta mensagens das seguintes fontes:

- Kernel
- Etapas iniciais do processo de inicialização
- Saída padrão e de erro dos demônios à medida que iniciam e rodam
- **Syslog**

O serviço **Rsyslog** ordena as mensagens **syslog** por tipo e prioridade e as grava nos arquivos do diretório **/var/log**. O diretório **/var/log** armazena de forma persistente as mensagens de registro.

1.14.2. Subdiretórios que armazenam mensagens do syslog

Os seguintes subdiretórios sob o **/var/log** directory store **syslog** mensagens.

- **/var/log/messages** - todas **syslog** mensagens exceto as seguintes
- **/var/log/secure** - mensagens e erros relacionados à segurança e autenticação
- **/var/log/maillog** - mensagens e erros relacionados ao servidor de e-mail
- **/var/log/cron** - arquivos de log relacionados a tarefas executadas periodicamente
- **/var/log/boot.log** - arquivos de log relacionados à inicialização do sistema

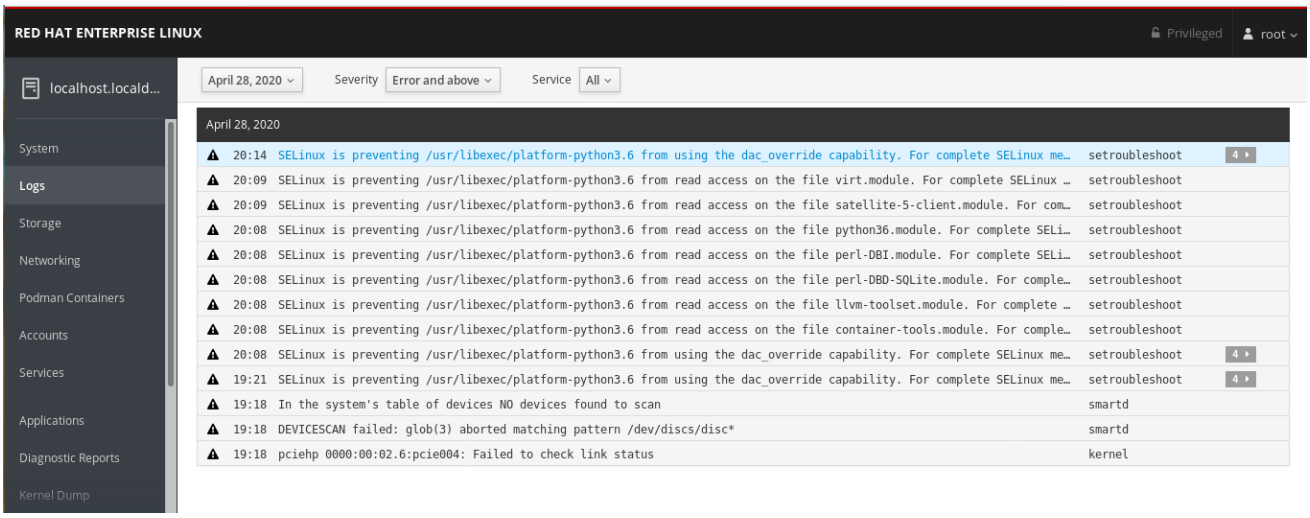
1.14.3. Inspeção de arquivos de log usando o console web

Siga as etapas deste procedimento para inspecionar os arquivos de registro usando o console web.

Procedimento

1. Acesse o console web do Red Hat Enterprise Linux 8.
   Para obter detalhes, consulte Login no console web.
2. Clique em **Logs**.

Figura 1.2. Inspeção dos arquivos de log no console web RHEL 8



1.14.4. Visualização de logs usando a linha de comando

O Journal é um componente do sistemad que ajuda a visualizar e gerenciar os arquivos de log. Ele aborda problemas ligados à extração tradicional, estreitamente integrado com o resto do sistema, e suporta várias tecnologias de extração e gerenciamento de acesso para os arquivos de log.

Você pode usar o comando **journalctl** para visualizar mensagens no diário do sistema usando a linha de comando, por exemplo:

```
$ journalctl -b | grep kvm
May 15 11:31:41 localhost.localdomain kernel: kvm-clock: Using msrs 4b564d01 and 4b564d00
May 15 11:31:41 localhost.localdomain kernel: kvm-clock: cpu 0, msr 76401001, primary cpu clock
...
```

Tabela 1.1. Visualização de informações do sistema

Comando	Descrição
journalctl	Mostra todos os lançamentos de diário coletados.
journalctl FILEPATH	Mostra os logs relacionados a um arquivo específico. Por exemplo, o comando **journalctl /dev/sda** exibe os logs relacionados ao sistema de arquivos **/dev/sda**.
journalctl -b	Mostra os logs para o boot atual.
journalctl -k -b -1	Mostra os logs do kernel para o boot atual.

Tabela 1.2. Visualização de informações sobre serviços específicos

Comando	Descrição
journalctl -b _SYSTEMD_UNIT=foo	Filtros de registro para ver os que correspondem ao serviço "foo" **systemd**.

Comando	Descrição
journalctl -b _SYSTEMD_UNIT=foo _PID=number	Combina fósforos. Por exemplo, este comando mostra logs para **systemd-units** que combinam **foo** e o PID **number**.
journalctl -b _SYSTEMD_UNIT=foo _PID=number _SYSTEMD_UNIT=foo1	O separador " " combina duas expressões em um OR lógico. Por exemplo, este comando mostra todas as mensagens do processo do serviço **foo** com o **PID** mais todas as mensagens do serviço **foo1** (de qualquer um de seus processos).
journalctl -b _SYSTEMD_UNIT=foo _SYSTEMD_UNIT=foo1	Este comando mostra todas as entradas correspondentes a qualquer uma das expressões, referindo-se ao mesmo campo. Aqui, este comando mostra registros que correspondem a uma unidade de sistema **foo** ou a uma unidade de sistema **foo1**.

Tabela 1.3. Visualização de logs relacionados a botas específicas

Comando	Descrição
journalctl --list-boots	Mostra uma lista tabular de números de inicialização, suas identificações e os carimbos de tempo da primeira e última mensagem referente à inicialização. Você pode usar o ID no próximo comando para visualizar informações detalhadas.
journalctl --boot=ID _SYSTEMD_UNIT=foo	Mostra informações sobre o ID de inicialização especificado.

1.14.5. Recursos adicionais

- Para detalhes sobre como configurar **Rsyslog** para registrar registros, veja Configurando uma solução de registro remoto.
- A página do homem **journalctl(1)**.
- Para mais informações em **systemd**, veja Gerenciamento de serviços com systemd .

1.15. ACESSO AO SUPORTE DA RED HAT

Esta seção descreve como solucionar efetivamente seus problemas usando o suporte da Red Hat e **sosreport**.

Para obter suporte da Red Hat, use o Portal do Cliente da Red Hat , que fornece acesso a tudo o que está disponível com sua assinatura.

1.15.1. Obtenção de suporte da Red Hat através do Portal do Cliente Red Hat

A seção seguinte descreve como usar o Portal do Cliente da Red Hat para obter ajuda.

Pré-requisitos

- Uma conta de usuário válida no Portal do Cliente da Red Hat. Veja Criar um Login da Red Hat .
- Uma assinatura ativa para o sistema RHEL.

Procedimento

1. Acesse o suporte da Red Hat :
   a. Abrir um novo caso de suporte.
   b. Iniciar uma conversa ao vivo com um especialista em Red Hat.
   c. Entre em contato com um especialista da Red Hat fazendo uma chamada ou enviando um e-mail.

1.15.2. Solução de problemas usando o sosreport

O comando **sosreport** coleta detalhes de configuração, informações do sistema e informações de diagnóstico de um sistema Red Hat Enterprise Linux.

A seção seguinte descreve como usar o comando **sosreport** para produzir relatórios para seus casos de suporte.

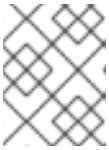
Pré-requisitos

- Uma conta de usuário válida no Portal do Cliente da Red Hat. Veja Criar um Login da Red Hat .
- Uma assinatura ativa para o sistema RHEL.
- Um número de caixa de apoio.

Procedimento

1. Instale o pacote **sos**:

```
# yum install sos
```



NOTA

A instalação mínima default do Red Hat Enterprise Linux não inclui o pacote **sos**, que fornece o comando **sosreport**.

2. Gerar um relatório:

```
# sosreport
```

3. Anexe o relatório ao seu caso de apoio.
   Veja o Como posso anexar um arquivo a um caso de suporte da Red Hat? Artigo da Red Hat Knowledgebase para mais informações.

   Observe que, ao anexar o relatório, você é solicitado a inserir o número do caso de suporte relevante.

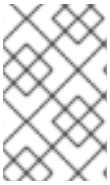
Recursos adicionais

- Para mais informações em **sosreport**, veja o que é um sosreport e como criar um no Red Hat Enterprise Linux 4.6 e posteriores? Artigo da Red Hat Knowledgebase.

CAPÍTULO 2. GERENCIAMENTO DE PACOTES DE SOFTWARE

2.1. FERRAMENTAS DE GERENCIAMENTO DE SOFTWARE NO RED HAT ENTERPRISE LINUX 8

No RHEL 8, a instalação do software é habilitada pela nova versão do **YUM** (**YUM v4**), que se baseia na ferramenta **DNF** tecnologia.



NOTA

A documentação a montante identifica a tecnologia como **DNF** e a ferramenta é referida como **DNF** no rio acima. Como resultado, alguns resultados retornados pelo novo **YUM** ferramenta no RHEL 8 menciona **DNF**.

Embora **YUM v4** usado no RHEL 8 é baseado em **DNF**é compatível com **YUM v3** usado no RHEL 7. Para instalação de software, o comando **yum** e a maioria de suas opções funcionam da mesma forma na RHEL 8 como funcionavam na RHEL 7.

Selecionado **yum** plug-ins e utilitários foram portados para o novo DNF back end, e podem ser instalados com os mesmos nomes que no RHEL 7. Os pacotes também fornecem links simbólicos de compatibilidade, de modo que os binários, arquivos de configuração e diretórios podem ser encontrados em locais habituais.

Note que o API Python API legado fornecido por **YUM v3** não está mais disponível. Você pode migrar seus plug-ins e scripts para a nova API fornecida por **YUM v4** (DNF Python API), que é estável e totalmente suportada. Veja DNF API Reference para mais informações.

2.2. FLUXOS DE APLICAÇÃO

O Red Hat Enterprise Linux 8 introduz o conceito de Fluxos de Aplicação. Múltiplas versões de componentes de espaço do usuário são agora entregues e atualizadas com mais freqüência do que os pacotes do sistema operacional principal. Isto proporciona maior flexibilidade para personalizar o Red Hat Enterprise Linux sem impactar a estabilidade subjacente da plataforma ou implementações específicas.

Os componentes disponibilizados como Application Streams podem ser empacotados como módulos ou pacotes RPM, e são entregues através do repositório AppStream no Red Hat Enterprise Linux 8. Cada Application Stream tem um determinado ciclo de vida, seja o mesmo que o RHEL 8 ou mais curto, mais adequado para a aplicação em particular. Os fluxos de aplicações com um ciclo de vida mais curto são listados na página do Red Hat Enterprise Linux 8 Application Streams Life Cycle Cycle .

Os módulos são coleções de pacotes que representam uma unidade lógica: uma aplicação, uma pilha de idiomas, um banco de dados ou um conjunto de ferramentas. Estes pacotes são construídos, testados e lançados juntos.

Os fluxos de módulos representam versões dos componentes do Application Stream. Por exemplo, dois fluxos (versões) do servidor de banco de dados PostgreSQL estão disponíveis no módulo postgresql: PostgreSQL 10 (o fluxo padrão) e PostgreSQL 9.6. Apenas um fluxo de módulo pode ser instalado no sistema. Versões diferentes podem ser utilizadas em containers separados.

Os comandos detalhados do módulo são descritos no documento Instalar, gerenciar e remover componentes de espaço do usuário. Para uma lista de módulos disponíveis no AppStream, veja o manifesto de pacotes.

2.3. BUSCA DE PACOTES DE SOFTWARE

yum permite realizar um conjunto completo de operações com pacotes de software.

A seção seguinte descreve como usar **yum** para:

- Busca de embalagens.
- Liste os pacotes.
- Lista de repositórios.
- Mostrar informações sobre os pacotes.
- Listar grupos de pacotes.
- Especificar as expressões globais na entrada do yum.

2.3.1. Buscando pacotes com yum

- Para pesquisar um pacote, use:

  ```
  # pesquisa de yum term
  ```

  Substituir *term* por um termo relacionado ao pacote.

  Note que o comando **yum search** retorna os termos de correspondência dentro do nome e resumo dos pacotes. Isto torna a busca mais rápida e permite que você procure por pacotes dos quais você não sabe o nome, mas para os quais você conhece um termo relacionado.

- Para incluir os termos "match" nas descrições dos pacotes, use:

  ```
  # pesquisa de yum -- tudo term
  ```

  Substitua *term* por um termo que você deseja procurar em um nome de pacote, resumo ou descrição.

  Note que **yum search --all** permite uma busca mais exaustiva, mas mais lenta.

2.3.2. Listagem de pacotes com yum

- Para listar informações sobre todos os pacotes instalados e disponíveis, use:

  ```
  # lista de yum -- todos
  ```

- Para listar todos os pacotes instalados em seu sistema, use:

  ```
  # lista yum --instalada
  ```

- Para listar todos os pacotes em todos os repositórios habilitados que estão disponíveis para instalação, use:

  ```
  # lista yum --disponível
  ```

Observe que você pode filtrar os resultados anexando expressões globais como argumentos. Consulte Seção 2.3.6, "Especificação de expressões globais na entrada do yum" para obter mais detalhes.

2.3.3. Listagem de repositórios com yum

- Para listar todos os repositórios habilitados em seu sistema, use:

```
# repolista de yum
```

- Para listar todos os repositórios desativados em seu sistema, use:

```
# Repolista de yum - deficiente
```

- Para listar tanto os repositórios habilitados como os desabilitados, use:

```
# yum repolist --tudo
```

- Para listar informações adicionais sobre os repositórios, use:

```
# yum repoinfo
```

Observe que você pode filtrar os resultados passando o ID ou nome dos repositórios como argumentos ou anexando expressões globais. Consulte Seção 2.3.6, "Especificação de expressões globais na entrada do yum" para obter mais detalhes.

2.3.4. Exibição de informações do pacote com yum

- Para exibir informações sobre um ou mais pacotes, use:

```
# info yum package-name
```

Substitua *package-name* pelo nome do pacote.

Observe que você pode filtrar os resultados anexando expressões globais como argumentos. Consulte Seção 2.3.6, "Especificação de expressões globais na entrada do yum" para obter mais detalhes.

2.3.5. Listagem de grupos de pacotes com yum

- Para ver o número de grupos instalados e disponíveis, use:

```
# resumo do grupo yum
```

- Para listar todos os grupos instalados e disponíveis, use:

```
# lista do grupo yum
```

Observe que você pode filtrar os resultados anexando opções de linha de comando para o comando **yum group list** (**--hidden**, **--available**). Para mais opções disponíveis, veja as páginas de manual.

- Para listar os pacotes obrigatórios e opcionais contidos em um determinado grupo, use:

```
# informações sobre o grupo yum group-name
```

Substituir *group-name* pelo nome do grupo.

Observe que você pode filtrar os resultados anexando expressões globais como argumentos. Consulte Seção 2.7.4, “Especificação de expressões globais na entrada do yum” para obter mais detalhes.

2.3.6. Especificação de expressões globais na entrada do yum

os comandos**yum** permitem filtrar os resultados, anexando um ou mais *glob expressions* como argumentos. Expressões globais devem ser evitadas quando passadas como argumentos para o comando **yum**. Para garantir que as expressões globais sejam passadas para **yum** como pretendido, use um dos seguintes métodos:

- Dupla ou única citação de toda a expressão global.

  ```
  # yum fornece "file-name
  ```

  Substituir *file-name* pelo nome do arquivo.

- Escape dos caracteres curinga precedendo-os com uma contrabarra (\).

  ```
  # yum fornece {\i1}/file-name
  ```

  Substituir *file-name* pelo nome do arquivo.

2.4. INSTALAÇÃO DE PACOTES DE SOFTWARE

A seção seguinte descreve como usar **yum** para:

- Instalar pacotes.
- Instalar um grupo de pacotes.
- Especifique um nome de pacote na entrada do yum.

2.4.1. Instalação de embalagens com yum

- Para instalar um pacote e todas as dependências do pacote, use:

  ```
  # instalação do yum package-name
  ```

  Substitua *package-name* pelo nome do pacote.

- Para instalar vários pacotes e suas dependências simultaneamente, use:

  ```
  # instalação do yum package-name-1 package-name-2
  ```

  Substituir *package-name-1* e *package-name-2* pelos nomes dos pacotes.

- Ao instalar pacotes em um sistema *multilib* (AMD64, máquina Intel 64), você pode especificar a arquitetura do pacote, anexando-o ao nome do pacote:

  ```
  # instalação do yum package-name.arch
  ```

Substituir *package-name.arch* pelo nome e arquitetura do pacote.

- Se você sabe o nome do binário que deseja instalar, mas não o nome do pacote, você pode usar o caminho para o binário como um argumento:

```
# instalação do yum /usr/sbin/binary-file
```

Substitua ***/usr/sbin/binary-file*** com um caminho para o arquivo binário.

yum busca através das listas de pacotes, encontra o pacote que fornece ***/usr/sbin/binary-file*** e lhe pergunta se você deseja instalá-lo.

- Para instalar um pacote previamente descarregado de um diretório local, use:

```
# instalação do yum /path/
```

Substitua */path/* pelo caminho para o pacote.

Observe que você pode otimizar a pesquisa de pacotes definindo explicitamente como analisar o argumento. Veja Seção 2.4.3, “Especificação de um nome de pacote na entrada do yum” para mais detalhes.

2.4.2. Instalação de um grupo de embalagens com yum

- Para instalar um grupo de pacotes por um nome de grupo, use:

```
# instalação do grupo yum group-name
```

Ou

```
# yum install @group-name
```

Substituir *group-name* pelo nome completo do grupo ou grupo ambiental.

- Para instalar um grupo de pacotes pelo groupID, use:

```
# instalação do grupo yum groupID
```

Substituir *groupID* pela identificação do grupo.

2.4.3. Especificação de um nome de pacote na entrada do yum

Para otimizar o processo de instalação e remoção, você pode anexar os sufixos **-n**, **-na**, ou **-nerva** aos comandos **yum install** e **yum remove** para definir explicitamente como analisar um argumento:

- Para instalar um pacote usando o seu nome exato, use:

```
# yum install-n name
```

Substituir *name* com o nome exato do pacote.

- Para instalar um pacote usando seu nome exato e arquitetura, use:

```
# yum install-na name.architecture
```

Substituir *name* e *architecture* pelo nome exato e arquitetura do pacote.

- Para instalar um pacote usando seu nome exato, época, versão, lançamento e arquitetura, use:

```
# yum install-nevra name-epoch:version-release.architecture
```

Substituir *name*, *epoch*, *version*, *release*, e *architecture* pelo nome exato, época, versão, lançamento, e arquitetura do pacote.

2.5. ATUALIZAÇÃO DE PACOTES DE SOFTWARE

yum permite que você verifique se seu sistema tem alguma atualização pendente. Você pode listar os pacotes que precisam de atualização e optar por atualizar um único pacote, vários pacotes ou todos os pacotes de uma só vez. Se algum dos pacotes que você escolher atualizar tiver dependências, eles também serão atualizados.

A seção seguinte descreve como usar **yum** para:

- Verificar por atualizações.
- Atualizar um único pacote.
- Atualizar um grupo de pacotes.
- Atualizar todos os pacotes e suas dependências.
- Aplicar atualizações de segurança.
- Automatizar atualizações de software.

2.5.1. Verificação de atualizações com o yum

- Para ver quais pacotes instalados em seu sistema têm atualizações disponíveis, use:

```
# yum check-update
```

A saída retorna a lista de pacotes e suas dependências que têm uma atualização disponível.

2.5.2. Atualização de um único pacote com yum

- Para atualizar um pacote, use:

```
# yum atualização package-name
```

Substitua *package-name* pelo nome do pacote.

IMPORTANTE

Ao aplicar atualizações no kernel, **yum** sempre **installs** um novo kernel, independentemente de você estar usando o comando **yum update** ou **yum install**.

2.5.3. Atualização de um grupo de pacotes com yum

- Para atualizar um grupo de pacotes, use:

```
# atualização do grupo yum group-name
```

  Substituir *group-name* pelo nome do grupo de pacotes.

2.5.4. Atualização de todos os pacotes e suas dependências com yum

- Para atualizar todos os pacotes e suas dependências, use:

```
# yum atualização
```

2.5.5. Atualização de pacotes relacionados à segurança com yum

- Para atualizar para os últimos pacotes disponíveis que tenham erratas de segurança, use:

```
# atualização do yum - segurança
```

- Para atualizar para os últimos pacotes de erratas de segurança, use:

```
# yum update-minimal -segurança
```

2.5.6. Automatização de atualizações de software

Para verificar e baixar atualizações de pacotes automática e regularmente, você pode usar a ferramenta **DNF Automatic** que é fornecida pelo pacote **dnf-automatic**.

DNF Automatic é uma interface alternativa de linha de comando para **yum** que é adequada para execução automática e regular usando temporizadores de sistema, cron jobs e outras ferramentas semelhantes.

DNF Automatic sincroniza os metadados do pacote conforme necessário e depois verifica se há atualizações disponíveis. Depois, a ferramenta pode realizar uma das seguintes ações, dependendo de como você a configura:

- Saída
- Download de pacotes atualizados
- Faça o download e aplique as atualizações

O resultado da operação é então relatado por um mecanismo selecionado, como a saída padrão ou e-mail.

2.5.6.1. Instalando o DNF Automático

O procedimento a seguir descreve como instalar a ferramenta **DNF Automatic**.

Procedimento

- Para instalar o pacote **dnf-automatic**, use:

```
# yum instalar dnf-automatic
```

Etapas de verificação

- Para verificar o sucesso da instalação, confirme a presença do pacote **dnf-automatic** executando o seguinte comando:

```
# rpm -qi dnf-automatic
```

2.5.6.2. DNF Arquivo de configuração automática

Por padrão, **DNF Automatic** usa **/etc/dnf/automatic.conf** como seu arquivo de configuração para definir seu comportamento.

O arquivo de configuração é separado nas seguintes seções tópicas:

- seção**[commands]**
  Define o modo de operação do **DNF Automatic**.
- seção**[emitters]**
  Define como os resultados de **DNF Automatic** são relatados.
- seção**[command_email]**
  Fornece a configuração do emissor de e-mail para um comando externo usado para enviar e-mail.
- seção**[email]**
  Fornece a configuração do emissor de e-mail.
- seção**[base]**
  Substitui os ajustes do arquivo principal de configuração do yum.

Com as configurações padrão do arquivo **/etc/dnf/automatic.conf**, **DNF Automatic** verifica as atualizações disponíveis, faz o download delas e relata os resultados como saída padrão.

ATENÇÃO

As configurações do modo de operação da seção **[commands]** são substituídas pelas configurações usadas por uma unidade de temporizador do sistema para todas as unidades de temporizador, exceto **dnf-automatic.timer**.

Recursos adicionais

- Para mais detalhes sobre determinadas seções, consulte a documentação DNF Automático .
- Para mais detalhes sobre unidades temporizadas do sistema, consulte as páginas do manual **man dnf-automatic**.

- Para a visão geral das unidades temporizadas do sistema incluídas no **dnf-automatic package**, veja a seção 2.5.6.4 Visão geral das unidades temporizadas do sistema incluídas no pacote dnf-automatic

2.5.6.3. Habilitando o DNF Automático

Para executar **DNF Automatic**, você sempre precisa habilitar e iniciar uma unidade temporizada específica do sistema. Você pode usar uma das unidades timer fornecidas no pacote **dnf-automatic**, ou você pode escrever sua própria unidade timer, dependendo de suas necessidades.

A seção seguinte descreve como habilitar **DNF Automatic**.

Pré-requisitos

- Você especificou o comportamento do DNF Automatic, modificando o arquivo de configuração **/etc/dnf/automatic.conf**.

Para mais informações sobre o arquivo de configuração **DNF Automatic**, consulte a seção 2.5.6.2, "DNF Automatic configuration file".

Procedimento

- Selecione, ative e inicie uma unidade de temporizador do sistema que se adapte às suas necessidades:

```
# Systemctl habilita -agora <unit>
```

  onde **<unit>** é um dos seguintes cronômetros:

  - **dnf-automatic-download.timer**
  - **dnf-automatic-install.timer**
  - **dnf-automatic-notifyonly.timer**
  - **dnf-automatic.timer**

Para as atualizações disponíveis em **downloading**, use:

```
# systemctl habilita o dnf-automatic-download.timer
```

```
# systemctl start dnf-automatic-download.timer
```

Para as atualizações disponíveis em **downloading and installing**, use:

```
# systemctl habilita o dnf-automatic-install.timer
```

```
# systemctl start dnf-automatic-install.timer
```

Para **reporting** sobre atualizações disponíveis, use:

```
# systemctl habilita dnf-automatico-nontifyonly.timer
```

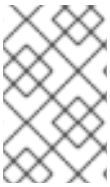
```
# systemctl start dnf-automatico-nontifyonly.timer
```

Opcionalmente, você pode usar:

```
# systemctl habilita o dnf-automatic.timer
```

```
# systemctl start dnf-automatic.timer
```

Em termos de download e aplicação de atualizações, esta unidade timer se comporta de acordo com as configurações do arquivo de configuração **/etc/dnf/automatic.conf**. O comportamento padrão é semelhante ao do **dnf-automatic-download.timer**: ele baixa os pacotes atualizados, mas não os instala.



NOTA

Alternativamente, você também pode executar **DNF Automatic** executando o arquivo **/usr/bin/dnf-automatic** diretamente da linha de comando ou a partir de um script personalizado.

Etapas de verificação

- Para verificar se o temporizador está habilitado, execute o seguinte comando:

```
# status systemctl <systemd timer unit>
```

Recursos adicionais

- Para mais informações sobre os temporizadores dnf-automáticos, consulte as páginas do manual **man dnf-automatic**.
- Para a visão geral das unidades temporizadas systemd incluídas no pacote **dnf-automatic**, veja a Seção 2.5.6.4 Visão geral das unidades temporizadas systemd incluídas no pacote **dnf-automatic**

2.5.6.4. Visão geral das unidades temporizadas do sistema incluídas no pacote dnf-automatic

As unidades temporizadas do sistema têm precedência e substituem as configurações do arquivo de configuração **/etc/dnf/automatic.conf** relativas ao download e à aplicação de atualizações.

Por exemplo, se você definir:

download_updates = yes

no arquivo de configuração **/etc/dnf/automatic.conf**, mas você ativou a unidade **dnf-automatic-notifyonly.timer**, os pacotes não serão baixados.

O pacote **dnf-automatic** inclui as seguintes unidades temporizadas do sistema:

Unidade do temporizador	Função	Substitui as configurações no arquivo **/etc/dnf/automatic.conf**?
dnf-automatic-download.timer	Faz o download dos pacotes para o cache e os torna disponíveis para atualização. Nota: Esta unidade de temporizador não instala os pacotes atualizados. Para realizar a instalação, é necessário executar o comando **dnf update**.	Sim
dnf-automatic-install.timer	Descarrega e instala pacotes atualizados.	Sim
dnf-automatic-notifyonly.timer	Faz o download apenas dos dados do repositório para manter o cache do repositório atualizado e notifica você sobre as atualizações disponíveis. Nota: Esta unidade timer não faz o download nem instala os pacotes atualizados	Sim
dnf-automatic.timer	O comportamento deste temporizador em relação ao download e à aplicação de atualizações é especificado pelas configurações no arquivo de configuração **/etc/dnf/automatic.conf**. O comportamento padrão é o mesmo que para a unidade **dnf-automatic-download.timer**: ela só baixa pacotes, mas não os instala.	Não

Recursos adicionais

- Para mais informações sobre os temporizadores **dnf-automatic**, consulte as páginas do manual **man dnf-automatic**.
- Para obter mais informações sobre o arquivo de configuração **/etc/dnf/automatic.conf**, consulte a seção 2.5.6.2. DNF Arquivo de configuração automática

2.6. DESINSTALAÇÃO DE PACOTES DE SOFTWARE

A seção seguinte descreve como usar **yum** para:

- Remover as embalagens.
- Remover um grupo de pacotes.
- Especifique um nome de pacote na entrada do yum.

2.6.1. Remoção de embalagens com yum

- Para remover um pacote em particular e todos os pacotes dependentes, use:

```
# yum remover package-name
```

  Substitua *package-name* pelo nome do pacote.

- Para remover vários pacotes e suas dependências simultaneamente, use:

```
# yum remover package-name-1 package-name-2
```

  Substituir *package-name-1* e *package-name-2* pelos nomes dos pacotes.

NOTA

yum não é capaz de remover um pacote sem remover pacotes dependentes.

Observe que você pode otimizar a pesquisa de pacotes definindo explicitamente como analisar o argumento. Veja Seção 2.6.3, “Especificação de um nome de pacote na entrada do yum” para mais detalhes.

2.6.2. Remoção de um grupo de embalagens com yum

- Para remover um grupo de pacotes pelo nome do grupo, use:

```
# grupo yum remover group-name
```

  Ou

```
# yum remove @group-name
```

  Substituir *group-name* pelo nome completo do grupo.

- Para remover um grupo de pacotes pelo groupID, use:

```
# grupo yum remover groupID
```

  Substituir *groupID* pela identificação do grupo.

2.6.3. Especificação de um nome de pacote na entrada do yum

Para otimizar o processo de instalação e remoção, você pode anexar os sufixos **-n**, **-na**, ou **-nerva** aos comandos **yum install** e **yum remove** para definir explicitamente como analisar um argumento:

- Para instalar um pacote usando o seu nome exato, use:

```
# yum install-n name
```

  Substituir *name* com o nome exato do pacote.

- Para instalar um pacote usando seu nome exato e arquitetura, use:

```
# yum install-na name.architecture
```

  Substituir *name* e *architecture* pelo nome exato e arquitetura do pacote.

- Para instalar um pacote usando seu nome exato, época, versão, lançamento e arquitetura, use:

```
# yum install-nevra name-epoch:version-release.architecture
```

  Substituir *name*, *epoch*, *version*, *release*, e *architecture* pelo nome exato, época, versão, lançamento, e arquitetura do pacote.

2.7. GERENCIAMENTO DE GRUPOS DE PACOTES DE SOFTWARE

Um grupo de pacotes é uma coleção de pacotes que servem a um propósito comum (**System Tools**, **Sound and Video**). A instalação de um grupo de pacotes puxa um conjunto de pacotes dependentes, o que economiza tempo consideravelmente.

A seção seguinte descreve como usar **yum** para:

- Listar grupos de pacotes.
- Instalar um grupo de pacotes.
- Remover um grupo de pacotes.
- Especificar as expressões globais na entrada do yum.

2.7.1. Listagem de grupos de pacotes com yum

- Para ver o número de grupos instalados e disponíveis, use:

```
# resumo do grupo yum
```

- Para listar todos os grupos instalados e disponíveis, use:

```
# lista do grupo yum
```

  Observe que você pode filtrar os resultados anexando opções de linha de comando para o comando **yum group list** (**--hidden**, **--available**). Para mais opções disponíveis, veja as páginas de manual.

- Para listar os pacotes obrigatórios e opcionais contidos em um determinado grupo, use:

```
# informações sobre o grupo yum group-name
```

  Substituir *group-name* pelo nome do grupo.

Observe que você pode filtrar os resultados anexando expressões globais como argumentos. Consulte Seção 2.7.4, “Especificação de expressões globais na entrada do yum” para obter mais detalhes.

2.7.2. Instalação de um grupo de embalagens com yum

- Para instalar um grupo de pacotes por um nome de grupo, use:

```
# instalação do grupo yum group-name
```

Ou

```
# yum install @group-name
```

Substituir *group-name* pelo nome completo do grupo ou grupo ambiental.

- Para instalar um grupo de pacotes pelo groupID, use:

```
# instalação do grupo yum groupID
```

Substituir *groupID* pela identificação do grupo.

2.7.3. Remoção de um grupo de embalagens com yum

- Para remover um grupo de pacotes pelo nome do grupo, use:

```
# grupo yum remover group-name
```

Ou

```
# yum remove @group-name
```

Substituir *group-name* pelo nome completo do grupo.

- Para remover um grupo de pacotes pelo groupID, use:

```
# grupo yum remover groupID
```

Substituir *groupID* pela identificação do grupo.

2.7.4. Especificação de expressões globais na entrada do yum

os comandos**yum** permitem filtrar os resultados, anexando um ou mais *glob expressions* como argumentos. Expressões globais devem ser evitadas quando passadas como argumentos para o comando **yum**. Para garantir que as expressões globais sejam passadas para **yum** como pretendido, use um dos seguintes métodos:

- Dupla ou única citação de toda a expressão global.

```
# yum fornece "file-name
```

Substituir *file-name* pelo nome do arquivo.

- Escape dos caracteres curinga precedendo-os com uma contrabarra (\).

```
# yum fornece {\i1}/file-name
```

Substituir *file-name* pelo nome do arquivo.

2.8. HISTÓRICO DE GERENCIAMENTO DE PACOTES DE MANUSEIO

O comando **yum history** permite a você rever informações sobre a linha do tempo de **yum** transações, datas e horários em que ocorreram, o número de pacotes afetados, se essas transações foram bem-sucedidas ou abortadas e se o banco de dados RPM foi alterado entre as transações. O comando **yum history** também pode ser usado para desfazer ou refazer as transações.

A seção seguinte descreve como usar **yum** para:

- Relacione as transações.
- Reverter transações.
- Repetir transações.
- Especificar as expressões globais na entrada do yum.

2.8.1. Listagem das transações com yum

- Para exibir uma lista de todas as últimas **yum** transações, uso:

```
# história do yum
```

- Para exibir uma lista de todas as últimas operações para um pacote selecionado, use:

```
# lista do histórico do yum package-name
```

Substitua *package-name* pelo nome do pacote. Você pode filtrar a saída do comando anexando expressões globais. Veja Seção 2.8.4, “Especificação de expressões globais na entrada do yum” para mais detalhes.

- Para examinar uma transação em particular, use:

```
# Informação sobre a história do yum transactionID
```

Substituir *transactionID* pela identificação da transação.

2.8.2. Revertendo transações com yum

- Para reverter uma determinada transação, use:

```
# yum history undo transactionID
```

Substituir *transactionID* pela identificação da transação.

- Para reverter a última transação, use:

```
# yum história desfazer último
```

Observe que o comando **yum history undo** reverte apenas os passos que foram executados durante a transação. Se a transação instalou um novo pacote, o comando **yum history undo** o desinstala. Se a transação desinstalou um pacote, o comando **yum history undo** o reinstala. **yum history undo** também tenta baixar todos os pacotes atualizados para suas versões anteriores, caso os pacotes mais antigos ainda estejam disponíveis.

2.8.3. Repetição de transações com yum

- Para repetir uma transação em particular, use:

  ```
  # história do yum refazer transactionID
  ```

  Substituir *transactionID* pela identificação da transação.

- Para repetir a última transação, use:

  ```
  # história do yum redo last
  ```

Note que o comando **yum history redo** apenas repete os passos que foram executados durante a transação.

2.8.4. Especificação de expressões globais na entrada do yum

os comandos**yum** permitem filtrar os resultados, anexando um ou mais *glob expressions* como argumentos. Expressões globais devem ser evitadas quando passadas como argumentos para o comando **yum**. Para garantir que as expressões globais sejam passadas para **yum** como pretendido, use um dos seguintes métodos:

- Dupla ou única citação de toda a expressão global.

  ```
  # yum fornece "file-name
  ```

  Substituir *file-name* pelo nome do arquivo.

- Escape dos caracteres curinga precedendo-os com uma contrabarra (\).

  ```
  # yum fornece {\i1}/file-name
  ```

  Substituir *file-name* pelo nome do arquivo.

2.9. GERENCIAMENTO DE REPOSITÓRIOS DE SOFTWARE

As informações de configuração para **yum** e utilidades relacionadas são armazenadas no arquivo **/etc/yum.conf**. Este arquivo contém um ou mais **[*repository*]** que lhe permitem definir opções específicas de repositório.

Recomenda-se definir repositórios individuais em arquivos novos ou existentes do **.repo** no diretório **/etc/yum.repos.d/**.

Observe que os valores que você define em cada **[*repository*]** das seções do arquivo **/etc/yum.conf** sobrepõem-se aos valores definidos na seção **[main]**.

A seção seguinte descreve como fazê-lo:

- Conjunto **[*repository*]** opções.
- Adicione um **yum** repositório.
- Habilitar um **yum** repositório.
- Desativar um **yum** repositório.

2.9.1. Definição de opções de repositório yum

O arquivo de configuração /**etc**/**yum.conf** contém o **[*repository*]** onde *repository* é uma identificação única de repositório. O **[*repository*]** permite que você defina seções individuais **yum** repositórios.

NOTA

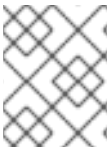
Não dê nomes de repositórios personalizados usados pelos repositórios da Red Hat para evitar conflitos.

Para uma lista completa dos disponíveis **[*repository*]** opções, veja as **[*repository*] OPTIONS** da seção **yum.conf**(5) da página do manual.

2.9.2. Adicionando um repositório yum

Para definir um novo repositório, você pode:

- Adicione um **[*repository*]** para o arquivo /**etc**/**yum.conf**.
- Adicione um **[*repository*]** para um arquivo **.repo** no diretório /**etc**/**yum.repos.d**/. **yum** os repositórios geralmente fornecem seu próprio arquivo **.repo**.



NOTA

Recomenda-se definir seus repositórios em um arquivo **.repo** ao invés de /**etc**/**yum.conf**, pois todos os arquivos com a extensão **.repo** neste diretório são lidos por **yum**.

- Para adicionar um repositório ao seu sistema e habilitá-lo, use:

```
# yum-config-manager --add-repo repository_URL
```

Substituir *repository_url* por uma URL que aponte para o repositório.

ATENÇÃO

A obtenção e instalação de pacotes de software de fontes não verificadas ou não confiáveis que não sejam baseadas em certificados da Red Hat **Content Delivery Network** (**CDN**) constitui um risco potencial de segurança e pode levar a problemas de segurança, estabilidade, compatibilidade e capacidade de manutenção.

2.9.3. Possibilitando um repositório yum

- Para habilitar um repositório, use:

```
# yum-config-manager --enable repositoryID
```

Substitua *repositoryID* pela identificação única do repositório.

Para listar as IDs de repositório disponíveis, veja Seção 2.3.2, “Listagem de pacotes com yum” .

2.9.4. Desabilitando um repositório yum

- Para desativar um repositório yum, use:

```
# yum-config-manager --disable repositoryID
```

Substitua *repositoryID* pela identificação única do repositório.

Para listar as IDs de repositório disponíveis, veja Seção 2.3.2, “Listagem de pacotes com yum” .

2.10. CONFIGURAÇÃO DO YUM

As informações de configuração para **yum** e utilidades relacionadas são armazenadas no arquivo **/etc/yum.conf**. Este arquivo contém uma seção obrigatória **[main]**, que permite definir **yum** opções que têm efeito global.

A seção seguinte descreve como fazê-lo:

- Veja as configurações atuais do **yum**.
- Conjunto **yum** [principal] opções.
- Use **yum** plug-ins.

2.10.1. Visualizando as configurações atuais do yum

- Para exibir os valores atuais das opções globais de yum especificados na seção **[main]** do arquivo **/etc/yum.conf**, use:

```
# yum config-manager --dump
```

2.10.2. Configuração das opções principais do yum

O arquivo de configuração **/etc/yum.conf** contém uma seção **[main]**. Os pares de valores-chave nesta seção afetam como **yum** opera e trata os repositórios.

Você pode adicionar opções adicionais sob o título da seção **[main]** em **/etc/yum.conf**.

Para uma lista completa das opções disponíveis em **[main]**, consulte a seção **[main] OPTIONS** da página do manual **yum.conf**(5).

2.10.3. Usando yum plug-ins

yum fornece plug-ins que ampliam e melhoram suas operações. Alguns plug-ins são instalados por padrão.

A seção seguinte descreve como habilitar, configurar e desabilitar **yum** plug-ins.

2.10.3.1. Gestão de plug-ins yum

Os arquivos de configuração do plug-in sempre contêm uma seção **[main]** onde a opção **enabled=** controla se o plug-in está habilitado quando você executa os comandos **yum**. Se esta opção estiver faltando, você pode adicioná-la manualmente ao arquivo.

Cada plug-in instalado tem seu próprio arquivo de configuração no diretório **/etc/dnf/plugins/**. Você pode ativar ou desativar as opções específicas de plug-in nestes arquivos.

2.10.3.2. Habilitação de plug-ins de yum

- Para habilitar todos os plug-ins de yum:

  1. Assegure-se de que uma linha começando com **plugins=** esteja presente na seção **[main]** do arquivo **/etc/yum.conf**.

  2. Defina o valor de **plugins=** para **1**.

     ```
     plugins=1
     ```

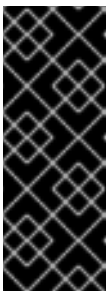
2.10.3.3. Desativação de plug-ins de yum

- Para desativar todos os plug-ins de yum:

  1. Assegure-se de que uma linha começando com **plugins=** esteja presente na seção **[main]** do arquivo **/etc/yum.conf**.

  2. Defina o valor de **plugins=** para **0**.

     ```
     plugins=0
     ```



IMPORTANTE

A desativação de todos os plug-ins é recomendada em **not**. Alguns plug-ins oferecem importantes serviços de yum. Em particular, o **product-id** e **subscription-manager** plug-ins fornecem suporte para o certificado baseado em **Content Delivery Network** (**CDN**). A desativação de plug-ins globalmente é fornecida como uma opção de conveniência, e é aconselhável apenas ao diagnosticar um problema potencial com **yum**.

- Para desativar todos os yum plug-ins para um determinado comando, anexar a opção **--noplugins** ao comando.

  ```
  # yum -- atualização dos plugins
  ```

- Para desativar certos yum plug-ins para um único comando, anexar **--disableplugin=*plugin-name*** opção para o comando.

  ```
  # yum update --disableplugin=plugin-name
  ```

Substitua *plugin-name* pelo nome do plug-in.

CAPÍTULO 3. GERENCIAMENTO DE SERVIÇOS COM SYSTEMD

3.1. INTRODUÇÃO AO SISTEMAD

Systemd é um gerente de sistemas e serviços para sistemas operacionais Linux. Ele foi projetado para ser retrocompatível com scripts de inicialização SysV e fornece uma série de recursos, tais como inicialização paralela de serviços de sistema no momento da inicialização, ativação de daemons sob demanda, ou lógica de controle de serviços baseada em dependência. Começando com o Red Hat Enterprise Linux 7, **systemd** substituiu Upstart como o sistema init padrão.

Systemd introduz o conceito de *systemd units*. Estas unidades são representadas por arquivos de configuração de unidades localizados em um dos diretórios listados na tabela a seguir.

Tabela 3.1. Localização dos arquivos da unidade Systemd

Diretório	Descrição
/usr/lib/systemd/system/	Arquivos da unidade Systemd distribuídos com pacotes de RPM instalados.
/run/systemd/system/	Arquivos da unidade Systemd criados em tempo de execução. Este diretório tem precedência sobre o diretório com os arquivos das unidades de serviço instaladas.
/etc/systemd/system/	Arquivos de unidades do sistema criados por **systemctl enable**, bem como arquivos de unidades adicionados para ampliar um serviço. Este diretório tem precedência sobre o diretório com arquivos unitários de tempo de execução.

As unidades encapsulam informações sobre:

- Serviços de sistema
- Tomadas de escuta
- Outros objetos que são relevantes para o sistema init

Para uma lista completa dos tipos de unidades do sistema disponíveis, consulte a tabela a seguir.

Tabela 3.2. Tipos de unidades do sistema disponíveis

Tipo de unidade	Extensão do arquivo	Descrição
Unidade de serviço	**.service**	Um serviço de sistema.
Unidade alvo	**.target**	Um grupo de unidades do sistema.

Tipo de unidade	Extensão do arquivo	Descrição
Unidade de montagem automática	**.automount**	Um ponto de montagem automática do sistema de arquivo.
Unidade do dispositivo	**.device**	Um arquivo de dispositivo reconhecido pelo kernel.
Unidade de montagem	**.mount**	Um ponto de montagem do sistema de arquivo.
Unidade de caminho	**.path**	Um arquivo ou diretório em um sistema de arquivos.
Unidade de escopo	**.scope**	Um processo criado externamente.
Unidade de fatias	**.slice**	Um grupo de unidades hierarquicamente organizadas que gerenciam os processos do sistema.
Unidade de soquetes	**.socket**	Uma tomada de comunicação inter-processo.
Unidade de troca	**.swap**	Um dispositivo swap ou um arquivo swap.
Unidade do temporizador	**.timer**	Um temporizador do sistema.

Substituindo a configuração padrão do systemd usando system.conf
A configuração padrão de **systemd** é definido durante a compilação e pode ser encontrado no arquivo de configuração do sistema em **/etc/systemd/system.conf**. Use este arquivo se você quiser se desviar desses valores padrão e substituir os valores padrão selecionados para unidades systemd globalmente.

Por exemplo, para anular o valor padrão do limite de tempo limite, que está definido para 90 segundos, use o parâmetro **DefaultTimeoutStartSec** para inserir o valor requerido em segundos.

```
DefaultTimeoutStartSec=required value
```

Para maiores informações, ver Exemplo 3.11, “Alteração do limite de tempo limite” .

3.1.1. Principais características

O sistema e o gerente de serviços do sistema fornecem as seguintes características principais:

- **Socket-based activation** - No momento da inicialização, **systemd** cria tomadas de escuta para todos os serviços do sistema que suportam este tipo de ativação, e passa as tomadas para estes

serviços assim que são iniciadas. Isto não só permite **systemd** para iniciar serviços em paralelo, mas também possibilita reiniciar um serviço sem perder nenhuma mensagem enviada a ele enquanto ele não estiver disponível: o soquete correspondente permanece acessível e todas as mensagens são enfileiradas.
Systemd usa *socket units* para ativação baseada em soquete.

- **Bus-based activation** - Os serviços de sistema que utilizam o D-Bus para comunicação entre processos podem ser iniciados sob demanda na primeira vez que uma aplicação do cliente tenta se comunicar com eles **Systemd** usa *D-Bus service files* para ativação baseada em ônibus.

- **Device-based activation** - Os serviços de sistema que suportam a ativação baseada em dispositivos podem ser iniciados sob demanda quando um determinado tipo de hardware é conectado ou fica disponível **Systemd** usa *device units* para a ativação baseada em dispositivo.

- **Path-based activation** - Os serviços de sistema que suportam ativação baseada em caminho podem ser iniciados sob demanda quando um determinado arquivo ou diretório muda seu estado **Systemd** usa *path units* para ativação baseada em caminho.

- **Mount and automount point management** - **Systemd** monitora e gerencia os pontos de montagem e montagem automática **Systemd** usa *mount units* para pontos de montagem e *automount units* para pontos de montagem automática.

- **Aggressive parallelization** - Por causa do uso de ativação baseada em soquetes, **systemd** pode iniciar os serviços do sistema em paralelo assim que todas as tomadas de escuta estiverem prontas. Em combinação com os serviços de sistema que suportam a ativação sob demanda, a ativação paralela reduz significativamente o tempo necessário para inicializar o sistema.

- **Transactional unit activation logic** - Antes de ativar ou desativar uma unidade, **systemd** calcula suas dependências, cria uma transação temporária e verifica se esta transação é consistente. Se uma transação for inconsistente, **systemd** automaticamente tenta corrigi-lo e remover trabalhos não essenciais dele antes de relatar um erro.

- **Backwards compatibility with SysV init** - **Systemd** suporta scripts de inicialização SysV, conforme descrito no *Linux Standard Base Core Specification* que facilita o caminho de atualização para as unidades de serviço do sistema.

3.1.2. Mudanças de compatibilidade

O sistema de sistema e o gerente de serviços foi projetado para ser na maior parte compatível com o SysV init e Upstart. A seguir estão as alterações de compatibilidade mais notáveis com relação ao sistema Red Hat Enterprise Linux 6 que usou o SysV init:

- **Systemd** tem apenas um apoio limitado para os níveis de execução. Ele fornece um número de unidades-alvo que podem ser mapeadas diretamente para esses níveis de execução e, por razões de compatibilidade, também é distribuído com o comando **runlevel** anterior. Nem todos os alvos do sistema podem ser mapeados diretamente para runlevels e, como conseqüência, este comando pode retornar **N** para indicar um runlevel desconhecido. Recomenda-se evitar o uso do comando **runlevel**, se possível. Para mais informações sobre alvos do sistema e sua comparação com runlevels, veja Seção 3.3, “Trabalhando com metas do sistema” .

- O utilitário **systemctl** não suporta comandos personalizados. Além dos comandos padrão como **start**, **stop** e **status**, os autores de scripts de inicialização SysV poderiam implementar suporte para qualquer número de comandos arbitrários a fim de fornecer funcionalidade adicional. Por exemplo, o script de inicialização para **iptables** poderia ser executado com o comando **panic**,

que imediatamente ativou o modo de pânico e reconfigurou o sistema para começar a descartar todos os pacotes de entrada e de saída. Isto não é suportado em **systemd** e o **systemctl** só aceita comandos documentados.
Para mais informações sobre o utilitário **systemctl** e sua comparação com o anterior **service**, ver Tabela 3.3, "Comparação da utilidade do serviço com o systemctl" .

- O serviço **systemctl** não se comunica com serviços que não tenham sido iniciados por **systemd**. Quando **systemd** inicia um serviço de sistema, ele armazena a identificação de seu processo principal, a fim de acompanhar o andamento do mesmo. O utilitário **systemctl** então usa este PID para consultar e gerenciar o serviço. Conseqüentemente, se um usuário inicia um determinado daemon diretamente na linha de comando, **systemctl** é incapaz de determinar seu status atual ou pará-lo.

- **Systemd** deixa de executar apenas serviços. Anteriormente, quando a seqüência de desligamento foi iniciada, o Red Hat Enterprise Linux 6 e versões anteriores do sistema usavam links simbólicos localizados no diretório **/etc/rc0.d/** para parar todos os serviços de sistema disponíveis, independentemente de seu status. Com **systemd** Só os serviços em funcionamento são interrompidos ao serem encerrados.

- Os serviços do sistema não conseguem ler a partir do fluxo de entrada padrão. Quando **systemd** inicia um serviço, ele conecta sua entrada padrão a **/dev/null** para evitar qualquer interação com o usuário.

- Os serviços de sistema não herdam nenhum contexto (como as variáveis de ambiente **HOME** e **PATH**) do usuário invocador e sua sessão. Cada serviço é executado em um contexto de execução limpa.

- Ao carregar um roteiro de inicialização do SysV, **systemd** lê as informações de dependência codificadas no cabeçalho da Base Padrão Linux (LSB) e as interpreta em tempo de execução.

- Todas as operações em unidades de serviço estão sujeitas a um timeout padrão de 5 minutos para evitar que um mau funcionamento do serviço congele o sistema. Este valor é codificado para serviços que são gerados a partir de initscripts e não podem ser alterados. Entretanto, arquivos de configuração individuais podem ser usados para especificar um valor de timeout mais longo por serviço, veja Exemplo 3.11, "Alteração do limite de tempo limite" .

Para uma lista detalhada das mudanças de compatibilidade introduzidas com **systemd**Veja o Guia de Planejamento de Migração para o Red Hat Enterprise Linux 7.

3.2. GERENCIAMENTO DE SERVIÇOS DE SISTEMA

As versões anteriores do Red Hat Enterprise Linux, que eram distribuídas com SysV init ou Upstart, usavam *init scripts* localizado no diretório **/etc/rc.d/init.d/**. Estes scripts init eram tipicamente escritos em Bash, e permitiam ao administrador de sistemas controlar o estado dos serviços e daemons em seu sistema. Começando com o Red Hat Enterprise Linux 7, estes scripts de inicialização foram substituídos por *service units*.

As unidades de serviço terminam com a extensão do arquivo **.service** e servem a um propósito similar ao dos scripts de inicialização. Para visualizar, iniciar, parar, reiniciar, ativar ou desativar serviços de sistema, use o comando **systemctl** como descrito em Comparação do utilitário de serviço com systemctl, Comparação do utilitário chkconfig com systemctl , e mais adiante nesta seção. Os comandos **service** e **chkconfig** ainda estão disponíveis no sistema e funcionam como esperado, mas só estão incluídos por razões de compatibilidade e devem ser evitados.

Tabela 3.3. Comparação da utilidade do serviço com o systemctl

serviço	systemctl	Descrição
service *name* start	**systemctl start *name*.service**	Inicia um serviço.
service *name* stop	**systemctl stop *name*.service**	Interrompe um serviço.
service *name* restart	**systemctl restart *name*.service**	Recomeça um serviço.
service *name* condrestart	**systemctl try-restart *name*.service**	Reinicia um serviço somente se ele estiver em funcionamento.
service *name* reload	**systemctl reload *name*.service**	Recarregar a configuração.
service *name* status	**systemctl status *name*.service** **systemctl is-active *name*.service**	Verifica se um serviço está funcionando.
service --status-all	**systemctl list-units --type service --all**	Exibe o status de todos os serviços.

Tabela 3.4. Comparação do utilitário chkconfig com o systemctl

chkconfig	systemctl	Descrição
chkconfig *name* on	**systemctl enable *name*.service**	Possibilita um serviço.
chkconfig *name* off	**systemctl disable *name*.service**	Desabilita um serviço.
chkconfig --list *name*	**systemctl status *name*.service** **systemctl is-enabled *name*.service**	Verifica se um serviço está habilitado.
chkconfig --list	**systemctl list-unit-files --type service**	Relaciona todos os serviços e verifica se eles estão habilitados.
chkconfig --list	**systemctl list-dependencies --after**	Lista os serviços que são ordenados para começar antes da unidade especificada.

chkconfig	systemctl	Descrição
chkconfig --list	**systemctl list-dependencies --before**	Lista os serviços que são encomendados para começar após a unidade especificada.

Especificação de unidades de serviço

Para maior clareza, todos os exemplos de comando no restante desta seção usam nomes completos de unidades com a extensão **.service**, por exemplo:

```
# systemctl stop nfs-server.service
```

Entretanto, a extensão do arquivo pode ser omitida, caso em que o utilitário **systemctl** assume que o argumento é uma unidade de serviço. O seguinte comando é equivalente ao acima:

```
# systemctl stop nfs-server
```

Além disso, algumas unidades têm nomes falsos. Esses nomes podem ter nomes mais curtos do que as unidades, que podem ser usados em vez dos nomes reais das unidades. Para encontrar todos os pseudônimos que podem ser usados para uma unidade em particular, use:

```
# systemctl show nfs-server.service -p Nomes
```

Comportamento do systemctl em um ambiente chroot

Se você mudar o diretório raiz usando o comando **chroot**, a maioria dos comandos **systemctl** se recusam a executar qualquer ação. A razão para isto é que o processo **systemd** e o usuário que usou o comando **chroot** não têm a mesma visão do sistema de arquivos. Isto acontece, por exemplo, quando **systemctl** é invocado a partir de um arquivo **kickstart**.

A exceção a isto são os comandos de arquivo de unidade como os comandos **systemctl enable** e **systemctl disable**. Estes comandos não precisam de um sistema em execução e não afetam os processos em execução, mas afetam os arquivos de unidade. Portanto, você pode executar estes comandos mesmo em ambiente **chroot**. Por exemplo, para habilitar o serviço **httpd** em um sistema sob o diretório /**srv**/**website1**/:

```
# chroot /srv/website1
# systemctl enable httpd.service
Created symlink /etc/systemd/system/multi-user.target.wants/httpd.service, pointing to
/usr/lib/systemd/system/httpd.service.
```

3.2.1. Serviços de listagem

Para listar todas as unidades de serviço atualmente carregadas, digite o seguinte em um prompt de shell:

```
systemctl list-units --type service
```

Para cada arquivo de unidade de serviço, este comando exibe seu nome completo (**UNIT**) seguido de uma nota se o arquivo da unidade foi carregado (**LOAD**), seu alto nível (**ACTIVE**) e baixo nível (**SUB**) estado de ativação do arquivo da unidade, e uma breve descrição (**DESCRIPTION**).

Por padrão, o comando **systemctl list-units** exibe apenas as unidades ativas. Se você quiser listar todas as unidades carregadas independentemente de seu estado, execute este comando com a opção de linha de comando **--all** ou **-a**:

```
systemctl list-units --type service --all
```

Você também pode listar todas as unidades de serviço disponíveis para ver se elas estão habilitadas. Para fazer isso, digite:

```
systemctl list-unit-files --type service
```

Para cada unidade de serviço, este comando exibe seu nome completo (**UNIT FILE**) seguido de informações se a unidade de serviço está habilitada ou não (**STATE**). Para informações sobre como determinar o status de cada unidade de serviço, consulte Exibindo o status do serviço.

Exemplo 3.1. Serviços de listagem

Para listar todas as unidades de serviço atualmente carregadas, execute o seguinte comando:

```
$ systemctl list-units --type service
UNIT                    LOAD   ACTIVE SUB     DESCRIPTION
abrt-ccpp.service       loaded active exited  Install ABRT coredump hook
abrt-oops.service       loaded active running ABRT kernel log watcher
abrt-vmcore.service     loaded active exited  Harvest vmcores for ABRT
abrt-xorg.service       loaded active running ABRT Xorg log watcher
abrtd.service           loaded active running ABRT Automated Bug Reporting Tool
…
systemd-vconsole-setup.service loaded active exited  Setup Virtual Console
tog-pegasus.service     loaded active running OpenPegasus CIM Server

LOAD   = Reflects whether the unit definition was properly loaded.
ACTIVE = The high-level unit activation state, i.e. generalization of SUB.
SUB    = The low-level unit activation state, values depend on unit type.

46 loaded units listed. Pass --all to see loaded but inactive units, too.
To show all installed unit files use 'systemctl list-unit-files'
```

Para listar todos os arquivos da unidade de serviço instalada para determinar se eles estão habilitados, digite:

```
$ systemctl list-unit-files --type service
UNIT FILE                              STATE
abrt-ccpp.service                      enabled
abrt-oops.service                      enabled
abrt-vmcore.service                    enabled
abrt-xorg.service                      enabled
abrtd.service                          enabled
…
wpa_supplicant.service                 disabled
ypbind.service                         disabled

208 unit files listed.
```

3.2.2. Exibição do status do serviço

Para exibir informações detalhadas sobre uma unidade de serviço que corresponda a um serviço do sistema, digite o seguinte em uma janela de comandos:

```
systemctl status name.service
```

Substitua *name* pelo nome da unidade de serviço que você deseja inspecionar (por exemplo, **gdm**). Este comando exibe o nome da unidade de serviço selecionada seguido de sua breve descrição, um ou mais campos descritos em Tabela 3.5, “Informações sobre a unidade de serviço disponível” , e se for executado pelo usuário **root**, também as entradas de registro mais recentes.

Tabela 3.5. Informações sobre a unidade de serviço disponível

Campo	Descrição
Loaded	Informação se a unidade de serviço foi carregada, o caminho absoluto para o arquivo da unidade e uma nota se a unidade está habilitada.
Active	Informação se a unidade de serviço está funcionando, seguida por um carimbo de tempo.
Main PID	O PID do serviço correspondente do sistema seguido de seu nome.
Status	Informações adicionais sobre o serviço de sistema correspondente.
Process	Informações adicionais sobre os processos relacionados.
CGroup	Informações adicionais sobre os Grupos de Controle relacionados (cgroups).

Para verificar apenas se uma determinada unidade de serviço está funcionando, execute o seguinte comando:

```
systemctl is-active name.service
```

Da mesma forma, para determinar se uma determinada unidade de serviço está habilitada, digite:

```
systemctl is-enabled name.service
```

Observe que tanto **systemctl is-active** quanto **systemctl is-enabled** retornam um status de saída de **0** se a unidade de serviço especificada estiver em funcionamento ou habilitada. Para informações sobre como listar todas as unidades de serviço atualmente carregadas, consulte Listagem de serviços .

Exemplo 3.2. Exibição do status do serviço

A unidade de serviço para o Gerenciador de Monitores do GNOME é denominada **gdm.service**. Para determinar o status atual desta unidade de serviço, digite o seguinte em uma janela de comandos:

```
# systemctl status gdm.service
gdm.service - GNOME Display Manager
   Loaded: loaded (/usr/lib/systemd/system/gdm.service; enabled)
   Active: active (running) since Thu 2013-10-17 17:31:23 CEST; 5min ago
 Main PID: 1029 (gdm)
   CGroup: /system.slice/gdm.service
           ├─1029 /usr/sbin/gdm
           ├─1037 /usr/libexec/gdm-simple-slave --display-id /org/gno…
           └─1047 /usr/bin/Xorg :0 -background none -verbose -auth /r…

Oct 17 17:31:23 localhost systemd[1]: Started GNOME Display Manager.
```

Exemplo 3.3. Exibição de serviços encomendados antes de um serviço

Para determinar quais serviços são pedidos para começar antes do serviço especificado, digite o seguinte em um prompt de shell:

```
# systemctl list-dependencies --after gdm.service
gdm.service
├─dbus.socket
├─getty@tty1.service
├─livesys.service
├─plymouth-quit.service
├─system.slice
├─systemd-journald.socket
├─systemd-user-sessions.service
└─basic.target
[output truncated]
```

Exemplo 3.4. Exibição de serviços encomendados para começar após um serviço

Para determinar quais serviços são pedidos para começar após o serviço especificado, digite o seguinte em um prompt de shell:

```
# systemctl list-dependencies --before gdm.service
gdm.service
├─dracut-shutdown.service
├─graphical.target
│ ├─systemd-readahead-done.service
│ ├─systemd-readahead-done.timer
│ └─systemd-update-utmp-runlevel.service
└─shutdown.target
  ├─systemd-reboot.service
  └─final.target
    └─systemd-reboot.service
```

3.2.3. Início de um serviço

Para iniciar uma unidade de serviço que corresponda a um serviço de sistema, digite o seguinte em um prompt de shell como **root**:

systemctl start *name*.service

Substitua *name* pelo nome da unidade de serviço que você deseja iniciar (por exemplo, **gdm**). Este comando inicia a unidade de serviço selecionada na sessão atual. Para informações sobre como habilitar uma unidade de serviço a ser iniciada no momento da inicialização, consulte Habilitação de um serviço. Para obter informações sobre como determinar o status de uma determinada unidade de serviço, consulte Exibindo o status do serviço.

Exemplo 3.5. Início de um serviço

A unidade de serviço para o Servidor HTTP Apache é denominada **httpd.service**. Para ativar esta unidade de serviço e iniciar o daemon **httpd** na sessão atual, execute o seguinte comando como **root**:

```
# systemctl start httpd.service
```

3.2.4. Interrupção de um serviço

Para parar uma unidade de serviço que corresponda a um serviço de sistema, digite o seguinte em um prompt de shell como **root**:

systemctl stop *name*.service

Substitua *name* pelo nome da unidade de serviço que você deseja parar (por exemplo, **bluetooth**). Este comando interrompe a unidade de serviço selecionada na sessão atual. Para informações sobre como desativar uma unidade de serviço e impedir que ela seja iniciada no momento da inicialização, consulte Desativação de um serviço . Para obter informações sobre como determinar o status de uma determinada unidade de serviço, consulte Exibindo o status do serviço.

Exemplo 3.6. Interrupção de um serviço

A unidade de serviço para o daemon **bluetoothd** é chamada **bluetooth.service**. Para desativar esta unidade de serviço e parar o daemon **bluetoothd** na sessão atual, execute o seguinte comando como **root**:

```
# systemctl stop bluetooth.service
```

3.2.5. Reinício de um serviço

Para reiniciar uma unidade de serviço que corresponda a um serviço de sistema, digite o seguinte em uma janela de comandos como **root**:

systemctl restart *name*.service

Substitua *name* pelo nome da unidade de serviço que você deseja reiniciar (por exemplo, **httpd**). Este comando interrompe a unidade de serviço selecionada na sessão atual e a reinicia imediatamente. É importante notar que, se a unidade de serviço selecionada não estiver em execução, este comando também a iniciará. Para dizer **systemd** para reiniciar uma unidade de serviço somente se o serviço correspondente já estiver em execução, execute o seguinte comando como **root**:

systemctl try-restart *name*.service

Certos serviços do sistema também permitem recarregar sua configuração sem interromper sua execução. Para fazer isso, digite como **root**:

```
systemctl reload name.service
```

Observe que os serviços do sistema que não suportam este recurso ignoram completamente este comando. Por conveniência, o comando **systemctl** também suporta os comandos **reload-or-restart** e **reload-or-try-restart** que, em seu lugar, reiniciam tais serviços. Para obter informações sobre como determinar o status de uma determinada unidade de serviço, consulte Exibindo o status do serviço.

Exemplo 3.7. Reinício de um serviço

A fim de evitar que os usuários encontrem mensagens de erro desnecessárias ou páginas web parcialmente renderizadas, o Servidor HTTP Apache permite editar e recarregar sua configuração sem a necessidade de reiniciá-la e interromper as solicitações processadas ativamente. Para fazer isso, digite o seguinte em uma janela de comandos como **root**:

```
# systemctl reload httpd.service
```

3.2.6. Permitindo um serviço

Para configurar uma unidade de serviço que corresponda a um serviço de sistema a ser iniciado automaticamente no momento da inicialização, digite o seguinte em uma janela de comandos como **root**:

```
systemctl enable name.service
```

Substitua *name* pelo nome da unidade de serviço que você deseja habilitar (por exemplo, **httpd**). Este comando lê a seção **[Install]** da unidade de serviço selecionada e cria links simbólicos apropriados para a **/usr/lib/systemd/system/*name*.service** no diretório **/etc/systemd/system/** e seus subdiretórios. Este comando não reescreve, no entanto, os links já existentes. Se você quiser garantir que os links simbólicos sejam recriados, use o seguinte comando como **root**:

```
systemctl reenable name.service
```

Este comando desabilita a unidade de serviço selecionada e a habilita imediatamente novamente. Para informações sobre como determinar se uma determinada unidade de serviço está habilitada para iniciar no momento da inicialização, consulte Exibição do status do serviço. Para obter informações sobre como iniciar um serviço na sessão atual, consulte Iniciando um serviço.

Exemplo 3.8. Permitindo um serviço

Para configurar o Servidor HTTP Apache para iniciar automaticamente no momento do boot, execute o seguinte comando como **root**:

```
# systemctl enable httpd.service
Created symlink from /etc/systemd/system/multi-user.target.wants/httpd.service to
/usr/lib/systemd/system/httpd.service.
```

3.2.7. Desabilitando um serviço

Para evitar que uma unidade de serviço que corresponde a um serviço de sistema seja automaticamente iniciada no momento da inicialização, digite o seguinte em uma janela de comandos como **root**:

```
systemctl disable name.service
```

Substitua *name* pelo nome da unidade de serviço que você deseja desativar (por exemplo, **bluetooth**). Este comando lê a seção **[Install]** da unidade de serviço selecionada e remove os links simbólicos apropriados para a **/usr/lib/systemd/system/*name*.service** do diretório **/etc/systemd/system/** e seus subdiretórios. Além disso, você pode mascarar qualquer unidade de serviço para evitar que ela seja iniciada manualmente ou por outro serviço. Para fazer isso, execute o seguinte comando como **root**:

```
systemctl mask name.service
```

Este comando substitui o **/etc/systemd/system/*name*.service** com um link simbólico para **/dev/null**, tornando o arquivo da unidade real inacessível a **systemd**. Para reverter esta ação e desmascarar uma unidade de serviço, digite como **root**:

```
systemctl unmask name.service
```

Para informações sobre como determinar se uma determinada unidade de serviço está habilitada para iniciar no momento da inicialização, consulte Exibição do status do serviço . Para obter informações sobre como parar um serviço na sessão atual, consulte Parando um serviço.

Exemplo 3.9. Desabilitando um serviço

Exemplo 3.6, “Interrupção de um serviço” ilustra como parar a unidade **bluetooth.service** na sessão atual. Para evitar que esta unidade de serviço comece no momento da inicialização, digite o seguinte em uma janela de comandos como **root**:

```
# systemctl disable bluetooth.service
Removed symlink /etc/systemd/system/bluetooth.target.wants/bluetooth.service.
Removed symlink /etc/systemd/system/dbus-org.bluez.service.
```

3.2.8. Iniciando um serviço conflituoso

Em **systemd**Existem dependências positivas e negativas entre os serviços. O início de um determinado serviço pode exigir o início de um ou mais serviços (dependência positiva) ou a interrupção de um ou mais serviços (dependência negativa).

Quando você tenta iniciar um novo serviço, **systemd** resolve automaticamente todas as dependências. Note que isto é feito sem notificação explícita ao usuário. Se você já estiver executando um serviço, e tentar iniciar outro serviço com uma dependência negativa, o primeiro serviço é automaticamente interrompido.

Por exemplo, se você estiver executando o serviço **postfix**, e tentar iniciar o serviço **sendmail**, **systemd** primeiro pára automaticamente **postfix**, porque estes dois serviços são conflitantes e não podem funcionar no mesmo porto.

3.3. TRABALHANDO COM METAS DO SISTEMA

Os alvos do sistema são representados por unidades-alvo. O arquivo de unidades alvo termina com a extensão **.target** e seu único objetivo é agrupar outras unidades do sistema através de uma cadeia de

dependências. Por exemplo, o **graphical.target unit**, que é usado para iniciar uma sessão gráfica, inicia serviços de sistema como o GNOME Display Manager **(gdm.service)** ou o Serviço de Contas **(accounts-daemon.service)** e também ativa o **multi-user.target unit**. Da mesma forma, a unidade multi-usuário.target inicia outros serviços essenciais do sistema como o NetworkManager **(NetworkManager.service)** ou o D-Bus **(dbus.service)** e ativa outra unidade alvo chamada basic.target.

Esta seção inclui procedimentos a serem implementados enquanto se trabalha com metas **systemd**.

3.3.1. Diferença entre os níveis de execução do SysV e as metas do sistema

As versões anteriores do Red Hat Enterprise Linux foram distribuídas com SysV init ou Upstart, e implementaram um conjunto pré-definido de níveis de execução que representavam modos específicos de operação. Estes níveis de execução foram numerados de 0 a 6 e foram definidos por uma seleção de serviços de sistema a serem executados quando um determinado nível de execução era habilitado pelo administrador de sistemas. Começando com o Red Hat Enterprise Linux 7, o conceito de níveis de execução foi substituído por alvos do sistema.

O Red Hat Enterprise Linux 7 foi distribuído com um número de alvos predefinidos que são mais ou menos similares ao conjunto padrão de níveis de execução dos lançamentos anteriores. Por razões de compatibilidade, ele também fornece alias para estes alvos que mapeiam diretamente para os níveis de execução do SysV.

A tabela a seguir fornece uma lista completa dos níveis de execução do SysV e seus correspondentes alvos do sistema:

Tabela 3.6. Comparação dos níveis de execução do SysV com as metas do sistema

Runlevel	Unidades Alvo	Descrição
0	**runlevel0.target**, **poweroff.target**	Desligue e desligue o sistema.
1	**runlevel1.target**, **rescue.target**	Preparar uma concha de salvamento.
2	**runlevel2.target**, **multi-user.target**	Configurar um sistema multi-usuário não gráfico.
3	**runlevel3.target**, **multi-user.target**	Configurar um sistema multi-usuário não gráfico.
4	**runlevel4.target**, **multi-user.target**	Configurar um sistema multi-usuário não gráfico.
5	**runlevel5.target**, **graphical.target**	Configurar um sistema gráfico multiusuário.
6	**runlevel6.target**, **reboot.target**	Desligar e reiniciar o sistema.

A tabela a seguir compara os comandos init SysV com systemctl. Use o utilitário systemctl para visualizar, alterar ou configurar alvos do sistema:

IMPORTANTE

Os comandos **runlevel** e **telinit** ainda estão disponíveis no sistema e funcionam como esperado, mas só estão incluídos por razões de compatibilidade e devem ser evitados.

Tabela 3.7. Comparação dos comandos init SysV com systemctl

Antigo Comando	Novo Comando	Descrição
runlevel	**systemctl list-units --type target**	Listas de unidades alvo atualmente carregadas.
telinit *runlevel*	**systemctl isolate *name*.target**	Muda a meta atual.

Recursos adicionais

- man sysv init
- o homem começa no init
- man systemctl

3.3.2. Visualizando o alvo padrão

A unidade alvo padrão é representada pelo arquivo **/etc/systemd/system/default.target**.

Procedimento

- Para determinar qual unidade alvo é usada por padrão:

```
$ systemctl get-default
graphical.target
```

- Para determinar o alvo padrão usando o link simbólico:

```
$  ls -l /lib/systemd/system/default.target
```

3.3.3. Visualizando as unidades-alvo

Por padrão, o comando **systemctl list-units** exibe apenas as unidades ativas.

Procedimento

- Para listar todas as unidades carregadas, independentemente de seu estado:

```
Unidades da lista do sistema -- tipo alvo -- tudo
```

- Para listar todas as unidades alvo atualmente carregadas:

```
$ systemctl list-units --type target

UNIT                   LOAD   ACTIVE SUB    DESCRIPTION
basic.target           loaded active active Basic System
cryptsetup.target      loaded active active Encrypted Volumes
getty.target           loaded active active Login Prompts
graphical.target       loaded active active Graphical Interface
local-fs-pre.target    loaded active active Local File Systems (Pre)
local-fs.target        loaded active active Local File Systems
multi-user.target      loaded active active Multi-User System
network.target         loaded active active Network
paths.target           loaded active active Paths
remote-fs.target       loaded active active Remote File Systems
sockets.target         loaded active active Sockets
sound.target           loaded active active Sound Card
spice-vdagentd.target loaded active active Agent daemon for Spice guests
swap.target            loaded active active Swap
sysinit.target         loaded active active System Initialization
time-sync.target       loaded active active System Time Synchronized
timers.target          loaded active active Timers

LOAD   = Reflects whether the unit definition was properly loaded.
ACTIVE = The high-level unit activation state, i.e. generalization of SUB.
SUB    = The low-level unit activation state, values depend on unit type.

17 loaded units listed.
```

3.3.4. Mudando o alvo padrão

A unidade alvo padrão é representada pelo arquivo **/etc/systemd/system/default.target**. O procedimento a seguir descreve como alterar o alvo padrão usando o comando systemctl:

Procedimento

1. Para determinar a unidade alvo padrão:

```
# sistema de falha
```

2. Para configurar o sistema para usar uma unidade alvo diferente por padrão:

```
# systemctl set-default multi-user.target
rm /etc/systemd/system/default.target
ln -s /usr/lib/systemd/system/multi-user.target /etc/systemd/system/default.target
```

   Este comando substitui o arquivo **/etc/systemd/system/default.target** por um link simbólico para **/usr/lib/systemd/system/name.target**, onde o nome é o nome da unidade alvo que você deseja usar. Substitua *multi-user* pelo nome da unidade alvo que você deseja usar por padrão.

3. Reinicialização

```
# reinicialização
```

3.3.5. Mudando o alvo padrão usando um link simbólico

O procedimento a seguir descreve como alterar o alvo padrão criando um vínculo simbólico com o alvo.

Procedimento

1. Para determinar a unidade alvo padrão:

```
# ls /lib/systemd/system/default.target -l
```

2. Para criar um vínculo simbólico:

```
# ln -sf /lib/systemd/system/graphical.target /etc/systemd/system/default.target
```

3. Reinicie o sistema:

```
# reinicialização
```

Etapas de verificação

- Verificar o recém-criado alvo.default.target:

```
$ systemctl get-default
multi-user.target
```

3.3.6. Mudando a meta atual

Este procedimento explica como mudar a unidade alvo na sessão atual usando o comando systemctl.

Procedimento

- Para mudar para uma unidade alvo diferente na sessão atual:

```
# systemctl isolar multi-user.target
```

Este comando inicia a unidade alvo chamada *multi-user* e todas as unidades dependentes, e imediatamente pára todas as outras.

Substitua *multi-user* pelo nome da unidade alvo que você deseja usar por padrão.

Etapas de verificação

- Verificar o recém-criado alvo.default.target:

```
$ systemctl get-default
multi-user.target
```

3.3.7. Modo de inicialização para resgate

Rescue mode fornece um ambiente conveniente de usuário único e permite reparar seu sistema em situações em que ele não é capaz de completar um processo de inicialização regular. No modo de resgate, o sistema tenta montar todos os sistemas de arquivos locais e iniciar alguns serviços

importantes do sistema, mas não ativa as interfaces de rede ou permite que mais usuários sejam logados no sistema ao mesmo tempo.

Procedimento

- Para mudar o alvo atual e entrar no modo de resgate na sessão atual:

```
# systemctl rescue

Broadcast message from root@localhost on pts/0 (Fri 2013-10-25 18:23:15 CEST):

The system is going down to rescue mode NOW!
```

Este comando é similar a **systemctl isolate rescue.target**, mas também envia uma mensagem informativa a todos os usuários que estão atualmente logados no sistema.

Para evitar que **systemd** envie uma mensagem, execute o seguinte comando com a opção de linha de comando **--no-wall # systemctl --no-wall rescue**

3.3.8. A inicialização para o modo de emergência

Emergency mode oferece o ambiente mais mínimo possível e permite reparar seu sistema mesmo em situações em que o sistema não consegue entrar no modo de resgate. No modo de emergência, o sistema monta o sistema de arquivo raiz apenas para leitura, não tenta montar qualquer outro sistema de arquivo local, não ativa interfaces de rede e apenas inicia alguns serviços essenciais.

Procedimento

- Para mudar o alvo atual e entrar no modo de emergência:

```
# systemctl emergência
```

Este comando é similar a **systemctl isolate emergency.target**, mas também envia uma mensagem informativa a todos os usuários que estão atualmente logados no sistema.

Para evitar que o systemd envie esta mensagem, execute o seguinte comando com a opção de linha de comando **--no-wall # systemctl --no-wall emergency**

3.4. ENCERRAMENTO, SUSPENSÃO E HIBERNAÇÃO DO SISTEMA

No Red Hat Enterprise Linux 7, o utilitário **systemctl** substituiu vários comandos de gerenciamento de energia usados em versões anteriores do Red Hat Enterprise Linux. Os comandos listados em Tabela 3.8, “Comparação dos comandos de gerenciamento de energia com o systemctl” ainda estão disponíveis no sistema por razões de compatibilidade, mas é aconselhável que você use **systemctl** quando possível.

Tabela 3.8. Comparação dos comandos de gerenciamento de energia com o systemctl

Antigo Comando	Novo Comando	Descrição
halt	**systemctl halt**	Pára o sistema.
poweroff	**systemctl poweroff**	Desliga os poderes do sistema.

Antigo Comando	Novo Comando	Descrição
reboot	**systemctl reboot**	Reinicia o sistema.
pm-suspend	**systemctl suspend**	Suspende o sistema.
pm-hibernate	**systemctl hibernate**	Hiberna o sistema.
pm-suspend-hybrid	**systemctl hybrid-sleep**	Hiberna e suspende o sistema.

3.4.1. Desligamento do sistema

O utilitário **systemctl** fornece comandos para desligar o sistema, porém o tradicional comando **shutdown** também é suportado. Embora o comando **shutdown** chamará o utilitário **systemctl** para executar o desligamento, ele tem a vantagem de suportar também um argumento de tempo. Isto é particularmente útil para manutenção programada e para dar mais tempo aos usuários para reagir ao aviso de que um desligamento do sistema foi programado. A opção de cancelar o desligamento também pode ser uma vantagem.

Usando comandos systemctl

Para desligar o sistema e desligar a máquina, digite o seguinte em uma janela de comandos como **root**:

```
systemctl poweroff
```

Para desligar e interromper o sistema sem desligar a máquina, execute o seguinte comando como **root**:

```
systemctl halt
```

Por padrão, a execução de qualquer um destes comandos causa **systemd** para enviar uma mensagem informativa a todos os usuários que estão atualmente logados no sistema. Para evitar **systemd** a partir do envio desta mensagem, execute o comando selecionado com a opção de linha de comando **--no-wall**, por exemplo:

```
systemctl --no-wall poweroff
```

Usando o comando de desligamento

Para desligar o sistema e desligar a máquina em um determinado momento, use um comando no seguinte formato: **root**:

```
shutdown --poweroff hh:mm
```

Onde *hh:mm* é a hora no formato de relógio 24 horas. O arquivo **/run/nologin** é criado 5 minutos antes do desligamento do sistema para evitar novos logins. Quando um argumento de tempo é usado, uma mensagem opcional, a *wall message*, pode ser anexada ao comando.

Para desligar e parar o sistema após um atraso, sem desligar a máquina, use um comando no seguinte formato: **root**:

```
desligamento --parar m
```

Onde *m* é o tempo de atraso em minutos. A palavra-chave **now** é um pseudônimo para **0**.

Um desligamento pendente pode ser cancelado pelo usuário **root** da seguinte forma:

```
shutdown -c
```

Consulte a página do manual **shutdown(8)** para mais opções de comando.

3.4.2. Reinicialização do sistema

Para reiniciar o sistema, execute o seguinte comando como **root**:

```
systemctl reboot
```

Por padrão, este comando causa **systemd** para enviar uma mensagem informativa a todos os usuários que estão atualmente logados no sistema. Para evitar **systemd** ao enviar esta mensagem, execute este comando com a opção de linha de comando **--no-wall**:

```
systemctl --no-wall reboot
```

3.4.3. Suspender o sistema

Para suspender o sistema, digite o seguinte em um prompt de shell como **root**:

```
systemctl suspend
```

Este comando salva o estado do sistema em RAM e, com exceção do módulo RAM, desliga a maioria dos dispositivos da máquina. Quando você liga a máquina novamente, o sistema então restaura seu estado a partir da RAM sem ter que inicializar novamente. Como o estado do sistema é salvo na RAM e não no disco rígido, restaurar o sistema do modo suspenso é significativamente mais rápido do que restaurá-lo da hibernação, mas como conseqüência, um estado de sistema suspenso também é vulnerável a quedas de energia.

Para obter informações sobre como hibernar o sistema, consulte Seção 3.4.4, “Hibernando o sistema”.

3.4.4. Hibernando o sistema

Para hibernar o sistema, digite o seguinte em um prompt de shell como **root**:

```
systemctl hibernate
```

Este comando salva o estado do sistema na unidade de disco rígido e desliga a máquina. Quando você liga a máquina novamente, o sistema então restaura seu estado a partir dos dados salvos sem ter que inicializar novamente. Como o estado do sistema é salvo no disco rígido e não na RAM, a máquina não precisa manter energia elétrica no módulo RAM, mas como conseqüência, restaurar o sistema da hibernação é significativamente mais lento do que restaurá-lo do modo suspenso.

Para hibernar e suspender o sistema, execute o seguinte comando como **root**:

```
systemctl hybrid-sleep
```

Para obter informações sobre como suspender o sistema, consulte Seção 3.4.3, “Suspender o sistema”.

3.5. TRABALHANDO COM ARQUIVOS DE UNIDADES DO SISTEMA

Este capítulo inclui a descrição dos arquivos de unidades do sistema. As seções seguintes lhe mostram como fazê-lo:

- Criar arquivos de unidade personalizados
- Converter scripts init SysV em arquivos unitários
- Modificar os arquivos de unidade existentes
- Trabalho com unidades instanciadas

3.5.1. Introdução aos arquivos unitários

Um arquivo de unidade contém diretrizes de configuração que descrevem a unidade e definem seu comportamento. Vários comandos **systemctl** trabalham com arquivos de unidade em segundo plano. Para fazer ajustes mais precisos, o administrador do sistema deve editar ou criar arquivos de unidade manualmente. Tabela 3.1, "Localização dos arquivos da unidade Systemd" lista três diretórios principais onde os arquivos de unidade são armazenados no sistema, o diretório **/etc/systemd/system/** é reservado para arquivos de unidade criados ou personalizados pelo administrador do sistema.

Os nomes dos arquivos da unidade assumem a seguinte forma:

> *unit_nametype_extension*

Aqui, *unit_name* significa o nome da unidade e *type_extension* identifica o tipo de unidade, ver Tabela 3.2, "Tipos de unidades do sistema disponíveis" para uma lista completa dos tipos de unidade. Por exemplo, geralmente há **sshd.service**, bem como **sshd.socket** unidade presente em seu sistema.

Os arquivos de unidade podem ser complementados com um diretório para arquivos de configuração adicionais. Por exemplo, para adicionar opções de configuração personalizadas a **sshd.service**, criar o arquivo **sshd.service.d/custom.conf** e inserir diretivas adicionais lá. Para mais informações sobre os diretórios de configuração, consulte Modificando os arquivos de unidade existentes.

Também podem ser criados os diretórios **sshd.service.wants/** e **sshd.service.requires/**. Estes diretórios contêm links simbólicos para arquivos unitários que são dependências do serviço **sshd**. Os links simbólicos são criados automaticamente durante a instalação de acordo com as opções [Instalar] arquivo de unidade ou em tempo de execução com base nas opções [Unidade]. Também é possível criar estes diretórios e links simbólicos manualmente. Para mais detalhes sobre as opções [Instalar] e [Unidade], veja as tabelas abaixo.

Muitas opções de arquivo de unidade podem ser definidas usando o chamado **unit specifiers** - wildcard strings que são substituídas dinamicamente por parâmetros de unidade quando o arquivo de unidade é carregado. Isto permite a criação de arquivos genéricos de unidades que servem como modelos para a geração de unidades instanciadas. Consulte Trabalhando com unidades instanciadas para obter detalhes.

3.5.2. Estrutura do arquivo da unidade

Os arquivos de unidade normalmente consistem de três seções:

- A seção **[Unit]** - contém opções genéricas que não dependem do tipo da unidade. Estas opções fornecem descrição da unidade, especificam o comportamento da unidade e definem dependências para outras unidades. Para uma lista das opções [Unidade] mais freqüentemente utilizadas, veja Tabela 3.9, "Opções de seção [Unidade] importantes" .

- A seção **[Unit type]** - se uma unidade tem diretrizes específicas de tipo, estas são agrupadas sob uma seção com o nome do tipo de unidade. Por exemplo, os arquivos da unidade de serviço contêm a seção **[Service]**.

- A seção **[Install]** - contém informações sobre a instalação da unidade utilizada pelos comandos **systemctl enable** e **disable**. Para uma lista de opções para a seção **[Install]**, veja Tabela 3.11, "Opções de seção [Instalar] importantes".

3.5.2.1. Opções de seção [Unidade] importantes

As seguintes tabelas listam opções importantes da seção [Unidade].

Tabela 3.9. Opções de seção [Unidade] importantes

Opção[a]	Descrição
Description	Uma descrição significativa da unidade. Este texto é exibido, por exemplo, na saída do comando **systemctl status**.
Documentation	Fornece uma lista de URIs de referência de documentação para a unidade.
After[b]	Define a ordem na qual as unidades são iniciadas. A unidade só começa depois que as unidades especificadas em **After** estiverem ativas. Ao contrário de **Requires**, **After** não ativa explicitamente as unidades especificadas. A opção **Before** tem a funcionalidade oposta a**After**.
Requires	Configura as dependências de outras unidades. As unidades listadas em **Requires** são ativadas em conjunto com a unidade. Se alguma das unidades necessárias falhar em iniciar, a unidade não é ativada.
Wants	Configura dependências mais fracas do que **Requires**. Se qualquer uma das unidades listadas não começar com sucesso, não terá impacto na ativação da unidade. Esta é a forma recomendada para estabelecer dependências de unidades personalizadas.
Conflicts	Configura dependências negativas, um oposto a **Requires**.

[a] Para uma lista completa de opções configuráveis na seção [Unidade], consulte a página do manual **systemd.unit(5)**.

[b] Na maioria dos casos, é suficiente definir apenas as dependências do pedido com **After** e **Before** opções de arquivo de unidade. Se você também definir uma dependência de pedidos com **Wants** (recomendado) ou **Requires**, a dependência de pedidos ainda precisa ser especificada. Isso porque as dependências de pedidos e requisitos funcionam independentemente uma da outra.

3.5.2.2. Opções importantes da seção [Serviço]

As seguintes tabelas listam opções importantes da seção [Serviço].

Tabela 3.10. Opções importantes da seção [Serviço]

Opção[a]	Descrição
Type	Configura o tipo de inicialização do processo da unidade que afeta a funcionalidade do **ExecStart** e opções relacionadas. Uma das: * **simple** - O valor padrão. O processo iniciado com **ExecStart** é o processo principal do serviço. * **forking** - O processo teve início com **ExecStart**, que se torna o principal processo do serviço. O processo dos pais sai quando a partida é concluída. * **oneshot** - Este tipo é semelhante a **simple**, mas o processo sai antes de iniciar as unidades conseqüentes. * **dbus** - Este tipo é semelhante a **simple**, mas as unidades conseqüentes só são iniciadas após o processo principal ganhar um nome D-Bus. * **notify** - Este tipo é semelhante a **simple**, mas as unidades conseqüentes só são iniciadas depois que uma mensagem de notificação é enviada através da função sd_notify(). * **idle** - similar a **simple**, a execução real do binário de serviço é atrasada até que todos os trabalhos estejam concluídos, o que evita misturar a saída de status com a saída de shell dos serviços.
ExecStart	Especifica comandos ou scripts a serem executados quando a unidade é iniciada. **ExecStartPre** e **ExecStartPost** especificam comandos personalizados a serem executados antes e depois **ExecStart**. **Type=oneshot** permite especificar vários comandos personalizados que são então executados sequencialmente.
ExecStop	Especifica comandos ou scripts a serem executados quando a unidade é parada.
ExecReload	Especifica comandos ou scripts a serem executados quando a unidade é recarregada.
Restart	Com esta opção ativada, o serviço é reiniciado após a saída de seu processo, com exceção de uma parada limpa pelo comando **systemctl**.

Opção[a]	Descrição
RemainAfterExit	Se definido para True, o serviço é considerado ativo mesmo quando todos os seus processos saíram. O valor padrão é Falso. Esta opção é especialmente útil se **Type=oneshot** estiver configurado.

[a] Para obter uma lista completa de opções configuráveis na seção [Serviço], consulte a página do manual **systemd.service(5)**.

3.5.2.3. Opções de seção [Instalar] importantes

As tabelas a seguir listam opções importantes da seção [Instalar].

Tabela 3.11. Opções de seção [Instalar] importantes

Opção[a]	Descrição
Alias	Fornece uma lista separada por espaço de nomes adicionais para a unidade. A maioria dos comandos **systemctl**, excluindo **systemctl enable**, pode usar pseudônimos ao invés do nome real da unidade.
RequiredBy	Uma lista de unidades que dependem da unidade. Quando esta unidade é habilitada, as unidades listadas em **RequiredBy** ganham uma dependência **Require** da unidade.
WantedBy	Uma lista de unidades que dependem pouco da unidade. Quando esta unidade é habilitada, as unidades listadas em **WantedBy** ganham uma dependência **Want** da unidade.
Also	Especifica uma lista de unidades a serem instaladas ou desinstaladas junto com a unidade.
DefaultInstance	Limitada a unidades instanciadas, esta opção especifica a instância padrão para a qual a unidade é habilitada. Veja Trabalhando com unidades instanciadas

[a] Para uma lista completa de opções configuráveis na seção [Instalar], consulte a página do manual **systemd.unit(5)**.

3.5.3. Criação de arquivos unitários personalizados

Há vários casos de uso para criar arquivos de unidade do zero: você poderia executar um daemon personalizado, criar uma segunda instância de algum serviço existente (como em Criar uma segunda instância do serviço sshd), ou importar um script de inicialização SysV (mais em Converter scripts de inicialização SysV em arquivos de unidade). Por outro lado, se você pretende apenas modificar ou

ampliar o comportamento de uma unidade existente, use as instruções de Modificação de arquivos de unidade existentes. O procedimento a seguir descreve o processo geral de criação de um serviço personalizado.

Procedimento

1. Preparar o arquivo executável com o serviço personalizado. Este pode ser um script personalizado, ou um executável entregue por um fornecedor de software. Se necessário, prepare um arquivo PID para manter um PID constante para o processo principal do serviço personalizado. Também é possível incluir arquivos de ambiente para armazenar variáveis shell para o serviço. Certifique-se de que o script fonte seja executável (executando o **chmod a x**) e não seja interativo.

2. Crie um arquivo unitário no diretório **/etc/systemd/system/** e certifique-se de que ele tenha as permissões corretas do arquivo. Executar como **root**:

   ```
   touch /etc/systemd/system/name.service

   chmod 664 /etc/systemd/system/name.service
   ```

   Substituir *name* por um nome do serviço a ser criado. Note que o arquivo não precisa ser executável.

3. Abra o ***name*.service** criado na etapa anterior, e adicionar as opções de configuração do serviço. Há uma variedade de opções que podem ser usadas dependendo do tipo de serviço que você deseja criar, veja Estrutura do arquivo de unidade . A seguir, um exemplo de configuração de unidade para um serviço relacionado à rede:

   ```
   [Unit]
   Description=service_description
   After=network.target

   [Service]
   ExecStart=path_to_executable
   Type=forking
   PIDFile=path_to_pidfile

   [Install]
   WantedBy=default.target
   ```

   Onde:

   - *service_description* é uma descrição informativa que é exibida em arquivos de diário de bordo e na saída do comando **systemctl status**.
   - a configuração **After** garante que o serviço seja iniciado somente depois que a rede estiver funcionando. Adicione uma lista separada por espaço de outros serviços ou alvos relevantes.
   - *path_to_executable* representa o caminho para o serviço executável real.
   - **Type=forking** é usado para daemons que fazem a chamada do sistema de garfos. O processo principal do serviço é criado com o PID especificado em *path_to_pidfile*. Encontre outros tipos de inicialização em Tabela 3.10, “Opções importantes da seção [Serviço]” .

- **WantedBy** declara a meta ou metas que o serviço deve ser iniciado. Pense nessas metas como uma substituição do antigo conceito de níveis de execução.

4. Notifique **systemd** que um novo ***name*.service** existe através da execução do seguinte comando como **root**:

```
systemctl daemon-reload

systemctl start name.service
```

ATENÇÃO

Sempre execute o comando **systemctl daemon-reload** após criar novos arquivos de unidade ou modificar os arquivos de unidade existentes. Caso contrário, os comandos **systemctl start** ou **systemctl enable** poderiam falhar devido a um descompasso entre os estados de **systemd** e arquivos de unidades de serviço reais em disco. Note que em sistemas com um grande número de unidades isso pode levar muito tempo, pois o estado de cada unidade tem que ser serializado e subseqüentemente deserializado durante a recarga.

3.5.3.1. Criação de um arquivo unitário personalizado utilizando a segunda instância do serviço sshd

Os administradores de sistema freqüentemente precisam configurar e executar várias instâncias de um serviço. Isto é feito criando cópias dos arquivos originais de configuração do serviço e modificando certos parâmetros para evitar conflitos com a instância principal do serviço. O procedimento a seguir mostra como criar uma segunda instância do serviço **sshd**.

Procedimento

1. Criar uma cópia do arquivo **sshd_config** que será usado pelo segundo daemon:

```
# cp /etc/ssh/sshd{,-second}_config
```

2. Edite o arquivo **sshd-second_config** criado na etapa anterior para atribuir um número de porta e um arquivo PID diferentes ao segundo daemon:

```
Port 22220
PidFile /var/run/sshd-second.pid
```

Consulte a página **sshd_config**(5) do manual para obter mais informações sobre as opções **Port** e **PidFile**. Certifique-se de que a porta escolhida não esteja em uso por nenhum outro serviço. O arquivo PID não precisa existir antes de executar o serviço, ele é gerado automaticamente no início do serviço.

3. Criar uma cópia do arquivo da unidade systemd para o serviço **sshd**:

```
# cp /usr/lib/systemd/systemd/system/sshd.service /etc/systemd/systemd/system/sshd-second.service
```

4. Altere o **sshd-second.service** criado na etapa anterior como segue:

   a. Modifique a opção **Description**:

   ```
   Descrição=OpenSSH servidor de segunda instância daemon
   ```

   b. Adicionar sshd.service aos serviços especificados na opção **After**, de modo que a segunda instância só comece após a primeira já ter começado:

   ```
   After=syslog.target network.target auditd.service sshd.service
   ```

   c. A primeira instância do sshd inclui a geração chave, portanto, remova a linha **ExecStartPre=/usr/sbin/sshd-keygen**.

   d. Adicione o parâmetro **-f /etc/ssh/sshd-second_config** ao comando **sshd**, para que seja utilizado o arquivo de configuração alternativa:

   ```
   ExecStart=/usr/sbin/sshd -D -f /etc/ssh/sshd-second_config $OPTIONS
   ```

   e. Após as modificações acima, o serviço sshd-segundo.serviço deve ter a seguinte aparência:

   ```
   [Unit]
   Description=OpenSSH server second instance daemon
   After=syslog.target network.target auditd.service sshd.service

   [Service]
   EnvironmentFile=/etc/sysconfig/sshd
   ExecStart=/usr/sbin/sshd -D -f /etc/ssh/sshd-second_config $OPTIONS
   ExecReload=/bin/kill -HUP $MAINPID
   KillMode=process
   Restart=on-failure
   RestartSec=42s

   [Install]
   WantedBy=multi-user.target
   ```

5. Se utilizar SELinux, adicione a porta para a segunda instância de sshd aos portos SSH, caso contrário, a segunda instância de sshd será rejeitada para se ligar ao porto:

   ```
   # porto semanage -a -t ssh_port_t -p tcp 22220
   ```

6. Habilite o sshd-second.service, para que ele comece automaticamente na inicialização:

   ```
   # systemctl habilita sshd-second.service
   ```

7. Verifique se o sshd-second.service está rodando usando o comando **systemctl status**.

8. Verificar se a porta está habilitada corretamente, conectando-se ao serviço:

   ```
   $ ssh -p 22220 user@server
   ```

   Se o firewall estiver em uso, certifique-se de que esteja configurado adequadamente para permitir conexões com a segunda instância de sshd.

3.5.3.2. Escolhendo um alvo para pedidos e dependências de arquivos de unidades personalizadas

Para saber como escolher corretamente um alvo para pedidos e dependências de seus arquivos de unidade personalizados, veja os seguintes artigos:

- Como escrever um arquivo de unidade de serviço que imponha que determinados serviços tenham que ser iniciados
- Como decidir que dependências uma definição de unidade de serviço de sistema deve ter

Informações adicionais com alguns exemplos reais de casos acionados pelo pedido e dependências em um arquivo de unidade estão disponíveis no artigo da Red Hat Knowledgebase Existe alguma informação útil sobre como escrever arquivos de unidade?

Se você quiser estabelecer limites para serviços iniciados por **systemd**, veja o artigo da Red Hat Knowledgebase Como estabelecer limites para serviços no RHEL 7 e systemd. Estes limites precisam ser definidos no arquivo de unidade do serviço. Note que **systemd** ignora os limites estabelecidos nos arquivos de configuração **/etc/security/limits.conf** e **/etc/security/limits.d/*.conf**. Os limites definidos nestes arquivos são definidos pelo PAM ao iniciar uma sessão de login, mas daemons iniciados por **systemd** não usam sessões de login do PAM.

3.5.4. Conversão de scripts init SysV em arquivos unitários

Antes de tomar tempo para converter um script de inicialização SysV em um arquivo de unidade, certifique-se de que a conversão ainda não foi feita em outro lugar. Todos os serviços centrais instalados no Red Hat Enterprise Linux vêm com arquivos de unidade padrão, e o mesmo se aplica a muitos pacotes de software de terceiros.

A conversão de um roteiro de inicialização em um arquivo de unidade requer a análise do roteiro e a extração das informações necessárias do mesmo. Com base nestes dados, você pode criar um arquivo unitário. Como os scripts de inicialização podem variar muito dependendo do tipo de serviço, você pode precisar empregar mais opções de configuração para a tradução do que as descritas neste capítulo. Observe que alguns níveis de personalização que estavam disponíveis com scripts de inicialização não são mais suportados por unidades do sistema.

A maioria das informações necessárias para a conversão é fornecida no cabeçalho do roteiro. O exemplo a seguir mostra a seção de abertura do script de inicialização usado para iniciar o serviço **postfix** no Red Hat Enterprise Linux 6:

> !/bin/bash # postfix Postfix Mail Transfer Agent # chkconfig: 2345 80 30 # description: Postfix is a Mail Transport Agent, which is the program that moves mail from one machine to another. # processname: master # pidfile: /var/spool/postfix/pid/master.pid # config: /etc/postfix/main.cf # config: /etc/postfix/master.cf BEGIN INIT INFO # Provides: postfix MTA # Required-Start: $local_fs $network $remote_fs # Required-Stop: $local_fs $network $remote_fs # Default-Start: 2 3 4 5 # Default-Stop: 0 1 6 # Short-Description: start and stop postfix # Description: Postfix is a Mail Transport Agent, which is the program that moves mail from one machine to another. # END INIT INFO

No exemplo acima, apenas as linhas que começam com **# chkconfig** e **# description** são obrigatórias, portanto, você pode não encontrar o resto em diferentes arquivos init. O texto incluído entre as linhas **BEGIN INIT INFO** e **END INIT INFO** se chama **Linux Standard Base (LSB) header**. Se especificado, os cabeçalhos da LSB contêm diretrizes que definem a descrição do serviço, as dependências e os níveis de execução padrão. O que segue é uma visão geral das tarefas analíticas com o objetivo de coletar os dados necessários para um novo arquivo de unidade. O script init postfix é usado como um exemplo.

3.5.4.1. Encontrando a descrição de serviço do sistema

Você pode encontrar informações descritivas sobre o roteiro na linha começando com **#description**. Use esta descrição junto com o nome do serviço na opção **Description** na seção [Unidade] do arquivo da unidade. O cabeçalho da LSB pode conter dados similares nas linhas **#Short-Description** e **#Description**.

3.5.4.2. Encontrar as dependências de serviço do sistema

O cabeçalho da LSB pode conter várias diretrizes que formam dependências entre serviços. A maioria delas é traduzível para opções de unidades do sistema, veja Tabela 3.12, “Opções de dependência do cabeçalho da LSB”

Tabela 3.12. Opções de dependência do cabeçalho da LSB

Opção LSB	Descrição	Equivalente de arquivo unitário
Provides	Especifica o nome da instalação de inicialização do serviço, que pode ser referenciado em outros scripts de inicialização (com o prefixo "$"). Isto não é mais necessário, pois os arquivos de unidade se referem a outras unidades pelos nomes de seus arquivos.	–
Required-Start	Contém os nomes das instalações de inicialização dos serviços necessários. Isto é traduzido como uma dependência de pedido, os nomes das instalações de boot são substituídos por nomes de arquivos de unidade dos serviços correspondentes ou alvos aos quais pertencem. Por exemplo, no caso de **postfix**, a dependência de início de rede requerida em $ foi traduzida para a dependência de início de rede Após dependência de rede.target.	**After**, **Before**
Should-Start	Constitui dependências mais fracas do que o início necessário. Dependências Falhadas Should-Start não afetam a inicialização do serviço.	**After**, **Before**
Required-Stop, **Should-Stop**	Constituir dependências negativas.	**Conflicts**

3.5.4.3. Encontrar alvos padrão do serviço

A linha que começa com **#chkconfig** contém três valores numéricos. O mais importante é o primeiro número que representa os níveis de execução padrão nos quais o serviço é iniciado. Mapear estes níveis de execução para metas equivalentes do sistema. Em seguida, liste estes alvos na opção **WantedBy** na seção [Instalar] do arquivo da unidade. Por exemplo, **postfix** foi iniciado anteriormente nos runlevels 2, 3, 4 e 5, o que se traduz para multi-usuário.target e gráfico.target. Note que o graphical.target depende do multiuser.target, portanto não é necessário especificar ambos. Você pode encontrar informações sobre níveis de execução padrão e proibidos também em **#Default-Start** e **#Default-Stop** linhas no cabeçalho da LSB.

Os outros dois valores especificados na linha **#chkconfig** representam as prioridades de inicialização e desligamento do init script. Estes valores são interpretados por **systemd** se ele carrega o roteiro de inicialização, mas não há arquivo de unidade equivalente.

3.5.4.4. Encontrar arquivos utilizados pelo serviço

Os scripts de inicialização exigem o carregamento de uma biblioteca de funções de um diretório dedicado e permitem a importação de configuração, ambiente e arquivos PID. As variáveis de ambiente são especificadas na linha começando com **#config** no cabeçalho do init script, que se traduz para a opção de arquivo de unidade **EnvironmentFile**. O arquivo PID especificado na linha de scripts init **#pidfile** é importado para o arquivo de unidade com a opção **PIDFile**.

A informação chave que não está incluída no cabeçalho do init script é o caminho para o executável do serviço, e potencialmente alguns outros arquivos requeridos pelo serviço. Em versões anteriores do Red Hat Enterprise Linux, os scripts de inicialização usavam uma declaração de caso Bash para definir o comportamento do serviço em ações default, tais como **start**, **stop** ou **restart**, bem como ações personalizadas. O seguinte trecho do script init **postfix** mostra o bloco de código a ser executado no início do serviço.

```
conf_check() {
    [ -x /usr/sbin/postfix ] || exit 5
    [ -d /etc/postfix ] || exit 6
    [ -d /var/spool/postfix ] || exit 5
}

make_aliasesdb() {
 if [ "$(/usr/sbin/postconf -h alias_database)" == "hash:/etc/aliases" ]
 then
  # /etc/aliases.db might be used by other MTA, make sure nothing
  # has touched it since our last newaliases call
  [ /etc/aliases -nt /etc/aliases.db ] ||
   [ "$ALIASESDB_STAMP" -nt /etc/aliases.db ] ||
   [ "$ALIASESDB_STAMP" -ot /etc/aliases.db ] || return
  /usr/bin/newaliases
  touch -r /etc/aliases.db "$ALIASESDB_STAMP"
 else
  /usr/bin/newaliases
 fi
}

start() {
 [ "$EUID" != "0" ] && exit 4
 # Check that networking is up.
 [ ${NETWORKING} = "no" ] && exit 1
 conf_check
 # Start daemons.
 echo -n $"Starting postfix: "
```

```
    make_aliasesdb >/dev/null 2>&1
    [ -x $CHROOT_UPDATE ] && $CHROOT_UPDATE
    /usr/sbin/postfix start 2>/dev/null 1>&2 && success || failure $"$prog start"
    RETVAL=$?
    [ $RETVAL -eq 0 ] && touch $lockfile
        echo
    return $RETVAL
}
```

A extensibilidade do init script permitiu especificar duas funções personalizadas, **conf_check()** e **make_aliasesdb()**, que são chamadas a partir do bloco de funções **start()**. Observando mais de perto, vários arquivos e diretórios externos são mencionados no código acima: o principal serviço executável **/usr/sbin/postfix**, os diretórios de configuração **/etc/postfix/** e **/var/spool/postfix/**, assim como o diretório **/usr/sbin/postconf/**.

Systemd suporta apenas as ações predefinidas, mas permite executar executáveis personalizados com **ExecStart**, **ExecStartPre**, **ExecStartPost**, **ExecStop**, e **ExecReload** opções. O **/usr/sbin/postfix** junto com os scripts de suporte são executados no início do serviço. A conversão de scripts complexos de inicialização requer a compreensão do propósito de cada declaração no script. Algumas das declarações são específicas para a versão do sistema operacional, portanto, não é necessário traduzi-las. Por outro lado, alguns ajustes podem ser necessários no novo ambiente, tanto no arquivo de unidade quanto no executável do serviço e nos arquivos de suporte.

3.5.5. Modificação de arquivos de unidades existentes

Os serviços instalados no sistema vêm com arquivos de unidade padrão que são armazenados no diretório **/usr/lib/systemd/system/**. Os administradores do sistema não devem modificar estes arquivos diretamente, portanto qualquer personalização deve ser confinada aos arquivos de configuração no diretório **/etc/systemd/system/**.

Procedimento

1. Dependendo da extensão das mudanças necessárias, escolha uma das seguintes abordagens:

   - Crie um diretório para arquivos de configuração suplementares em **/etc/systemd/system/*unit*.d/**. Este método é recomendado para a maioria dos casos de uso. Ele permite estender a configuração padrão com funcionalidade adicional, enquanto ainda se refere ao arquivo original da unidade. As mudanças na unidade padrão introduzidas com uma atualização de pacote são, portanto, aplicadas automaticamente. Veja Estendendo a configuração padrão da unidade para maiores informações.

   - Criar uma cópia do arquivo original da unidade **/usr/lib/systemd/system/** em **/etc/systemd/system/** e fazer alterações lá. A cópia substitui o arquivo original, portanto as alterações introduzidas com a atualização do pacote não são aplicadas. Este método é útil para fazer mudanças significativas na unidade que devem persistir independentemente das atualizações do pacote. Consulte Substituição da configuração padrão da unidade para obter detalhes.

2. Para retornar à configuração padrão da unidade, exclua os arquivos de configuração criados sob medida em **/etc/systemd/system/**.

3. Para aplicar mudanças nos arquivos da unidade sem reiniciar o sistema, execute:

   ```
   systemctl daemon-reload
   ```

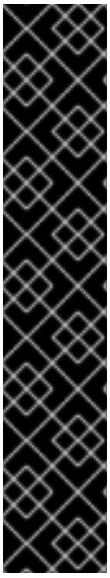
   A opção **daemon-reload** recarrega todos os arquivos unitários e recria toda a árvore de

dependência, que é necessária para aplicar imediatamente qualquer mudança em um arquivo unitário. Como alternativa, é possível obter o mesmo resultado com o seguinte comando, que deve ser executado sob o usuário **root**:

```
init q
```

4. Se o arquivo da unidade modificada pertence a um serviço em execução, este serviço deve ser reiniciado para aceitar novas configurações:

```
systemctl restart name.service
```



IMPORTANTE

Para modificar propriedades, tais como dependências ou timeouts, de um serviço que é tratado por um initscript SysV, não modifique o initscript em si. Em vez disso, crie um arquivo de configuração drop-in **systemd** para o serviço, conforme descrito em Estendendo a configuração padrão da unidade e Substituindo a configuração padrão da unidade. Em seguida, gerencie este serviço da mesma forma que um serviço normal **systemd**.

Por exemplo, para estender a configuração do serviço **network**, não modifique o arquivo initscript **/etc/rc.d/init.d/network**. Em vez disso, crie um novo diretório **/etc/systemd/system/network.service.d/** e um arquivo drop-in **systemd** **/etc/systemd/system/network.service.d/*my_config*.conf**. Em seguida, coloque os valores modificados no arquivo drop-in. Nota: **systemd** conhece o serviço **network** como **network.service**, e é por isso que o diretório criado deve ser chamado **network.service.d**

3.5.5.1. Estendendo a configuração padrão da unidade

Esta seção descreve como estender o arquivo de unidade padrão com opções adicionais de configuração.

Procedimento

1. Para estender o arquivo de unidade padrão com opções adicionais de configuração, primeiro crie um diretório de configuração em **/etc/systemd/system/**. Se estiver ampliando uma unidade de serviço, execute o seguinte comando como **root**:

```
mkdir /etc/systemd/system/name.service.d/
```

Substitua *name* pelo nome do serviço que você deseja estender. A sintaxe acima se aplica a todos os tipos de unidade.

2. Criar um arquivo de configuração no diretório feito na etapa anterior. Note que o nome do arquivo deve terminar com o sufixo **.conf**. Digite:

```
touch /etc/systemd/system/nome.serviço.d/config_name.conf
```

Substituir *config_name* pelo nome do arquivo de configuração. Este arquivo adere à estrutura normal do arquivo de unidade, portanto, todas as diretrizes devem ser especificadas nas seções apropriadas, ver Estrutura do arquivo de unidade .

Por exemplo, para adicionar uma dependência personalizada, crie um arquivo de configuração com o seguinte conteúdo:

```
[Unit]
Requires=new_dependency
After=new_dependency
```

Onde *new_dependency* significa que a unidade deve ser marcada como uma dependência. Outro exemplo é um arquivo de configuração que reinicia o serviço após a saída de seu processo principal, com um atraso de 30 segundos:

```
[Service]
Restart=always
RestartSec=30
```

É recomendado criar pequenos arquivos de configuração focados apenas em uma tarefa. Tais arquivos podem ser facilmente movidos ou ligados a diretórios de configuração de outros serviços.

3. Para aplicar as mudanças feitas na unidade, execute como **root**:

```
systemctl daemon-reload
systemctl restart name.service
```

Exemplo 3.10. Estendendo a configuração do httpd.service

Para modificar a unidade de serviço httpd.service para que um script de shell personalizado seja executado automaticamente ao iniciar o serviço Apache, execute os seguintes passos.

1. Criar um diretório e um arquivo de configuração personalizado:

```
# mkdir /etc/systemd/systemd/httpd.service.d/
```

```
# touch /etc/systemd/system/httpd.service.d/custom_script.conf
```

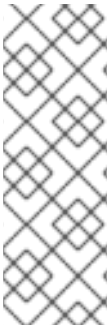
2. Desde que o script que você queira iniciar automaticamente com o Apache esteja localizado em **/usr/local/bin/custom.sh**, insira o seguinte texto no arquivo **custom_script.conf**:

```
[Service]
ExecStartPost=/usr/local/bin/custom.sh
```

3. Para aplicar as mudanças de unidade, execute:

```
# systemctl daemon-reload
```

```
# systemctl restart httpd.service
```



NOTA

Os arquivos de configuração dos diretórios de configuração em **/etc/systemd/system/** têm precedência sobre os arquivos de unidade em **/usr/lib/systemd/system/**. Portanto, se os arquivos de configuração contêm uma opção que pode ser especificada apenas uma vez, como **Description** ou **ExecStart**, o valor padrão desta opção é anulado. Observe que na saída do comando **systemd-delta**, descrito em Monitoramento de unidades sobrescritas, tais unidades são sempre marcadas como [EXTENDIDAS], mesmo que, em suma, certas opções sejam de fato sobrescritas.

3.5.5.2. Substituindo a configuração padrão da unidade

Esta seção descreve como substituir a configuração padrão da unidade.

Procedimento

1. Para fazer mudanças que persistirão após a atualização do pacote que fornece o arquivo da unidade, primeiro copie o arquivo para o diretório **/etc/systemd/system/**. Para isso, execute o seguinte comando como **root**:

   ```
   cp /usr/lib/systemd/system/name.service /etc/systemd/system/name.service
   ```

   Onde *name* representa o nome da unidade de serviço que você deseja modificar. A sintaxe acima se aplica a todos os tipos de unidade.

2. Abra o arquivo copiado com um editor de texto, e faça as mudanças desejadas. Para aplicar as mudanças na unidade, execute como **root**:

   ```
   systemctl daemon-reload
   systemctl restart name.service
   ```

Exemplo 3.11. Alteração do limite de tempo limite

Você pode especificar um valor de timeout por serviço para evitar que um mau funcionamento do serviço congele o sistema. Caso contrário, o timeout é definido por padrão para 90 segundos para serviços normais e para 300 segundos para serviços compatíveis com SysV.

Por exemplo, para estender o limite de tempo para o serviço **httpd**:

1. Copie o arquivo da unidade **httpd** para o diretório **/etc/systemd/system/**:

   ```
   cp /usr/lib/systemd/system/httpd.service /etc/systemd/system/httpd.service
   ```

2. Abra o arquivo **/etc/systemd/system/httpd.service** e especifique o valor **TimeoutStartUSec** na seção **[Service]**:

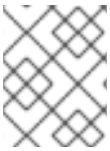
   ```
   …
   [Service]
   …
   PrivateTmp=true
   TimeoutStartSec=10

   [Install]
   WantedBy=multi-user.target
   …
   ```

3. Recarregue o daemon **systemd**:

```
systemctl daemon-reload
```

4. **Optional.** Verificar o novo valor de timeout:

```
systemctl show httpd -p TimeoutStartUSec
```



NOTA

Para alterar o limite de tempo limite globalmente, insira o arquivo **DefaultTimeoutStartSec** no arquivo **/etc/systemd/system.conf**.

3.5.5.3. Monitoramento de unidades anuladas

Esta seção descreve como exibir uma visão geral dos arquivos de unidade anulados ou modificados.

Procedimento

1. Para exibir uma visão geral dos arquivos de unidade anulados ou modificados, use o seguinte comando:

```
systemd-delta
```

Por exemplo, a saída do comando acima pode parecer como se segue:

```
[EQUIVALENT] /etc/systemd/system/default.target → /usr/lib/systemd/system/default.target
[OVERRIDDEN] /etc/systemd/system/autofs.service →
/usr/lib/systemd/system/autofs.service

--- /usr/lib/systemd/system/autofs.service       2014-10-16 21:30:39.000000000 -0400
+ /etc/systemd/system/autofs.service  2014-11-21 10:00:58.513568275 -0500
@@ -8,7 +8,8 @@
 EnvironmentFile=-/etc/sysconfig/autofs
 ExecStart=/usr/sbin/automount $OPTIONS --pid-file /run/autofs.pid
 ExecReload=/usr/bin/kill -HUP $MAINPID
-TimeoutSec=180
+TimeoutSec=240
+Restart=Always

 [Install]
 WantedBy=multi-user.target

[MASKED]     /etc/systemd/system/cups.service → /usr/lib/systemd/system/cups.service
[EXTENDED]   /usr/lib/systemd/system/sssd.service →
/etc/systemd/system/sssd.service.d/journal.conf

4 overridden configuration files found.
```

3.5.6. Trabalhando com unidades instanciadas

É possível instanciar múltiplas unidades a partir de um único arquivo de configuração de modelo em tempo de execução. O caractere "@" é usado para marcar o modelo e para associar unidades a ele. As unidades instantâneas podem ser iniciadas a partir de outro arquivo de unidade (usando as opções **Requires** ou **Wants**), ou com o comando **systemctl start**. As unidades de serviço instantâneas são nomeadas da seguinte forma:

```
template_name@instance_name.serviço
```

Onde *template_name* representa o nome do arquivo de configuração do modelo. Substituir *instance_name* pelo nome da instância da unidade. Várias instâncias podem apontar para o mesmo arquivo de modelo com opções de configuração comuns para todas as instâncias da unidade. O nome da unidade de modelo tem a forma de:

```
unit_name@.service
```

Por exemplo, a seguinte configuração **Wants** em um arquivo de unidade:

```
Wants=getty@ttyA.service getty@ttyB.service
```

primeiro faz a busca por unidades de serviço. Se tais unidades não forem encontradas, a parte entre "@" e o sufixo de tipo é ignorada e **systemd** pesquisa o arquivo **getty@.service**, lê a configuração a partir dele, e inicia os serviços.

Por exemplo, o modelo **getty@.service** contém as seguintes diretrizes:

```
[Unit]
Description=Getty on %I
…
[Service]
ExecStart=-/sbin/agetty --noclear %I $TERM
…
```

Quando os modelos getty@ttyA.service e getty@ttyB.service são instanciados a partir do modelo acima, **Description**= é resolvido como **Getty on ttyA** e **Getty on ttyB**.

3.5.6.1. Especificadores importantes de unidades

Os caracteres curinga, chamados **unit specifiers**, podem ser usados em qualquer arquivo de configuração de unidade. Os especificadores de unidade substituem certos parâmetros de unidade e são interpretados em tempo de execução. Tabela 3.13, “Especificadores importantes de unidades” lista os especificadores de unidade que são particularmente úteis para unidades modelo.

Tabela 3.13. Especificadores importantes de unidades

Especificador da unidade	Significado	Descrição
%n	Nome completo da unidade	Representa o nome completo da unidade incluindo o sufixo do tipo. **%N** tem o mesmo significado, mas também substitui os caracteres proibidos por códigos ASCII.

Especificador da unidade	Significado	Descrição
%p	Nome do prefixo	Significa um nome de unidade com o sufixo do tipo removido. Para unidades instanciadas, %p significa a parte do nome da unidade antes do caractere "@\ @".
%i	Nome da instância	É a parte do nome da unidade instanciada entre o caractere "@" e o sufixo de tipo. **%I** tem o mesmo significado, mas também substitui os caracteres proibidos para os códigos ASCII.
%H	Nome do anfitrião	Representa o nome do sistema em funcionamento no momento em que a configuração da unidade é carregada.
%t	Diretório de tempo de execução	Representa o diretório de tempo de execução, que é /**run** para o usuário **root**, ou o valor da variável XDG_RUNTIME_DIR para usuários não privilegiados.

Para uma lista completa dos especificadores de unidades, consulte a página do manual **systemd.unit(5)**.

3.6. OTIMIZAÇÃO DO SISTEMA PARA ENCURTAR O TEMPO DE INICIALIZAÇÃO

Há uma lista de arquivos de unidade do sistema que são ativados por padrão. Os serviços de sistema que são definidos por esses arquivos unitários são executados automaticamente na inicialização, o que influencia o tempo de inicialização.

Esta seção descreve:

- As ferramentas para examinar o desempenho da inicialização do sistema.
- A finalidade das unidades do sistema habilitadas por padrão, e as circunstâncias sob as quais você pode desabilitar tais unidades do sistema com segurança a fim de encurtar o tempo de inicialização.

3.6.1. Examinando o desempenho da inicialização do sistema

Para examinar o desempenho da inicialização do sistema, você pode usar o comando **systemd-analyze**. Este comando tem muitas opções disponíveis. No entanto, esta seção cobre apenas as selecionadas que podem ser importantes para o ajuste do sistema a fim de encurtar o tempo de inicialização.

Para uma lista completa e uma descrição detalhada de todas as opções, consulte a página de manual **systemd-analyze**.

Pré-requisitos

Antes de começar a examinar o sistemad a fim de afinar o tempo de inicialização, você pode querer listar todos os serviços habilitados:

```
$ systemctl list-unit-files --state=enabled
```

Analisando o tempo total de inicialização

Procedimento

- Para as informações gerais sobre o tempo que levou o último boot bem sucedido, use:

```
Análise do sistema
```

Analisando o tempo de inicialização da unidade

Procedimento

- Para obter informações sobre o tempo de inicialização de cada unidade do sistema, use:

```
$ sistema-analisar a culpa
```

A saída lista as unidades em ordem decrescente de acordo com o tempo que levaram para inicializar durante a última inicialização bem-sucedida.

Identificação de unidades críticas

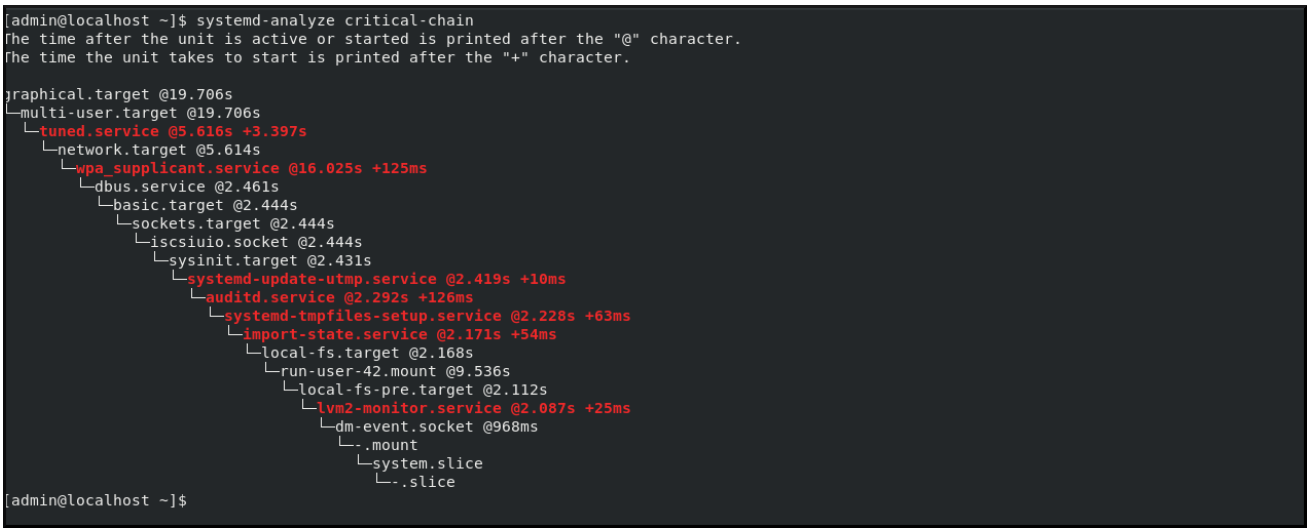
Procedimento

- Para identificar as unidades que levaram mais tempo para inicializar na última inicialização bem sucedida, use:

```
Analisar a cadeia crítica do sistema
```

A saída destaca as unidades que desaceleram criticamente a inicialização com a cor vermelha.

Figura 3.1. A saída do comando sistema-analisar a cadeia crítica



3.6.2. Um guia para selecionar serviços que podem ser desativados com segurança

Se você achar que o tempo de inicialização de seu sistema é longo, você pode encurtá-lo desativando alguns dos serviços habilitados na inicialização por padrão.

Para listar tais serviços, execute:

```
$ systemctl list-unit-files --state=enabled
```

Para desativar um serviço, execute:

```
# systemctl desativar service_name
```

Entretanto, certos serviços devem permanecer habilitados para que seu sistema operacional seja seguro e funcione da maneira que você precisa.

Você pode usar a tabela abaixo como um guia para selecionar os serviços que você pode desativar com segurança. A tabela lista todos os serviços habilitados por default em uma instalação mínima do Red Hat Enterprise Linux 8, e para cada serviço ela declara se este serviço pode ser desabilitado com segurança.

A tabela também fornece mais informações sobre as circunstâncias nas quais o serviço pode ser desativado, ou a razão pela qual você não deve desativar o serviço.

Tabela 3.14. Serviços habilitados por padrão em uma instalação mínima do RHEL 8

Nome do serviço	Pode ser desativado?	Mais informações

Nome do serviço	Pode ser desativado?	Mais informações
serviço auditd.service	sim	Desabilite **auditd.service** somente se você não precisar de mensagens de auditoria do kernel. Esteja ciente de que se você desativar **auditd.service**, o arquivo **/var/log/audit/audit.log** não será produzido. Conseqüentemente, você não é capaz de rever retroativamente algumas ações ou eventos comumente revisados, tais como logins de usuários, início de serviço ou mudanças de senha. Observe também que a auditoria tem duas partes: uma parte do kernel, e um serviço em si. Usando o comando **systemctl disable auditd**, você só desabilita o serviço, mas não a parte do kernel. Para desabilitar a auditoria do sistema em sua totalidade, configure **audit=0** na linha de comando do kernel.
autovt@.service	não	Este serviço funciona somente quando é realmente necessário, portanto, não precisa ser desativado.
crond.service	sim	Esteja ciente de que nenhum item da crontab funcionará se você desativar o crond.service.
dbus-org.fedoraproject.FirewallD1.service	sim	Um link simbólico para **firewalld.service**
dbus-org.freedesktop.NetworkManager.service	sim	Um link simbólico para **NetworkManager.service**
dbus-org.freedesktop.nm-dispatcher.service	sim	Um link simbólico para **NetworkManager-dispatcher.service**
firewalld.service	sim	Desabilite **firewalld.service** somente se você não precisar de firewall.

Nome do serviço	Pode ser desativado?	Mais informações
getty@.service	não	Este serviço funciona somente quando é realmente necessário, portanto, não precisa ser desativado.
import-state.service	sim	Desabilite **import-state.service** somente se você não precisar inicializar a partir de um armazenamento em rede.
irqbalance.service	sim	Desabilite **irqbalance.service** somente se você tiver apenas uma CPU. Não desabilite **irqbalance.service** em sistemas com várias CPUs.
kdump.service	sim	Desabilite **kdump.service** somente se você não precisar de relatórios de falhas do kernel.
loadmodules.service	sim	Este serviço não é iniciado a menos que o diretório **/etc/rc.modules** ou **/etc/sysconfig/modules** exista, o que significa que ele não é iniciado em uma instalação mínima do RHEL 8.
lvm2-monitor.service	sim	Desabilite **lvm2-monitor.service** somente se você não usar o Logical Volume Manager (LVM).
microcódigo.serviço	não	Não desabilite o serviço porque ele fornece atualizações do software do microcódigo na CPU.
NetworkManager-dispatcher.service	sim	Desative **NetworkManager-dispatcher.service** somente se você não precisar de notificações sobre mudanças na configuração da rede (por exemplo, em redes estáticas).

Nome do serviço	Pode ser desativado?	Mais informações
NetworkManager-wait-online.service	sim	Desabilite **NetworkManager-wait-online.service** somente se você não precisar de uma conexão de rede em funcionamento disponível logo após a inicialização. Se o serviço estiver habilitado, o sistema não finaliza a inicialização antes que a conexão de rede esteja funcionando. Isto pode prolongar significativamente o tempo de inicialização.
NetworkManager.service	sim	Desabilite **NetworkManager.service** somente se você não precisar de conexão a uma rede.
nis-domainname.service	sim	Desabilite **nis-domainname.service** somente se você não utilizar o Serviço de Informações de Rede (NIS).
rhsmcertd.service	não	
rngd.service	sim	Desabilite **rngd.service** somente se você não precisar de muita entropia em seu sistema, ou se não tiver nenhum tipo de gerador de hardware. Note que o serviço é necessário em ambientes que requerem muita entropia boa, como sistemas usados para geração de certificados X.509 (por exemplo, o servidor FreeIPA).
rsyslog.service	sim	Desabilite **rsyslog.service** somente se você não precisar de registros persistentes, ou se você definir **systemd-journald** para modo persistente.
selinux-autorelabel-mark.service	sim	Desabilite **selinux-autorelabel-mark.service** somente se você não utilizar SELinux.

Nome do serviço	Pode ser desativado?	Mais informações
sshd.service	sim	Desabilite **sshd.service** somente se você não precisar de logins remotos pelo servidor OpenSSH.
sssd.service	sim	Desabilitar **sssd.service** somente se não houver usuários que façam login no sistema pela rede (por exemplo, usando LDAP ou Kerberos). A Red Hat recomenda desabilitar todas as unidades **sssd-*** se você desabilitar **sssd.service**.
syslog.service	sim	Um pseudônimo para **rsyslog.service**
tuned.service	sim	Desabilite **tuned.service** somente se você precisar usar o ajuste de desempenho.
lvm2-lvmpolld.socket	sim	Desabilite **lvm2-lvmpolld.socket** somente se você não usar o Logical Volume Manager (LVM).
dnf-makecache.timer	sim	Desabilite **dnf-makecache.timer** somente se você não precisar que seus metadados do pacote sejam atualizados automaticamente.
temporizador sem limite de ancoragem	sim	Desabilite **unbound-anchor.timer** somente se você não precisar atualizar diariamente a âncora de confiança raiz para Extensões de Segurança DNS (DNSSEC). Esta âncora de confiança raiz é utilizada pela biblioteca Unbound resolver and resolver para validação do DNSSEC.

Para encontrar mais informações sobre um serviço, você pode executar um dos seguintes comandos:

```
$ systemctl cat -YRFFGUNA service_name>
```

```
$ ajuda systemctl -YRFFGUNA service_name>
```

O comando **systemctl cat** fornece o conteúdo do arquivo de serviço localizado sob **/usr/lib/systemd/system/<service>**, assim como todas as anulações aplicáveis. As substituições aplicáveis incluem as substituições do arquivo de unidade do arquivo **/etc/systemd/system/<service>** ou arquivos drop-in de um diretório **unit.type.d** correspondente.

Para mais informações sobre os arquivos drop-in, consulte a página de manual **systemd.unit**.

O comando **systemctl help** mostra a página de homem do serviço em particular.

3.7. RECURSOS ADICIONAIS

Para mais informações sobre o systemd e seu uso no Red Hat Enterprise Linux, veja os recursos listados abaixo.

3.7.1. Documentação Instalada

- **systemctl**(1) - A página de manual do utilitário de linha de comando **systemctl** fornece uma lista completa de opções e comandos suportados.
- **systemd**(1) - A página de manual do gerente de sistema e serviços **systemd** fornece mais informações sobre seus conceitos e documentos disponíveis opções de linha de comando e variáveis de ambiente, arquivos e diretórios de configuração suportados, sinais reconhecidos e opções de kernel disponíveis.
- **systemd-delta**(1) - A página de manual do utilitário **systemd-delta** que permite encontrar arquivos de configuração estendidos e anulados.
- **systemd.directives(7)** - A página do manual chamada **systemd.directives** fornece informações detalhadas sobre as diretrizes do sistema.
- **systemd.unit**(5) - A página de manual chamada **systemd.unit** fornece informações detalhadas sobre os arquivos e documentos da unidade systemd, todas as opções de configuração disponíveis.
- **systemd.service**(5) - A página do manual chamada **systemd.service** documenta o formato dos arquivos das unidades de serviço.
- **systemd.target**(5) - A página do manual chamada **systemd.target** documenta o formato dos arquivos da unidade de destino.
- **systemd.kill**(5) - A página do manual chamada **systemd.kill** documenta a configuração do processo de matança.

3.7.2. Documentação on-line

- systemd Home Page - A home page do projeto fornece mais informações sobre o systemd.

CAPÍTULO 4. INTRODUÇÃO À GESTÃO DE CONTAS DE USUÁRIOS E GRUPOS

O controle de usuários e grupos é um elemento central da administração do sistema Red Hat Enterprise Linux (RHEL). Cada usuário RHEL tem credenciais de login distintas e pode ser atribuído a vários grupos para personalizar seus privilégios de sistema.

Um usuário que cria um arquivo é o proprietário desse arquivo *and* o proprietário do grupo desse arquivo. O arquivo é designado separadamente para ler, escrever e executar permissões para o proprietário, o grupo e aqueles fora desse grupo. O proprietário do arquivo só pode ser alterado pelo usuário do **root**. As permissões de acesso ao arquivo podem ser alteradas tanto pelo usuário **root** quanto pelo proprietário do arquivo. Um usuário regular pode mudar a propriedade do grupo de um arquivo que ele possui para um grupo do qual é membro.

Cada usuário está associado a um número de identificação numérico único chamado *user ID* (**UID**). Cada grupo está associado a um *group ID* (**GID**). Os usuários dentro de um grupo compartilham as mesmas permissões para ler, escrever e executar arquivos de propriedade desse grupo.

4.1. INTRODUÇÃO AOS USUÁRIOS E GRUPOS

Um usuário que cria um arquivo é o proprietário desse arquivo *and* o proprietário do grupo desse arquivo. O arquivo é designado separadamente para ler, escrever e executar permissões para o proprietário, o grupo e aqueles fora desse grupo. O proprietário do arquivo só pode ser alterado pelo usuário do **root**. As permissões de acesso ao arquivo podem ser alteradas tanto pelo usuário **root** quanto pelo proprietário do arquivo. Um usuário regular pode mudar a propriedade do grupo de um arquivo que ele possui para um grupo do qual é membro.

Cada usuário está associado a um número de identificação numérico único chamado *user ID* (**UID**). Cada grupo está associado a um *group ID* (**GID**). Os usuários dentro de um grupo compartilham as mesmas permissões para ler, escrever e executar arquivos de propriedade desse grupo.

4.2. CONFIGURAÇÃO DE IDS DE USUÁRIOS E GRUPOS RESERVADOS

A RHEL reserva IDs de usuários e grupos abaixo de 1000 para usuários e grupos do sistema. Você pode encontrar os IDs de usuário e de grupo reservados no pacote **setup**. Para visualizar os IDs de usuário e grupo reservados, use:

```
cat /usr/share/doc/setup*/uidgid
```

É recomendável atribuir IDs aos novos usuários e grupos a partir de 5000, pois a faixa reservada pode aumentar no futuro.

Para que os IDs atribuídos a novos usuários comecem em 5000 por padrão, modifique os parâmetros **UID_MIN** e **GID_MIN** no arquivo **/etc/login.defs**.

Procedimento

Para modificar os IDs atribuídos a novos usuários, comece em 5000 por padrão, use:

1. Abra o arquivo **/etc/login.defs** em um editor de sua escolha.
2. Encontre as linhas que definem o valor mínimo para a seleção automática da UID.

```
# Min/max values for automatic uid selection in useradd
#
UID_MIN                  1000
```

3. Modificar o valor **UID_MIN** para começar em 5000.

```
# Min/max values for automatic uid selection in useradd
#
UID_MIN                  5000
```

4. Encontre as linhas que definem o valor mínimo para a seleção automática de GID.

```
# Min/max values for automatic gid selection in groupadd
#
GID_MIN                  1000
```

Observe que para usuários e grupos criados antes de você alterar os valores **UID_MIN** e **GID_MIN**, os UIDs e GIDs ainda começam com o valor padrão 1000.

ATENÇÃO

Não aumente as IDs reservadas pelo sistema acima de 1000 mudando **SYS_UID_MAX** para evitar conflitos com sistemas que mantêm o limite de 1000.

4.3. GRUPOS PRIVADOS DE USUÁRIOS

A RHEL utiliza a configuração do sistema *user private group* (**UPG**), o que torna os grupos UNIX mais fáceis de gerenciar. Um grupo privado de usuários é criado sempre que um novo usuário é adicionado ao sistema. O grupo privado de usuário tem o mesmo nome do usuário para o qual foi criado e esse usuário é o único membro do grupo privado de usuário.

As UPGs simplificam a colaboração em um projeto entre vários usuários. Além disso, a configuração do sistema UPG torna seguro definir permissões padrão para um arquivo ou diretório recém-criado, pois permite tanto ao usuário, como ao grupo do qual este usuário faz parte, fazer modificações no arquivo ou diretório.

Uma lista de todos os grupos é armazenada no arquivo de configuração **/etc/group**.

CAPÍTULO 5. GERENCIAMENTO DE CONTAS DE USUÁRIO NO CONSOLE WEB

O console web RHEL oferece uma interface gráfica que lhe permite executar uma ampla gama de tarefas administrativas sem acessar diretamente seu terminal. Por exemplo, você pode adicionar, editar ou remover contas de usuário do sistema.

Depois de ler esta seção, você saberá:

- De onde vêm as contas existentes.
- Como adicionar novas contas.
- Como definir a expiração da senha.
- Como e quando encerrar as sessões dos usuários.

Pré-requisitos

- Configure o console web RHEL. Para obter detalhes, consulte Começando a usar o console web RHEL,
- Acesse o console web da RHEL com uma conta que tenha permissões de administrador atribuídas. Para detalhes, consulte Login no console web da RHEL .

5.1. CONTAS DE USUÁRIO DO SISTEMA GERENCIADAS NO CONSOLE WEB

Com as contas de usuário exibidas no console web RHEL, você pode:

- Autenticar os usuários ao acessar o sistema.
- Defina os direitos de acesso ao sistema.

O console web RHEL exibe todas as contas de usuário localizadas no sistema. Portanto, você pode ver pelo menos uma conta de usuário logo após o primeiro login no console web.

Após o login no console web RHEL, você pode realizar as seguintes operações:

- Criar novas contas de usuários.
- Alterar seus parâmetros.
- Bloquear contas.
- Encerrar sessões de usuários.

5.2. ADICIONANDO NOVAS CONTAS USANDO O CONSOLE WEB

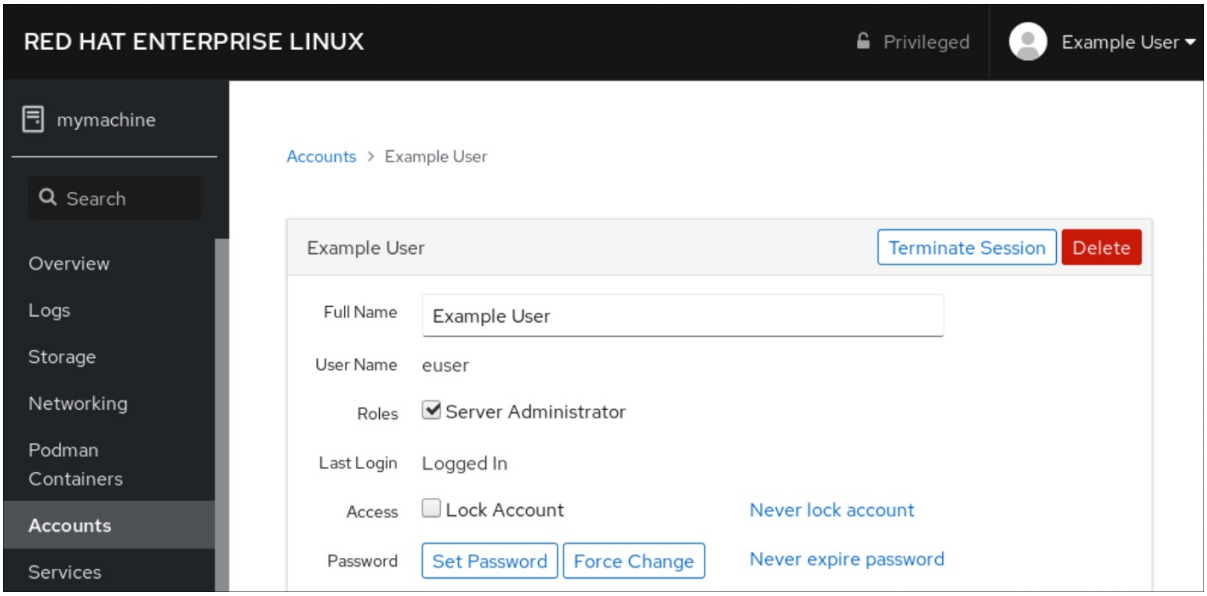
Use os seguintes passos para adicionar contas de usuário ao sistema e definir os direitos de administração das contas através do console web RHEL.

Pré-requisitos

- O console web RHEL deve ser instalado e acessível. Para detalhes, consulte Instalando o console web.

Procedimento

1. Acesse o console web RHEL.
2. Clique em **Contas**.
3. Clique em **Criar nova conta**.
1. No campo **Full Name**, digite o nome completo do usuário.
   O console web RHEL sugere automaticamente um nome de usuário a partir do nome completo e o preenche no campo **User Name**. Se você não quiser usar a convenção de nomenclatura original que consiste na primeira letra do nome e no sobrenome completo, atualize a sugestão.
2. Nos campos **Password/Confirm**, digite a senha e digite-a novamente para verificar se sua senha está correta.
   A barra colorida colocada abaixo dos campos mostra o nível de segurança da senha digitada, o que não permite criar um usuário com uma senha fraca.
1. Clique em **Criar** para salvar as configurações e fechar a caixa de diálogo.
2. Selecione a conta recém-criada.
3. Selecione **Server Administrator** no item **Roles**.



Agora você pode ver a nova conta nas configurações do **Accounts** e pode usar as credenciais para se conectar ao sistema.

5.3. APLICAR A EXPIRAÇÃO DA SENHA NO CONSOLE WEB

Por padrão, as contas de usuário definiram senhas para nunca expirar. Você pode definir que as senhas do sistema expirem após um número definido de dias. Quando a senha expirar, a próxima tentativa de login irá solicitar uma mudança de senha.

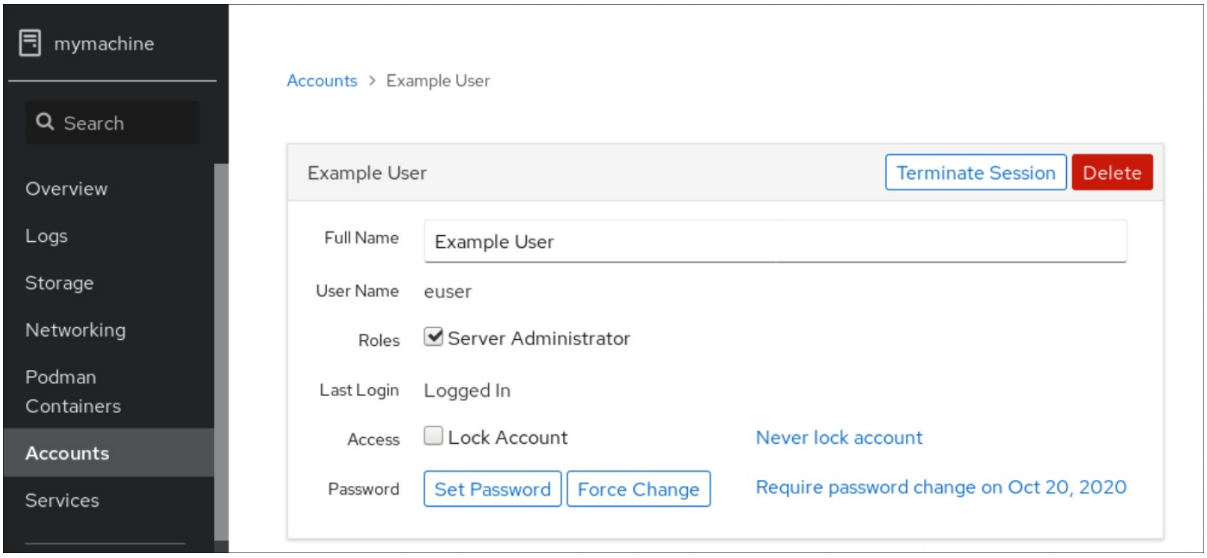
Procedimento

1. Acesse o console web RHEL 8.

2. Clique em **Contas**.
3. Selecione a conta de usuário para a qual deseja fazer valer a expiração da senha.
4. Nas configurações da conta de usuário, clique em **Nunca expirar senha**.
5. Na caixa de diálogo **Password Expiration**, selecione **Require password change every ... days**e digite um número inteiro positivo representando o número de dias em que a senha expira.
1. Clique em **Change**.

Etapas de verificação

- Para verificar se a expiração da senha está definida, abra as configurações da conta.
  O console web RHEL 8 exibe um link com a data de expiração.



5.4. ENCERRAMENTO DAS SESSÕES DO USUÁRIO NO CONSOLE WEB

Um usuário cria sessões de usuário ao entrar no sistema. Encerrar sessões de usuário significa fazer o logout do usuário no sistema. Pode ser útil se você precisar realizar tarefas administrativas sensíveis a mudanças de configuração, por exemplo, atualizações do sistema.

Em cada conta de usuário no console web RHEL 8, você pode encerrar todas as sessões da conta, exceto a sessão do console web que você está utilizando atualmente. Isto evita que você perca o acesso ao seu sistema.

Procedimento

1. Acesse o console web RHEL 8.
2. Clique em **Contas**.
3. Clique na conta de usuário para a qual você deseja encerrar a sessão.
4. Clique em **Encerrar sessão**.
   Se o botão **Terminar Sessão** estiver inativo, o usuário não está logado no sistema.

   O console web RHEL encerra as sessões.

CAPÍTULO 6. GERENCIANDO USUÁRIOS A PARTIR DA LINHA DE COMANDO

Você pode gerenciar usuários e grupos usando a interface de linha de comando (**CLI**). Isto permite adicionar, remover e modificar usuários e grupos de usuários no ambiente Red Hat Enterprise Linux.

6.1. ADICIONANDO UM NOVO USUÁRIO A PARTIR DA LINHA DE COMANDO

Esta seção descreve como usar o utilitário **useradd** para adicionar um novo usuário.

Pré-requisitos

- **Root** acesso

Procedimento

- Para adicionar um novo usuário, use:

```
# useradd options username
```

Substituir *options* pelas opções de linha de comando para o comando **useradd**, e substituir *username* pelo nome do usuário.

Exemplo 6.1. Adicionando um novo usuário

Para adicionar o usuário **sarah** com o ID de usuário **5000**, use:

```
# useradd -u 5000 sarah
```

Etapas de verificação

- Para verificar se o novo usuário é adicionado, use o utilitário **id**.

```
# id sarah
```

A saída retorna:

```
uid=5000(sarah) gid=5000(sarah) grupos=5000(sarah)
```

Recursos adicionais

- **useradd** página do homem

6.2. ADICIONANDO UM NOVO GRUPO A PARTIR DA LINHA DE COMANDO

Esta seção descreve como usar o utilitário **groupadd** para adicionar um novo grupo.

Pré-requisitos

Pré-requisitos

- **Root** acesso

Procedimento

- Para adicionar um novo grupo, use:

```
# groupadd options group-name
```

Substituir *options* pelas opções de linha de comando para o comando **groupadd**, e substituir *group-name* pelo nome do grupo.

Exemplo 6.2. Adicionando um novo grupo

Para adicionar o grupo **sysadmins** com o ID do grupo **5000**, use:

```
# grupadd -g 5000 sysadmins
```

Etapas de verificação

- Para verificar se o novo grupo foi adicionado, use o utilitário **tail**.

```
# cauda /etc/grupo
```

A saída retorna:

```
sysadmins:x:5000:
```

Recursos adicionais

- **groupadd** página do homem

6.3. ADICIONANDO UM USUÁRIO A UM GRUPO A PARTIR DA LINHA DE COMANDO

Esta seção descreve como usar o utilitário **usermod** para adicionar um grupo aos grupos suplementares do usuário.

Pré-requisitos

- **Root** acesso

Procedimento

- Para adicionar um grupo aos grupos suplementares do usuário, use:

```
# usermod --append -G group-name username
```

Substituir *group-name* pelo nome do grupo, e substituir *username* pelo nome do usuário.

Exemplo 6.3. Adicionando um usuário a um grupo

Para adicionar o usuário **sysadmin** ao grupo **system-administrators**, use:

```
# usermod --append -G system-administrators sysadmin
```

Etapas de verificação

- Para verificar os novos grupos é adicionado aos grupos suplementares do usuário **sysadmin**, use:

```
# grupos sysadmin
```

A saída retorna:

```
sysadmin: administradores de sistemas sysadmin
```

6.4. CRIAÇÃO DE UM DIRETÓRIO DE GRUPOS

Sob a configuração do sistema UPG, você pode aplicar o *set-group identification permission* (**setgid** bit) a um diretório. O bit **setgid** torna o gerenciamento de projetos de grupo que compartilham um diretório mais simples. Quando você aplica o bit **setgid** a um diretório, os arquivos criados dentro desse diretório são automaticamente atribuídos a um grupo que possui o diretório. Qualquer usuário que tenha a permissão de escrever e executar dentro deste grupo pode agora criar, modificar e excluir arquivos no diretório.

A seção seguinte descreve como criar diretórios de grupos.

Pré-requisitos

- **Root** acesso

Procedimento

1. Criar um diretório:

```
# mkdir directory-name
```

Substituir *directory-name* pelo nome do diretório.

2. Criar um grupo:

```
# groupadd group-name
```

Substituir *group-name* pelo nome do grupo.

3. Acrescentar usuários ao grupo:

```
# usermod --append -G group-name username
```

Substituir *group-name* pelo nome do grupo, e substituir [role=" *abstract"]e _username* pelo nome do usuário.

4. Associar o usuário e o grupo proprietário do diretório com o grupo *group-name*:

```
# chown group-name directory-name
```

Substituir *group-name* pelo nome do grupo, e substituir *directory-name* pelo nome do diretório.

5. Defina as permissões de escrita para permitir que os usuários criem e modifiquem arquivos e diretórios e defina o bit **setgid** para que essa permissão seja aplicada dentro do diretório *directory-name*:

```
# chmod g rwxs directory-name
```

Substituir *directory-name* pelo nome do diretório.

Agora todos os membros do ***group-name*** grupo pode criar e editar arquivos no ***directory-name*** diretório. Arquivos recém-criados mantêm a propriedade do grupo de ***group-name*** grupo.

Etapas de verificação

- Para verificar a exatidão das permissões estabelecidas, use:

```
# ls -ld directory-name
```

Substituir *directory-name* pelo nome do diretório.

A saída retorna:

```
drwxrwsr-x. 2 raíz group-name 6 Nov 25 08:45 directory-name
```

CAPÍTULO 7. REMOÇÃO DE UM USUÁRIO DE UM GRUPO USANDO A LINHA DE COMANDO

Você pode remover um usuário de um grupo primário ou suplementar substituindo os grupos aos quais o usuário pertence com um novo conjunto de grupos que não contém o grupo do qual você deseja remover o usuário.

7.1. SUBSTITUINDO O GRUPO PRIMÁRIO DE UM USUÁRIO

Esta seção descreve como usar o utilitário **usermod** para substituir o grupo primário do usuário.

Pré-requisitos

- **Root** acesso

Procedimento

- Para anular o grupo primário do usuário, use:

```
# usermod -g group-name username
```

Substituir *group-name* pelo nome do grupo, e substituir *username* pelo nome do usuário.

Exemplo 7.1. Mudando o grupo primário de um usuário

Se o usuário **sarah** pertence aos grupos primários **sarah1**, e você deseja mudar o grupo primário do usuário para **sarah2**, use:

```
# usermod -g sarah2 sarah
```

Etapas de verificação

- Para verificar se o grupo primário do usuário está anulado, use:

```
# grupos sarah
```

A saída retorna:

```
sarah : sarah2
```

7.2. SUBSTITUINDO OS GRUPOS SUPLEMENTARES UM USUÁRIO

Esta seção descreve como usar o utilitário **usermod** para substituir os grupos suplementares do usuário.

Pré-requisitos

- **Root** acesso

Procedimento

- Para anular os grupos suplementares do usuário, use:

```
# usermod -G group-name username
```

Substituir *group-name* pelo nome do grupo, e substituir *username* pelo nome do usuário.

Exemplo 7.2. Mudança do grupo suplementar de um usuário

Se o usuário **sarah** pertence ao grupo **system-administrator** e ao grupo **developer** e você deseja remover o usuário **sarah** do grupo **system-administrator**, você pode fazer isso substituindo a antiga lista de grupos por uma nova. Para fazer isso, use:

```
# usermod -G developer sarah
```

Etapas de verificação

- Para verificar se os grupos suplementares do usuário são anulados, use:

```
# grupos sarah
```

A saída retorna:

```
sarah : desenvolvedor sarah
```

CAPÍTULO 8. CONCEDER ACESSO AO SUDO A UM USUÁRIO

Os administradores do sistema podem conceder acesso a **sudo** para permitir que usuários não root executem comandos administrativos. O comando **sudo** fornece aos usuários acesso administrativo sem utilizar a senha do usuário **root**.

Quando os usuários precisam executar um comando administrativo, eles podem preceder esse comando com **sudo**. O comando é então executado como se eles fossem o usuário **root**.

Esteja ciente das seguintes limitações:

- Somente os usuários listados no arquivo de configuração **/etc/sudoers** podem usar o comando **sudo**.
- O comando é executado na concha do usuário, não na concha **root**.

Pré-requisitos

- **Root** acesso

Procedimento

1. Abra o arquivo **/etc/sudoers**.

```
# visudo
```

   O arquivo **/etc/sudoers** define as políticas aplicadas pelo comando **sudo**.

2. No arquivo **/etc/sudoers** encontram-se as linhas que permitem o acesso **sudo** aos usuários do grupo administrativo **wheel**.

```
## Allows people in group wheel to run all commands
%wheel        ALL=(ALL)       ALL
```

3. Certifique-se de que a linha que começa com **%wheel** não tenha **#** comentar o caráter antes dela.
4. Salve quaisquer mudanças e saia do editor.
5. Adicione usuários que você deseja conceder a **sudo** acesso ao grupo administrativo **wheel**.

```
# usermod --append -G wheel username
```

   Substituir *username* pelo nome do usuário.

Exemplo 8.1. Adicionando um usuário ao grupo de rodas

Para adicionar o usuário **sarah** ao grupo administrativo **wheel**, use:

```
# usermod --append -G wheel sarah
```

Etapas de verificação

- Para verificar se o usuário é adicionado ao grupo administrativo **wheel**, use o utilitário **id**.

```
# id sarah
```

A saída retorna:

```
uid=5000(sarah) gid=5000(sarah) grupos=5000(sarah),10(roda)
```

CAPÍTULO 9. MUDANDO E REDEFININDO A SENHA DE ROOT

Se a senha root existente não for mais satisfatória ou for esquecida, você pode alterá-la ou redefini-la tanto como usuário **root** quanto como usuário não-root.

9.1. MUDANDO A SENHA DE ROOT COMO USUÁRIO ROOT

Esta seção descreve como usar o comando **passwd** para mudar a senha **root** como usuário **root**.

Pré-requisitos

- **Root** acesso

Procedimento

- Para alterar a senha **root**, use:

```
# senha
```

  Você é solicitado a digitar sua senha atual antes de poder alterá-la.

9.2. ALTERAR OU REDEFINIR A SENHA DE ROOT ESQUECIDA COMO USUÁRIO NÃO-ROOT

Esta seção descreve como usar o comando **passwd** para alterar ou redefinir a senha esquecida **root** como um usuário não-root.

Pré-requisitos

- Você pode entrar como um usuário não-rootal.
- Você é um membro do grupo administrativo **wheel**.

Procedimento

- Para alterar ou redefinir a senha **root** como um usuário não-root que pertence ao grupo **wheel**, use:

```
$ sudo passwd raiz
```

  Você é solicitado a digitar sua senha atual não-root antes de poder alterar a senha **root**.

9.3. REDEFINIÇÃO DA SENHA DE ROOT NO BOOT

Se você não conseguir fazer login como usuário não root ou não pertencer ao grupo administrativo **wheel**, você pode redefinir a senha root na inicialização, mudando para um ambiente especializado **chroot jail**.

Procedimento

1. Reinicie o sistema e, na tela de inicialização do GRUB 2, pressione a tecla **e** para interromper o processo de inicialização.

Os parâmetros de inicialização do kernel aparecem.

```
load_video
set gfx_payload=keep
insmod gzio
linux ($root)/vmlinuz-4.18.0-80.e18.x86_64 root=/dev/mapper/rhel-root ro crash\
kernel=auto resume=/dev/mapper/rhel-swap rd.lvm.lv/swap rhgb quiet
initrd ($root)/initramfs-4.18.0-80.e18.x86_64.img $tuned_initrd
```

2. Vá para o final da linha que começa com **linux**.

```
linux ($root)/vmlinuz-4.18.0-80.e18.x86_64 root=/dev/mapper/rhel-root ro crash\
kernel=auto resume=/dev/mapper/rhel-swap rd.lvm.lv/swap rhgb quiet
```

Pressione **Ctrl e** para pular para o final da linha.

3. Adicione **rd.break** ao final da linha que começa com **linux**.

```
linux ($root)/vmlinuz-4.18.0-80.e18.x86_64 root=/dev/mapper/rhel-root ro crash\
kernel=auto resume=/dev/mapper/rhel-swap rd.lvm.lv/swap rhgb quiet rd.break
```

4. Pressione **Ctrl x** para iniciar o sistema com os parâmetros alterados.
O prompt **switch_root** aparece.

5. Remonte o sistema de arquivo como gravável:

```
montar -o remount,rw /sysroot
```

O sistema de arquivo é montado como somente leitura no diretório /**sysroot**. Remontar o sistema de arquivo como gravável permite mudar a senha.

6. Entre no ambiente **chroot**:

```
croot /sysroot
```

O prompt **sh-4.4#** aparece.

7. Redefinir a senha **root**:

```
senhas
```

Siga as instruções exibidas pela linha de comando para finalizar a mudança da senha **root**.

8. Habilitar o processo de reetiquetagem SELinux na próxima inicialização do sistema:

```
toque /.autorelabel
```

9. Sair do ambiente **chroot**:

```
saída
```

10. Saia do prompt **switch_root**:

```
saída
```

11. Aguarde até que o processo de reetiquetagem SELinux esteja concluído. Note que a reetiquetagem de um disco grande pode demorar muito tempo. O sistema é reinicializado automaticamente quando o processo é concluído.

Etapas de verificação

1. Para verificar se a senha **root** foi alterada com sucesso, faça o login como um usuário normal e abra o Terminal.
2. Execute a casca interativa como raiz:

```
$ su
```

3. Digite sua nova senha **root**.
4. Imprimir o nome do usuário associado com o ID do usuário efetivo atual:

```
whoami
```

A saída retorna:

```
raiz
```

CAPÍTULO 10. GERENCIAMENTO DE PERMISSÕES DE ARQUIVOS

10.1. INTRODUÇÃO ÀS PERMISSÕES DE ARQUIVO

Cada arquivo ou diretório tem três níveis de propriedade:

- Proprietário do usuário (**u**).
- Proprietário do grupo (**g**).
- Outros (**o**).

A cada nível de propriedade podem ser atribuídas as seguintes permissões:

- Leia (**r**).
- Escreva (**w**).
- Executar (**x**).

Note que a permissão de execução de um arquivo permite que você execute esse arquivo. A permissão de execução de um diretório permite que você acesse o conteúdo do diretório, mas não o execute.

Quando um novo arquivo ou diretório é criado, o conjunto padrão de permissão é automaticamente atribuído a ele. A permissão padrão para um arquivo ou diretório é baseada em dois fatores:

- Permissão de base.
- O *user file-creation mode mask* (**umask**).

10.1.1. Permissões de base

Sempre que um novo arquivo ou diretório é criado, uma permissão de base é automaticamente atribuída a ele.

As permissões básicas para um arquivo ou diretório podem ser expressas nos valores *symbolic* ou *octal*.

Permission	Symbolic value	Octal value
Sem permissão	---	0
Executar	--x	1
Escreva para	-w-	2
Escrever e executar	-wx	3
Leia	r--	4
Ler e executar	r-x	5

Ler e escrever	rw-	6
Ler, escrever, executar	rwx	7

A permissão básica para um diretório é **777** (**drwxrwxrwx**), que concede a todos as permissões para ler, escrever e executar. Isto significa que o proprietário do diretório, o grupo e outros podem listar o conteúdo do diretório, criar, apagar e editar itens dentro do diretório e descer até ele.

Note que arquivos individuais dentro de um diretório podem ter sua própria permissão que pode impedi-lo de editá-los, apesar de ter acesso irrestrito ao diretório.

A permissão básica para um arquivo é **666** (**-rw-rw-rw-**), que concede a todos as permissões de leitura e escrita. Isto significa que o proprietário do arquivo, o grupo e outros podem ler e editar o arquivo.

Exemplo 1

Se um arquivo tiver as seguintes permissões:

```
$ ls -l
-rwxrw----. 1 sysadmins sysadmins 2 Mar 2 08:43 file
```

- **-** indica que se trata de um arquivo.
- **rwx** indica que o proprietário do arquivo tem permissões para ler, escrever e executar o arquivo.
- **rw-** indica que o grupo tem permissões para ler e escrever, mas não para executar o arquivo.
- **---** indica que outros usuários não têm permissão para ler, escrever, ou executar o arquivo.
- **.** indica que o contexto de segurança da SELinux está definido para o arquivo.

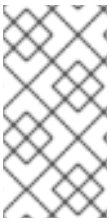
Exemplo 2

Se um diretório tem as seguintes permissões:

```
$ ls -dl directory
drwxr-----. 1 sysadmins sysadmins 2 Mar 2 08:43 directory
```

- **d** indica que se trata de um diretório.
- **rwx** indica que o proprietário do diretório tem as permissões para ler, escrever e acessar o conteúdo do diretório.
  Como proprietário de um diretório, você pode listar os itens (arquivos, subdiretórios) dentro do diretório, acessar o conteúdo desses itens, e modificá-los.
- **r--** indica que o grupo tem permissões para ler, mas não para escrever ou acessar o conteúdo do diretório.
  Como membro do grupo que possui o diretório, você pode listar os itens dentro do diretório. Você não pode acessar informações sobre os itens dentro do diretório ou modificá-los.
- **---** indica que outros usuários não têm permissão para ler, escrever, ou acessar o conteúdo do diretório.
  Como alguém que não é proprietário do usuário, ou como proprietário do grupo do diretório, você não pode listar os itens dentro do diretório, acessar informações sobre esses itens, ou modificá-los.

- **.** indica que o contexto de segurança da SELinux está definido para o diretório.



NOTA

A permissão básica que é automaticamente atribuída a um arquivo ou diretório é **not** a permissão padrão que o arquivo ou diretório acaba por ter. Quando você cria um arquivo ou diretório, a permissão base é alterada pelo site *umask*. A combinação da permissão base e o *umask* cria a permissão padrão para arquivos e diretórios.

10.1.2. Máscara de modo de criação de arquivo de usuário

O *umask* é uma variável que remove automaticamente as permissões do valor da permissão base sempre que um arquivo ou diretório é criado para aumentar a segurança geral de um sistema linux.

O *umask* pode ser expresso em *symbolic* ou *octal*.

Permission	Symbolic value	Octal value
Ler, escrever e executar	rwx	0
Ler e escrever	rw-	1
Ler e executar	r-x	2
Leia	r--	3
Escrever e executar	-wx	4
Escreva para	-w-	5
Executar	--x	6
Sem permissões	---	7

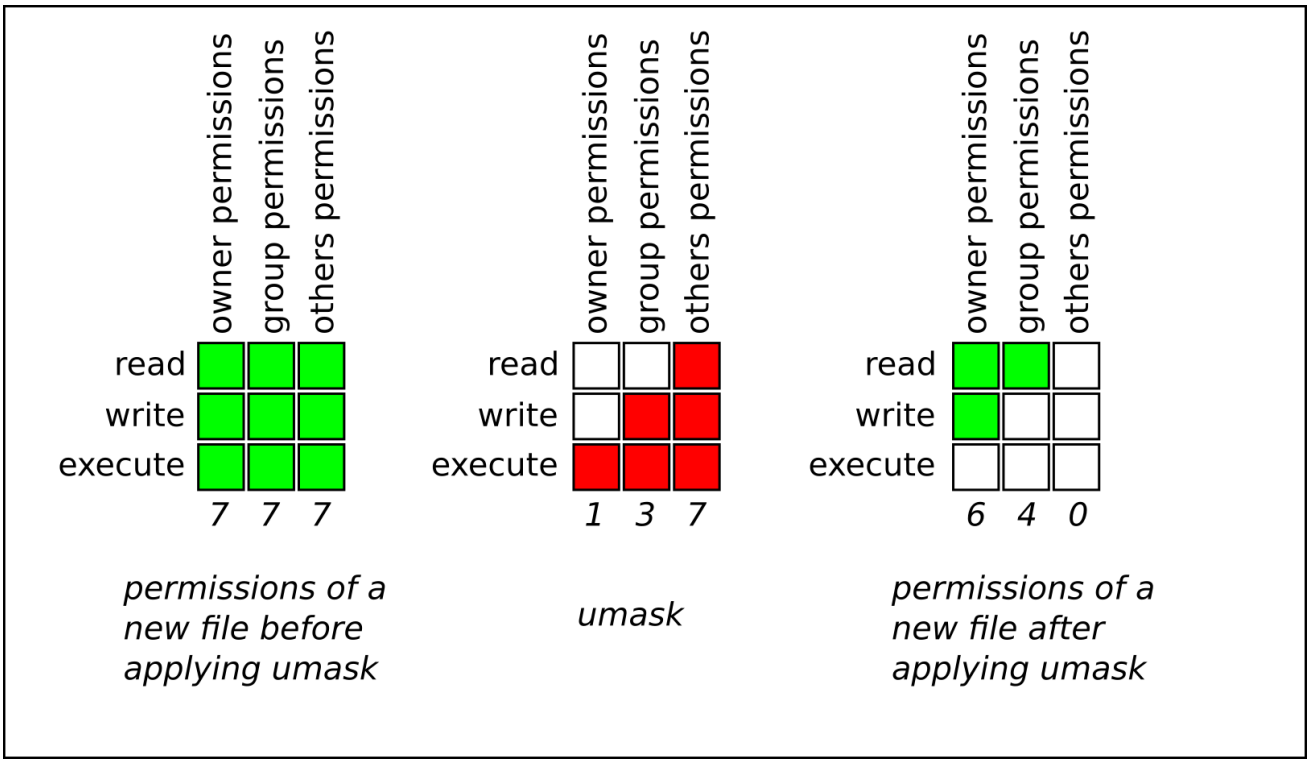
O padrão *umask* para um usuário padrão é **0002**. O padrão *umask* para um usuário **root** é **0022**.

O primeiro dígito do *umask* representa permissões especiais (sticky bit,). Os três últimos dígitos do *umask* representam as permissões que são removidas do proprietário do usuário (**u**), proprietário do grupo (**g**), e outros (**o**) respectivamente.

Exemplo

O exemplo a seguir ilustra como o *umask* com um valor octal de **0137** é aplicado ao arquivo com a permissão básica de **777**, para criar o arquivo com a permissão padrão de **640**.

Figura 10.1. Aplicando a umask ao criar um arquivo



10.1.3. Permissões padrão

A permissão padrão para um novo arquivo ou diretório é determinada através da aplicação da permissão de base *umask*.

Exemplo 1

Quando um **standard user** cria um novo **directory**, o *umask* é definido para **002** (**rwxrwxr-x**), e a permissão básica para um diretório é definida para **777** (**rwxrwxrwx**). Isto traz a permissão padrão para **775** (**drwxrwxr-x**).

	Symbolic value	Octal value
Base permission	rwxrwxrwxrwx	777
Umask	rwxrwxr-x	002
Default permission	rwxrwxr-x	775

Isto significa que o proprietário do diretório e o grupo podem listar o conteúdo do diretório, criar, apagar e editar itens dentro do diretório e descer até ele. Outros usuários só podem listar o conteúdo do diretório e descer para dentro dele.

Exemplo 2

Quando um **standard user** cria um novo **file**, o *umask* é definido para **002** (**rwxrwxr-x**), e a permissão básica para um arquivo é definida para **666** (**rw-rw-rw-**). Isto traz a permissão padrão para **664** (**-rw-rw-r--**).

	Symbolic value	Octal value
Base permission	rw-rw-rw-	666
Umask	rwxrwxr-x	002
Default permission	rw-rw-r--	664

Isto significa que o proprietário do arquivo e o grupo podem ler e editar o arquivo, enquanto outros usuários só podem ler o arquivo.

Exemplo 3

Quando um **root user** cria um novo **directory**, o *umask* é definido para **022** (**rwxr-xr-x**), e a permissão básica para um diretório é definida para **777** (**rwxrwxrwx**). Isto traz a permissão padrão para **755** (**rwxr-xr-x**).

	Symbolic value	Octal value
Base permission	rwxrwxrwxrwx	777
Umask	rwxr-xr-x	022
Default permission	rwxr-xr-x	755

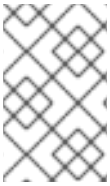
Isto significa que o proprietário do diretório pode listar o conteúdo do diretório, criar, apagar e editar itens dentro do diretório, e descer até ele. O grupo e outros só podem listar o conteúdo do diretório e descer para dentro dele.

Exemplo 4

Quando um **root user** cria um novo **file**, o *umask* é definido para **022** (**rwxr-xr-x**), e a permissão básica para um arquivo é definida para **666** (**rw-rw-rw-**). Isto traz a permissão padrão para **644** (**-rw-r—r--**).

	Symbolic value	Octal value
Base permission	rw-rw-rw-	666
Umask	rwxr-xr-x	022
Default permission	rw-r-r--	644

Isto significa que o proprietário do arquivo pode ler e editar o arquivo, enquanto o grupo e outros só podem ler o arquivo.



NOTA

Por razões de segurança, os arquivos regulares não podem ter permissões de execução por padrão, mesmo que o *umask* esteja configurado para **000** (**rwxrwxrwx**). Entretanto, os diretórios podem ser criados com permissões de execução.

10.2. EXIBIÇÃO DAS PERMISSÕES DOS ARQUIVOS

A seção seguinte descreve como usar o comando **ls** para exibir as permissões de diretórios, arquivos, arquivos dentro de diretórios.

Procedimento

- Para ver as permissões de um determinado diretório, use:

```
$ ls -dl directory-name
```

  Substituir *directory-name* pelo nome do diretório.

- Para ver as permissões de todos os arquivos dentro de um determinado diretório, use:

```
$ ls -l directory-name
```

  Substituir *directory-name* pelo nome do diretório.

- Para ver as permissões de um determinado arquivo, use:

```
$ ls -l file-name
```

  Substituir *file-name* pelo nome do arquivo.

Informações adicionais

- Veja a página de manual **ls** para mais detalhes.

10.3. MUDANÇA DE PERMISSÕES DE ARQUIVO

A seção seguinte descreve como fazê-lo:

- Alterar as permissões dos arquivos usando valores simbólicos.
- Alterar as permissões dos arquivos usando valores octal.

10.3.1. Mudança de permissões de arquivo usando valores simbólicos

Você pode atribuir as seguintes permissões:

- Leia (**r**).
- Escreva (**w**).
- Executar (**x**).

As permissões podem ser atribuídas a:

- Proprietário do usuário (**u**).
- Proprietário do grupo (**g**).
- Outros (**o**).
- Todos (**a**).

Para adicionar ou retirar as permissões, você pode usar os seguintes sinais:

- para adicionar as permissões em cima das permissões existentes.
- **-** para retirar as permissões da permissão existente.
- **=** para omitir as permissões existentes e definir explicitamente as novas permissões.

A seção seguinte descreve como definir e remover permissões de arquivo usando os valores simbólicos.

Procedimento

- Para alterar as permissões de arquivo para um arquivo ou diretório existente, use:

```
$ chmod u=symbolic_value,g symbolic_value,o-symbolic_value file-name
```

Substituir *file-name* pelo nome do arquivo ou diretório, e substituir *symbolic_value* por usuários, grupos e outros com valores simbólicos correspondentes. Veja Seção 10.1.1, “Permissões de base” para mais detalhes.

Exemplo

Para alterar as permissões de arquivo para **my-file.txt** de **664** (**-rw-rw-r--**) para **740** (**-rwx-r---**), use:

```
$ chmod u x,g-w,o= my-file.txt
```

Note que qualquer permissão que não seja especificada após o sinal de igual (**=**) é automaticamente proibida.

- Para definir as mesmas permissões de uso para usuário, grupo, e outros:

```
$ chmod a=symbolic_value file-name
```

Substituir *file-name* pelo nome do arquivo ou diretório, e substituir *symbolic_value* por um valor simbólico. Veja Seção 10.1.1, “Permissões de base” para mais detalhes.

Exemplo

Para definir a permissão para **my-file.txt** para **777** (**-rwxrwxrwx** ou **drwxrwxrwx**), use:

```
$ chmod a=rwx my-file
```

- Para alterar as permissões de um diretório e todos os seus subdiretórios, adicione a opção **-R**:

```
$ chmod -R symbolic_value directory-name
```

Substituir *directory-name* pelo nome do diretório, e substituir *symbolic_value* por um valor simbólico. Veja Seção 10.1.1, "Permissões de base" para mais detalhes.

Exemplo

Para alterar as permissões para /**my-directory**/ e todos os seus subdiretórios de **775** (**drwxrwxr-x**) para **740** (**drwx-r---**), use:

```
$ chmod -R g-wx,o= /my-diretório
```

10.3.2. Mudança de permissões de arquivo usando valores octal

A seção seguinte descreve como usar o comando **chmod** para alterar as permissões de um arquivo ou diretório.

Procedimento

- Para alterar as permissões de arquivo para um arquivo ou diretório existente, use:

  ```
  $ chmod octal_value file-name
  ```

  Substituir *file-name* pelo nome do arquivo ou diretório, e substituir *octal_value* por um valor octal. Veja Seção 10.1.1, "Permissões de base" para mais detalhes.

10.4. EXIBINDO A UMASK

A seção seguinte descreve como fazê-lo:

- Mostrar o valor octal atual do *umask*.
- Mostrar o valor simbólico atual do *umask*.
- Mostrar a base padrão *umask*.

10.4.1. Exibindo o valor octal atual da umask

A seção seguinte descreve como usar o comando **umask** para exibir o atual *umask*.

Procedimento:

- Para exibir o valor octal atual do *umask* para um usuário padrão, use:

  ```
  $ umask
  ```

- Para exibir o valor octal atual do *umask* para um usuário do site **root**, use:

  ```
  $ sudo umask
  ```

  Ou:

  ```
  # umask
  ```

10.4.2. Exibindo o valor simbólico atual da máscara

A seção seguinte descreve como usar o comando **umask** para exibir o atual *umask*.

Procedimento

- Para exibir o valor simbólico atual do *umask*, use:

```
$ umask -S
```

- Para exibir o valor simbólico atual do *umask* para um usuário **root**, use:

```
$ sudo umask -S
```

Ou:

```
# umask -S
```

10.4.3. Exibindo a máscara de bash padrão

Há uma série de conchas que você pode usar, tais como **bash**, **ksh**, **zsh** e **tcsh**.

Esses shells podem se comportar como shells de login ou shells sem login. A shell de login é normalmente invocada abrindo um terminal nativo ou um GUI.

Para determinar se você está executando um comando em um login ou em uma shell sem login, use o comando **echo $0**.

Em **bash** shell, se a saída retornar **bash**, você está executando um comando em uma shell não-login.

```
$ echo $0
bash
```

O padrão *umask* para o shell não-login é definido no arquivo de configuração **/etc/bashrc**.

Se a saída retornar **-bash**, você está executando um comando em uma shell de login.

```
# echo $0
-bash
```

O padrão *umask* para a shell de login é definido no arquivo de configuração **/etc/profile**.

Procedimento

- Para exibir o padrão **bash** *umask* para o shell não-login, use:

```
$ grep umask /etc/bashrc
```

A saída retorna:

```
# By default, we want umask to get set. This sets it for non-login shell.
      umask 002
      umask 022
```

- Para exibir o padrão **bash** *umask* para a shell de login, use:

```
$ grep umask /etc/profile
```

A saída retorna:

```
# By default, we want umask to get set. This sets it for login shell
      umask 002
      umask 022
```

10.5. PREPARANDO A MÁSCARA PARA A ATUAL SESSÃO DE SHELL

A seção seguinte descreve como configurar o *umask* para a atual sessão de shell:

- Usando valores simbólicos.
- Usando valores octálicos.

Note que o *umask* é válido somente durante a atual sessão de shell e volta ao padrão *umask* após a conclusão da sessão.

10.5.1. Definição da umask usando valores simbólicos

A seção seguinte descreve como definir o *umask* com valores simbólicos.

Procedimento

- Para definir ou remover permissões para a atual sessão de shell, você pode usar sinais menos (**-**), mais ( ), e iguais (**=**) em combinação com valores simbólicos.

```
$ umask -S u=symbolic_value,g symbolic_value,o-symbolic_value
```

Substituir *symbolic_value* por usuários, grupos e outros com valores simbólicos. Veja Seção 10.1.2, “Máscara de modo de criação de arquivo de usuário” para mais detalhes.

Exemplo

Se seu atual *umask* está configurado para **113** (**u=rw-,g=rw-,o=r--**) e você deseja configurá-lo para **037** (**u=rwx,g=-r-,o=---**), use:

```
$ umask -S u x,g-w,o=
```

Note que qualquer permissão que não seja especificada após o sinal de igual (**=**) é automaticamente proibida.

- Para definir as mesmas permissões de uso para usuário, grupo, e outros:

```
$ umask a=symbolic_value
```

Substituir *symbolic_value* por um valor simbólico. Veja Seção 10.1.2, “Máscara de modo de criação de arquivo de usuário” para mais detalhes.

Exemplo

Para definir o *umask* para **000** (**u=rwx,g=rwx,o=rwx**), use:

```
$ umask a=rwx
```

Note que o *umask* é válido apenas para a atual sessão de shell.

10.5.2. Ajuste da umask usando valores octal

A seção seguinte descreve como definir o *umask* com valores octal.

Procedimento

- Para definir o *umask* para a atual sessão de shell usando valores octal, use:

  ```
  $ umask octal_value
  ```

  Substituir *octal_value* por um valor octal. Veja Seção 10.1.2, “Máscara de modo de criação de arquivo de usuário” para mais detalhes.

Note que o *umask* é válido apenas para a atual sessão de shell.

10.6. ALTERANDO A UMASK PADRÃO

A seção seguinte descreve como fazê-lo:

- Mude a base padrão *umask* para a shell não-login.
- Mude a base padrão *umask* para a shell de login.
- Mude a base padrão *umask* para um usuário específico.
- Definir permissões padrão para diretórios residenciais recém-criados.

Pré-requisitos

- **Root** acesso.

10.6.1. Alterando a máscara padrão para a casca de não-login

A seção seguinte descreve como alterar o padrão **bash** *umask* para usuários padrão.

Procedimento

1. Como **root**, abra o arquivo **/etc/bashrc** em um editor de sua escolha.
2. Modifique as seguintes seções para definir uma nova base padrão *umask*:

   ```
   if [ $UID -gt 199 ] && [ “id -gn” = “id -un” ]; then
     umask 002
   else
     umask 022
   fi
   ```

Substituir o valor octal padrão do *umask* (**002**) por outro valor octal. Veja Seção 10.1.2, "Máscara de modo de criação de arquivo de usuário" para mais detalhes.

3. Economize as mudanças.

10.6.2. Alterando a máscara padrão para a shell de login

A seção seguinte descreve como alterar o padrão **bash** *umask* para o usuário **root**.

Procedimento

1. Como **root**, abra o arquivo **/etc/profile** em um editor de sua escolha.
2. Modifique as seguintes seções para definir uma nova base padrão *umask*:

```
if [ $UID -gt 199 ] && [ "/usr/bin/id -gn" = "/usr/bin/id -un" ]; then
    umask 002
else
    umask 022
fi
```

Substituir o valor octal padrão do *umask* (**022**) por outro valor octal. Veja Seção 10.1.2, "Máscara de modo de criação de arquivo de usuário" para mais detalhes.

3. Economize as mudanças.

10.6.3. Alterando a máscara padrão para um usuário específico

A seção seguinte descreve como alterar o padrão *umask* para um usuário específico.

Procedimento

- Coloque a linha que especifica o valor octal do *umask* no arquivo **.bashrc** para o usuário em particular.

```
$ echo 'umask octal_value' >> /home/username/.bashrc
```

Substituir *octal_value* por um valor octal e substituir *username* pelo nome do usuário. Veja Seção 10.1.2, "Máscara de modo de criação de arquivo de usuário" para mais detalhes.

10.6.4. Definindo UMASK padrão para diretórios residenciais recém-criados

A seção seguinte descreve como alterar as permissões que especificam o *UMASK* para diretórios residenciais de usuários recém-criados.

Procedimento

1. Como **root**, abra o arquivo **/etc/login.defs** em um editor de sua escolha.
2. Modifique a seção a seguir para definir um novo padrão *UMASK*:

```
# The permission mask is initialized to this value. If not specified,
# the permission mask will be initialized to 022.
UMASK 077
```

Substituir o valor octal padrão (**077**) por outro valor octal. Veja Seção 10.1.2, "Máscara de modo de criação de arquivo de usuário" para mais detalhes.

3. Economize as mudanças.

10.7. LISTA DE CONTROLE DE ACESSO

Tradicionalmente, cada arquivo e diretório pode ter apenas um proprietário de usuário e um proprietário de grupo por vez. Se você quiser aplicar um conjunto mais específico de permissões a um arquivo ou diretório (permitir que certos usuários fora do grupo tenham acesso a um arquivo específico dentro de um diretório, mas não a outros arquivos) sem alterar a propriedade e as permissões de um arquivo ou diretório, você pode usar as listas de controle de acesso (ACL).

A seção seguinte descreve como fazê-lo:

- Exibir a LCA atual.
- Configure a LCA.

10.7.1. Exibindo a LCA atual

A seção seguinte descreve como exibir a LCA atual.

Procedimento

- Para exibir a ACL atual para um determinado arquivo ou diretório, use:

```
$ getfacl file-name
```

Substituir *file-name* pelo nome do arquivo ou diretório.

10.7.2. Ajuste do LCA

A seção seguinte descreve como definir o LCA.

Pré-requisitos

- **Root** acesso

Procedimento

- Para configurar o ACL para um arquivo ou diretório, use:

```
# setfacl -m u:username:symbolic_value file-name
```

Substituir *username* pelo nome do usuário, *symbolic_value* por um valor simbólico, e *file-name* pelo nome do arquivo ou diretório. Para mais informações, consulte a página de manual **setfacl**.

Exemplo

O exemplo seguinte descreve como modificar as permissões para o arquivo **group-project** de propriedade do usuário **root** que pertence ao grupo **root** para que este arquivo seja:

- Não pode ser executado por ninguém.

- O usuário **andrew** tem a permissão **rw-**.
- O usuário **susan** tem a permissão **---**.
- Outros usuários têm a permissão **r--**.

Procedimento

```
# setfacl -m u:andrew:rw- group-project
# setfacl -m u:susan:--- group-project
```

Etapas de verificação

- Para verificar se o usuário **andrew** tem a permissão **rw-**, o usuário **susan** tem a permissão **---**, e outros usuários têm a permissão **r--**, use:

  ```
  $ getfacl grupo-projeto
  ```

  A saída retorna:

  ```
  # file: group-project
  # owner: root
  # group: root
  user:andrew:rw-
  user:susan:---
  group::r--
  mask::rw-
  other::r--
  ```

CAPÍTULO 11. USANDO O CONJUNTO CHRONY PARA CONFIGURAR O NTP

11.1. INTRODUÇÃO À CONFIGURAÇÃO DO NTP COM CHRONY

O registro preciso do tempo é importante por uma série de razões em TI. Em rede, por exemplo, são necessários carimbos de tempo precisos em pacotes e registros. Em sistemas Linux, o protocolo **NTP** é implementado por um daemon rodando no espaço do usuário.

O daemon de espaço do usuário atualiza o relógio do sistema rodando no kernel. O relógio do sistema pode manter o tempo usando várias fontes de relógio. Normalmente, é usado o *Time Stamp Counter* (**TSC**). O TSC é um registro de CPU que conta o número de ciclos desde que foi reiniciado pela última vez. Ele é muito rápido, tem uma alta resolução e não há interrupções.

No Red Hat Enterprise Linux 8, o protocolo **NTP** é implementado pelo daemon **chronyd**, disponível nos repositórios do pacote **chrony**.

Estas seções descrevem o uso do **chrony** suíte.

11.2. INTRODUÇÃO À SUÍTE CHRONY

chrony é uma implementação do **Network Time Protocol (NTP)**. Você pode usar o **chrony**:

- Para sincronizar o relógio do sistema com os servidores **NTP**
- Para sincronizar o relógio do sistema com um relógio de referência, por exemplo, um receptor GPS
- Para sincronizar o relógio do sistema com uma entrada de tempo manual
- Como um servidor **NTPv4(RFC 5905)** ou par para fornecer um serviço de tempo a outros computadores na rede

chrony funciona bem em uma ampla gama de condições, incluindo conexões de rede intermitentes, redes muito congestionadas, temperaturas variáveis (os relógios comuns de computador são sensíveis à temperatura), e sistemas que não funcionam continuamente, ou funcionam em uma máquina virtual.

A precisão típica entre duas máquinas sincronizadas pela Internet está dentro de alguns milissegundos, e para máquinas em uma LAN dentro de dezenas de microssegundos. O registro de tempo do hardware ou um relógio de referência de hardware pode melhorar a precisão entre duas máquinas sincronizadas a um nível de sub-microssegundo.

chrony consiste em **chronyd**, um daemon que funciona no espaço do usuário, e **chronyc**Um programa de linha de comando que pode ser usado para monitorar o desempenho do **chronyd** e para alterar vários parâmetros operacionais quando ele está em execução.

O **chrony** daemon, **chronyd**, pode ser monitorado e controlado pelo utilitário de linha de comando **chronyc**. Este utilitário fornece um prompt de comando que permite inserir uma série de comandos para consultar o estado atual de **chronyd** e fazer alterações em sua configuração. Por padrão, **chronyd** aceita apenas comandos de uma instância local de **chronyc**mas pode ser configurado para aceitar comandos de monitoramento também de hosts remotos. O acesso remoto deve ser restrito.

11.2.1. Usando chronyc para controlar chronyd

Para fazer mudanças na instância local de **chronyd** usando o utilitário de linha de comando **chronyc** no modo interativo, digite o seguinte comando como **root**:

```
# cronico
```

chronyc deve funcionar como **root** para que alguns dos comandos restritos possam ser usados.

O **chronyc** será exibido como se segue:

```
chronyc>
```

Você pode digitar **help** para listar todos os comandos.

A utilidade também pode ser invocada em modo de comando não-interativo se chamada juntamente com um comando como a seguir:

```
chronyc command
```

NOTA

Alterações feitas usando **chronyc** não são permanentes, eles serão perdidos após um reinício do **chronyd**. Para mudanças permanentes, modificar **/etc/chrony.conf**.

11.3. DIFERENÇAS ENTRE O CHRONY E A NTP

Network Time Protocol (NTP) tem duas implementações diferentes com funcionalidades básicas similares - **ntp** e **chrony**.

Ambos **ntp** e **chrony** podem operar como um cliente **NTP** a fim de sincronizar o relógio do sistema com os servidores **NTP** e podem operar como um servidor **NTP** para outros computadores na rede. Cada implementação tem algumas características únicas. Para comparação de **ntp** e **chrony**Ver Comparação das implementações do NTP.

A configuração específica para um cliente **NTP** é idêntica na maioria dos casos. **NTP** Os servidores são especificados com a diretiva **server**. Um pool de servidores pode ser especificado com a diretriz **pool**.

A configuração específica para um servidor **NTP** difere em como o acesso do cliente é controlado. Por padrão, **ntpd** responde às solicitações do cliente a partir de qualquer endereço. O acesso pode ser restrito com a diretiva **restrict**, mas não é possível desativar o acesso completamente se **ntpd** usar qualquer servidor como cliente. **chronyd** não permite acesso por padrão e opera apenas como um cliente **NTP**. Para fazer **chrony** operar como um servidor **NTP**, você precisa especificar alguns endereços dentro da diretiva **allow**.

ntpd e **chronyd** diferem também no comportamento padrão com relação às correções do relógio do sistema. **ntpd** corrige o relógio por etapas quando o offset é maior que 128 milissegundos. Se o offset for maior que 1000 segundos, **ntpd** sai, a menos que seja a primeira correção do relógio e **ntpd** é iniciado com a opção **-g**. **chronyd** não faz o passo padrão do relógio, mas o arquivo padrão **chrony.conf** fornecido no pacote **chrony** permite passos nas três primeiras atualizações do relógio. Depois disso, todas as correções são feitas lentamente, acelerando ou diminuindo a velocidade do relógio. O comando **chronyc makestep** pode ser emitido para forçar o **chronyd** a pisar o relógio a qualquer momento.

11.4. MIGRANDO PARA O CRONO

No Red Hat Enterprise Linux 7, os usuários podem escolher entre **ntp** e **chrony** para assegurar uma cronometragem precisa. Para diferenças entre **ntp** e **chrony**, **ntpd** e **chronyd**, ver Diferenças entre ntpd e chronyd.

No Red Hat Enterprise Linux 8, **ntp** não é mais suportado **chrony** é ativado por padrão. Por este motivo, você pode precisar migrar de **ntp** para **chrony**.

Migrando de **ntp** para **chrony** é simples, na maioria dos casos. Os nomes correspondentes dos programas, arquivos de configuração e serviços são:

Tabela 11.1. Nomes correspondentes dos programas, arquivos de configuração e serviços ao migrar de ntp para chrony

ntp nome	nome de cristo
/etc/ntp.conf	/etc/chrony.conf
/etc/ntp/keys	/etc/chrony.keys
ntpd	chronyd
ntpq	chronyc
ntpd.service	chronyd.service
ntp-wait.service	chrony-wait.service

O **ntpdate** e **sntp** que estão incluídas na distribuição **ntp**, podem ser substituídas por **chronyd** usando a opção **-q** ou a opção **-t**. A configuração pode ser especificada na linha de comando para evitar a leitura de **/etc/chrony.conf**. Por exemplo, em vez de rodar **ntpdate ntp.example.com**, **chronyd** poderia ser iniciado como:

```
# chronyd -q 'server ntp.example.com iburst'
2018-05-18T12:37:43Z chronyd version 3.3 starting (+CMDMON +NTP +REFCLOCK +RTC
+PRIVDROP +SCFILTER +SIGND +ASYNCDNS +SECHASH +IPV6 +DEBUG)
2018-05-18T12:37:43Z Initial frequency -2.630 ppm
2018-05-18T12:37:48Z System clock wrong by 0.003159 seconds (step)
2018-05-18T12:37:48Z chronyd exiting
```

O **ntpstat** que anteriormente estava incluído no pacote **ntp** e suportava apenas **ntpd**, agora suporta tanto **ntpd** como **chronyd**. Ele está disponível no pacote **ntpstat**.

11.4.1. Roteiro migratório

Um script Python chamado **ntp2chrony.py** está incluído na documentação do pacote **chrony** (**/usr/share/doc/chrony**). O script converte automaticamente uma configuração existente **ntp** para **chrony**. Ele suporta as diretrizes e opções mais comuns no arquivo **ntp.conf**. Quaisquer linhas que são ignoradas na conversão são incluídas como comentários no arquivo **chrony.conf** gerado para revisão. As chaves que são especificadas no arquivo **ntp**, mas não são marcadas como chaves confiáveis no **ntp.conf**, são incluídas no arquivo gerado **chrony.keys** como comentários.

Por padrão, o script não sobrescreve nenhum arquivo. Se **/etc/chrony.conf** ou **/etc/chrony.keys** já existirem, a opção **-b** pode ser usada para renomear o arquivo como backup. O script suporta outras opções. A opção **--help** imprime todas as opções suportadas.

Um exemplo de uma invocação do roteiro com o padrão **ntp.conf** fornecido no pacote **ntp** é:

```
# python3 /usr/share/doc/chrony/ntp2chrony.py -b -v
Reading /etc/ntp.conf
Reading /etc/ntp/crypto/pw
Reading /etc/ntp/keys
Writing /etc/chrony.conf
Writing /etc/chrony.keys
```

A única diretriz ignorada neste caso é **disable monitor**, que tem um equivalente cronológico na diretriz **noclientlog**, mas foi incluída no padrão **ntp.conf** apenas para mitigar um ataque de amplificação.

O arquivo **chrony.conf** gerado normalmente inclui uma série de diretivas **allow** correspondentes às linhas restritas em **ntp.conf**. Se você não quiser executar **chronyd** como um servidor **NTP**, remova todas as diretivas **allow** de **chrony.conf**.

11.4.2. O papel do Timesync

Note que o uso da função timesync em seu sistema Red Hat Enterprise Linux 7 facilita a migração para **chrony**porque você pode usar o mesmo playbook em todas as versões do RHEL começando pelo RHEL 6, independentemente de o sistema usar **ntp** ou **chrony** para implementar o protocolo NTP.

Recursos adicionais

- Para uma referência detalhada sobre as variáveis de função **timesync**, instale o pacote **rhel-system-roles**, e veja os arquivos **README.md** ou **README.html** no diretório **/usr/share/doc/rhel-system-roles/timesync**.
- Para mais informações sobre os papéis do Sistema RHEL, veja Introdução aos papéis do Sistema RHEL.

11.5. CONFIGURANDO O CHRONY

O arquivo de configuração padrão para **chronyd** é **/etc/chrony.conf**. A opção **-f** pode ser usada para especificar um caminho alternativo para o arquivo de configuração. Consulte a página de manual **chrony.conf(5)** para mais opções. Para uma lista completa das diretrizes que podem ser usadas, veja O arquivo de configuração chronyd.

Abaixo está uma seleção de opções de configuração do **chronyd**:

Comentários

Os comentários devem ser precedidos de #, %, ; ou !

permitir

Opcionalmente, especificar um host, sub-rede ou rede a partir da qual seja possível conectar **NTP** a uma máquina atuando como servidor **NTP**. O padrão é não permitir conexões.
Exemplos:

```
permitir 192.0.2.0/24
```

Use este comando para conceder acesso a uma rede específica.

```
allow 2001:0db8:85a3::8a2e:0370:7334
```

Use este comando para conceder acesso a um **IPv6**.

A porta UDP número 123 precisa estar aberta no firewall a fim de permitir o acesso do cliente:

```
# firewall-cmd --zone=public --add-port=123/udp
```

Se você quiser abrir a porta 123 permanentemente, use a opção **--permanent**:

```
# firewall-cmd --permanent --zone=public --add-port=123/udp
```

cmdallow

Isto é semelhante à diretiva **allow** (ver seção **allow**), exceto que ela permite o controle de acesso (em vez de **NTP** acesso do cliente) a uma determinada sub-rede ou host. (Por "controle de acesso", entende-se que **chronyc** pode ser executado nesses hosts e conectado com sucesso a **chronyd** neste computador) A sintaxe é idêntica. Há também uma diretiva **cmddeny all** com comportamento semelhante ao da diretiva **cmdallow all**.

dumpdir

Caminho até o diretório para salvar o histórico de medições através das reinicializações do **chronyd** (assumindo que nenhuma mudança seja feita no comportamento do relógio do sistema enquanto ele não estiver funcionando). Se esta capacidade for usada (através do comando **dumponexit** no arquivo de configuração, ou do comando **dump** em **chronyc**), o comando **dumpdir** deve ser usado para definir o diretório onde os históricos de medição são salvos.

dumponexit

Se este comando estiver presente, indica que **chronyd** deve salvar o histórico de medição para cada uma de suas fontes de tempo registradas sempre que o programa sair. (Veja o comando **dumpdir** acima).

hwtimestamp

A diretiva **hwtimestamp** permite o registro de tempo do hardware para uma sincronização extremamente precisa. Para mais detalhes, consulte a página do manual **chrony.conf(5)**.

local

A palavra-chave **local** é usada para permitir que **chronyd** apareça sincronizado em tempo real do ponto de vista dos clientes que o pesquisam, mesmo que não tenha uma fonte de sincronização atual. Esta opção é normalmente usada no computador "master" em uma rede isolada, onde vários computadores são necessários para sincronizar entre si, e o "master" é mantido em linha com o tempo real por entrada manual.
Um exemplo do comando é:

```
estrato local 10
```

Um grande valor de 10 indica que o relógio está tão distante de um relógio de referência que seu tempo não é confiável. Se o computador alguma vez tiver acesso a outro computador que esteja finalmente sincronizado a um relógio de referência, ele estará quase certamente em um estrato inferior a 10. Portanto, a escolha de um valor alto como 10 para o comando **local** evita que o próprio tempo da máquina seja confundido com o tempo real, caso ele venha a vazar para clientes que tenham visibilidade de servidores reais.

log

O comando **log** indica que certas informações devem ser registradas. Ele aceita as seguintes opções:

medidas

Esta opção registra as medidas brutas **NTP** e informações relacionadas em um arquivo chamado **measurements.log**.

estatísticas

Esta opção registra informações sobre o processamento da regressão em um arquivo chamado **statistics.log**.

rastreamento

Esta opção registra mudanças na estimativa da taxa de ganho ou perda do sistema, e quaisquer alterações feitas, em um arquivo chamado **tracking.log**.

rtc

Esta opção registra informações sobre o relógio em tempo real do sistema.

relógios

Esta opção registra as medidas do relógio de referência bruto e filtrado em um arquivo chamado **refclocks.log**.

tempcomp

Esta opção registra as medições de temperatura e as compensações de taxa do sistema em um arquivo chamado **tempcomp.log**.
Os arquivos de log são escritos no diretório especificado pelo comando **logdir**.

Um exemplo do comando é:

```
acompanhamento estatístico das medidas de registro
```

logdir

Esta diretiva permite especificar o diretório onde os arquivos de registro são escritos.
Um exemplo do uso desta diretriz é:

```
logdir /var/log/chrony
```

makestep

Normalmente, **chronyd** fará com que o sistema corrija gradualmente qualquer compensação de tempo, diminuindo ou acelerando o relógio conforme necessário. Em certas situações, o relógio do sistema pode estar tão à deriva que este processo de giro demoraria muito tempo para corrigir o relógio do sistema. Esta diretriz força o **chronyd** a pisar no relógio do sistema se o ajuste for maior que um valor limite, mas somente se não houver mais atualizações do relógio desde que **chronyd** foi iniciado do que um limite especificado (um valor negativo pode ser usado para desabilitar o limite). Isto é particularmente útil quando se usa um relógio de referência, porque a diretiva **initstepslew** só funciona com fontes **NTP**.
Um exemplo do uso desta diretriz é:

```
makestep 1000 10
```

Isto faria com que o relógio do sistema passasse se o ajuste fosse maior que 1000 segundos, mas somente nas primeiras dez atualizações do relógio.

maxchange

Esta diretiva estabelece a compensação máxima permitida corrigida em uma atualização do relógio. A verificação é realizada somente após o número especificado de atualizações para permitir um grande ajuste inicial do relógio do sistema. Quando ocorrer um deslocamento maior que o máximo especificado, ele será ignorado para o número especificado de vezes e então **chronyd** desistirá e sairá (um valor negativo pode ser usado para nunca sair). Em ambos os casos, uma mensagem é enviada ao syslog.
Um exemplo do uso desta diretriz é:

```
maxchange 1000 1 2
```

Após a primeira atualização do relógio, **chronyd** irá verificar o offset em cada atualização do relógio, ignorará dois ajustes maiores que 1000 segundos e sairá em outro.

maxupdateskew

Uma das tarefas do **chronyd** é saber quão rápido ou lento é o relógio do computador em relação a suas fontes de referência. Além disso, ele computa uma estimativa dos limites de erro em torno do valor estimado.
Se o intervalo de erro for muito grande, indica que as medidas ainda não foram estabelecidas, e que a taxa estimada de ganho ou perda não é muito confiável.

O parâmetro **maxupdateskew** é o limiar para determinar se uma estimativa não é confiável demais para ser usada. Por padrão, o limite é de 1000 ppm.

O formato da sintaxe é:

```
maxupdateskew skew-in-ppm
```

Os valores típicos para *skew-in-ppm* podem ser 100 para uma conexão discada para servidores através de uma linha telefônica, e 5 ou 10 para um computador em uma LAN.

Deve-se notar que este não é o único meio de proteção contra o uso de estimativas não confiáveis. Em todos os momentos, **chronyd** mantém um registro tanto da taxa de ganho ou perda estimada, quanto do erro vinculado à estimativa. Quando uma nova estimativa é gerada após outra medição de uma das fontes, um algoritmo de combinação ponderada é usado para atualizar a estimativa principal. Portanto, se **chronyd** tiver uma estimativa mestre altamente confiável e uma nova estimativa for gerada com grandes limites de erro, a estimativa mestre existente dominará na nova estimativa mestre.

minsources

A diretiva **minsources** estabelece o número mínimo de fontes que precisam ser consideradas como selecionáveis no algoritmo de seleção de fontes antes que o relógio local seja atualizado.
O formato da sintaxe é:

```
minsources number-of-sources
```

Por padrão, *number-of-sources* é 1. A definição de fontes mínimas para um número maior pode ser usada para melhorar a confiabilidade, porque várias fontes precisarão corresponder umas com as outras.

noclientlog

Esta diretiva, que não aceita argumentos, especifica que os acessos dos clientes não devem ser registrados. Normalmente eles são registrados, permitindo que estatísticas sejam relatadas usando o comando de clientes em **chronyc** e permitindo aos clientes usar o modo intercalado com a opção

xleave na diretiva **server**.

resselecionador

Quando **chronyd** selecionar a fonte de sincronização das fontes disponíveis, preferirá aquela com distância mínima de sincronização. Entretanto, para evitar a re-seleção freqüente quando há fontes com distância semelhante, uma distância fixa é adicionada à distância para fontes que atualmente não são selecionadas. Isto pode ser definido com a opção **reselectdist**. Por padrão, a distância é de 100 microssegundos.
O formato da sintaxe é:

```
resselecionador dist-in-seconds
```

peso estrato

A diretiva **stratumweight** define a distância que deve ser adicionada por estrato à distância de sincronização quando **chronyd** seleciona a fonte de sincronização a partir das fontes disponíveis.
O formato da sintaxe é:

```
peso estrato dist-in-seconds
```

Por padrão, *dist-in-seconds* é de 1 milissegundo. Isto significa que as fontes com estrato mais baixo são geralmente preferidas às fontes com estrato mais alto, mesmo quando sua distância é significativamente pior. A definição de **stratumweight** a 0 faz com que **chronyd** ignore o estrato ao selecionar a fonte.

rtcfile

A diretiva **rtcfile** define o nome do arquivo no qual **chronyd** pode salvar parâmetros associados ao rastreamento da precisão do relógio em tempo real (RTC) do sistema.
O formato da sintaxe é:

```
rtcfile /var/lib/chrony/rtc
```

chronyd salva informações neste arquivo quando ele sai e quando o comando **writertc** é emitido em **chronyc**. A informação guardada é o erro do RTC em alguma época, aquela época (em segundos desde 1 de janeiro de 1970), e a taxa na qual o RTC ganha ou perde tempo. Nem todos os relógios em tempo real são suportados, pois seu código é específico do sistema. Observe que se esta diretriz for usada, então o relógio em tempo real não deve ser ajustado manualmente, pois isso interferiria com **chrony**A necessidade de medir a taxa de variação do relógio em tempo real se o relógio foi ajustado em intervalos aleatórios.

rtcsync

A diretiva **rtcsync** está presente no arquivo **/etc/chrony.conf** por padrão. Isto informará ao kernel que o relógio do sistema é mantido sincronizado e o kernel atualizará o relógio em tempo real a cada 11 minutos.

11.5.1. Configuração de chrony para segurança

chronyc pode acessar **chronyd** de duas maneiras:

- Protocolo Internet, IPv4 ou IPv6.
- Soquete de domínio Unix, que é acessível localmente pelo usuário **root** ou **chrony**.

Por padrão, **chronyc** se conecta com o soquete de domínio Unix. O caminho padrão é

/var/run/chrony/chronyd.sock. Se esta conexão falhar, o que pode acontecer, por exemplo, quando **chronyc** está funcionando sob um usuário não-rootal, **chronyc** tenta se conectar ao 127.0.0.1 e depois ::1.

Somente os seguintes comandos de monitoramento, que não afetam o comportamento do **chronyd**, são permitidos a partir da rede:

- atividade
- lista manual
- rtcdata
- suavização
- fontes
- sourcestats
- rastreamento
- waitsync

O conjunto de hosts de onde **chronyd** aceita estes comandos pode ser configurado com a diretiva **cmdallow** no arquivo de configuração de **chronyd**, ou com o comando **cmdallow** em **chronyc**. Por padrão, os comandos são aceitos somente do localhost (127.0.0.1 ou ::1).

Todos os outros comandos são permitidos somente através do soquete de domínio Unix. Quando enviado através da rede, **chronyd** responde com um erro **Not authorised**, mesmo que seja do localhost.

Acesso remoto ao chronyd com chronyc

1. Permitir o acesso tanto de endereços IPv4 como IPv6, adicionando o seguinte ao arquivo **/etc/chrony.conf**:

   ```
   bindcmdaddress 0.0.0.0
   ```

   ou

   ```
   bindcmdaddress ::
   ```

2. Permitir comandos a partir do endereço IP remoto, rede ou sub-rede, utilizando a diretiva **cmdallow**.
   Adicione o seguinte conteúdo ao arquivo **/etc/chrony.conf**:

   ```
   cmdallow 192.168.1.0/24
   ```

3. Abra a porta 323 no firewall para se conectar a partir de um sistema remoto.

   ```
   # firewall-cmd --zone=public --add-port=323/udp
   ```

   Se você quiser abrir a porta 323 permanentemente, use o site **--permanent**.

   ```
   # firewall-cmd --permanent --zone=public --add-port=323/udp
   ```

Note que a diretiva **allow** é para acesso **NTP** enquanto que a diretiva **cmdallow** é para permitir o recebimento de comandos remotos. É possível fazer estas mudanças temporariamente usando **chronyc** funcionando localmente. Edite o arquivo de configuração para fazer alterações permanentes.

11.6. USANDO O CHRONY

11.6.1. Instalando o chrony

O **chrony** é instalado por default no Red Hat Enterprise Linux. Para garantir que assim seja, execute o seguinte comando como **root**:

```
# yum instalar chrony
```

O local padrão para o **chrony** daemon é **/usr/sbin/chronyd**. O utilitário de linha de comando será instalado para **/usr/bin/chronyc**.

11.6.2. Verificação do status de chronyd

Para verificar o status de **chronyd**, emita o seguinte comando:

```
$ systemctl status chronyd
chronyd.service - NTP client/server
   Loaded: loaded (/usr/lib/systemd/system/chronyd.service; enabled)
   Active: active (running) since Wed 2013-06-12 22:23:16 CEST; 11h ago
```

11.6.3. Iniciando chronyd

Para iniciar **chronyd**, emita o seguinte comando como **root**:

```
# systemctl start chronyd
```

Para garantir que **chronyd** comece automaticamente no início do sistema, emita o seguinte comando como **root**:

```
# systemctl habilita o chronyd
```

11.6.4. Parando a chronyd

Para parar **chronyd**, emita o seguinte comando como **root**:

```
# systemctl stop chronyd
```

Para evitar que **chronyd** comece automaticamente no início do sistema, emita o seguinte comando como **root**:

```
# Systemctl desativar chronyd
```

11.6.5. Verificando se o chrony está sincronizado

Para verificar se **chrony** é sincronizado, faça uso dos comandos **tracking**, **sources**, e **sourcestats**.

11.6.5.1. Verificação do rastreamento do chrony

Para verificar **chrony** rastreamento, emitir o seguinte comando:

```
$ chronyc tracking
Reference ID    : CB00710F (foo.example.net)
Stratum         : 3
Ref time (UTC)  : Fri Jan 27 09:49:17 2017
System time     :  0.000006523 seconds slow of NTP time
Last offset     : -0.000006747 seconds
RMS offset      : 0.000035822 seconds
Frequency       : 3.225 ppm slow
Residual freq   : 0.000 ppm
Skew            : 0.129 ppm
Root delay      : 0.013639022 seconds
Root dispersion : 0.001100737 seconds
Update interval : 64.2 seconds
Leap status     : Normal
```

Os campos são os seguintes:

ID de referência

Este é o ID de referência e nome (ou endereço **IP**), se disponível, do servidor para o qual o computador está atualmente sincronizado. O ID de referência é um número hexadecimal para evitar confusão com endereços IPv4.

Stratum

O stratum indica a quantidade de lúpulos que estamos a uma distância de um computador com um relógio de referência anexo. Tal computador é um computador stratum-1, portanto o computador no exemplo está a dois lúpulos de distância (ou seja, a.b.c é um stratum-2 e é sincronizado a partir de um stratum-1).

Tempo de reflexão

Este é o momento (UTC) em que a última medição da fonte de referência foi processada.

Tempo do sistema

Em operação normal, **chronyd** nunca pisa o relógio do sistema, porque qualquer salto na escala de tempo pode ter conseqüências adversas para certos programas de aplicação. Em vez disso, qualquer erro no relógio do sistema é corrigido através de uma ligeira aceleração ou desaceleração do relógio do sistema até que o erro tenha sido removido e, em seguida, voltando à velocidade normal do relógio do sistema. Uma conseqüência disto é que haverá um período em que o relógio do sistema (como lido por outros programas usando a chamada do sistema **gettimeofday()**, ou pelo comando de data na shell) será diferente da estimativa do tempo real atual (que reporta aos clientes **NTP** quando estiver operando em modo servidor) da **chronyd**. O valor reportado nesta linha é a diferença devido a este efeito.

Última compensação

Esta é a compensação local estimada na última atualização do relógio.

Compensação RMS

Esta é uma média de longo prazo do valor da compensação.

Freqüência

A "freqüência" é a taxa pela qual o relógio do sistema estaria errado se **chronyd** não estivesse corrigindo-o. Ela é expressa em ppm (partes por milhão). Por exemplo, um valor de 1 ppm significaria que quando o relógio do sistema pensa que avançou 1 segundo, ele realmente avançou 1,000001 segundos em relação ao tempo real.

Freq residual

Isto mostra a "freqüência residual" para a fonte de referência atualmente selecionada. Isto reflete qualquer diferença entre o que as medidas da fonte de referência indicam que a freqüência deve ser e a freqüência que está sendo usada atualmente.
O motivo nem sempre é zero é que um procedimento de suavização é aplicado à freqüência. Cada vez que uma medição da fonte de referência é obtida e uma nova freqüência residual é computada, a precisão estimada deste residual é comparada com a precisão estimada (ver **skew**) do valor da freqüência existente. Uma média ponderada é computada para a nova freqüência, com pesos dependendo destas precisões. Se as medições da fonte de referência seguirem uma tendência consistente, o residual será conduzido a zero ao longo do tempo.

Tendência

Este é o erro estimado vinculado à freqüência.

Atraso na raiz

Este é o total dos atrasos do caminho da rede para o computador stratum-1 a partir do qual o computador é finalmente sincronizado. Os valores dos atrasos de raiz são impressos em resolução de nanossegundos. Em certas situações extremas, este valor pode ser negativo. (Isto pode surgir em um arranjo simétrico de pares onde as freqüências dos computadores não estão se rastreando e o atraso da rede é muito curto em relação ao tempo de retorno em cada computador)

Dispersão radicular

Esta é a dispersão total acumulada através de todos os computadores de volta ao computador stratum-1 a partir do qual o computador é finalmente sincronizado. A dispersão é devida à resolução do relógio do sistema, variações de medição estatística, etc. Os valores de dispersão da raiz são impressos em resolução de nanossegundos.

Salto de status

Este é o status de salto, que pode ser Normal, Inserir segundo, Apagar segundo ou Não sincronizado.

11.6.5.2. Verificação das fontes do cristo

O comando de fontes exibe informações sobre as fontes de tempo atuais que **chronyd** está acessando.

O argumento opcional -v pode ser especificado, significando verboso. Neste caso, linhas extras de legenda são mostradas como um lembrete dos significados das colunas.

```
$ chronyc sources
 210 Number of sources = 3
MS Name/IP address         Stratum Poll Reach LastRx Last sample
===============================================================================

#* GPS0                          0   4   377    11   -479ns[ -621ns] /-  134ns
^? a.b.c                        2   6   377    23   -923us[ -924us] +/-   43ms
^ d.e.f                       1   6   377    21  -2629us[-2619us] +/-   86ms
```

As colunas são as seguintes:

M

Isto indica o modo da fonte. **^** significa um servidor, **=** significa um par e **#** indica um relógio de referência conectado localmente.

S

Esta coluna indica o estado das fontes. \A coluna "*" indica a fonte à qual **chronyd** está atualmente sincronizada. "\Indica as fontes aceitáveis que são combinadas com a fonte selecionada. "-" indica as

fontes aceitáveis que são excluídas pelo algoritmo de combinação. "\Indica fontes para as quais a conectividade foi perdida ou cujos pacotes não passaram em todos os testes. "\x" indica um relógio que **chronyd** acha que é um *falseticker* (seu tempo é inconsistente com a maioria das outras fontes). "~" indica uma fonte cujo tempo parece ter demasiada variabilidade. A condição "?~" também é mostrada no início, até que pelo menos 3 amostras tenham sido coletadas dela.

Nome/endereço IP

Isto mostra o nome ou o endereço **IP** da fonte, ou o ID de referência para o relógio de referência.

Stratum

Isto mostra o estrato da fonte, como relatado em sua amostra recebida mais recentemente. O Stratum 1 indica um computador com um relógio de referência acoplado localmente. Um computador que está sincronizado com um computador do estrato 1 está no estrato 2. Um computador que está sincronizado com um computador do estrato 2 está no estrato 3, e assim por diante.

Enquete

Isto mostra a taxa na qual a fonte está sendo pesquisada, como um logaritmo de base 2 do intervalo em segundos. Assim, um valor de 6 indica que está sendo feita uma medição a cada 64 segundos. **chronyd** varia automaticamente a taxa de votação em resposta às condições prevalecentes.

Alcance

Isto mostra o registro de alcance da fonte impresso como um número octal. O registro tem 8 bits e é atualizado em cada pacote recebido ou perdido da fonte. Um valor de 377 indica que foi recebida uma resposta válida para todas as últimas oito transmissões.

LastRx

Esta coluna mostra há quanto tempo a última amostra foi recebida da fonte. Isto normalmente é em segundos. As cartas **m**, **h**, **d** ou **y** indicam minutos, horas, dias ou anos. Um valor de 10 anos indica que ainda não foram recebidas amostras desta fonte.

Última amostra

Esta coluna mostra o offset entre o relógio local e a fonte na última medição. O número entre parênteses rectos mostra o deslocamento real medido. Isto pode ser sufixado por **ns** (indicando nanossegundos), **us** (indicando microssegundos), **ms** (indicando milissegundos), ou **s** (indicando segundos). O número à esquerda dos colchetes mostra a medida original, ajustada para permitir qualquer deslocamento aplicado ao relógio local desde então. O número que segue o indicador **/-** mostra a margem de erro na medição. Os offsets positivos indicam que o relógio local está à frente da fonte.

11.6.5.3. Verificação das estatísticas da fonte chrony

O comando **sourcestats** exibe informações sobre a taxa de deriva e o processo de estimativa de compensação para cada uma das fontes atualmente em exame pelo **chronyd**.

O argumento opcional **-v** pode ser especificado, significando verboso. Neste caso, linhas extras de legenda são mostradas como um lembrete dos significados das colunas.

```
$ chronyc sourcestats
210 Number of sources = 1
Name/IP Address            NP  NR  Span  Frequency  Freq Skew  Offset  Std Dev
==============================================================================

abc.def.ghi                11   5   46m     -0.001      0.045     1us    25us
```

As colunas são as seguintes:

Nome/endereço IP

Este é o nome ou endereço **IP** do servidor **NTP** (ou par) ou ID de referência do relógio de referência com o qual o resto da linha se relaciona.

NP

Este é o número de pontos de amostra que estão sendo retidos atualmente para o servidor. A taxa de deriva e a compensação de corrente são estimadas através da realização de uma regressão linear através desses pontos.

NR

Este é o número de corridas de resíduos com o mesmo sinal após a última regressão. Se este número começar a ficar muito pequeno em relação ao número de amostras, indica que uma linha reta não é mais um bom ajuste para os dados. Se o número de corridas for muito baixo, **chronyd** descarta amostras mais antigas e executa novamente a regressão até que o número de corridas se torne aceitável.

Espanhol

Este é o intervalo entre as amostras mais antigas e as mais recentes. Se nenhuma unidade for mostrada, o valor é em segundos. No exemplo, o intervalo é de 46 minutos.

Freqüência

Esta é a freqüência residual estimada para o servidor, em partes por milhão. Neste caso, estima-se que o relógio do computador esteja rodando 1 parte em 10^9 lento em relação ao servidor.

Tirolesa Freq

Este é o erro estimado para a Freq (novamente em partes por milhão).

Offset

Esta é a compensação estimada da fonte.

Std Dev

Este é o desvio padrão estimado da amostra.

11.6.6. Ajuste manual do relógio do sistema

Para pisar o relógio do sistema imediatamente, contornando quaisquer ajustes em andamento por giro, emitir o seguinte comando como **root**:

```
# cronyc makestep
```

Se a diretiva **rtcfile** for usada, o relógio em tempo real não deve ser ajustado manualmente. Ajustes aleatórios interfeririam com **chrony**A necessidade de medir a taxa de deriva do relógio em tempo real.

11.7. ESTABELECENDO O CHRONY PARA DIFERENTES AMBIENTES

11.7.1. Estabelecendo o chrony para um sistema em uma rede isolada

Para uma rede que nunca está conectada à Internet, um computador é selecionado para ser o master timeserver. Os outros computadores são ou clientes diretos do mestre, ou clientes de clientes. No master, o arquivo de drift deve ser configurado manualmente com a taxa média de drift do relógio do sistema. Se o master for reinicializado, ele obterá o tempo dos sistemas vizinhos e calculará uma média para definir seu relógio do sistema. Em seguida, ele retoma a aplicação de ajustes com base no arquivo de derivação. O arquivo de derivação será atualizado automaticamente quando o comando **settime** for usado.

No sistema selecionado para ser o mestre, usando um editor de texto rodando como **root**, edite **/etc/chrony.conf** como segue:

```
driftfile /var/lib/chrony/drift
commandkey 1
keyfile /etc/chrony.keys
initstepslew 10 client1 client3 client6
local stratum 8
manual
allow 192.0.2.0
```

Onde **192.0.2.0** é o endereço de rede ou sub-rede a partir do qual os clientes têm permissão de se conectar.

Nos sistemas selecionados para serem clientes diretos do mestre, usando um editor de texto rodando como **root**, edite o **/etc/chrony.conf** da seguinte forma:

```
server master
driftfile /var/lib/chrony/drift
logdir /var/log/chrony
log measurements statistics tracking
keyfile /etc/chrony.keys
commandkey 24
local stratum 10
initstepslew 20 master
allow 192.0.2.123
```

Onde **192.0.2.123** é o endereço do mestre, e **master** é o nome do anfitrião do mestre. Os clientes com esta configuração irão ressincronizar o master se ele for reiniciado.

Nos sistemas de clientes que não devem ser clientes diretos do mestre, o arquivo **/etc/chrony.conf** deve ser o mesmo, exceto que as diretrizes **local** e **allow** devem ser omitidas.

Em uma rede isolada, você também pode usar a diretiva **local** que permite um modo de referência local, que permite que **chronyd** operando como um servidor **NTP** apareça sincronizado em tempo real, mesmo quando nunca foi sincronizado ou quando a última atualização do relógio aconteceu há muito tempo.

Para permitir que vários servidores na rede utilizem a mesma configuração local e sejam sincronizados entre si, sem confundir clientes que pesquisam mais de um servidor, utilize a opção **orphan** da diretiva **local** que permite o modo órfão. Cada servidor precisa ser configurado para pesquisar todos os outros servidores com **local**. Isto garante que somente o servidor com a menor ID de referência tenha a referência local ativa e outros servidores estejam sincronizados com ela. Quando o servidor falhar, outro assumirá o controle.

11.8. CRONOLOGIA COM HW TIMESTAMPING

11.8.1. Entendendo o hardware de marcação temporal

O timestamping do hardware é um recurso suportado em alguns Controladores de Interface de Rede (NICs) que fornece timestamping preciso dos pacotes que entram e saem. **NTP** timestamps são normalmente criados pelo kernel e **chronyd** com o uso do relógio do sistema. Entretanto, quando o HW timestamping é ativado, o NIC usa seu próprio relógio para gerar os timestamps quando os pacotes entram ou saem da camada de ligação ou da camada física. Quando usado com **NTP**, o registro de

tempo de hardware pode melhorar significativamente a precisão da sincronização. Para melhor precisão, tanto os servidores **NTP** quanto os clientes **NTP** precisam usar o timestamping de hardware. Sob condições ideais, pode ser possível uma precisão de sub-microssegundo.

Outro protocolo para sincronização de tempo que utiliza a marcação de tempo por hardware é **PTP**.

Ao contrário de **NTP**, **PTP** conta com assistência em switches e roteadores de rede. Se você quiser alcançar a melhor precisão de sincronização, use **PTP** em redes que tenham switches e roteadores com suporte **PTP**, e prefira **NTP** em redes que não tenham tais switches e roteadores.

11.8.2. Verificação do suporte para o registro de tempo do hardware

Para verificar se o timestamping do hardware com **NTP** é suportado por uma interface, use o comando **ethtool -T**. Uma interface pode ser usada para o timestamping do hardware com **NTP** se **ethtool** listar os recursos **SOF_TIMESTAMPING_TX_HARDWARE** e **SOF_TIMESTAMPING_TX_SOFTWARE** e também o modo de filtro **HWTSTAMP_FILTER_ALL**.

Exemplo 11.1. Verificação de suporte para a marcação temporal do hardware em uma interface específica

```
# ethtool -T eth0
```

Saída:

```
Timestamping parameters for eth0:
Capabilities:
        hardware-transmit     (SOF_TIMESTAMPING_TX_HARDWARE)
        software-transmit     (SOF_TIMESTAMPING_TX_SOFTWARE)
        hardware-receive      (SOF_TIMESTAMPING_RX_HARDWARE)
        software-receive      (SOF_TIMESTAMPING_RX_SOFTWARE)
        software-system-clock (SOF_TIMESTAMPING_SOFTWARE)
        hardware-raw-clock    (SOF_TIMESTAMPING_RAW_HARDWARE)
PTP Hardware Clock: 0
Hardware Transmit Timestamp Modes:
        off                   (HWTSTAMP_TX_OFF)
        on                    (HWTSTAMP_TX_ON)
Hardware Receive Filter Modes:
        none                  (HWTSTAMP_FILTER_NONE)
        all                   (HWTSTAMP_FILTER_ALL)
        ptpv1-l4-sync         (HWTSTAMP_FILTER_PTP_V1_L4_SYNC)
        ptpv1-l4-delay-req    (HWTSTAMP_FILTER_PTP_V1_L4_DELAY_REQ)
        ptpv2-l4-sync         (HWTSTAMP_FILTER_PTP_V2_L4_SYNC)
        ptpv2-l4-delay-req    (HWTSTAMP_FILTER_PTP_V2_L4_DELAY_REQ)
        ptpv2-l2-sync         (HWTSTAMP_FILTER_PTP_V2_L2_SYNC)
        ptpv2-l2-delay-req    (HWTSTAMP_FILTER_PTP_V2_L2_DELAY_REQ)
        ptpv2-event           (HWTSTAMP_FILTER_PTP_V2_EVENT)
        ptpv2-sync            (HWTSTAMP_FILTER_PTP_V2_SYNC)
        ptpv2-delay-req       (HWTSTAMP_FILTER_PTP_V2_DELAY_REQ)
```

11.8.3. Permitindo o registro de tempo do hardware

Para habilitar a marcação temporal do hardware, use a diretiva **hwtimestamp** no arquivo **/etc/chrony.conf**. A diretiva pode especificar uma única interface, ou um caractere curinga pode ser

usado para habilitar o timestamping do hardware em todas as interfaces que o suportam. Use a especificação de curinga no caso de nenhuma outra aplicação, como **ptp4l** do pacote **linuxptp**, está usando a marcação de tempo do hardware em uma interface. Várias diretivas **hwtimestamp** são permitidas no arquivo de configuração do chrony.

Exemplo 11.2. Permitindo o registro de tempo do hardware usando a diretiva hwtimestamp

```
hwtimestamp eth0
hwtimestamp eth1
hwtimestamp *
```

11.8.4. Configuração do intervalo de votação do cliente

O intervalo padrão de um intervalo de sondagem (64-1024 segundos) é recomendado para servidores na Internet. Para servidores locais e hardware de marcação de tempo, um intervalo de votação mais curto precisa ser configurado a fim de minimizar o offset do relógio do sistema.

A seguinte diretiva em **/etc/chrony.conf** especifica um servidor local **NTP** usando um segundo intervalo de sondagem:

```
servidor ntp.local minpoll 0 maxpoll 0
```

11.8.5. Habilitando o modo intercalado

NTP servidores que não são dispositivos de hardware **NTP**, mas sim computadores de uso geral rodando um software **NTP** implementação, como **chrony**O equipamento só receberá um carimbo de data/hora de transmissão após o envio de um pacote. Este comportamento impede que o servidor guarde o carimbo de data/hora no pacote ao qual ele corresponde. A fim de habilitar **NTP** clientes recebendo carimbos de data/hora de transmissão que foram gerados após a transmissão, configure os clientes para usar o modo intercalado **NTP** adicionando a opção **xleave** à diretiva do servidor em **/etc/chrony.conf**:

```
servidor ntp.local minpoll 0 maxpoll 0 xleave
```

11.8.6. Configuração de servidor para um grande número de clientes

A configuração padrão do servidor permite que alguns milhares de clientes, no máximo, usem o modo intercalado simultaneamente. Para configurar o servidor para um número maior de clientes, aumente a diretiva **clientloglimit** em **/etc/chrony.conf**. Esta diretiva especifica o tamanho máximo de memória alocada para o registro do acesso dos clientes no servidor:

```
clientloglimit 100000000
```

11.8.7. Verificação da marcação temporal do hardware

Para verificar se a interface habilitou com sucesso o registro de tempo do hardware, verifique o registro do sistema. O log deve conter uma mensagem de **chronyd** para cada interface com timestamping de hardware habilitado com sucesso.

Exemplo 11.3. Mensagens de registro para interfaces com hardware habilitado para registro de tempo

```
-
```

```
chronyd[4081]: Enabled HW timestamping on eth0
chronyd[4081]: Enabled HW timestamping on eth1
```

Quando **chronyd** estiver configurado como um cliente **NTP** ou par, você poderá ter os modos de transmissão e recebimento de timestamping e o modo intercalado reportados para cada fonte **NTP** pelo comando **chronyc ntpdata**:

Exemplo 11.4. Relatar o modo de transmissão, recebimento de timestamping e interleaved para cada fonte NTP

```
# cronyc ntpdata
```

Saída:

```
Remote address  : 203.0.113.15 (CB00710F)
Remote port     : 123
Local address   : 203.0.113.74 (CB00714A)
Leap status     : Normal
Version         : 4
Mode            : Server
Stratum         : 1
Poll interval   : 0 (1 seconds)
Precision       : -24 (0.000000060 seconds)
Root delay      : 0.000015 seconds
Root dispersion : 0.000015 seconds
Reference ID    : 47505300 (GPS)
Reference time  : Wed May 03 13:47:45 2017
Offset          : -0.000000134 seconds
Peer delay      : 0.000005396 seconds
Peer dispersion : 0.000002329 seconds
Response time   : 0.000152073 seconds
Jitter asymmetry: +0.00
NTP tests       : 111 111 1111
Interleaved     : Yes
Authenticated   : No
TX timestamping : Hardware
RX timestamping : Hardware
Total TX        : 27
Total RX        : 27
Total valid RX  : 27
```

Exemplo 11.5. Relatar a estabilidade das medições de NTP

```
# Fontes cronicas
```

Com o hardware com timestamping habilitado, a estabilidade das medições do **NTP** deve ser em dezenas ou centenas de nanossegundos, sob carga normal. Esta estabilidade é relatada na coluna **Std Dev** da saída do comando **chronyc sourcestats**:

Saída:

```
210 Number of sources = 1
Name/IP Address            NP  NR  Span  Frequency  Freq Skew  Offset  Std Dev
ntp.local                  12   7    11     +0.000      0.019     +0ns    49ns
```

11.8.8. Configurando a ponte PTP-NTP

Se um Precision Time Protocol (**PTP**) altamente preciso estiver disponível em uma rede que não tenha switches ou roteadores com suporte **PTP**, um computador pode ser dedicado a operar como um escravo **PTP** e um servidor stratum-1 **NTP**. Tal computador precisa ter duas ou mais interfaces de rede, e estar próximo do grão-mestre ou ter uma conexão direta com ele. Isto garantirá uma sincronização altamente precisa na rede.

Configure o **ptp4l** e **phc2sys** dos pacotes **linuxptp** para usar uma interface para sincronizar o relógio do sistema usando **PTP**.

Configurar **chronyd** para fornecer o tempo do sistema usando a outra interface:

Exemplo 11.6. Configuração do chronyd para fornecer o tempo do sistema usando a outra interface

```
bindaddress 203.0.113.74
hwtimestamp eth1
local stratum 1
```

11.9. ALCANÇANDO ALGUNS AJUSTES ANTERIORMENTE SUPORTADOS PELA NTP EM CHRONY

Algumas configurações que estavam na versão principal anterior do Red Hat Enterprise Linux suportadas por **ntp**não são apoiados por **chrony**. Esta seção lista essas configurações e descreve maneiras de alcançá-las em um sistema com **chrony**.

11.9.1. Monitoramento por ntpq e ntpdc

chronyd não pode ser monitorado pelo **ntpq** e **ntpdc** utilidades do **ntp** distribuição, porque **chrony** não suporta os modos 6 e 7 do site **NTP**. Ele apóia um protocolo diferente e **chronyc** é a implementação do cliente. Para mais informações, consulte a página de manual **chronyc(1)**.

Para monitorar o status do relógio do sistema sincronizado por **chronyd**, você pode:

- Use o comando de rastreamento
- Use o **ntpstat** utilidade, que suporta **chrony** e fornece uma saída semelhante à que costumava ter com **ntpd**

Exemplo 11.7. Usando o comando de rastreamento

```
$ chronyc -n tracking
Reference ID    : 0A051B0A (10.5.27.10)
Stratum         : 2
Ref time (UTC)  : Thu Mar 08 15:46:20 2018
```

```
System time     : 0.000000338 seconds slow of NTP time
Last offset     : +0.000339408 seconds
RMS offset      : 0.000339408 seconds
Frequency       : 2.968 ppm slow
Residual freq   : +0.001 ppm
Skew            : 3.336 ppm
Root delay      : 0.157559142 seconds
Root dispersion : 0.001339232 seconds
Update interval : 64.5 seconds
Leap status     : Normal
```

Exemplo 11.8. Usando a utilidade ntpstat

```
$ ntpstat
synchronised to NTP server (10.5.27.10) at stratum 2
   time correct to within 80 ms
   polling server every 64 s
```

11.9.2. Usando mecanismo de autenticação baseado em criptografia de chave pública

No Red Hat Enterprise Linux 7, **ntp** apoiado **Autokey**que é um mecanismo de autenticação baseado em criptografia de chave pública **Autokey** não é suportado em **chronyd**.

Em um sistema Red Hat Enterprise Linux 8, é recomendado o uso de chaves simétricas. Gerar as chaves com o comando **chronyc keygen**. Um cliente e um servidor precisam compartilhar uma chave especificada em /**etc**/**chrony.keys**. O cliente pode habilitar a autenticação usando a opção **key** na diretiva **server**, **pool**, ou **peer**.

11.9.3. Usando associações efêmeras simétricas

No Red Hat Enterprise Linux 7, **ntpd** suportava associações efêmeras simétricas, que podem ser mobilizadas por pacotes de pares que não estão especificados no arquivo de configuração **ntp.conf**. No Red Hat Enterprise Linux 8, **chronyd** precisa que todos os pares sejam especificados em **chrony.conf**. Associações efêmeras simétricas não são suportadas.

Observe que o uso do modo cliente/servidor ativado pela diretiva **server** ou **pool** é mais seguro em comparação com o modo simétrico ativado pela diretiva **peer**.

11.9.4. cliente multicast/broadcast

O Red Hat Enterprise Linux 7 suportava o modo broadcast/multicast **NTP**, o que simplifica a configuração dos clientes. Com este modo, os clientes podem ser configurados para apenas ouvir pacotes enviados para um endereço multicast/broadcast ao invés de ouvir nomes ou endereços específicos de servidores individuais, que podem mudar com o tempo.

No Red Hat Enterprise Linux 8, **chronyd** não suporta o modo de transmissão/multicast. A principal razão é que ele é menos preciso e menos seguro que o modo cliente/servidor comum e os modos simétricos.

Há várias opções de migração a partir de uma configuração **NTP** broadcast/multicast:

- Configurar o DNS para traduzir um único nome, como ntp.example.com, para múltiplos endereços de diferentes servidores
  Os clientes podem ter uma configuração estática usando apenas uma única diretriz de pool para sincronizar com vários servidores. Se um servidor do pool se tornar inalcançável, ou inadequado para sincronização, os clientes o substituem automaticamente por outro servidor do pool.

- Distribua a lista de servidores **NTP** sobre DHCP
  Quando o NetworkManager recebe uma lista de servidores **NTP** do servidor DHCP, **chronyd** é automaticamente configurado para usá-los. Este recurso pode ser desativado adicionando **PEERNTP=no** ao arquivo /**etc**/**sysconfig**/**network**.

- Use o **Precision Time Protocol (PTP)**
  Esta opção é adequada principalmente para ambientes onde os servidores mudam freqüentemente, ou se um grupo maior de clientes precisa ser capaz de se sincronizar uns com os outros sem ter um servidor designado.

  PTP foi projetado para mensagens multicast e funciona de forma semelhante ao modo de transmissão **NTP**. Uma implementação de **PTP** está disponível no pacote **linuxptp**.

  PTP normalmente requer hardware timestamping e suporte em switches de rede para ter um bom desempenho. Entretanto, espera-se que **PTP** funcione melhor que **NTP** no modo de transmissão, mesmo com timestamping de software e sem suporte em switches de rede.

  Em redes com um número muito grande de escravos **PTP** em um caminho de comunicação, recomenda-se configurar o **PTP** escravos com a opção **hybrid_e2e** a fim de reduzir a quantidade de tráfego de rede gerada pelos escravos. Pode-se configurar um computador rodando **chronyd** como cliente **NTP**, e possivelmente **NTP** servidor, para operar também como um **PTP** grandmaster para distribuir tempo sincronizado para um grande número de computadores usando mensagens multicast.

11.10. RECURSOS ADICIONAIS

As seguintes fontes de informação fornecem recursos adicionais em relação a **chrony**.

11.10.1. Documentação Instalada

- **chronyc(1)** man page - Descreve o **chronyc** ferramenta de interface de linha de comando, incluindo comandos e opções de comando.

- **chronyd(8)** man page - Descreve o daemon **chronyd** incluindo comandos e opções de comando.

- **chrony.conf(5)** man page - Descreve o **chrony** arquivo de configuração.

11.10.2. Documentação on-line

- https://chrony.tuxfamily.org/doc/3.3/chronyc.html

- https://chrony.tuxfamily.org/doc/3.3/chronyd.html

- https://chrony.tuxfamily.org/doc/3.3/chrony.conf.html

Para respostas às perguntas freqüentes, veja https://chrony.tuxfamily.org/faq.html

11.11. GERENCIANDO A SINCRONIZAÇÃO DE TEMPO USANDO AS FUNÇÕES DO SISTEMA RHEL

Você pode gerenciar a sincronização de tempo em várias máquinas alvo usando a função **timesync**.

A função **timesync** instala e configura uma implementação NTP ou PTP para operar como um cliente NTP ou escravo PTP a fim de sincronizar o relógio do sistema com servidores NTP ou grandmasters nos domínios PTP.

Note que o uso da função **timesync** também facilita a migração para o chrony, pois você pode usar o mesmo playbook em todas as versões do Red Hat Enterprise Linux começando com o RHEL 6, independentemente de o sistema usar **ntp** ou **chrony** para implementar o protocolo NTP.

ATENÇÃO

A função **timesync** substitui a configuração do serviço de provedor dado ou detectado no host administrado. As configurações anteriores são perdidas, mesmo que não estejam especificadas nas variáveis de função. A única configuração preservada é a escolha do provedor se a variável **timesync_ntp_provider** não estiver definida.

O exemplo a seguir mostra como aplicar o papel **timesync** em uma situação com apenas um pool de servidores.

Exemplo 11.9. Um exemplo de playbook aplicando o papel de timesync para um único pool de servidores

```
---
- hosts: timesync-test
  vars:
    timesync_ntp_servers:
      - hostname: 2.rhel.pool.ntp.org
        pool: yes
        iburst: yes
  roles:
    - rhel-system-roles.timesync
```

Recursos adicionais

- Para uma referência detalhada sobre as variáveis de função **timesync**, instale o pacote **rhel-system-roles**, e veja os arquivos **README.md** ou **README.html** no diretório **/usr/share/doc/rhel-system-roles/timesync**.
- Para mais informações sobre os papéis do Sistema RHEL, veja Introdução aos papéis do Sistema RHEL.

CAPÍTULO 12. USANDO COMUNICAÇÕES SEGURAS ENTRE DOIS SISTEMAS COM OPENSSH

SSH (Secure Shell) é um protocolo que fornece comunicações seguras entre dois sistemas usando uma arquitetura cliente-servidor e permite que os usuários façam login em sistemas host de servidores remotamente. Ao contrário de outros protocolos de comunicação remota, como FTP ou Telnet, o SSH criptografa a sessão de login, o que impede que intrusos coletem senhas não criptografadas da conexão.

O Red Hat Enterprise Linux inclui os pacotes básicos **OpenSSH**: o pacote geral **openssh**, o pacote **openssh-server** e o pacote **openssh-clients**. Note que os pacotes **OpenSSH** requerem o pacote **OpenSSL openssl-libs** , que instala várias bibliotecas criptográficas importantes que permitem que **OpenSSH** forneça comunicações criptografadas.

12.1. SSH E OPENSSH

SSH (Secure Shell) é um programa para efetuar login em uma máquina remota e executar comandos nessa máquina. O protocolo SSH fornece comunicações criptografadas seguras entre dois hosts não confiáveis através de uma rede insegura. Você também pode encaminhar conexões X11 e portas TCP/IP arbitrárias através do canal seguro.

O protocolo SSH atenua as ameaças à segurança, tais como interceptação da comunicação entre dois sistemas e imitação de um determinado host, quando você o utiliza para login remoto ou cópia de arquivo. Isto porque o cliente e o servidor SSH usam assinaturas digitais para verificar suas identidades. Além disso, toda a comunicação entre os sistemas cliente e servidor é criptografada.

OpenSSH é uma implementação do protocolo SSH suportada por uma série de sistemas operacionais Linux, UNIX e similares. Ele inclui os arquivos centrais necessários tanto para o cliente OpenSSH quanto para o servidor. A suíte OpenSSH consiste das seguintes ferramentas de espaço do usuário:

- **ssh** é um programa de login remoto (cliente SSH)
- **sshd** é um **OpenSSH** daemon SSH
- **scp** é um programa seguro de cópia remota de arquivos
- **sftp** é um programa seguro de transferência de arquivos
- **ssh-agent** é um agente de autenticação para o cache de chaves privadas
- **ssh-add** adiciona identidades chave privadas a **ssh-agent**
- **ssh-keygen** gera, gerencia e converte chaves de autenticação para **ssh**
- **ssh-copy-id** é um script que adiciona chaves públicas locais ao arquivo **authorized_keys** em um servidor SSH remoto
- **ssh-keyscan** - reúne as chaves de anfitrião público do SSH

Existem atualmente duas versões do SSH: a versão 1, e a mais recente versão 2. A suíte **OpenSSH** no Red Hat Enterprise Linux 8 suporta apenas o SSH versão 2, que tem um algoritmo melhorado de troca de chaves não vulnerável a explorações conhecidas na versão 1.

OpenSSH, como um dos subsistemas criptográficos centrais da RHEL, utiliza políticas de criptografia em todo o sistema. Isto garante que os conjuntos de cifras fracas e algoritmos criptográficos sejam desativados na configuração padrão. Para ajustar a política, o administrador deve usar o comando

update-crypto-policies para fazer configurações mais rígidas ou mais frouxas ou optar manualmente pela exclusão das políticas criptográficas de todo o sistema.

A suíte **OpenSSH** utiliza dois conjuntos diferentes de arquivos de configuração: aqueles para programas de clientes (ou seja, **ssh**, **scp** e **sftp**), e aqueles para o servidor (o daemon **sshd**). As informações de configuração SSH de todo o sistema são armazenadas no diretório **/etc/ssh/**. As informações de configuração do SSH específicas do usuário são armazenadas em **~/.ssh/**, no diretório home do usuário. Para uma lista detalhada dos arquivos de configuração do OpenSSH, consulte a seção **FILES** na página de manual **sshd(8)**.

Recursos adicionais

- Páginas de homens para o tópico **ssh** listado pelo comando **man -k ssh**.
- Usando políticas criptográficas de todo o sistema .

12.2. CONFIGURANDO E INICIANDO UM SERVIDOR OPENSSH

Use o seguinte procedimento para uma configuração básica que possa ser necessária para seu ambiente e para iniciar um servidor **OpenSSH**. Observe que após a instalação padrão da RHEL, o daemon **sshd** já foi iniciado e as chaves do servidor são criadas automaticamente.

Pré-requisitos

- O pacote **openssh-server** está instalado.

Procedimento

1. Inicie o daemon **sshd** na sessão atual e configure-o para iniciar automaticamente no momento da inicialização:

```
# systemctl start sshd
# systemctl enable sshd
```

2. Para especificar endereços diferentes do padrão **0.0.0.0** (IPv4) ou **::** (IPv6) para a diretiva **ListenAddress** no arquivo de configuração **/etc/ssh/sshd_config** e para usar uma configuração de rede dinâmica mais lenta, adicione a dependência da unidade alvo **network-online.target** ao arquivo de unidade **sshd.service**. Para conseguir isso, crie o arquivo **/etc/systemd/system/sshd.service.d/local.conf** com o seguinte conteúdo:

```
[Unit]
Wants=network-online.target
After=network-online.target
```

3. Analise se as configurações do servidor **OpenSSH** no arquivo de configuração **/etc/ssh/sshd_config** atendem aos requisitos de seu cenário.

4. Opcionalmente, altere a mensagem de boas-vindas que seu servidor **OpenSSH** exibe antes que um cliente se autentique, editando o arquivo **/etc/issue**, por exemplo:

```
Welcome to ssh-server.example.com
Warning: By accessing this server, you agree to the referenced terms and conditions.
```

Certifique-se de que a opção **Banner** não seja comentada em **/etc/ssh/sshd_config** e seu valor contenha **/etc/issue**:

```
# less /etc/ssh/sshd_config | grep Banner
Banner /etc/issue
```

Note que para alterar a mensagem exibida após um login bem sucedido, você tem que editar o arquivo **/etc/motd** no servidor. Consulte a página de manual **pam_motd** para maiores informações.

5. Recarregue a configuração **systemd** e reinicie **sshd** para aplicar as mudanças:

```
# systemctl daemon-reload
# systemctl restart sshd
```

Etapas de verificação

1. Verifique se o daemon **sshd** está funcionando:

```
# systemctl status sshd
● sshd.service - OpenSSH server daemon
   Loaded: loaded (/usr/lib/systemd/system/sshd.service; enabled; vendor preset: enabled)
   Active: active (running) since Mon 2019-11-18 14:59:58 CET; 6min ago
     Docs: man:sshd(8)
           man:sshd_config(5)
 Main PID: 1149 (sshd)
    Tasks: 1 (limit: 11491)
   Memory: 1.9M
   CGroup: /system.slice/sshd.service
           └─1149 /usr/sbin/sshd -D -oCiphers=aes128-ctr,aes256-ctr,aes128-cbc,aes256-cbc -
oMACs=hmac-sha2-256,>

Nov 18 14:59:58 ssh-server-example.com systemd[1]: Starting OpenSSH server daemon...
Nov 18 14:59:58 ssh-server-example.com sshd[1149]: Server listening on 0.0.0.0 port 22.
Nov 18 14:59:58 ssh-server-example.com sshd[1149]: Server listening on :: port 22.
Nov 18 14:59:58 ssh-server-example.com systemd[1]: Started OpenSSH server daemon.
```

2. Conecte-se ao servidor SSH com um cliente SSH.

```
# ssh user@ssh-server-example.com
ECDSA key fingerprint is SHA256:dXbaS0RG/UzlTTku8GtXSz0S1++lPegSy31v3L/FAEc.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added 'ssh-server-example.com' (ECDSA) to the list of known hosts.

user@ssh-server-example.com's password:
```

Recursos adicionais

- **sshd(8)** e **sshd_config(5)** páginas man

12.3. USANDO PARES DE CHAVES AO INVÉS DE SENHAS PARA AUTENTICAÇÃO SSH

Para melhorar ainda mais a segurança do sistema, gerar pares de chaves SSH e depois impor a autenticação baseada em chaves, desativando a autenticação por senha.

12.3.1. Configurando um servidor OpenSSH para autenticação baseada em chaves

Siga estes passos para configurar seu servidor OpenSSH para fazer cumprir a autenticação baseada em chaves.

Pré-requisitos

- O pacote **openssh-server** está instalado.
- O daemon **sshd** está rodando no servidor.

Procedimento

1. Abra a configuração **/etc/ssh/sshd_config** em um editor de texto, por exemplo:

```
# vi /etc/ssh/sshd_config
```

2. Mude a opção **PasswordAuthentication** para **no**:

```
SenhaAutenticação não
```

   Em um sistema que não seja uma nova instalação padrão, verifique se **PubkeyAuthentication no** não foi definido e se a diretiva **ChallengeResponseAuthentication** está definida para **no**. Se você estiver conectado remotamente, não usando console ou acesso fora da banda, teste o processo de login baseado em chave antes de desativar a autenticação da senha.

3. Para utilizar a autenticação baseada em chaves com diretórios domésticos montados em NFS, habilite o **use_nfs_home_dirs** SELinux boolean:

```
# setsebool -P use_nfs_home_dirs 1
```

4. Recarregue o daemon **sshd** para aplicar as mudanças:

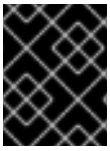
```
# systemctl reload sshd
```

Recursos adicionais

- **sshd(8)**, **sshd_config(5)**, e **setsebool(8)** páginas man

12.3.2. Geração de pares de chaves SSH

Use este procedimento para gerar um par de chaves SSH em um sistema local e para copiar a chave pública gerada para um servidor **OpenSSH**. Se o servidor estiver configurado de acordo, você pode entrar no servidor **OpenSSH** sem fornecer nenhuma senha.



IMPORTANTE

Se você completar os seguintes passos como **root**, somente **root** é capaz de usar as chaves.

Procedimento

1. Para gerar um par de chaves ECDSA para a versão 2 do protocolo SSH:

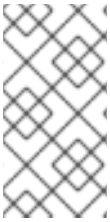
```
$ ssh-keygen -t ecdsa
Generating public/private ecdsa key pair.
Enter file in which to save the key (/home/joesec/.ssh/id_ecdsa):
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /home/joesec/.ssh/id_ecdsa.
Your public key has been saved in /home/joesec/.ssh/id_ecdsa.pub.
The key fingerprint is:
SHA256:Q/x+qms4j7PCQ0qFd09iZEFHA+SqwBKRNaU72oZfaCI
joesec@localhost.example.com
The key's randomart image is:
+---[ECDSA 256]---+
|.oo..o=++        |
|.. o .oo .       |
|. .. o. o        |
|....o.+...       |
|o.oo.o +S .      |
|.=.+.   .o       |
|E.*+.  .  . .    |
|.=..+ +..  o     |
|  .  oo*+o.      |
+----[SHA256]-----+
```

Você também pode gerar um par de chaves RSA usando a opção **-t rsa** com o comando **ssh-keygen** ou um par de chaves Ed25519, digitando o comando **ssh-keygen -t ed25519**.

2. Para copiar a chave pública para uma máquina remota:

```
$ ssh-copy-id joesec@ssh-server-example.com
/usr/bin/ssh-copy-id: INFO: attempting to log in with the new key(s), to filter out any that are already installed
joesec@ssh-server-example.com's password:
...
Number of key(s) added: 1

Now try logging into the machine, with: "ssh 'joesec@ssh-server-example.com'" and check to make sure that only the key(s) you wanted were added.
```

Se você não usar o programa **ssh-agent** em sua sessão, o comando anterior copia a chave pública ~/**.ssh**/**id*.pub** modificada mais recentemente, caso ainda não esteja instalada. Para especificar outro arquivo de chave pública ou para priorizar chaves em arquivos sobre chaves armazenadas em cache na memória por **ssh-agent**, use o comando **ssh-copy-id** com a opção **-i**.



NOTA

Se você reinstalar seu sistema e quiser manter os pares de chaves gerados anteriormente, faça backup do diretório ~/**.ssh**/. Após a reinstalação, copie-o de volta para seu diretório home. Você pode fazer isso para todos os usuários em seu sistema, incluindo **root**.

Etapas de verificação

1. Acesse o servidor OpenSSH sem fornecer nenhuma senha:

```
$ ssh joesec@ssh-server-example.com
Welcome message.
...
Last login: Mon Nov 18 18:28:42 2019 from ::1
```

Recursos adicionais

- **ssh-keygen(1)** e **ssh-copy-id(1)** páginas man

12.4. USANDO CHAVES SSH ARMAZENADAS EM UM CARTÃO INTELIGENTE

O Red Hat Enterprise Linux 8 permite que você use chaves RSA e ECDSA armazenadas em um cartão inteligente em clientes OpenSSH. Use este procedimento para habilitar a autenticação usando um Cartão Smart Card ao invés de usar uma senha.

Pré-requisitos

- No lado do cliente, o pacote **opensc** está instalado e o serviço **pcscd** está funcionando.

Procedimento

1. Liste todas as chaves fornecidas pelo módulo OpenSC PKCS #11 incluindo seus PKCS #11 URIs e salve a saída para o arquivo *keys.pub*:

```
$ ssh-keygen -D pkcs11: > keys.pub
$ ssh-keygen -D pkcs11:
ssh-rsa AAAAB3NzaC1yc2E...KKZMzcQZzx
pkcs11:id=%02;object=SIGN%20pubkey;token=SSH%20key;manufacturer=piv_II?module-
path=/usr/lib64/pkcs11/opensc-pkcs11.so
ecdsa-sha2-nistp256 AAA...J0hkYnnsM=
pkcs11:id=%01;object=PIV%20AUTH%20pubkey;token=SSH%20key;manufacturer=piv_II?
module-path=/usr/lib64/pkcs11/opensc-pkcs11.so
```

2. Para permitir a autenticação usando um cartão inteligente em um servidor remoto (*example.com*), transfira a chave pública para o servidor remoto. Use o comando **ssh-copy-id** com *keys.pub* criado na etapa anterior:

```
$ ssh-copy-id -f -i keys.pub username@example.com
```

3. Para conectar-se a *example.com* usando a chave ECDSA da saída do comando **ssh-keygen -D** no passo 1, você pode usar apenas um subconjunto do URI, que faz referência única à sua chave, por exemplo:

```
$ ssh -i "pkcs11:id=%01?module-path=/usr/lib64/pkcs11/opensc-pkcs11.so" example.com
Enter PIN for 'SSH key':
[example.com] $
```

4. Você pode usar a mesma cadeia URI no arquivo **~/.ssh/config** para tornar a configuração permanente:

```
$ cat ~/.ssh/config
IdentityFile "pkcs11:id=%01?module-path=/usr/lib64/pkcs11/opensc-pkcs11.so"
$ ssh example.com
Enter PIN for 'SSH key':
[example.com] $
```

Como o OpenSSH usa a embalagem **p11-kit-proxy** e o módulo OpenSC PKCS #11 está registrado no Kit PKCS #11, você pode simplificar os comandos anteriores:

```
$ ssh -i "pkcs11:id=%01" example.com
Enter PIN for 'SSH key':
[example.com] $
```

Se você pular a parte **id=** de um PKCS #11 URI, o OpenSSH carrega todas as chaves que estão disponíveis no módulo proxy. Isto pode reduzir a quantidade de digitação necessária:

```
$ ssh -i pkcs11: example.com
Enter PIN for 'SSH key':
[example.com] $
```

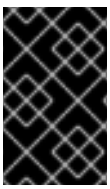
Recursos adicionais

- Fedora 28: Melhor suporte a cartões inteligentes no OpenSSH
- **p11-kit(8)** página do homem
- **ssh(1)** página do homem
- **ssh-keygen(1)** página do homem
- **opensc.conf(5)** página do homem
- **pcscd(8)** página do homem

12.5. TORNANDO O OPENSSH MAIS SEGURO

As seguintes dicas o ajudam a aumentar a segurança ao usar o OpenSSH. Note que as mudanças no arquivo de configuração **/etc/ssh/sshd_config** OpenSSH requerem o recarregamento do daemon **sshd** para ter efeito:

```
# systemctl reload sshd
```



IMPORTANTE

A maioria das mudanças de configuração de endurecimento de segurança reduz a compatibilidade com clientes que não suportam algoritmos atualizados ou conjuntos de cifras.

Desabilitando protocolos de conexão inseguros

- Para tornar o SSH verdadeiramente eficaz, evite o uso de protocolos de conexão inseguros que são substituídos pela suíte **OpenSSH**. Caso contrário, a senha de um usuário pode ser protegida usando SSH para apenas uma sessão, a ser capturada posteriormente ao fazer o login usando Telnet. Por este motivo, considere desativar protocolos inseguros, tais como telnet, rsh, rlogin e ftp.

Habilitação da autenticação baseada em chave e desativação da autenticação baseada em senha

- Desabilitar senhas para autenticação e permitir apenas pares de chaves reduz a superfície de ataque e também pode poupar o tempo dos usuários. Em clientes, gerar pares de chaves usando a ferramenta **ssh-keygen** e usar o utilitário **ssh-copy-id** para copiar chaves públicas de clientes no servidor **OpenSSH**. Para desativar a autenticação baseada em senhas em seu servidor OpenSSH, edite **/etc/ssh/sshd_config** e mude a opção **PasswordAuthentication** para **no**:

```
SenhaAutenticação não
```

Tipos de chaves

- Embora o comando **ssh-keygen** gere um par de chaves RSA por padrão, você pode instruí-lo a gerar chaves ECDSA ou Ed25519 usando a opção **-t**. O ECDSA (Elliptic Curve Digital Signature Algorithm) oferece melhor desempenho do que o RSA com a força simétrica equivalente da chave. Ele também gera chaves mais curtas. O algoritmo de chave pública Ed25519 é uma implementação de curvas Edwards retorcidas que é mais segura e também mais rápida que RSA, DSA, e ECDSA.
  O OpenSSH cria automaticamente chaves de servidor RSA, ECDSA e Ed25519 se elas estiverem faltando. Para configurar a criação da chave de host no RHEL 8, use o serviço instanciado **sshd-keygen@.service**. Por exemplo, para desativar a criação automática do tipo de chave RSA:

```
# systemctl mask sshd-keygen@rsa.service
```

- Para excluir tipos-chave específicos para conexões SSH, comente as linhas relevantes em **/etc/ssh/sshd_config**, e recarregue o serviço **sshd**. Por exemplo, para permitir apenas as chaves de host Ed25519:

```
# HostKey /etc/ssh/ssh_host_rsa_key
# HostKey /etc/ssh/ssh_host_ecdsa_key
HostKey /etc/ssh/ssh_host_ed25519_key
```

Porto sem falta

- Por padrão, o daemon **sshd** ouve na porta TCP 22. A mudança da porta reduz a exposição do sistema a ataques baseados em varredura automatizada da rede e, assim, aumenta a segurança através da obscuridade. Você pode especificar a porta usando a diretiva **Port** no arquivo de configuração **/etc/ssh/sshd_config**.
  Você também tem que atualizar a política padrão da SELinux para permitir o uso de uma porta não inadimplente. Para fazer isso, utilize a ferramenta **semanage** do pacote **policycoreutils-python-utils**:

```
# semanage port -a -t ssh_port_t -p tcp port_number
```

  Além disso, atualizar a configuração **firewalld**:

```
# firewall-cmd --add-port port_number/tcp
# firewall-cmd --runtime-to-permanent
```

Nos comandos anteriores, substituir *port_number* pelo novo número de porta especificado usando a diretiva **Port**.

Sem login de raiz

- Se seu caso particular de uso não exigir a possibilidade de login como usuário root, você deve considerar a possibilidade de configurar a diretiva de configuração **PermitRootLogin** para **no** no arquivo **/etc/ssh/sshd_config**. Desativando a possibilidade de logar-se como usuário root, o administrador pode auditar quais usuários executam que comandos privilegiados depois de logar-se como usuários regulares e depois ganhar direitos de root.
  Alternativamente, defina **PermitRootLogin** para **prohibit-password**:

  ```
  PermitRootLogin proíbe palavras-passe
  ```

  Isto reforça o uso de autenticação baseada em chaves em vez do uso de senhas para o login como raiz e reduz os riscos ao prevenir ataques de força bruta.

Usando a extensão X Security

- O servidor X em clientes Red Hat Enterprise Linux não fornece a extensão X Security. Portanto, os clientes não podem solicitar outra camada de segurança ao conectar-se a servidores SSH não confiáveis com o encaminhamento X11. A maioria das aplicações não é capaz de rodar com esta extensão habilitada de qualquer forma.
  Por padrão, a opção **ForwardX11Trusted** no arquivo **/etc/ssh/ssh_config.d/05-redhat.conf** está definida para **yes**, e não há diferença entre o comando **ssh -X remote_machine** (host não confiável) e **ssh -Y remote_machine** (trust host).

  Se seu cenário não exigir o recurso de encaminhamento X11, defina a diretiva **X11Forwarding** no arquivo de configuração **/etc/ssh/sshd_config** para **no**.

Restringir o acesso a usuários, grupos ou domínios específicos

- As diretrizes **AllowUsers** e **AllowGroups** no servidor de arquivos de configuração **/etc/ssh/sshd_config** permitem que você permita que somente determinados usuários, domínios ou grupos se conectem ao seu servidor OpenSSH. Você pode combinar **AllowUsers** e **AllowGroups** para restringir o acesso de forma mais precisa, por exemplo:

  ```
  AllowUsers *@192.168.1.*,*@10.0.0.*,!*@192.168.1.2
  AllowGroups example-group
  ```

  As linhas de configuração anteriores aceitam conexões de todos os usuários dos sistemas em 192.168.1.* e 10.0.0.* sub-redes, exceto do sistema com o endereço 192.168.1.2. Todos os usuários devem estar no grupo **example-group**. O servidor OpenSSH nega todas as outras conexões.

  Observe que o uso de listas de permissão (diretrizes que começam com Allow) é mais seguro do que o uso de listas de bloco (opções que começam com Deny) porque as listas de permissão também bloqueiam novos usuários ou grupos não autorizados.

Mudando as políticas criptográficas de todo o sistema

- **OpenSSH** utiliza as políticas criptográficas do sistema RHEL, e o nível padrão de políticas criptográficas do sistema oferece configurações seguras para os modelos de ameaça atuais. Para tornar suas configurações criptográficas mais rígidas, altere o nível da política atual:

```
# update-crypto-policies --set FUTURE
Setting system policy to FUTURE
```

- Para optar pela exclusão das políticas de criptografia de todo o sistema para seu servidor **OpenSSH**, descomente a linha com a variável **CRYPTO_POLICY=** no arquivo **/etc/sysconfig/sshd**. Após esta mudança, os valores que você especificar nas seções **Ciphers**, **MACs**, **KexAlgoritms**, e **GSSAPIKexAlgorithms** no arquivo **/etc/ssh/sshd_config** não serão sobrepostos. Note que esta tarefa requer profunda experiência na configuração de opções criptográficas.

- Veja Utilizando políticas criptográficas em todo o sistema no título de endurecimento de segurança RHEL 8 para mais informações.

Recursos adicionais

- **sshd_config(5)**, **ssh-keygen(1)**, **crypto-policies(7)**, e **update-crypto-policies(8)** páginas man

12.6. CONEXÃO A UM SERVIDOR REMOTO USANDO UM HOST SSH JUMP

Use este procedimento para conectar-se a um servidor remoto através de um servidor intermediário, também chamado de jump host.

Pré-requisitos

- Um host de salto aceita conexões SSH de seu sistema.
- Um servidor remoto aceita conexões SSH somente a partir do host de salto.

Procedimento

1. Defina o host de salto, editando o arquivo **~/.ssh/config**, por exemplo:

```
Host jump-server1
  HostName jump1.example.com
```

2. Adicione a configuração de salto do servidor remoto com a diretiva **ProxyJump** para **~/.ssh/config**, por exemplo:

```
Host remote-server
  HostName remote1.example.com
  ProxyJump jump-server1
```

3. Conecte-se ao servidor remoto através do servidor de salto:

```
$ ssh remote-server
```

O comando anterior é equivalente ao comando **ssh -J jump-server1 remote-server** se você omitir os passos de configuração 1 e 2.

NOTA

Você pode especificar mais servidores de salto e também pode pular a adição de definições de host ao arquivo de configurações quando você fornecer seus nomes de host completos, por exemplo:

```
$ ssh -J jump1.example.com,jump2.example.com,jump3.example.com
remote1.example.com
```

Mude a notação do nome do host apenas no comando anterior se os nomes dos usuários ou portas SSH nos servidores de salto forem diferentes dos nomes e portas no servidor remoto, por exemplo:

```
$ ssh -J
johndoe@jump1.example.com:75,johndoe@jump2.example.com:75,johndoe@jump3.e
xample.com:75 joesec@remote1.example.com:220
```

Recursos adicionais

- **ssh_config(5)** e **ssh(1)** páginas man

12.7. CONEXÃO A MÁQUINAS REMOTAS COM CHAVES SSH USANDO O SSH-AGENT

Para evitar a entrada de uma senha cada vez que você inicia uma conexão SSH, você pode usar o utilitário **ssh-agent** para fazer o cache da chave SSH privada. A chave privada e a frase-senha permanecem seguras.

Pré-requisitos

- Você tem um host remoto com daemon SSH rodando e alcançável através da rede.
- Você sabe o endereço IP ou o nome do host e as credenciais para fazer o login no host remoto.
- Você gerou um par de chaves SSH com uma frase-chave e transferiu a chave pública para a máquina remota. Para mais informações, consulte Gerando pares de chaves SSH .

Procedimento

1. Opcional: Verifique se você pode usar a chave para autenticar no host remoto:
   a. Conecte-se ao host remoto usando SSH:

      ```
      $ ssh example.user1@198.51.100.1 hostname
      ```

   b. Digite a frase-chave que você definiu ao criar a chave para conceder acesso à chave privada.

      ```
      $ ssh example.user1@198.51.100.1 hostname
       host.example.com
      ```

2. Inicie o **ssh-agent**.

```
$ eval $(ssh-agent)
Agent pid 20062
```

3. Adicione a chave a **ssh-agent**.

```
$ ssh-add ~/.ssh/id_rsa
Enter passphrase for ~/.ssh/id_rsa:
Identity added: ~/.ssh/id_rsa (example.user0@198.51.100.12)
```

Etapas de verificação

- Opcional: Faça o login na máquina host usando SSH.

```
$ ssh example.user1@198.51.100.1

Last login: Mon Sep 14 12:56:37 2020
```

Note que você não precisou digitar a senha.

12.8. RECURSOS ADICIONAIS

Para mais informações sobre configuração e conexão com servidores e clientes do **OpenSSH** no Red Hat Enterprise Linux, veja os recursos listados abaixo.

Documentação instalada

- **sshd(8)** página man documenta as opções de linha de comando disponíveis e fornece uma lista completa de arquivos de configuração e diretórios suportados.
- a página de manual**ssh(1)** fornece uma lista completa de opções de linha de comando disponíveis e arquivos de configuração e diretórios suportados.
- a página de manual**scp(1)** fornece uma descrição mais detalhada da utilidade **scp** e seu uso.
- a página de manual**sftp(1)** fornece uma descrição mais detalhada da utilidade **sftp** e seu uso.
- **ssh-keygen(1)** man page documents in detail the use of the **ssh-keygen** utility to generate, manage, and convert authentication keys used by ssh.
- a página de manual**ssh-copy-id(1)** descreve o uso do roteiro **ssh-copy-id**.
- **ssh_config(5)** página man documentos disponíveis opções de configuração do cliente SSH.
- a página de manual**sshd_config(5)** fornece uma descrição completa das opções de configuração do daemon SSH disponíveis.
- a página de manual**update-crypto-policies(8)** fornece orientação sobre o gerenciamento de políticas criptográficas de todo o sistema
- **crypto-policies(7)** página man fornece uma visão geral dos níveis de política criptográfica de todo o sistema

Documentação on-line

- OpenSSH Home Page - contém mais documentação, perguntas freqüentes, links para as listas de discussão, relatórios de bugs e outros recursos úteis.
- Configurando o SELinux para aplicações e serviços com configurações não-padrão - você pode aplicar procedimentos análogos para OpenSSH em uma configuração não-padrão com SELinux em modo de aplicação.
- Controle do tráfego da rede usando firewalld - fornece orientação sobre a atualização das configurações **firewalld** após a mudança de uma porta SSH

CAPÍTULO 13. CONFIGURAÇÃO DE UMA SOLUÇÃO DE REGISTRO REMOTO

Para garantir que os logs de várias máquinas em seu ambiente sejam registrados centralmente em um servidor de logs, você pode configurar o aplicativo **Rsyslog** para registrar logs que se ajustem a critérios específicos do sistema cliente para o servidor.

13.1. O SERVIÇO DE REGISTRO RSYSLOG

A aplicação Rsyslog, em combinação com o serviço **systemd-journald**, fornece suporte local e remoto de registro no Red Hat Enterprise Linux. O daemon **rsyslogd** lê continuamente **syslog** mensagens recebidas pelo serviço **systemd-journald** da revista. **rsyslogd** então filtra e processa estes eventos **syslog** e os registra em **rsyslog** arquivos de log ou os encaminha para outros serviços de acordo com sua configuração.

O daemon **rsyslogd** também fornece filtragem estendida, retransmissão de mensagens protegida por criptografia, módulos de entrada e saída, e suporte para transporte usando os protocolos TCP e UDP.

Em **/etc/rsyslog.conf**, que é o arquivo principal de configuração para **rsyslog**, você pode especificar as regras de acordo com as quais **rsyslogd** trata as mensagens. Geralmente, você pode classificar as mensagens por sua fonte e tópico (facilidade) e urgência (prioridade), e então atribuir uma ação que deve ser executada quando uma mensagem se encaixa nestes critérios.

Em **/etc/rsyslog.conf**, você também pode ver uma lista de arquivos de log mantidos por **rsyslogd**. A maioria dos arquivos de log estão localizados no diretório **/var/log/**. Algumas aplicações, tais como **httpd** e **samba**, armazenam seus arquivos de log em um subdiretório dentro de **/var/log/**.

Recursos adicionais

- As páginas de manual **rsyslogd(8)** e **rsyslog.conf(5)**
- Documentação instalada com o pacote **rsyslog-doc** em file:///usr/share/doc/rsyslog/html/index.html

13.2. INSTALANDO A DOCUMENTAÇÃO RSYSLOG

O aplicativo Rsyslog tem uma extensa documentação que está disponível em https://www.rsyslog.com/doc/, mas você também pode instalar o pacote de documentação **rsyslog-doc** localmente, seguindo este procedimento.

Pré-requisitos

- Você ativou o repositório **AppStream** em seu sistema
- Você está autorizado a instalar novos pacotes usando **sudo**

Procedimento

- Instale o pacote **rsyslog-doc**:

```
$ sudo yum install rsyslog-doc
```

Verificação

- Abra o arquivo:///usr/share/doc/rsyslog/html/index.html em um navegador de sua escolha, por exemplo:

```
$ firefox file:///usr/share/doc/rsyslog/html/index.html
```

13.3. CONFIGURAÇÃO DE REGISTRO REMOTO SOBRE TCP

A aplicação Rsyslog permite que você execute um servidor de registro e configure sistemas individuais para enviar seus arquivos de registro para o servidor de registro. Para usar o registro remoto através de TCP, configure tanto o servidor quanto o cliente. O servidor coleta e analisa os logs enviados por um ou mais sistemas clientes.

Com o aplicativo Rsyslog, você pode manter um sistema de registro centralizado onde as mensagens de registro são encaminhadas para um servidor através da rede. Para evitar a perda de mensagens quando o servidor não está disponível, você pode configurar uma fila de ação para a ação de encaminhamento. Desta forma, as mensagens que não puderam ser enviadas são armazenadas localmente até que o servidor esteja novamente disponível. Note que tais filas não podem ser configuradas para conexões usando o protocolo UDP.

O plug-in **omfwd** permite o encaminhamento sobre UDP ou TCP. O protocolo padrão é o UDP. Como o plug-in está embutido, não precisa ser carregado.

13.3.1. Configuração de um servidor para o logon remoto sobre TCP

Siga este procedimento para configurar um servidor para coleta e análise dos logs enviados por um ou mais sistemas clientes.

Por padrão, **rsyslog** usa TCP na porta **514**.

Pré-requisitos

- **rsyslog** está instalado no sistema do servidor
- Você está logado como root no servidor

Procedimento

1. Opcional: Para utilizar uma porta diferente para o tráfego **rsyslog**, adicione o tipo **syslogd_port_t** SELinux à porta. Por exemplo, habilite a porta **30514**:

```
# semanage port -a -t syslogd_port_t -p tcp 30514
```

2. Opcional: Para utilizar uma porta diferente para o tráfego **rsyslog**, configure **firewalld** para permitir o tráfego de entrada **rsyslog** nessa porta. Por exemplo, permitir o tráfego TCP na porta **30514** na zona **zone**:

```
# firewall-cmd --zone=zone --permanent --add-port=30514/tcp
success
```

3. Criar um novo arquivo no diretório **/etc/rsyslog.d/** chamado, por exemplo, **remotelog.conf**, e inserir o seguinte conteúdo:

```
# Define templates before the rules that use them
### Per-Host Templates for Remote Systems ###
```

```
template(name="TmplAuthpriv" type="list") {
    constant(value="/var/log/remote/auth/")
    property(name="hostname")
    constant(value="/")
    property(name="programname" SecurePath="replace")
    constant(value=".log")
    }

template(name="TmplMsg" type="list") {
    constant(value="/var/log/remote/msg/")
    property(name="hostname")
    constant(value="/")
    property(name="programname" SecurePath="replace")
    constant(value=".log")
    }

# Provides TCP syslog reception
module(load="imtcp")
# Adding this ruleset to process remote messages
ruleset(name="remote1"){
     authpriv.*   action(type="omfile" DynaFile="TmplAuthpriv")
      *.info;mail.none;authpriv.none;cron.none
action(type="omfile" DynaFile="TmplMsg")
}

input(type="imtcp" port="30514" ruleset="remote1")
```

4. Salvar as mudanças no arquivo **/etc/rsyslog.d/remotelog.conf**.

5. Certifique-se de que o serviço **rsyslog** esteja rodando e ativado no servidor de registro:

```
# systemctl status rsyslog
```

6. Reinicie o serviço **rsyslog**.

```
# systemctl restart rsyslog
```

7. Opcional: Se **rsyslog** não estiver habilitado, certifique-se de que o serviço **rsyslog** seja iniciado automaticamente após a reinicialização:

```
# systemctl enable rsyslog
```

Seu servidor de log está agora configurado para receber e armazenar arquivos de log de outros sistemas em seu ambiente.

Verificação

- Teste a sintaxe do arquivo **/etc/rsyslog.conf**:

```
# rsyslogd -N 1
rsyslogd: version 8.1911.0-2.el8, config validation run (level 1), master config
/etc/rsyslog.conf
rsyslogd: End of config validation run. Bye.
```

Recursos adicionais

- As páginas **rsyslogd(8)**, **rsyslog.conf(5)**, **semanage(8)**, e **firewall-cmd(1)** man
- Documentação instalada com o pacote **rsyslog-doc** em file:///usr/share/doc/rsyslog/html/index.html

13.3.2. Configuração de registro remoto para um servidor através de TCP

Siga este procedimento para configurar um sistema de encaminhamento de mensagens de registro para um servidor através do protocolo TCP. O plug-in **omfwd** permite o encaminhamento sobre o UDP ou TCP. O protocolo padrão é o UDP. Como o plug-in está embutido, não é necessário carregá-lo.

Pré-requisitos

- O pacote **rsyslog** é instalado nos sistemas do cliente que devem se reportar ao servidor.
- Você configurou o servidor para o registro remoto.
- A porta especificada é permitida no SELinux e aberta no firewall.

Procedimento

1. Criar um novo arquivo no diretório **/etc/rsyslog.d/** chamado, por exemplo, **remotelog.conf**, e inserir o seguinte conteúdo:

```
*.* action(type="omfwd"
     queue.type="linkedlist"
     queue.filename="example_fwd"
     action.resumeRetryCount="-1"
     queue.saveOnShutdown="on"
     target="example.com" port="30514" protocol="tcp"
    )
```

Onde:

- **queue.type="linkedlist"** permite uma fila in-memory da LinkedList,
- **queue.filename** define um armazenamento em disco. Os arquivos de backup são criados com o prefixo **example_fwd** no diretório de trabalho especificado pela diretiva global **workDirectory** anterior,
- a configuração **action.resumeRetryCount -1** impede que **rsyslog** deixe cair mensagens ao tentar conectar novamente se o servidor não estiver respondendo,
- habilitado **queue.saveOnShutdown="on"** salva dados in-memory se **rsyslog** for desligado,
- a última linha encaminha todas as mensagens recebidas para o servidor de registro, a especificação da porta é opcional.

Com esta configuração, **rsyslog** envia mensagens para o servidor, mas mantém mensagens na memória se o servidor remoto não for alcançável. Um arquivo em disco é criado somente se **rsyslog** ficar sem espaço na fila de memória configurada ou precisar ser desligado, o que beneficia o desempenho do sistema.

2. Reinicie o serviço **rsyslog**.

```
# systemctl restart rsyslog
```

Verificação

Para verificar se o sistema cliente envia mensagens para o servidor, siga estes passos:

1. No sistema do cliente, envie uma mensagem de teste:

```
# logger test
```

2. No sistema do servidor, veja o log **/var/log/messages**, por exemplo:

```
# cat /var/log/remote/msg/hostname/root.log
Feb 25 03:53:17 hostname root[6064]: test
```

Onde *hostname* é o nome do host do sistema do cliente. Note que o log contém o nome do usuário que digitou o comando **logger**, neste caso **root**.

Recursos adicionais

- As páginas de manual **rsyslogd(8)** e **rsyslog.conf(5)**
- Documentação instalada com o pacote **rsyslog-doc** em file:///usr/share/doc/rsyslog/html/index.html

13.4. CONFIGURAÇÃO DO REGISTRO REMOTO SOBRE O UDP

O aplicativo **Rsyslog** permite que você configure um sistema para receber informações de registro de dados de sistemas remotos. Para utilizar o registro remoto através do UDP, configure tanto o servidor quanto o cliente. O servidor receptor coleta e analisa os logs enviados por um ou mais sistemas clientes. Por padrão, **rsyslog** usa o UDP na porta **514** para receber informações de logs de sistemas remotos.

13.4.1. Configuração de um servidor para receber informações de registro remoto sobre o UDP

Siga este procedimento para configurar um servidor para coleta e análise dos logs enviados por um ou mais sistemas clientes sobre o protocolo UDP.

Pré-requisitos

- O utilitário **rsyslog** está instalado.

Procedimento

1. Opcional: Para utilizar uma porta diferente para o tráfego **rsyslog** do que a porta padrão **514**:
   a. Adicionar o tipo **syslogd_port_t** SELinux à configuração da política SELinux, substituindo ***portno*** com o número do porto que você deseja utilizar **rsyslog**:

```
# semanage port -a -t syslogd_port_t -p udp portno
```

b. Configurar **firewalld** para permitir o tráfego de entrada **rsyslog**, substituindo ***portno*** com o número do porto e **zone** com a zona que você deseja utilizar **rsyslog**:

```
# firewall-cmd --zone=zone --permanent --add-port=portno/udp
success
```

c. Recarregue as regras de firewall:

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

2. Crie um novo arquivo **.conf** no diretório **/etc/rsyslog.d/**, por exemplo, **remotelogserv.conf**, e insira o seguinte conteúdo:

```
# Define templates before the rules that use them
### Per-Host Templates for Remote Systems ###
template(name="TmplAuthpriv" type="list") {
    constant(value="/var/log/remote/auth/")
    property(name="hostname")
    constant(value="/")
    property(name="programname" SecurePath="replace")
    constant(value=".log")
    }

template(name="TmplMsg" type="list") {
    constant(value="/var/log/remote/msg/")
    property(name="hostname")
    constant(value="/")
    property(name="programname" SecurePath="replace")
    constant(value=".log")
    }

# Provides UDP syslog reception
module(load="imudp")

# This ruleset processes remote messages
ruleset(name="remote1"){
     authpriv.*   action(type="omfile" DynaFile="TmplAuthpriv")
      *.info;mail.none;authpriv.none;cron.none
action(type="omfile" DynaFile="TmplMsg")
}

input(type="imudp" port="514" ruleset="remote1")
```

Onde **514** é o número da porta **rsyslog** usa por padrão. Você pode especificar uma porta diferente em seu lugar.

3. Reinicie o serviço **rsyslog**.

```
# systemctl restart rsyslog
```

4. Opcional: Se **rsyslog** não estiver habilitado, certifique-se de que o serviço **rsyslog** seja iniciado automaticamente após a reinicialização:

```
# systemctl enable rsyslog
```

Verificação

1. Verificar a sintaxe do arquivo **/etc/rsyslog.conf** e de todos os arquivos **.conf** no diretório **/etc/rsyslog.d/**:

```
# rsyslogd -N 1
rsyslogd: version 8.1911.0-2.el8, config validation run (level 1), master config
/etc/rsyslog.conf
rsyslogd: End of config validation run. Bye.
```

Recursos adicionais

- As páginas **rsyslogd(8)**, **rsyslog.conf(5)**, **semanage(8)**, e **firewall-cmd(1)** man
- Documentação baseada no navegador, que você pode instalar a partir do pacote **rsyslog-doc**, em **file:///usr/share/doc/rsyslog/html/index.html**

13.4.2. Configuração do registro remoto para um servidor sobre UDP

Siga este procedimento para configurar um sistema de encaminhamento de mensagens de registro para um servidor sobre o protocolo UDP. O plug-in **omfwd** permite o encaminhamento sobre o UDP ou TCP. O protocolo padrão é o UDP. Como o plug-in está embutido, não é necessário carregá-lo.

Pré-requisitos

- O pacote **rsyslog** é instalado nos sistemas do cliente que devem se reportar ao servidor.
- Você configurou o servidor para registro remoto como descrito em Configuração de um servidor para receber informações de registro remoto sobre UDP.

Procedimento

1. Crie um novo arquivo **.conf** no diretório **/etc/rsyslog.d/**, por exemplo, **remotelogcli.conf**, e insira o seguinte conteúdo:

```
*.* action(type="omfwd"
     queue.type="linkedlist"
     queue.filename="example_fwd"
     action.resumeRetryCount="-1"
     queue.saveOnShutdown="on"
     target="example.com" port="portno" protocol="udp"
    )
```

Onde:

- **queue.type="linkedlist"** permite uma fila in-memory da LinkedList.
- **queue.filename** define um armazenamento em disco. Os arquivos de backup são criados com o ***example_fwd*** prefixo no diretório de trabalho especificado pela diretiva global anterior **workDirectory**.
- A configuração **action.resumeRetryCount -1** impede que **rsyslog** deixe cair mensagens ao tentar conectar novamente se o servidor não estiver respondendo.
- **enabled queue.saveOnShutdown="on"** salva dados in-memory se **rsyslog** se desligar.

- ***portno*** é o número da porta que você deseja utilizar **rsyslog**. O valor padrão é **514**.
- A última linha encaminha todas as mensagens recebidas para o servidor de registro, a especificação da porta é opcional.
  Com esta configuração, **rsyslog** envia mensagens para o servidor, mas mantém mensagens na memória se o servidor remoto não for alcançável. Um arquivo em disco é criado somente se **rsyslog** ficar sem espaço na fila de memória configurada ou precisar ser desligado, o que beneficia o desempenho do sistema.

2. Reinicie o serviço **rsyslog**.

```
# systemctl restart rsyslog
```

3. Opcional: Se **rsyslog** não estiver habilitado, certifique-se de que o serviço **rsyslog** seja iniciado automaticamente após a reinicialização:

```
# systemctl enable rsyslog
```

Verificação

Para verificar se o sistema cliente envia mensagens para o servidor, siga estes passos:

1. No sistema do cliente, envie uma mensagem de teste:

```
# logger test
```

2. No sistema do servidor, veja o **/var/log/remote/msg/*hostname*/root.log** log, por exemplo:

```
# cat /var/log/remote/msg/hostname/root.log
Feb 25 03:53:17 hostname root[6064]: test
```

Onde **hostname** é o nome do host do sistema do cliente. Note que o log contém o nome do usuário que digitou o comando logger, neste caso **root**.

Recursos adicionais

- As páginas de manual **rsyslogd(8)** e **rsyslog.conf(5)**
- Documentação baseada no navegador, que você pode instalar a partir do pacote **rsyslog-doc**, em **file:///usr/share/doc/rsyslog/html/index.html**

13.5. CONFIGURAÇÃO DE REGISTRO REMOTO CONFIÁVEL

Com o Protocolo de Registro de Eventos Confiáveis (RELP), você pode enviar e receber mensagens **syslog** através do TCP com um risco muito reduzido de perda de mensagens. O RELP proporciona a entrega confiável de mensagens de eventos, o que o torna útil em ambientes onde a perda de mensagens não é aceitável. Para usar o RELP, configure o módulo de entrada **imrelp**, que roda no servidor e recebe os logs, e o módulo de saída **omrelp**, que roda no cliente e envia os logs para o servidor de logs.

Pré-requisitos

- Você instalou os pacotes **rsyslog**, **librelp**, e **rsyslog-relp** no servidor e nos sistemas do cliente.

- A porta especificada é permitida no SELinux e aberta no firewall.

Procedimento

1. Configurar o sistema do cliente para registro remoto confiável:

   a. No sistema do cliente, criar um novo arquivo **.conf** no diretório /**etc**/**rsyslog.d**/ chamado, por exemplo, **relpcli.conf**, e inserir o seguinte conteúdo:

   ```
   module(load="omrelp")
   *.* action(type="omrelp" target="target_IP" port="target_port")
   ```

   Onde:

   - ***target_IP*** é o endereço IP do servidor de registro.
   - ***target_port*** é a porta do servidor de registro de dados.

   b. Salvar as mudanças no arquivo /**etc**/**rsyslog.d**/**relpserv.conf**.

   c. Reinicie o serviço **rsyslog**.

   ```
   # systemctl restart rsyslog
   ```

   d. Opcional: Se **rsyslog** não estiver habilitado, certifique-se de que o serviço **rsyslog** seja iniciado automaticamente após a reinicialização:

   ```
   # systemctl enable rsyslog
   ```

2. Configurar o sistema do servidor para registro remoto confiável:

   a. No sistema do servidor, criar um novo arquivo **.conf** no diretório /**etc**/**rsyslog.d**/ chamado, por exemplo, **relpserv.conf**, e inserir o seguinte conteúdo:

   ```
   ruleset(name="relp"){
   *.* action(type="omfile" file="log_path")
   }


   module(load="imrelp")
   input(type="imrelp" port="target_port" ruleset="relp")
   ```

   Onde:

   - ***log_path*** especifica o caminho para o armazenamento das mensagens.
   - ***target_port*** é a porta do servidor de registro de dados. Use o mesmo valor que no arquivo de configuração do cliente.

   b. Salvar as mudanças no arquivo /**etc**/**rsyslog.d**/**relpserv.conf**.

   c. Reinicie o serviço **rsyslog**.

   ```
   # systemctl restart rsyslog
   ```

d. Opcional: Se **rsyslog** não estiver habilitado, certifique-se de que o serviço **rsyslog** seja iniciado automaticamente após a reinicialização:

```
# systemctl enable rsyslog
```

Verificação

Para verificar se o sistema cliente envia mensagens para o servidor, siga estes passos:

1. No sistema do cliente, envie uma mensagem de teste:

```
# logger test
```

2. No sistema do servidor, visualizar o log no ***log_path***por exemplo:

```
# cat /var/log/remote/msg/hostname/root.log
Feb 25 03:53:17 hostname root[6064]: test
```

Onde **hostname** é o nome do host do sistema do cliente. Note que o log contém o nome do usuário que digitou o comando logger, neste caso **root**.

Recursos adicionais

- As páginas de manual **rsyslogd(8)** e **rsyslog.conf(5)**
- Documentação baseada no navegador, que você pode instalar a partir do pacote **rsyslog-doc**, em **file:///usr/share/doc/rsyslog/html/index.html**

13.6. MÓDULOS RSYSLOG SUPORTADOS

Para expandir a funcionalidade do utilitário **Rsyslog**, você pode usar módulos adicionais específicos. Os módulos fornecem entradas adicionais (Módulos de Entrada), saídas (Módulos de Saída), e outras funcionalidades específicas. Um módulo também pode fornecer diretrizes de configuração adicionais que ficam disponíveis depois que você carrega esse módulo.

Liste os módulos de entrada e saída instalados em seu sistema com o seguinte comando:

```
# ls /usr/lib64/rsyslog/{i,o}m* *
```

Para ver a lista de todos os módulos **rsyslog** disponíveis, abra a seguinte página a partir da documentação instalada a partir do pacote **rsyslog-doc**.

```
$ firefox file:///usr/share/doc/rsyslog/html/configuration/modules/idx_output.html
```

13.7. RECURSOS ADICIONAIS

- Documentação instalada com o pacote **rsyslog-doc** em file:///usr/share/doc/rsyslog/html/index.html
- As páginas de manual **rsyslog.conf(5)** e **rsyslogd(8)**
- A configuração do sistema de registro sem journald ou com uso minimizado de journald Artigo da Knowledgebase

- Os efeitos negativos da configuração de registro padrão da RHEL sobre o desempenho e suas atenuações artigo

CAPÍTULO 14. USANDO O PAPEL DO SISTEMA DE REGISTRO

Como administrador do sistema, você pode usar a função de sistema de registro para configurar um host RHEL como um servidor de registro para coletar registros de muitos sistemas clientes.

14.1. O PAPEL DO SISTEMA DE REGISTRO

Com o papel do sistema de registro, você pode implantar configurações de registro em hosts locais e remotos.

Para aplicar uma função de sistema de registro em um ou mais sistemas, você define a configuração de registro em um *playbook*. Um playbook é uma lista de uma ou mais peças. Os playbooks são legíveis por humanos, e são escritos no formato YAML. Para mais informações sobre playbooks, consulte Trabalhando com playbooks em Documentação possível.

O conjunto de sistemas que você deseja Ansible para configurar de acordo com o playbook está definido em um *inventory file*. Para mais informações sobre como criar e utilizar inventários, veja Como construir seu inventário na documentação do Ansible.

As soluções de registro fornecem múltiplas formas de leitura de registros e múltiplas saídas de registro.

Por exemplo, um sistema de registro pode receber as seguintes entradas:

- arquivos locais,
- **systemd/journal**,
- outro sistema de registro através da rede.

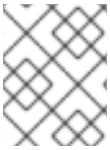
Além disso, um sistema de registro pode ter as seguintes saídas:

- os logs são armazenados nos arquivos locais no diretório /**var**/**log**,
- os logs são enviados para a Elasticsearch,
- os logs são encaminhados para outro sistema de extração.

Com o papel do sistema de registro, você pode combinar as entradas e saídas para atender às suas necessidades. Por exemplo, você pode configurar uma solução de registro que armazena entradas de **journal** em um arquivo local, enquanto as entradas lidas dos arquivos são ambas encaminhadas para outro sistema de registro e armazenadas nos arquivos de registro locais.

14.2. PARÂMETROS DE PAPEL DO SISTEMA DE REGISTRO

Em um livro de jogo de funções do sistema de registro, você define as entradas no parâmetro **logging_inputs**, as saídas no parâmetro **logging_outputs** e as relações entre as entradas e saídas no parâmetro **logging_flows**. O Sistema de Logging Role processa estas variáveis com opções adicionais para configurar o sistema de registro. Você também pode habilitar a criptografia.



NOTA

Atualmente, o único sistema de registro disponível na função de sistema de registro é **Rsyslog**.

- **logging_inputs** - Lista de insumos para a solução de extração.

- **name** - Nome único da entrada. Usado na lista de entradas **logging_flows** e uma parte do nome do arquivo **config** gerado.
- **type** - Tipo do elemento de entrada. O tipo especifica um tipo de tarefa que corresponde a um nome de diretório em **roles/rsyslog/{tasks,vars}/inputs/**.
  - **basics** - Entradas configurando entradas do jornal **systemd** ou do soquete **unix**.
    - **kernel_message** - Carregar **imklog** se estiver definido para **true**. Default para **false**.
    - **use_imuxsock** - Use **imuxsock** ao invés de **imjournal**. Por omissão para **false**.
    - **ratelimit_burst** - Número máximo de mensagens que podem ser emitidas dentro de **ratelimit_interval**. Por omissão para **20000** se **use_imuxsock** for falso. Predefinição para **200** se **use_imuxsock** for verdadeiro.
    - **ratelimit_interval** - Intervalo para avaliação **ratelimit_burst**. Por omissão para 600 segundos se **use_imuxsock** for falso. Default para 0 se **use_imuxsock** for verdadeiro. 0 indica que a limitação da taxa está desligada.
    - **persist_state_interval** - O estado do periódico é persistido a cada **value** mensagens. Por omissão para **10**. Efetivo somente quando **use_imuxsock** é falso.
  - **files** - Entradas configurando entradas a partir de arquivos locais.
  - **remote** - Entradas configurando as entradas do outro sistema de registro através da rede.
- **state** - Estado do arquivo de configuração. **present** ou **absent**. Padrão para **present**.

- **logging_outputs** - Lista de saídas para a solução de extração.
  - **files** - Saídas configurando saídas para arquivos locais.
  - **forwards** - Saídas configurando saídas para outro sistema de registro.
  - **remote_files** - Saídas configurando as saídas de outro sistema de registro em arquivos locais.
- **logging_flows** - Lista de fluxos que definem as relações entre **logging_inputs** e **logging_outputs**. A variável **logging_flows** tem as seguintes chaves:
  - **name** - Nome único do fluxo
  - **inputs** - Lista de valores do nome **logging_inputs**
  - **outputs** - Lista de valores do nome **logging_outputs**.

Recursos adicionais

- Documentação instalada com o pacote **rhel-system-roles** em **/usr/share/ansible/roles/rhel-system-roles.logging/README.html**

14.3. APLICAÇÃO DE UMA FUNÇÃO DE SISTEMA DE REGISTRO LOCAL

Siga estes passos para preparar e aplicar um livro de jogo do Red Hat Ansible Engine para configurar uma solução de registro em um conjunto de máquinas separadas. Cada máquina registrará os logs localmente.

Pré-requisitos

- Você tem o Red Hat Ansible Engine instalado no sistema a partir do qual você deseja executar o playbook.

NOTA

Você não precisa ter o Red Hat Ansible Engine instalado nos sistemas nos quais você deseja implantar a solução de registro.

- Você tem o pacote **rhel-system-roles** sobre o sistema a partir do qual você deseja executar o playbook.

NOTA

Você não precisa ter **rsyslog** instalado, porque a função do sistema instala **rsyslog** quando implantado.

- Você tem um arquivo de inventário listando os sistemas nos quais você deseja configurar a solução de registro.

Procedimento

1. Criar um playbook que defina o papel exigido:

   a. Criar um novo arquivo YAML e abri-lo em um editor de texto, por exemplo:

   ```
   # vi logging-playbook.yml
   ```

   b. Insira o seguinte conteúdo:

   ```
   ---
   - name: Deploying basics input and implicit files output
     hosts: all
     roles:
       - linux-system-roles.logging
     vars:
       logging_inputs:
         - name: system_input
           type: basics
       logging_outputs:
         - name: files_output
           type: files
       logging_flows:
         - name: flow1
           inputs: [system_input]
           outputs: [files_output]
   ```

2. Executar o playbook em um inventário específico:

```
# ansible-playbook -i inventory-file /path/to/file/logging-playbook.yml
```

Onde:

- ***inventory-file*** é o arquivo do inventário.
- ***logging-playbook.yml*** é o livro de jogo que você usa.

Verificação

1. Teste a sintaxe do arquivo **/etc/rsyslog.conf**:

```
# rsyslogd -N 1
rsyslogd: version 8.1911.0-6.el8, config validation run (level 1), master config
/etc/rsyslog.conf
rsyslogd: End of config validation run. Bye.
```

2. Verifique se o sistema envia mensagens para o log:
   a. Envie uma mensagem de teste:

```
# logger test
```

   b. Veja o registro **/var/log/messages**, por exemplo:

```
# cat /var/log/messages
Aug  5 13:48:31 hostname root[6778]: test
```

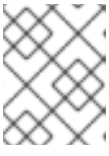
   Onde `*hostname*` é o nome do host do sistema do cliente. Note que o log contém o nome do usuário que digitou o comando logger, neste caso **root**.

14.4. APLICAÇÃO DE UMA SOLUÇÃO DE REGISTRO REMOTO UTILIZANDO O PAPEL DO SISTEMA DE REGISTRO

Siga estes passos para preparar e aplicar um livro de exercícios do Red Hat Ansible Engine para configurar uma solução de registro remoto. Neste playbook, um ou mais clientes pegam logs de **systemd-journal** e os encaminham para um servidor remoto. O servidor recebe entradas remotas de **remote_rsyslog** e **remote_files** e envia os logs para arquivos locais em diretórios nomeados por nomes de hosts remotos.

Pré-requisitos

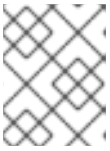
- Você tem o Red Hat Ansible Engine instalado no sistema a partir do qual você deseja executar o playbook.



NOTA

Você não precisa ter o Red Hat Ansible Engine instalado nos sistemas nos quais você deseja implantar a solução de registro.

- Você tem o pacote **rhel-system-roles** sobre o sistema a partir do qual você deseja executar o playbook.



NOTA

Você não precisa ter **rsyslog** instalado, porque a função do sistema instala **rsyslog** quando implantado.

- Você tem pelo menos dois sistemas:
  - Pelo menos um será o servidor de registro.
  - Pelo menos um será o cliente madeireiro.

Procedimento

1. Criar um playbook que defina o papel exigido:
   a. Criar um novo arquivo YAML e abri-lo em um editor de texto, por exemplo:

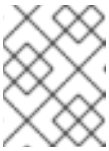
```
# vi logging-playbook.yml
```

   b. Insira o seguinte conteúdo no arquivo:

```
---
- name: Deploying remote input and remote_files output
  hosts: server
  roles:
    - linux-system-roles.logging
  vars:
    logging_inputs:
      - name: remote_udp_input
        type: remote
        udp_ports: [ 601 ]
      - name: remote_tcp_input
        type: remote
        tcp_ports: [ 601 ]
    logging_outputs:
      - name: remote_files_output
        type: remote_files
    logging_flows:
      - name: flow_0
        inputs: [remote_udp_input, remote_tcp_input]
        outputs: [remote_files_output]

- name: Deploying basics input and forwards output
  hosts: clients
  roles:
    - linux-system-roles.logging
  vars:
    logging_inputs:
      - name: basic_input
        type: basics
    logging_outputs:
      - name: forward_output0
        type: forwards
        severity: info
        target: host1.example.com
        udp_port: 601
```

```
    - name: forward_output1
      type: forwards
      facility: mail
      target: host1.example.com
      tcp_port: 601
  logging_flows:
    - name: flows0
      inputs: [basic_input]
      outputs: [forward_output0, forward_output1]

[basic_input]
[forward_output0, forward_output1]
```

Onde ***host1.example.com*** é o servidor de registro.



NOTA

Você pode modificar os parâmetros no livro de jogo para atender às suas necessidades.

ATENÇÃO

A solução de registro funciona somente com as portas definidas na política SELinux do sistema servidor ou cliente e abertas no firewall. A política padrão do SELinux inclui as portas 601, 514, 6514, 10514, e 20514. Para utilizar uma porta diferente, modifique a política SELinux no sistema cliente e no sistema servidor. A configuração do firewall através das funções do sistema ainda não é suportada.

2. Crie um arquivo de inventário que lista seus servidores e clientes:

   a. Criar um novo arquivo e abri-lo em um editor de texto, por exemplo:

   ```
   # vi inventory.ini
   ```

   b. Insira o seguinte conteúdo no arquivo do inventário:

   ```
   [servers]
   server ansible_host=host1.example.com
   [clients]
   client ansible_host=host2.example.com
   ```

   Where: * ***host1.example.com*** is the logging server. * ***host2.example.com*** is the logging client.

3. Execute o playbook em seu inventário.

   ```
   # ansible-playbook -i /path/to/file/inventory.ini /path/to/file/_logging-playbook.yml
   ```

   Onde:

- ***inventory.ini*** é o arquivo do inventário.
- ***logging-playbook.yml*** é o livro de jogo que você criou.

Etapas de verificação

1. Tanto no sistema cliente quanto no servidor, teste a sintaxe do arquivo **/etc/rsyslog.conf**:

```
# rsyslogd -N 1
rsyslogd: version 8.1911.0-6.el8, config validation run (level 1), master config
/etc/rsyslog.conf
rsyslogd: End of config validation run. Bye.
```

2. Verificar se o sistema cliente envia mensagens para o servidor:
   a. No sistema do cliente, envie uma mensagem de teste:

```
# logger test
```

   b. No sistema do servidor, veja o log **/var/log/messages**, por exemplo:

```
# cat /var/log/messages
Aug  5 13:48:31 host2.example.com root[6778]: test
```

   Onde ***host2.example.com*** é o nome do host do sistema cliente. Note que o log contém o nome do usuário que digitou o comando logger, neste caso **root**.

Recursos adicionais

- Começando com os papéis do Sistema RHEL
- Documentação instalada com o pacote **rhel-system-roles** em **/usr/share/ansible/roles/rhel-system-roles.logging/README.html**
- Artigo KB daRHEL System Roles

14.5. RECURSOS ADICIONAIS

- Começando com os papéis do Sistema RHEL
- Documentação instalada com o pacote **rhel-system-roles** em **/usr/share/ansible/roles/rhel-system-roles.logging/README.html**
- Artigo KB daRHEL System Roles

CAPÍTULO 15. USANDO PYTHON

15.1. INTRODUÇÃO À PYTHON

Python é uma linguagem de programação de alto nível que suporta múltiplos paradigmas de programação, tais como orientada a objetos, imperativa, funcional e processual. Python tem uma semântica dinâmica e pode ser usada para programação de propósito geral.

Com o Red Hat Enterprise Linux, muitos pacotes que são instalados no sistema, tais como pacotes que fornecem ferramentas de sistema, ferramentas para análise de dados ou aplicações web, são escritos em Python. Para poder usar estes pacotes, você precisa ter os pacotes **python** instalados.

15.1.1. Versões Python

Duas versões incompatíveis de Python são amplamente utilizadas, Python 2.x e Python 3.x.

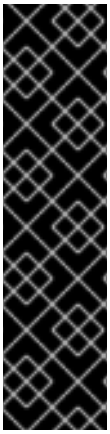
A RHEL 8 fornece as seguintes versões do Python.

Versão	Pacote para instalar	Exemplos de comando	Disponível desde	Ciclo de vida
Python 3.6	**python3**	**python3**, **pip3**	RHEL 8.0	rHEL 8 completo
Python 2.7	**python2**	**python2**, **pip2**	RHEL 8.0	mais curto
Python 3.8	**python38**	**python3.8**, **pip3.8**	RHEL 8.2	mais curto

Veja Red Hat Enterprise Linux Life Cycle e Red Hat Enterprise Linux 8 Application Streams Life Cycle para obter detalhes sobre a duração do suporte.

Cada uma das versões Python é distribuída em um módulo separado, e pelo projeto, é possível instalar vários módulos em paralelo no mesmo sistema.

O módulo **python38** não inclui as mesmas ligações com as ferramentas do sistema (RPM, DNF, SELinux e outras) que são fornecidas para o módulo **python36**.



IMPORTANTE

Sempre especifique a versão do Python ao instalá-lo, invocá-lo ou interagir de outra forma com ele. Por exemplo, use **python3** ao invés de **python** em nomes de pacotes e comandos. Todos os comandos relacionados ao Python devem incluir também a versão, por exemplo, **pip3**, **pip2**, ou **pip3.8**.

O comando não versionado **python** (**/usr/bin/python**) não está disponível por padrão no RHEL 8. Você pode configurá-lo usando o comando **alternatives**; para instruções, veja Configurando o Python não versionado. Qualquer alteração manual em **/usr/bin/python**, exceto alterações feitas usando o comando **alternatives**, podem ser sobrescritas após uma atualização.

Como administrador de sistemas, recomenda-se usar preferencialmente o Python 3 pelas seguintes razões:

- A Python 3 representa a principal direção de desenvolvimento do projeto Python.
- O apoio ao Python 2 na comunidade a montante termina em 2020.
- As bibliotecas populares Python estão deixando cair o suporte Python 2 no rio acima.
- Python 2 no Red Hat Enterprise Linux 8 terá um ciclo de vida mais curto e seu objetivo é facilitar uma transição mais suave para **Python 3** para os clientes.

Para os desenvolvedores, Python 3 tem as seguintes vantagens sobre Python 2:

- Python 3 permite escrever código expressivo, de fácil manutenção e correto.
- O código escrito em Python 3 terá uma longevidade maior.
- Python 3 tem novas características, incluindo asyncio, f-strings, desembalagem avançada, argumentos apenas com palavras-chave, exceções encadeadas e muito mais.

Entretanto, o software existente tende a exigir que **/usr/bin/python** seja o Python 2. Por esta razão, nenhum pacote padrão **python** é distribuído com o Red Hat Enterprise Linux 8, e você pode escolher entre usar Python 2 e 3 como **/usr/bin/python**, conforme descrito em Seção 15.2.5, "Configurando o Python não versionado".

15.1.2. O pacote plataforma interna - pitão

As ferramentas do sistema no Red Hat Enterprise Linux 8 usam uma versão 3.6 do Python fornecida pelo pacote interno **platform-python**. A Red Hat aconselha os clientes a usar o pacote **python36** em seu lugar.

15.2. INSTALANDO E USANDO PYTHON

ATENÇÃO

O uso do comando **python** não versionado para instalar ou executar Python não funciona por padrão devido à ambigüidade. Sempre especifique a versão do Python, ou configure a versão padrão do sistema usando o comando **alternatives**.

15.2.1. Instalando o Python 3

No Red Hat Enterprise Linux 8, o Python 3 é distribuído nas versões 3.6 e 3.8, fornecidas pelos módulos **python36** e **python38** no repositório AppStream.

Procedimento

- Para instalar o Python 3.6 a partir do módulo **python36**, execute o seguinte comando:

```
# yum instalar python3
```

O fluxo do módulo python36:3.6 é ativado automaticamente.

- Para instalar o Python 3.8 a partir do módulo **python38**, use:

```
# yum instalar python38
```

O fluxo do módulo python38:3.8 é ativado automaticamente.

Para detalhes sobre os módulos no RHEL 8, consulte Instalação, gerenciamento e remoção de componentes de espaço do usuário.

NOTA

Por projeto, os módulos RHEL 8 podem ser instalados em paralelo, incluindo os módulos **python27**, **python36**, e **python38**. Note que a instalação paralela não é suportada para múltiplas correntes dentro de um único módulo.

Python 3.8 e pacotes construídos para ele podem ser instalados em paralelo com o Python 3.6 no mesmo sistema, com a exceção do módulo **mod_wsgi**. Devido a uma limitação do Servidor HTTP Apache, apenas um dos pacotes **python3-mod_wsgi** e **python38-mod_wsgi** pode ser instalado em um sistema.

Os pacotes com módulos adicionais para Python 3.6 geralmente usam o prefixo **python3-**; os pacotes para Python 3.8 incluem o prefixo **python38-**. Sempre inclua o prefixo ao instalar pacotes Python adicionais, como mostrado nos exemplos abaixo.

Procedimento

- Para instalar o módulo **Requests** para Python 3.6, execute este comando:

```
# yum instalar pedidos de python3
```

- Para instalar a extensão **Cython** para Python 3.8, use:

```
# yum instalar python38-Cython
```

15.2.1.1. Instalação de pacotes Python 3 adicionais para desenvolvedores

Pacotes Python 3.8 adicionais para desenvolvedores são distribuídos através do repositório CodeReady Linux Builder no módulo **python38-devel**. Este módulo contém o pacote **python38-pytest** e suas dependências: os pacotes **pyparsing**, **atomicwrites**, **attrs**, **packaging**, **py**, **more-itertools**, **pluggy**, e **wcwidth**.

IMPORTANTE

O repositório CodeReady Linux Builder e seu conteúdo não tem o suporte da Red Hat.

Para instalar pacotes do módulo **python38-devel**, siga o procedimento abaixo.

Procedimento

- Habilitar o repositório não suportado CodeReady Linux Builder:

```
# assinatura-manager repos --enable codeready-builder-for-rhel-8-x86_64-rpms
```

- Habilite o módulo **python38-devel**:

```
# módulo yum habilita o python38-devel
```

- Instale o pacote **python38-pytest**:

```
# yum instalar python38-pytest
```

Para mais informações sobre o repositório CodeReady Linux Builder, veja Como habilitar e fazer uso do conteúdo dentro do CodeReady Linux Builder.

15.2.2. Instalando o Python 2

Alguns softwares ainda não foram totalmente portados para Python 3, e precisam do Python 2 para operar. O Red Hat Enterprise Linux 8 permite a instalação paralela do Python 3 e do Python 2. Se você precisar da funcionalidade Python 2, instale o módulo **python27**, que está disponível no repositório AppStream.

ATENÇÃO

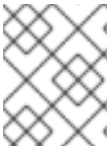
Observe que Python 3 é a principal direção de desenvolvimento do projeto Python. O suporte para Python 2 está sendo gradualmente eliminado. O módulo **python27** tem um período de suporte mais curto do que o Red Hat Enterprise Linux 8.

Procedimento

- Para instalar o Python 2.7 a partir do módulo **python27**, execute este comando:

```
# yum instalar python2
```

O fluxo do módulo python27:2.7 é ativado automaticamente.



NOTA

Por projeto, os módulos RHEL 8 podem ser instalados em paralelo, incluindo os módulos **python27**, **python36**, e **python38**.

Para detalhes sobre os módulos, consulte Instalação, gerenciamento e remoção de componentes de espaço do usuário.

Os pacotes com módulos adicionais para Python 2 geralmente usam o prefixo **python2-**. Sempre inclua o prefixo ao instalar pacotes Python adicionais, como mostrado nos exemplos abaixo.

Procedimento

- Para instalar o módulo **Requests** para Python 2, execute este comando:

```
# yum instalar pedidos de python2
```

- Para instalar a extensão **Cython** para Python 2, use:

```
# yum instalar python2-Cython
```

15.2.3. Usando Python 3

Ao executar o intérprete Python ou comandos relacionados ao Python, especifique sempre a versão.

Procedimento

- Para executar o intérprete Python 3.6 ou comandos relacionados, use, por exemplo:

```
$ python3
$ python3 -m cython --help
$ pip3 install <package>
```

- Para executar o intérprete Python 3.8 ou comandos relacionados, use, por exemplo:

```
$ python3.8
$ python3.8 -m cython --help
$ pip3.8 install <package>
```

15.2.4. Usando Python 2

Ao executar o intérprete Python 2 ou comandos relacionados ao Python2, especifique sempre a versão.

Procedimento

- Para executar o intérprete Python 2 ou comandos relacionados, use, por exemplo:

```
$ python2
$ python2 -m cython --help
$ pip2 install <package>
```

15.2.5. Configurando o Python não versionado

Os administradores do sistema podem configurar o comando **python** não versionado, localizado em **/usr/bin/python**, usando o comando **alternatives**. Note que o pacote necessário, **python3**, **python38**, ou **python2**, precisa ser instalado antes de configurar o comando não versionado para a respectiva versão.

IMPORTANTE

O executável **/usr/bin/python** é controlado pelo sistema **alternatives**. Quaisquer alterações manuais podem ser sobrescritas após uma atualização.

Comandos adicionais relacionados ao Python, tais como **pip3**, não têm variantes não versionadas configuráveis.

15.2.5.1. Configurando o comando python não versionado diretamente

Para configurar o comando não versionado **python** diretamente para uma versão selecionada do Python, use este procedimento.

Procedimento

- Para configurar o comando não versionado **python** para Python 3.6, execute este comando:

```
# alternativas --set python /usr/bin/python3
```

- Para configurar o comando não versionado **python** para Python 3.8, use o seguinte comando:

```
# alternativas --set python /usr/bin/python3.8
```

- Para configurar o comando não versionado **python** para Python 2, use:

```
# alternativas --set python /usr/bin/python2
```

15.2.5.2. Configurando o comando python não versionado para a versão Python requerida interativamente

Você também pode configurar o comando **python** não versionado para a versão Python exigida de forma interativa.

Para configurar o comando não versionado **python** de forma interativa, use este procedimento.

Procedimento

1. Execute o seguinte comando:

```
# alternativas --config python
```

2. Selecione a versão desejada da lista fornecida.
3. Para reiniciar esta configuração e remover o comando **python** não versionado, execute:

```
# alternativas --auto python
```

15.3. MIGRAÇÃO DE PYTHON 2 PARA PYTHON 3

Como desenvolvedor, você pode querer migrar seu antigo código escrito em Python 2 para Python 3. Para mais informações sobre como migrar grandes bases de código para Python 3, veja The Conservative Python 3 Porting Guide.

Note que após esta migração, o código Python 2 original torna-se interpretável pelo intérprete Python 3 e permanece interpretável também para o intérprete Python 2.

15.4. EMBALAGEM DE PYTHON 3 RPMS

A maioria dos projetos Python usa Setuptools para embalagem, e define as informações de embalagem no arquivo **setup.py**. Para mais informações sobre embalagens do Setuptools, consulte a documentação do Setuptools.

Você também pode embalar seu projeto Python em um pacote RPM, que oferece as seguintes vantagens em comparação com o pacote Setuptools:

- Especificação das dependências de um pacote em outras RPMs (mesmo não-Python)
- Assinatura criptográfica
  Com a assinatura criptográfica, o conteúdo dos pacotes de RPM pode ser verificado, integrado e testado com o resto do sistema operacional.

15.4.1. Descrição do arquivo SPEC para um pacote Python

Um arquivo SPEC contém instruções que o utilitário **rpmbuild** usa para construir um RPM. As instruções estão incluídas em uma série de seções. Um arquivo SPEC tem duas partes principais nas quais as seções são definidas:

- Preâmbulo (contém uma série de itens de metadados que são usados no Corpo)
- Corpo (contém a parte principal das instruções)

Para mais informações sobre os arquivos SPEC, consulte Embalagem e distribuição de software .

Um arquivo RPM SPEC para projetos Python tem algumas especificidades em comparação com arquivos SPEC não-Python RPM. Mais notavelmente, um nome de qualquer pacote RPM de uma biblioteca Python deve sempre incluir o prefixo que determina a versão, por exemplo, **python3** para Python 3.6 ou **python38** para Python 3.8.

Outras especificações são mostradas no seguinte arquivo da SPEC **example for the python3-detox package**. Para a descrição de tais especificidades, veja as notas abaixo do exemplo.

```
%global modname detox                                                    1

Name:           python3-detox                                            2
Version:        0.12
Release:        4%{?dist}
Summary:        Distributing activities of the tox tool
License:        MIT
URL:            https://pypi.io/project/detox
Source0:        https://pypi.io/packages/source/d/%{modname}/%{modname}-%{version}.tar.gz

BuildArch:      noarch

BuildRequires:  python36-devel                                           3
BuildRequires:  python3-setuptools
BuildRequires:  python36-rpm-macros
BuildRequires:  python3-six
BuildRequires:  python3-tox
BuildRequires:  python3-py
BuildRequires:  python3-eventlet

%?python_enable_dependency_generator                                     4

%description

Detox is the distributed version of the tox python testing tool. It makes efficient use of multiple CPUs
by running all possible activities in parallel.
```

Detox has the same options and configuration that tox has, so after installation you can run it in the same way and with the same options that you use for tox.

```
$ detox
```

```
%prep
%autosetup -n %{modname}-%{version}

%build
%py3_build                                                    5

%install
%py3_install

%check
%{__python3} setup.py test                                          6

%files -n python3-%{modname}
%doc CHANGELOG
%license LICENSE
%{_bindir}/detox
%{python3_sitelib}/%{modname}/
%{python3_sitelib}/%{modname}-%{version}*

%changelog
...
```

1. A macro **modname** contém o nome do projeto Python. Neste exemplo, ele é **detox**.
2. Ao embalar um projeto Python em RPM, o prefixo **python3** sempre precisa ser adicionado ao nome original do projeto. O nome original aqui é **detox** e o **name of the RPM** é **python3-detox**.
3. **BuildRequires** especifica quais pacotes são necessários para construir e testar este pacote. No BuildRequires, inclua sempre itens fornecendo ferramentas necessárias para a construção de pacotes Python: **python36-devel** e **python3-setuptools**. O pacote **python36-rpm-macros** é necessário para que os arquivos com **/usr/bin/python3** shebangs sejam automaticamente alterados para **/usr/bin/python3.6**. Para maiores informações, veja Seção 15.4.4, “Manuseio de hashbangs em scripts Python”.
4. Cada pacote Python requer alguns outros pacotes para funcionar corretamente. Tais pacotes também precisam ser especificados no arquivo da SPEC. Para especificar o **dependencies**, você pode usar a macro **%python_enable_dependency_generator** para usar automaticamente as dependências definidas no arquivo **setup.py**. Se um pacote tem dependências que não são especificadas usando Setuptools, especifique-as dentro das diretivas adicionais **Requires**.
5. As macros **%py3_build** e **%py3_install** executam os comandos **setup.py build** e **setup.py install**, respectivamente, com argumentos adicionais para especificar os locais de instalação, o intérprete a utilizar e outros detalhes.
6. A seção **check** fornece uma macro que executa a versão correta do Python. A macro **%{__python3}** contém um caminho para o intérprete Python 3, por exemplo **/usr/bin/python3**. Recomendamos usar sempre a macro em vez de um caminho literal.

15.4.2. Macros comuns para Python 3 RPMs

Em um arquivo SPEC, utilize sempre as macros abaixo, em vez de codificar rigidamente seus valores.

Em nomes macro, use sempre **python3** ou **python2** em vez de **python** não versionado. Configurar a versão particular do Python 3 no **BuildRequires** do arquivo SPEC para **python36-rpm-macros** ou **python38-rpm-macros**.

Macro	Definição normal	Descrição
%{__pitão3}	/usr/bin/python3	Intérprete Python 3
%{python3_version}	3.6	A versão completa do intérprete Python 3.
%{python3_sitelib}	/usr/lib/python3.6/site-packages	Onde são instalados módulos Python puros.
%{python3_sitearch}	/usr/lib64/pithon3.6/site-packages	Onde módulos contendo extensões específicas de arquitetura são instalados.
%py3_build		Executa o comando **setup.py build** com argumentos adequados para um pacote de sistema.
%py3_instalar		Executa o comando **setup.py install** com argumentos adequados para um pacote de sistema.

15.4.3. O sistema automático prevê Python RPMs

Ao embalar um projeto Python, certifique-se de que, se presente, os seguintes diretórios estejam incluídos no RPM resultante:

- **.dist-info**
- **.egg-info**
- **.egg-link**

A partir destes diretórios, o processo de construção do RPM gera automaticamente o virtual **pythonX.Ydist** fornece, por exemplo, **python3.6dist(detox)**. Estes provimentos virtuais são utilizados por pacotes que são especificados pela macro **%python_enable_dependency_generator**.

15.4.4. Manuseio de hashbangs em scripts Python

No Red Hat Enterprise Linux 8, espera-se que scripts Python executáveis usem hashbangs (shebangs) especificando explicitamente pelo menos a versão principal do Python.

O script **/usr/lib/rpm/redhat/brp-mangle-shebangs** buildroot policy (BRP) é executado automaticamente ao construir qualquer pacote RPM, e tenta corrigir hashbangs em todos os arquivos executáveis.

NOTA

O script BRP gera erros ao encontrar um script Python com um hashbang ambíguo, como por exemplo:

```
#! /usr/bin/pithon
```

ou

```
#! /usr/bin/env python
```

15.4.4.1. Modificando hashbangs em scripts Python

Para modificar os hashbangs nos scripts Python que causam os erros de construção no tempo de construção RPM, use este procedimento.

Procedimento

- Aplique o script **pathfix.py** do pacote **platform-python-devel**:

  ```
  # pathfix.py -pn -i %{__python3} PATH...
  ```

  Note que múltiplos ***PATH*s** pode ser especificado. Se um ***PATH*** é um diretório, **pathfix.py** escaneia recursivamente qualquer script Python que corresponda ao padrão **^[a-zA-Z0-9_] \.py$**, não apenas aqueles com um hashbang ambíguo. Adicione este comando à seção **%prep** ou ao final da seção **%install**.

Alternativamente, modifique os scripts Python empacotados para que eles estejam de acordo com o formato esperado. Para este fim, **pathfix.py** também pode ser usado fora do processo de construção RPM. Ao rodar **pathfix.py** fora de uma construção RPM, substitua ***__python3*** do exemplo acima com um caminho para o hashbang, tal como **/usr/bin/python3**.

Se os scripts Python empacotados exigirem uma versão diferente da Python 3.6, ajuste os comandos acima para incluir a respectiva versão.

15.4.4.2. Trocar /usr/bin/python3 hashbangs em seus pacotes personalizados

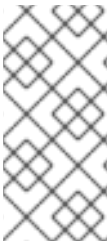
Adicionalmente, hashbangs no formulário **/usr/bin/python3** são por default substituídos por hashbangs apontando para Python do pacote **platform-python** usado para ferramentas de sistema com o Red Hat Enterprise Linux.

Para alterar o hashbangs **/usr/bin/python3** em seus pacotes personalizados para apontar para uma versão do Python instalada a partir do Application Stream, no formulário **/usr/bin/python3.6**, use o seguinte procedimento.

Procedimento

- Adicione o pacote **python36-rpm-macros** na seção **BuildRequires** do arquivo da SPEC, incluindo a seguinte linha:

```
BuildRequires:  python36-rpm-macros
```



NOTA

Para evitar verificação de hashbang e modificação pelo roteiro do BRP, use a seguinte diretiva RPM:

```
%undefine p_mangle_shebangs
```

Se você estiver usando outra versão que não a Python 3.6, ajuste os comandos acima para incluir a respectiva versão.

15.4.5. Recursos adicionais

- Para mais informações sobre embalagens RPM, consulte Embalagem e distribuição de software .

CAPÍTULO 16. USANDO A LINGUAGEM PHP SCRIPTING

Hypertext Preprocessor (PHP) é uma linguagem de script de uso geral utilizada principalmente para script do lado do servidor, que permite executar o código PHP usando um servidor web.

No RHEL 8, a linguagem PHP scripting é fornecida pelo módulo **php**, que está disponível em múltiplas correntes (versões).

Dependendo de seu caso de uso, você pode instalar um perfil específico do fluxo do módulo selecionado:

- **common** - O perfil padrão para scripting do lado do servidor usando um servidor web. Ele inclui várias extensões amplamente utilizadas.
- **minimal** - Este perfil instala apenas a interface de linha de comando para scripting com PHP sem utilizar um servidor web.
- **devel** - Este perfil inclui pacotes do perfil **common** e pacotes adicionais para fins de desenvolvimento.

16.1. INSTALANDO A LINGUAGEM PHP SCRIPTING

Esta seção descreve como instalar uma versão selecionada do módulo **php**.

Procedimento

- Para instalar um fluxo de módulos **php** com o perfil padrão, use:

```
#módulo yum instalar php stream
```

  Substitua *stream* pela versão do PHP que você deseja instalar.

  Por exemplo, para instalar o PHP 7.4:

```
#módulo yum instalar php:7.4
```

  O perfil padrão **common** instala também o pacote **php-fpm**, e pré-configura o PHP para uso com o **Apache HTTP Server** ou **nginx**.

- Para instalar um perfil específico de um fluxo de módulos **php**, use:

```
#módulo yum instalar php stream/profile
```

  Substitua *stream* pela versão desejada e *profile* pelo nome do perfil que você deseja instalar.

  Por exemplo, para instalar o PHP 7.4 para uso sem um servidor web:

```
#módulo yum instalar php:7.4/minimal
```

Recursos adicionais

- Se você quiser atualizar de uma versão anterior do PHP disponível no RHEL 8, veja Mudando para uma versão posterior.

- Para mais informações sobre o gerenciamento dos módulos e fluxos RHEL 8, consulte Instalação, gerenciamento e remoção de componentes de espaço do usuário .

16.2. USANDO A LINGUAGEM PHP SCRIPTING COM UM SERVIDOR WEB

16.2.1. Usando PHP com o Servidor HTTP Apache

No RHEL 8, o **Apache HTTP Server** permite executar PHP como um servidor de processo FastCGI. FastCGI Process Manager (FPM) é um daemon PHP alternativo FastCGI que permite a um website gerenciar altas cargas. O PHP usa FastCGI Process Manager por padrão no RHEL 8.

Esta seção descreve como executar o código PHP usando o servidor de processo FastCGI.

Pré-requisitos

- A linguagem PHP scripting está instalada em seu sistema.
  Ver Seção 16.1, "Instalando a linguagem PHP scripting" .

Procedimento

1. Instale o módulo **httpd**:

```
# yum instalar módulo httpd:2.4
```

2. Inicie o **Apache HTTP Server**:

```
# systemctl start httpd
```

   Ou, se o **Apache HTTP Server** já estiver rodando em seu sistema, reinicie o serviço **httpd** após a instalação do PHP:

```
# systemctl restart httpd
```

3. Iniciar o serviço **php-fpm**:

```
# systemctl start php-fpm
```

4. Opcional: Permitir que ambos os serviços comecem no momento da inicialização:

```
# systemctl habilita o php-fpm httpd
```

5. Para obter informações sobre suas configurações de PHP, crie o arquivo **index.php** com o seguinte conteúdo no diretório **/var/www/html/**:

```
echo '<?php phpinfo(); ?>' > /var/wwww/html/index.php
```

6. Para executar o arquivo **index.php**, aponte o navegador para:

```
http://<hostname>/
```

7. Opcional: Ajuste a configuração se você tiver requisitos específicos:
   - **/etc/httpd/conf/httpd.conf** - genérico **httpd** configuração
   - **/etc/httpd/conf.d/php.conf** - Configuração específica para PHP para **httpd**
   - **/usr/lib/systemd/system/httpd.service.d/php-fpm.conf** - por padrão, o serviço **php-fpm** é iniciado com **httpd**
   - **/etc/php-fpm.conf** - Configuração principal do FPM
   - **/etc/php-fpm.d/www.conf** - padrão **www** configuração do pool

Exemplo 16.1. Executando um "Olá, Mundo!" Script PHP usando o Servidor HTTP Apache

1. Crie um diretório **hello** para seu projeto no diretório **/var/www/html/**:

```
# mkdir olá
```

2. Crie um arquivo **hello.php** no diretório **/var/www/html/hello/** com o seguinte conteúdo:

```
# <!DOCTYPE html>
<html>
<head>
<title>Hello, World! Page</title>
</head>
<body>
<?php
    echo 'Hello, World!';
?>
</body>
</html>
```

3. Inicie o **Apache HTTP Server**:

```
# systemctl start httpd
```

4. Para executar o arquivo **hello.php**, aponte o navegador para:

```
http://<hostname>/hello/hello.php
```

   Como resultado, é exibida uma página web com o texto "Olá, Mundo!

Recursos adicionais

- Configurando o servidor web Apache HTTP

16.2.2. Usando PHP com o servidor web nginx

Esta seção descreve como executar o código PHP através do servidor web **nginx**.

Pré-requisitos

- A linguagem PHP scripting está instalada em seu sistema.
  Ver Seção 16.1, “Instalando a linguagem PHP scripting” .

Procedimento

1. Instale um fluxo de módulos **nginx**:

```
# yum instalar módulo nginx stream
```

   Substitua *stream* pela versão de **nginx** que você deseja instalar.

   Por exemplo, para instalar **nginx** versão 1.18:

```
# yum instalar módulo nginx:1.18
```

2. Inicie o servidor **nginx**:

```
# systemctl start nginx
```

   Ou, se o servidor **nginx** já estiver rodando em seu sistema, reinicie o serviço **nginx** após a instalação do PHP:

```
# systemctl restart nginx
```

3. Iniciar o serviço **php-fpm**:

```
# systemctl start php-fpm
```

4. Opcional: Permitir que ambos os serviços comecem no momento da inicialização:

```
# systemctl habilita php-fpm nginx
```

5. Para obter informações sobre suas configurações de PHP, crie o arquivo **index.php** com o seguinte conteúdo no diretório **/usr/share/nginx/html/**:

```
echo '<?php phpinfo(); ?>' > /usr/share/nginx/html/index.php
```

6. Para executar o arquivo **index.php**, aponte o navegador para:

```
http://<hostname>/
```

7. Opcional: Ajuste a configuração se você tiver requisitos específicos:
   - **/etc/nginx/nginx.conf** - **nginx** configuração principal
   - **/etc/nginx/conf.d/php-fpm.conf** - Configuração FPM para **nginx**
   - **/etc/php-fpm.conf** - Configuração principal do FPM
   - **/etc/php-fpm.d/www.conf** - padrão **www** configuração do pool

Exemplo 16.2. Executando um "Olá, Mundo!" Script PHP usando o servidor nginx

1. Crie um diretório **hello** para seu projeto no diretório **/usr/share/nginx/html/**:

```
# mkdir olá
```

2. Crie um arquivo **hello.php** no diretório **/usr/share/nginx/html/hello/** com o seguinte conteúdo:

```
# <!DOCTYPE html>
<html>
<head>
<title>Hello, World! Page</title>
</head>
<body>
<?php
    echo 'Hello, World!';
?>
</body>
</html>
```

3. Inicie o servidor **nginx**:

```
# systemctl start nginx
```

4. Para executar o arquivo **hello.php**, aponte o navegador para:

```
http://<hostname>/hello/hello.php
```

Como resultado, é exibida uma página web com o texto "Olá, Mundo!

16.3. EXECUTANDO UM SCRIPT PHP USANDO A INTERFACE DE LINHA DE COMANDO

Um script PHP é normalmente executado usando um servidor web, mas também pode ser executado usando a interface de linha de comando.

Se você quiser executar scripts **php** usando apenas a linha de comando, instale o perfil **minimal** de um fluxo de módulos **php**.

Veja Seção 16.1, “Instalando a linguagem PHP scripting” para detalhes.

Pré-requisitos

- A linguagem PHP scripting está instalada em seu sistema.
  Ver Seção 16.1, “Instalando a linguagem PHP scripting” .

Procedimento

1. Em um editor de texto, crie um ***filename*.php** arquivo
   Substitua *filename* por um nome de seu arquivo.

2. Executar o criado ***filename*.php** arquivo a partir da linha de comando:

```
# php filename.php
```

Exemplo 16.3. Executando um "Olá, Mundo!" Script PHP usando a interface de linha de comando

1. Crie um arquivo **hello.php** com o seguinte conteúdo usando um editor de texto:

```
<?php
    echo 'Hello, World!';
?>
```

2. Execute o arquivo **hello.php** a partir da linha de comando:

```
# php olá.php
```

Como resultado, "Olá, Mundo!" é impresso.

16.4. RECURSOS ADICIONAIS

- **httpd(8)** - A página do manual do serviço **httpd** contendo a lista completa de suas opções de linha de comando.
- **httpd.conf(5)** - A página de manual de configuração **httpd**, descrevendo a estrutura e localização dos arquivos de configuração **httpd**.
- **nginx(8)** - A página de manual para o servidor web **nginx** contendo a lista completa de suas opções de linha de comando e lista de sinais.
- **php-fpm(8)** - A página de manual do PHP FPM descrevendo a lista completa de suas opções de linha de comando e arquivos de configuração.

CAPÍTULO 17. USANDO LANCHEIRAS

Langpacks são meta-pacotes que instalam pacotes adicionais contendo traduções, dicionários e locales para cada pacote instalado no sistema.

Em um sistema Red Hat Enterprise Linux 8, **langpacks** a instalação é baseada nos meta-pacotes de linguagem **langpacks-<langcode>** e nas dependências fracas de RPM (tag Suplements).

Há dois pré-requisitos para poder usar **langpacks** para um idioma selecionado. Se estes pré-requisitos forem cumpridos, os meta-pacotes de idiomas puxam automaticamente os seus pacotes para o idioma selecionado no conjunto de transações.

Pré-requisitos

- O meta-pacote do idioma **langpacks-<langcode>** para o idioma selecionado foi instalado no sistema.
  No Red Hat Enterprise Linux 8, os meta pacotes langpacks são instalados automaticamente com a instalação inicial do sistema operacional usando o instalador Anaconda, porque estes pacotes estão disponíveis no repositório Application Stream.

  Para mais informações, veja Seção 17.1, “Verificação de idiomas que fornecem lancheiras”

- O pacote básico, para o qual você deseja pesquisar os pacotes locais, já foi instalado no sistema.

17.1. VERIFICAÇÃO DE IDIOMAS QUE FORNECEM LANCHEIRAS

Folheie este procedimento para verificar quais os idiomas que fornecem as lancheiras.

Procedimento

- Execute o seguinte comando:

```
*langpacks da lista yum-*
```

17.2. TRABALHANDO COM LANCHES BASEADOS NA DEPENDÊNCIA FRACA DE RPM

Esta seção descreve várias ações que você pode querer realizar ao consultar os pacotes de lanche baseados em dependência fraca RPM, instalando ou removendo o suporte de linguagem.

17.2.1. Listagem do suporte linguístico já instalado

Para listar o suporte linguístico já instalado, use este procedimento.

Procedimento

- Execute o seguinte comando:

```
# lista yum instalada langpacks*
```

17.2.2. Verificação da disponibilidade de suporte de idiomas

Para verificar se o suporte linguístico está disponível para qualquer idioma, use o procedimento a seguir.

Procedimento

- Execute o seguinte comando:

```
# lista yum disponível langpacks*
```

17.2.3. Listagem de pacotes instalados para um idioma

Para listar quais pacotes são instalados para qualquer idioma, use o seguinte procedimento:

Procedimento

- Execute o seguinte comando:

```
# yum repoquery --agrupamentos de mochilas de lanche<locale_code>
```

17.2.4. Instalação de suporte linguístico

Para acrescentar um novo suporte linguístico, use o seguinte procedimento.

Procedimento

- Execute o seguinte comando:

```
# yum install langpacks-<locale_code>
```

17.2.5. Removendo o suporte linguístico

Para remover qualquer suporte de idioma instalado, use o seguinte procedimento.

Procedimento

- Execute o seguinte comando:

```
# yum remove langpacks-<locale_code>
```

17.3. ECONOMIZANDO ESPAÇO EM DISCO USANDO GLIBC-LANGPACK-<LOCALE_CODE>

Atualmente, todos os locais são armazenados no arquivo **/usr/lib/locale/locale-archive**, o que requer muito espaço em disco.

Em sistemas onde o espaço em disco é um problema crítico, tais como containers e imagens em nuvem, ou apenas alguns locais são necessários, pode-se usar os pacotes glibc locale langpack (**glibc-langpack-<locale_code>**).

Para instalar os locais individualmente, e assim ganhar uma menor área de instalação do pacote, use o seguinte procedimento

Procedimento

Procedimento

- Execute o seguinte comando:

```
# yum install glibc-langpack-<locale_code>
```

Ao instalar o sistema operacional com Anaconda, **glibc-langpack-<locale_code>** é instalado para o idioma utilizado durante a instalação e também para os idiomas que você selecionou como idiomas adicionais. Note que **glibc-all-langpacks**, que contém todos os locales, é instalado por padrão, portanto, alguns locales são duplicados. Se você instalou **glibc-langpack-<locale_code>** para um ou mais idiomas selecionados, você pode excluir **glibc-all-langpacks** após a instalação para economizar o espaço em disco.

Observe que a instalação de apenas pacotes selecionados **glibc-langpack-<locale_code>** ao invés de **glibc-all-langpacks** tem impacto no desempenho em tempo de execução.

NOTA

Se o espaço em disco não for um problema, mantenha todos os locais instalados, utilizando o pacote **glibc-all-langpacks**.

CAPÍTULO 18. COMEÇANDO COM TCL/TK

18.1. INTRODUÇÃO À TCL/TK

Tool command language (Tcl) é uma linguagem de programação dinâmica. O intérprete para esta linguagem, juntamente com a biblioteca C, é fornecido pelo pacote **tcl**.

Usando **Tcl** emparelhado com **Tk** (**Tcl/Tk**) permite a criação de aplicações GUI em várias plataformas **Tk** é fornecido pelo pacote **tk**.

Note que **Tk** pode se referir a qualquer um dos seguintes itens:

- Um kit de ferramentas de programação para várias linguagens
- Uma biblioteca Tk C disponível para várias línguas, tais como C, Ruby, Perl e Python
- Um intérprete de desejos que instancia um console Tk
- Uma extensão Tk que acrescenta uma série de novos comandos a um determinado intérprete Tcl

Para mais informações sobre Tcl/Tk, consulte o manual Tcl/Tk ou a página web de documentação Tcl/Tk.

18.2. MUDANÇAS NOTÁVEIS NO TCL/TK 8.6

Red Hat Enterprise Linux 7 utilizado **Tcl/Tk 8.5**. Com o Red Hat Enterprise Linux 8, **Tcl/Tk version 8.6** é fornecido no repositório do sistema operacional básico.

Grandes mudanças em **Tcl/Tk 8.6** em comparação com **Tcl/Tk 8.5** são:

- Apoio à programação orientada a objetos
- Implementação de avaliação sem empilhamento
- Tratamento melhorado de exceções
- Coleta de pacotes de terceiros construídos e instalados com Tcl
- Operações com múltiplas roscas habilitadas
- Suporte a scripts com base em banco de dados SQL
- Suporte a redes IPv6
- Compressão Zlib embutida
- Processamento de listas
  Dois novos comandos, **lmap** e **dict map** estão disponíveis, que permitem a expressão de transformações sobre **Tcl** contêineres.
- Canais empilhados por roteiro
  Dois novos comandos, **chan push** e **chan pop** estão disponíveis, que permitem adicionar ou remover transformações de ou para os canais de E/S.

Grandes mudanças em **Tk** incluir:

- Suporte de imagem PNG embutido
- Janelas ocupadas
  Um novo comando, **tk busy**, está disponível, que desativa a interação do usuário para uma janela ou um widget e mostra o cursor ocupado.
- Nova interface de diálogo de seleção de fontes
- Suporte de texto em ângulo
- Movendo as coisas em um suporte de lona

Para a lista detalhada das mudanças entre **Tcl 8.5** e **Tcl 8.6**ver Mudanças no Tcl/Tk 8.6.

18.3. MIGRANDO PARA TCL/TK 8.6

Red Hat Enterprise Linux 7 utilizado **Tcl/Tk 8.5**. Com o Red Hat Enterprise Linux 8, **Tcl/Tk version 8.6** é fornecido no repositório do sistema operacional básico.

Esta seção descreve o caminho de migração para **Tcl/Tk 8.6** para:

- Redação dos desenvolvedores **Tcl** extensões ou embutimento **Tcl** intérprete em suas aplicações
- Tarefas de roteirização dos usuários com **Tcl/Tk**

18.3.1. Caminho de migração para desenvolvedores de extensões Tcl

Para tornar seu código compatível com **Tcl 8.6**Use o procedimento a seguir.

Procedimento

1. Reescreva o código para usar a estrutura **interp**. Por exemplo, se seu código ler **interp→errorLine**, reescreva-o para usar a seguinte função:

   ```
   Tcl_GetErrorLine(interp)
   ```

   Isto é necessário porque **Tcl 8.6** limita o acesso direto aos membros da estrutura **interp**.

2. Para tornar seu código compatível com ambos **Tcl 8.5** e **Tcl 8.6**Use o seguinte trecho de código em um arquivo de cabeçalho de sua aplicação ou extensão C ou C que inclua o **Tcl** biblioteca:

   ```
   # include <tcl.h>
   # if !defined(Tcl_GetErrorLine)
   # define Tcl_GetErrorLine(interp) (interp→errorLine)
   # endif
   ```

18.3.2. Caminho de migração para os usuários que escrevem suas tarefas com Tcl/Tk

Em **Tcl 8.6**a maioria dos scripts funciona da mesma forma que na versão anterior de **Tcl**.

Para migrar seu código para **Tcl 8.6**Use este procedimento.

Procedimento

- Ao escrever um código portátil, certifique-se de não usar os comandos que não são mais suportados em **Tk 8.6**:

```
tkIconList_Arrange
tkIconList_AutoScan
tkIconList_Btn1
tkIconList_Config
tkIconList_Create
tkIconList_CtrlBtn1
tkIconList_Curselection
tkIconList_DeleteAll
tkIconList_Double1
tkIconList_DrawSelection
tkIconList_FocusIn
tkIconList_FocusOut
tkIconList_Get
tkIconList_Goto
tkIconList_Index
tkIconList_Invoke
tkIconList_KeyPress
tkIconList_Leave1
tkIconList_LeftRight
tkIconList_Motion1
tkIconList_Reset
tkIconList_ReturnKey
tkIconList_See
tkIconList_Select
tkIconList_Selection
tkIconList_ShiftBtn1
tkIconList_UpDown
```

Observe que você pode verificar a lista de comandos não suportados também no arquivo **/usr/share/tk8.6/unsupported.tcl**.