Red Hat Enterprise Linux 8

8.0 Notas de Lançamento

Notas de lançamento do Red Hat Enterprise Linux 8.0

Last Updated: 2021-02-23

Red Hat Enterprise Linux 8 8.0 Notas de Lançamento

Notas de lançamento do Red Hat Enterprise Linux 8.0

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

As notas de lançamento fornecem uma cobertura de alto nível das melhorias e adições que foram implementadas no Red Hat Enterprise Linux 8.0 e documentam problemas conhecidos neste lançamento, bem como correções de bugs notáveis, previsões tecnológicas, funcionalidades obsoletas e outros detalhes.

Índice

2021-02-22Red Hat

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, certifique-se de estar visualizando a documentação no formato Multi-página HTML. Destaque a parte do texto que você deseja comentar. Em seguida, clique no pop-up **Add Feedback** que aparece abaixo do texto destacado, e siga as instruções exibidas.
- 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. VISÃO GERAL

Baseado no Fedora 28 e no kernel 4.18 upstream, o Red Hat Enterprise Linux 8.0 oferece aos usuários uma base estável, segura e consistente através de implementações de nuvens híbridas com as ferramentas necessárias para suportar cargas de trabalho tradicionais e emergentes. Os destaques do lançamento incluem:

Distribuição

- O conteúdo está disponível através dos repositórios **BaseOS** e Application Stream (**AppStream**).
- O repositório **AppStream** suporta uma nova extensão do formato tradicional RPM - *modules*. Isto permite que várias versões principais de um componente estejam disponíveis para instalação.

Veja Capítulo 3, *Distribuição do conteúdo no RHEL 8* para mais informações.

Gerenciamento de software

- O gerenciador de pacotes **YUM** é agora baseado na tecnologia **DNF** e fornece suporte para conteúdo modular, maior desempenho e uma API estável e bem projetada para integração com ferramentas.

Veja Seção 5.1.4, "Gestão de software" para mais detalhes.

Conchas e ferramentas de linha de comando

- RHEL 8 fornece o seguinte **version control systems**: **Git 2.18**, **Mercurial 4.8**, e **Subversion 1.10**.

Veja ??? para detalhes.

Linguagens de programação dinâmica, servidores web e de banco de dados

- **Python 3.6** é a implementação padrão **Python** no RHEL 8; suporte limitado para **Python 2.7** é fornecido. Nenhuma versão do Python é instalada por padrão.
- O**Node.js** é novo na RHEL. Outros **dynamic programming languages** foram atualizados desde a RHEL 7: **PHP 7.2**, Ruby **2.5**, **Perl 5.26**, **SWIG 3.0** estão agora disponíveis.
- O seguinte **database servers** é distribuído com a RHEL 8: **MariaDB 10.3**, **MySQL 8.0**, **PostgreSQL 10**, **PostgreSQL 9.6**, e **Redis 5**.
- A RHEL 8 fornece o **Servidor HTTP Apache 2.4** e introduz um novo **web server**, **nginx 1.14**.
- **O Squid** foi atualizado para a versão 4.4, e um novo **proxy caching server** está agora incluído: **Cache de Verniz 6.0**.

Veja Seção 5.1.7, "Linguagens de programação dinâmica, servidores web e de banco de dados" para mais informações.

Desktop

- **GNOME Shell** foi rebaseado para a versão 3.28.

- A sessão GNOME e o Gerenciador de monitores GNOME usam **Wayland** como seu servidor de exibição padrão. O servidor **X.Org**, que é o servidor de exibição padrão no RHEL 7, também está disponível.

Veja Seção 5.1.8, "Desktop" para mais informações.

Instalador e criação de imagem

- O instalador **Anaconda** pode utilizar a criptografia de disco **LUKS2**, e instalar o sistema em **NVDIMM** dispositivos.
- A ferramenta **Image Builder** permite aos usuários criar imagens personalizadas do sistema em uma variedade de formatos, incluindo imagens preparadas para implantação em nuvens de vários fornecedores.
- Instalação a partir de um DVD usando o Console de Gerenciamento de Hardware (**HMC**) e Elemento de Suporte (**SE**) em **IBM Z** estão disponíveis em RHEL 8.

Veja Seção 5.1.2, "Instalador e criação de imagem" para mais detalhes.

Kernel

- A filtragem de pacotes ampliada de Berkeley (**eBPF)** permite ao espaço do usuário anexar programas personalizados em uma variedade de pontos (soquetes, pontos de rastreamento, recepção de pacotes) para receber e processar dados. Este recurso está disponível como um **Technology Preview**.
- BPF Compiler Collection (**BCC**), uma ferramenta para criar programas eficientes de rastreamento e manipulação de kernel, está disponível como um **Technology Preview**.

Veja Seção 5.3.1, "Kernel" para mais informações.

Sistemas de arquivo e armazenamento

- O formato LUKS versão 2 (**LUKS2**) substitui o formato antigo LUKS (LUKS1). O subsistema **dm-crypt** e a ferramenta **cryptsetup** agora usa o LUKS2 como o formato padrão para volumes criptografados.

Veja Seção 5.1.12, "Sistemas de arquivo e armazenamento" para mais informações.

Segurança

- O sistema **cryptographic policies**, que configura os subsistemas criptográficos centrais, cobrindo os protocolos TLS, IPsec, SSH, DNSSEC e Kerberos, são aplicados por padrão. Com o novo comando **update-crypto-policies**, o administrador pode mudar facilmente entre os modos: padrão, legado, futuro e fips.
- O suporte para **smart cards** e Módulos de Segurança de Hardware (**HSM**) com **PKCS #11** é agora consistente em todo o sistema.

Veja Seção 5.1.15, "Segurança" para mais informações.

Trabalho em rede

- A estrutura **nftables** substitui o **iptables** no papel do recurso padrão de filtragem de pacotes de rede.
- O daemon **firewalld** agora usa o **nftables** como seu backend padrão.

- Foi introduzido o suporte para os drivers de rede virtual **IPVLAN** que permitem a conectividade de rede para vários contêineres.

- O eXpress Data Path (**XDP**), XDP para Controle de Tráfego (**tc**), e Address Family eXpress Data Path (**AF_XDP**), como partes do pacote estendido de Filtragem de Pacotes de Berkeley (**eBPF)** recurso, estão disponíveis como **Technology Previews**. Para mais detalhes, veja Seção 5.3.7, "Trabalho em rede".

Veja Seção 5.1.14, "Trabalho em rede" para recursos adicionais.

Virtualização

- Um tipo de máquina mais moderno baseado em PCI Express (**Q35**) é agora suportado e configurado automaticamente em máquinas virtuais criadas no RHEL 8. Isto fornece uma variedade de melhorias nas características e compatibilidade de dispositivos virtuais.

- Máquinas virtuais podem agora ser criadas e gerenciadas usando o console web RHEL 8, também conhecido como **Cockpit**.

- O emulador **QEMU** introduz o recurso **sandboxing**, que fornece limitações configuráveis ao que os sistemas chamados QEMU podem realizar, e assim torna as máquinas virtuais mais seguras.

Veja Seção 5.1.16, "Virtualização" para mais informações.

Compiladores e ferramentas de desenvolvimento

- O compilador **GCC** baseado na versão 8.2 traz suporte para versões mais recentes em linguagem C padrão, melhores otimizações, novas técnicas de endurecimento de código, avisos melhorados e novas características de hardware.

- Várias ferramentas para geração, manipulação e depuração de códigos podem agora tratar experimentalmente o formato **DWARF5** de informações de depuração.

- O suporte do Kernel para rastreamento **eBPF** está disponível para algumas ferramentas, tais como **BCC**, **PCP**, e **SystemTap**.

- As bibliotecas **glibc** baseadas na versão 2.28 adicionam suporte ao Unicode 11, novas chamadas ao sistema Linux, principais melhorias no resolvedor de stub DNS, endurecimento adicional da segurança e melhor desempenho.

- A RHEL 8 fornece OpenJDK 11, OpenJDK 8, IcedTea-Web, e várias ferramentas **Java**, tais como **Ant**, **Maven**, ou **Scala**.

Consulte Seção 5.1.11, "Compiladores e ferramentas de desenvolvimento" para obter mais detalhes.

Alta disponibilidade e clusters

- O gerente de recursos do cluster **Pacemaker** foi atualizado para a versão 2.0.0, que fornece uma série de correções e melhorias de bugs.

- No RHEL 8, o sistema de configuração **pcs** suporta totalmente Corosync 3, **knet**, e nomes de nós.

Veja Seção 5.1.13, "Alta disponibilidade e clusters" para mais informações.

Recursos adicionais

- **Capabilities and limits** do Red Hat Enterprise Linux 8 em comparação com outras versões do sistema estão disponíveis no artigo da Base de Conhecimento Capacidades e limites da tecnologia Red Hat Enterprise Linux.
- Informações sobre o Red Hat Enterprise Linux **life cycle** são fornecidas no documento Red Hat Enterprise Linux Life Cycle (Ciclo de Vida do Red Hat Enterprise Linux).
- O documento de manifesto do Pacote fornece um **package listing** para a RHEL 8.
- A maior parte **differences between RHEL 7 and RHEL 8** está documentada em Considerações sobre a adoção do RHEL 8.
- As instruções sobre como realizar um **in-place upgrade from RHEL 7 to RHEL 8** são fornecidas pelo documento Upgrading from RHEL 7 to RHEL 8 .
- Os caminhos de atualização com suporte irresistível estão listados em Caminhos de atualização suportados no local para o Red Hat Enterprise Linux.
- O serviço **Red Hat Insights**, que lhe permite identificar, examinar e resolver proativamente questões técnicas conhecidas, está agora disponível com todas as assinaturas RHEL. Para instruções sobre como instalar o cliente Red Hat Insights e registrar seu sistema no serviço, consulte a página Red Hat Insights Get Started .

Laboratórios do Portal do Cliente Red Hat

Red Hat Customer Portal Labs é um conjunto de ferramentas em uma seção do Portal do Cliente disponível em https://access.redhat.com/labs/. As aplicações nos laboratórios do Portal do Cliente da Red Hat podem ajudar a melhorar o desempenho, solucionar rapidamente problemas, identificar problemas de segurança e implementar e configurar rapidamente aplicações complexas. Algumas das aplicações mais populares são:

- Assistente de Registro
- Gerador de pontapé de saída
- Verificador do ciclo de vida do produto
- Certificados de Produto Red Hat
- Red Hat Satellite Upgrade Helper
- Checador de CVE Red Hat
- Ferramenta de configuração das opções JVM
- Ferramenta de Configuração do Balanceador de Carga
- Navegador de Código Red Hat
- Ajudante de configuração do Yum Repository

CAPÍTULO 2. ARQUITETURAS

O Red Hat Enterprise Linux 8.0 é distribuído com o kernel versão 4.18.0-80, que fornece suporte para as seguintes arquiteturas:

- Arquiteturas AMD e Intel de 64 bits
- A arquitetura ARM de 64 bits
- IBM Power Systems, Little Endian
- IBM Z

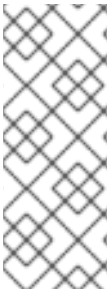
Certifique-se de adquirir a assinatura apropriada para cada arquitetura. Para mais informações, veja Get Started with Red Hat Enterprise Linux - arquiteturas adicionais. Para uma lista de assinaturas disponíveis, consulte Utilização de Assinaturas no Portal do Cliente.

CAPÍTULO 3. DISTRIBUIÇÃO DO CONTEÚDO NO RHEL 8

3.1. INSTALAÇÃO

O Red Hat Enterprise Linux 8 é instalado usando imagens ISO. Dois tipos de imagem ISO estão disponíveis para as arquiteturas AMD64, Intel 64-bit, 64-bit ARM, IBM Power Systems e IBM Z:

- DVD binário ISO: Uma imagem de instalação completa que contém os repositórios BaseOS e AppStream e permite que você complete a instalação sem repositórios adicionais.



NOTA

A imagem ISO do DVD Binário é maior que 4,7 GB, e como resultado, pode não caber em um DVD de uma única camada. Um DVD de camada dupla ou uma chave USB é recomendado quando se usa a imagem ISO do DVD Binário para criar uma mídia de instalação inicializável. Você também pode usar a ferramenta Image Builder para criar imagens RHEL personalizadas. Para mais informações sobre o Image Builder, consulte a *Composing a customized RHEL system image* documento.

- ISO de inicialização: Uma imagem ISO mínima de inicialização que é usada para iniciar no programa de instalação. Esta opção requer acesso aos repositórios BaseOS e AppStream para instalar os pacotes de software. Os repositórios são parte da imagem ISO do DVD Binário.

Consulte o documento Executando uma instalação padrão da RHEL para obter instruções sobre como baixar imagens ISO, criar mídia de instalação e concluir uma instalação RHEL. Para instalações Kickstart automatizadas e outros tópicos avançados, veja o documento Executando uma instalação RHEL avançada.

3.2. REPOSITÓRIOS

O Red Hat Enterprise Linux 8 é distribuído através de dois repositórios principais:

- BaseOS
- AppStream

Ambos os repositórios são necessários para uma instalação básica da RHEL, e estão disponíveis com todas as assinaturas RHEL.

O conteúdo no repositório BaseOS destina-se a fornecer o conjunto central da funcionalidade do SO subjacente que fornece a base para todas as instalações. Este conteúdo está disponível no formato RPM e está sujeito a termos de suporte semelhantes aos de versões anteriores da RHEL. Para uma lista de pacotes distribuídos através do BaseOS, veja o manifesto de pacotes.

O conteúdo no repositório Application Stream inclui aplicações adicionais de espaço do usuário, linguagens de tempo de execução e bancos de dados em apoio às diversas cargas de trabalho e casos de uso. O conteúdo no AppStream está disponível em um dos dois formatos - o familiar formato RPM e uma extensão para o formato RPM chamada *modules*. Para uma lista de pacotes disponíveis no AppStream, veja o manifesto de pacotes.

Além disso, o repositório CodeReady Linux Builder está disponível com todas as assinaturas RHEL. Ele fornece pacotes adicionais para uso dos desenvolvedores. Os pacotes incluídos no repositório do CodeReady Linux Builder não são suportados.

Para mais informações sobre os repositórios RHEL 8, consulte o manifesto do pacote.

3.3. FLUXOS DE APLICAÇÃO

O Red Hat Enterprise Linux 8.0 introduz o conceito de Application Streams. 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 RHEL 8. Cada componente do Application Stream tem um determinado ciclo de vida, veja o ciclo de vida do Red Hat Enterprise Linux 8 Application Streams.

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.

CAPÍTULO 4. LANÇAMENTO RHEL 8.0.1

4.1. NOVAS CARACTERÍSTICAS

Funções do Sistema RHEL atualizadas

Os pacotes **rhel-system-roles**, que fornecem uma interface de configuração para os subsistemas RHEL, foram atualizados. Mudanças notáveis incluem:

- O tratamento dos perfis ausentes no papel da **rede** foi melhorado. Ao excluir uma configuração existente do perfil do NetworkManager no disco, definindo o estado persistente como **ausente**, apenas a configuração persistente para o perfil é agora removida, e a configuração atual do tempo de execução permanece inalterada. Como resultado, o dispositivo de rede correspondente não é mais derrubado na situação descrita.

- A especificação de um tamanho máximo de unidade de transmissão (MTU) para as interfaces VLAN e MACVLAN na função de **rede** foi fixada. Como resultado, a definição do tamanho MTU nas interfaces VLAN e MACVLAN usando a função de **rede** não mais falha com a seguinte mensagem de erro:

  ```
  failure: created connection failed to normalize: nm-connection-error-quark:
  connection.type: property is missing (6)
  ```

- As funções **selinux** e **timesync** agora incluem todas as variáveis de entrada documentadas em seus arquivos padrão**(defaults/main.yml**). Isto facilita determinar quais variáveis de entrada são suportadas pelas funções, examinando o conteúdo de seus respectivos arquivos padrão.

- Os papéis **kdump** e **timesync** foram corrigidos para não falharem no modo de verificação.

(BZ#1685902, BZ#1674004, BZ#1685904)

sos-collector rebaseado para a versão 1.7

Os pacotes **sos-collector** foram atualizados para a versão 1.7 no RHEL 8.0.1. Mudanças notáveis incluem:

- **sos-collector** pode agora coletar sosreports dos nós do Red Hat Enterprise Linux CoreOS (RHCOS) da mesma forma que dos nós regulares da RHEL. Os usuários não precisam fazer nenhuma mudança na maneira como eles executam **o sos-collector**. A identificação de quando um nó é RHCOS ou RHEL é automática.

- Ao coletar dos nós RHCOS, **o sos-collector** criará um recipiente temporário no nó e utilizará o recipiente **das ferramentas de apoio** para gerar um sosreport. Este contêiner será removido após a conclusão.

- Usando a opção **--cluster-type=nenhuma** opção permite aos usuários pular todas as verificações ou modificações relacionadas ao comando **sosreport** que é executado nos nós, e simplesmente coletar de uma lista estática de nós passados através do parâmetro **--nodes**.

- O Red Hat Satellite é agora um tipo de cluster suportado para permitir a coleta de sosreports do Satellite e quaisquer Capsules.

(BZ#1695764)

Conjuntos de ferramentas de compilação melhorados

Os seguintes conjuntos de ferramentas de compilação, distribuídos como Application Streams, foram atualizados com o RHEL 8.0.1:

- Rust Toolset, que fornece o compilador de linguagem de programação Rust **rustc**, a ferramenta de construção de **carga** e o gerenciador de dependência, e as bibliotecas necessárias, para a versão 1.35
- Go Toolset, que fornece as ferramentas da linguagem de programação Go**(golang**) e bibliotecas, para a versão 1.11.6.

(BZ#1731500)

Habilitação e desativação de SMT

A configuração simultânea Multi-Threading (SMT) está agora disponível no RHEL 8. A desativação do SMT no console web permite mitigar uma classe de vulnerabilidades de segurança da CPU, como por exemplo:

- Amostragem de dados microarquitetônicos
- L1 Ataque de falha do terminal

(BZ#1713186)

4.2. PROBLEMAS CONHECIDOS

Deterioração do desempenho nos túneis IPSec

O uso do **aes256_sha2** ou da cifra IPSec **aes-gcm256** definida no RHEL 8.0.1 tem um impacto negativo no desempenho dos túneis IPSec. Os usuários com configurações VPN específicas experimentarão uma deterioração de 10% no desempenho dos túneis IPSec. Esta regressão não é causada pelas mitigações da Microarquitetura de Amostragem de Dados (MDS); ela pode ser observada com as mitigações ativadas e desativadas.

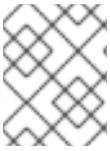
(BZ#1731362)

CAPÍTULO 5. LANÇAMENTO RHEL 8.0.0

5.1. NOVAS CARACTERÍSTICAS

Esta parte descreve os novos recursos e as principais melhorias introduzidas no Red Hat Enterprise Linux 8.

5.1.1. O console web



NOTA

A página de Assinaturas do console web é agora fornecida pelo novo pacote de **gerenciador de assinaturas-cockpit**.

Uma interface de firewall foi adicionada ao console web

A página **Networking** no console web RHEL 8 agora inclui uma seção **Firewall**. Nesta seção, os usuários podem ativar ou desativar o firewall, assim como adicionar, remover e modificar as regras do firewall.

(BZ#1647110)

O console web agora está disponível por padrão

Os pacotes para o console web RHEL 8, também conhecido como Cockpit, agora fazem parte dos repositórios default do Red Hat Enterprise Linux, e podem, portanto, ser imediatamente instalados em um sistema RHEL 8 registrado.

Além disso, em uma instalação não mínima do RHEL 8, o console web é instalado automaticamente e as portas de firewall exigidas pelo console são automaticamente abertas. Uma mensagem do sistema também foi adicionada antes do login que fornece informações sobre como habilitar ou acessar o console web.

(JIRA:RHELPLAN-10355)

Melhor integração IdM para o console web

Se seu sistema estiver registrado em um domínio de Gerenciamento de Identidade (IdM), o console web RHEL 8 agora utiliza os recursos de gerenciamento central de IdM do domínio por padrão. Isto inclui os seguintes benefícios:

- Os administradores do domínio IdM podem usar o console web para gerenciar a máquina local.
- O servidor web do console muda automaticamente para um certificado emitido pela autoridade certificadora da IdM (CA) e aceito pelos navegadores.
- Os usuários com um bilhete Kerberos no domínio IdM não precisam fornecer credenciais de login para acessar o console web.
- Os hosts SSH conhecidos para o domínio IdM são acessíveis ao console web sem a necessidade de adicionar manualmente uma conexão SSH.

Note que para que a integração do IdM com o console web funcione corretamente, o usuário primeiro precisa executar o utilitário **ipa-advise** com a opção **enable-admins-sudo** no sistema mestre do IdM.

(JIRA:RHELPLAN-3010)

O console web agora é compatível com navegadores móveis

Com esta atualização, os menus e páginas do console web podem ser navegados em variantes de navegadores móveis. Isto torna possível gerenciar sistemas usando o console web RHEL 8 a partir de um dispositivo móvel.

(JIRA:RHELPLAN-10352)

A página inicial do console web agora exibe atualizações e assinaturas em falta

Se um sistema gerenciado pelo console web RHEL 8 tiver pacotes desatualizados ou uma assinatura caducada, um aviso é agora exibido na página principal do console web do sistema.

(JIRA:RHELPLAN-10353)

O console web agora suporta a inscrição de PBD

Com esta atualização, você pode usar a interface do console web RHEL 8 para aplicar as regras de Decriptação Baseada em Políticas (PBD) a discos em sistemas gerenciados. Isto utiliza o cliente de decriptação Clevis para facilitar uma variedade de funções de gerenciamento de segurança no console web, como o desbloqueio automático de partições de disco criptografadas pelo LUKS.

(JIRA:RHELPLAN-10354)

As máquinas virtuais podem agora ser gerenciadas usando o console web

A página de **Máquinas Virtuais** pode agora ser adicionada à interface do console web RHEL 8, que permite ao usuário criar e gerenciar máquinas virtuais baseadas em libvírus.

(JIRA:RHELPLAN-2896)

5.1.2. Instalador e criação de imagem

A instalação da RHEL a partir de um DVD usando SE e HMC é agora totalmente suportada no IBM Z

A instalação do Red Hat Enterprise Linux 8 no hardware IBM Z a partir de um DVD usando o **Support Element (SE)** e **Hardware Management Console (HMC)** é agora totalmente suportada. Esta adição simplifica o processo de instalação no IBM Z com **SE** e **HMC**.

Ao inicializar a partir de um DVD binário, o instalador solicita ao usuário que introduza parâmetros adicionais do kernel. Para definir o DVD como fonte de instalação, anexe **inst.repo=hmc** aos parâmetros do kernel. O instalador então habilita o acesso aos arquivos **SE** e **HMC**, vai buscar as imagens para o estágio2 do DVD e fornece acesso aos pacotes no DVD para seleção de software.

O novo recurso elimina a necessidade de uma configuração de rede externa e expande as opções de instalação.

(BZ#1500792)

O instalador agora suporta o formato de criptografia de disco LUKS2

O instalador do Red Hat Enterprise Linux 8 agora usa o formato LUKS2 por default, mas você pode selecionar uma versão LUKS a partir da janela **Anaconda's** Particionamento Personalizado ou usando as novas opções nos comandos Kickstart's **autopart**, **logvol**, **part** e **RAID**.

LUKS2 oferece muitas melhorias e características, por exemplo, amplia as capacidades do formato em disco e oferece formas flexíveis de armazenamento de metadados.

(BZ#1547908)

Anaconda suporta Sistema Propósito no RHEL 8

Anteriormente, **Anaconda** não fornecia informações sobre a finalidade do sistema para **Subscription Manager**. No Red Hat Enterprise Linux 8.0, você pode definir o propósito pretendido do sistema durante a instalação, usando a janela **Propósito do Sistema Anaconda's** ou o comando Kickstart's **syspurpose**. Quando a instalação estiver completa, **Subscription Manager** usa as informações sobre o propósito do sistema ao assinar o sistema.

(BZ#1612060)

Pykickstart suporta Sistema Objetivo no RHEL 8

Anteriormente, não era possível para a biblioteca **pykickstart** fornecer informações sobre a finalidade do sistema para **Subscription Manager**. No Red Hat Enterprise Linux 8.0, o **pykickstart** analisa o novo comando **syspurpose** e registra o propósito pretendido do sistema durante a instalação automatizada e parcialmente automatizada. As informações são então passadas para **Anaconda**, salvas no sistema recém-instalado, e disponíveis para **Subscription Manager** ao assinar o sistema.

(BZ#1612061)

Anaconda suporta um novo parâmetro de inicialização do kernel no RHEL 8

Anteriormente, você só podia especificar um repositório base a partir dos parâmetros de inicialização do kernel. No Red Hat Enterprise Linux 8, um novo parâmetro do kernel, **inst.addrepo=<name>,<url>**, permite que você especifique um repositório adicional durante a instalação.

Este parâmetro tem dois valores obrigatórios: o nome do repositório e a URL que aponta para o repositório. Para mais informações, veja https://anaconda-installer.readthedocs.io/en/latest/boot-options.html#inst-addrepo

(BZ#1595415)

Anaconda suporta uma ISO unificada no RHEL 8

No Red Hat Enterprise Linux 8.0, uma ISO unificada carrega automaticamente os repositórios de fontes de instalação do BaseOS e AppStream.

Este recurso funciona para o primeiro repositório base que é carregado durante a instalação. Por exemplo, se você iniciar a instalação sem nenhum repositório configurado e tiver a ISO unificada como o repositório base na GUI, ou se você iniciar a instalação usando a opção **inst.repo=** que aponta para a ISO unificada. Como resultado, o repositório AppStream é habilitado na seção **Additional Repositories** da janela da GUI **Installation Source**. Você não pode remover o repositório AppStream ou alterar suas configurações, mas você pode desativá-lo em **Installation Source**. Este recurso não funciona se você iniciar a instalação usando um repositório base diferente e depois alterá-lo para a ISO unificada. Se você fizer isso, o repositório base é substituído. Entretanto, o repositório AppStream não é substituído e aponta para o arquivo original.

(BZ#1610806)

O Anaconda pode instalar pacotes modulares em scripts Kickstart

O instalador Anaconda foi ampliado para lidar com todas as características relacionadas aos fluxos de aplicação: módulos, fluxos e perfis. Os Kickstart scripts podem agora habilitar combinações de módulos e fluxos, instalar perfis de módulos e instalar pacotes modulares. Para mais informações, consulte Execução de uma instalação RHEL avançada .

(JIRA:RHELPLAN-1943)

A opção de inicialização nosmt está agora disponível nas opções de instalação do RHEL 8

A opção de inicialização **nosmt** está disponível nas opções de instalação que são passadas para um sistema RHEL 8 recém-instalado.

(BZ#1677411)

O RHEL 8 suporta a instalação a partir de um repositório em um disco rígido local

Anteriormente, a instalação da RHEL a partir de um disco rígido exigia uma imagem ISO como fonte de instalação. Entretanto, a imagem ISO RHEL 8 pode ser muito grande para alguns sistemas de arquivos; por exemplo, o sistema de arquivos FAT32 não pode armazenar arquivos maiores que 4 GiB.

No RHEL 8, você pode permitir a instalação a partir de um repositório em um disco rígido local. Você só precisa especificar o diretório em vez da imagem ISO. Por exemplo: `inst.repo=hd:<device>:<path para o repositório >``

(BZ#1502323)

Criação de imagens personalizadas do sistema com o Image Builder está disponível no RHEL 8

A ferramenta Image Builder permite que os usuários criem imagens RHEL personalizadas. O Image Builder está disponível em AppStream no **lorax-composer** pacote.

Com o Image Builder, os usuários podem criar imagens personalizadas do sistema que incluem pacotes adicionais. A funcionalidade do Image Builder pode ser acessada através dele:

- uma interface gráfica do usuário no console web
- uma interface de linha de comando na ferramenta **composer-cli**.

Os formatos de saída do Image Builder incluem, entre outros:

- imagem de disco ISO ao vivo
- arquivo qcow2 para uso direto com uma máquina virtual ou OpenStack
- arquivo de imagem do sistema de arquivos
- imagens de nuvens para Azure, VMWare e AWS

Para saber mais sobre o Image Builder, veja o título da documentação Composição de uma imagem personalizada do sistema RHEL.

(JIRA:RHELPLAN-7291, BZ#1628645, BZ#1628646, BZ#1628647, BZ#1628648)

5.1.3. Kernel

Versão do Kernel em RHEL 8.0

O Red Hat Enterprise Linux 8.0 é distribuído com o kernel versão 4.18.0-80.

(BZ#1797671)

O endereçamento físico ARM de 52 bits já está disponível

Com esta atualização, está disponível suporte para endereçamento físico de 52 bits (PA) para a arquitetura ARM de 64 bits. Isto fornece um espaço de endereçamento maior do que o anterior PA de 48 bits.

(BZ#1643522)

O código IOMMU suporta tabelas de 5 níveis de página no RHEL 8

O código da unidade de gerenciamento de memória I/O (IOMMU) no kernel Linux foi atualizado para suportar tabelas de páginas de 5 níveis no Red Hat Enterprise Linux 8.

(BZ#1485546)

Suporte para paginação em 5 níveis

O novo tipo de tabela de páginas de software **P4d_t** foi adicionado ao kernel Linux para suportar paginação em 5 níveis no Red Hat Enterprise Linux 8.

(BZ#1485532)

O gerenciamento de memória suporta tabelas de 5 níveis de página

Com o Red Hat Enterprise Linux 7, o barramento de memória existente tinha 48/46 bits de capacidade de endereçamento de memória virtual/física, e o kernel Linux implementou 4 níveis de tabelas de páginas para gerenciar esses endereços virtuais para endereços físicos. A linha de endereçamento do barramento físico colocou a capacidade limite superior de memória física em 64 TB.

Estes limites foram estendidos para 57/52 bits de endereçamento de memória virtual/física com 128 PiB de espaço de endereçamento virtual e 4 PB de capacidade de memória física.

Com a faixa de endereços ampliada, o gerenciamento de memória no Red Hat Enterprise Linux 8 adiciona suporte para a implementação da tabela de 5 níveis de página, para poder lidar com a faixa de endereços ampliada.

(BZ#1485525)

kernel-signing-ca.cer é movido para **kernel-core** no RHEL 8

Em todas as versões do Red Hat Enterprise Linux 7, a chave pública do **kernel-signing-ca.cer** estava localizada no pacote **kernel-doc**. Entretanto, no Red Hat Enterprise Linux 8, o **kernel-signing-ca.cer** foi realocado para o pacote **kernel-core** para cada arquitetura.

(BZ#1638465)

O padrão de mitigação Spectre V2 mudou de IBRS para Retpolines

A mitigação padrão da vulnerabilidade do Spectre V2 (CVE-2017-5715) para sistemas com Processadores Intel Core da 6ª Geração e seus derivados próximos [1] mudou de Indirect Branch Restricted Speculation (IBRS) para Retpolines no Red Hat Enterprise Linux 8. A Red Hat implementou esta mudança como resultado das recomendações da Intel para se alinhar com os padrões usados na comunidade Linux e para restaurar o desempenho perdido. Entretanto, observe que o uso de Retpolines em alguns casos pode não mitigar totalmente o Spectre V2. O documento Retpoline da Intel [2] descreve quaisquer casos de exposição. Este documento também declara que o risco de um ataque é baixo.

Para casos de uso onde a mitigação completa do Spectre V2 é desejada, um usuário pode selecionar o IBRS através da linha de inicialização do kernel adicionando a bandeira **spectre_v2=ibrs**.

Se um ou mais módulos do kernel não foram construídos com o suporte Retpoline, o arquivo

/sys/devices/system/cpu/vulnerabilities/spectre_v2 indicará vulnerabilidade e o arquivo **/var/log/messages** identificará os módulos ofensivos. Veja Como determinar quais módulos são responsáveis pelo retorno de spectre_v2 "Vulnerável": Retpoline com módulo(s) inseguro(s)}"? para maiores informações.

1] "Processadores Intel Core da 6ª geração e seus derivados próximos" são o que o documento Retpolines da Intel se refere como "Skylake-generation".

[2] Retpolina: A Branch Target Injection Mitigation - White Paper

(BZ#1651806)

Software hospedeiro Intel® Omni-Path Architecture (OPA)

O software hospedeiro Intel Omni-Path Architecture (OPA) é totalmente suportado no Red Hat Enterprise Linux 8.

A Intel OPA fornece hardware Host Fabric Interface (HFI) com inicialização e configuração para transferências de dados de alto desempenho (alta largura de banda, alta taxa de mensagens, baixa latência) entre os nós de computação e de E/S em um ambiente agrupado.

Para instruções sobre a instalação da documentação da arquitetura Intel Omni-Path, consulte: https://www.intel.com/content/dam/support/us/en/documents/network-and-i-o/fabric-products/Intel_OP_Software_RHEL_8_RN_K51383.pdf

(BZ#1683712)

NUMA suporta mais nós no RHEL 8

Com esta atualização, a contagem de nós NUMA (Non-Uniform Memory Access) foi aumentada de 4 nós NUMA para 8 nós NUMA no Red Hat Enterprise Linux 8 em sistemas com a arquitetura ARM de 64 bits.

(BZ#1550498)

A IOMMU passthrough está agora habilitada por padrão no RHEL 8

A passagem da Unidade de Gerenciamento de Memória de Entrada/Saída (IOMMU) foi habilitada por padrão. Isto proporciona melhor desempenho para os sistemas AMD porque o remapeamento do Acesso Direto à Memória (DMA) está desativado para o host. Esta atualização traz consistência com os sistemas Intel onde o remapeamento do DMA também está desabilitado por padrão. Os usuários podem desabilitar tal comportamento (e habilitar o remapeamento do DMA) especificando os parâmetros **iommu.passthrough=off** ou **iommu=nopt** na linha de comando do kernel, incluindo o hypervisor.

(BZ#1658391)

O kernel RHEL8 agora suporta tabelas de 5 níveis de página

O núcleo do Red Hat Enterprise Linux agora suporta totalmente os futuros processadores Intel com até 5 níveis de tabelas de páginas. Isto permite que os processadores suportem até 4PB de memória física e 128PB de espaço de endereços virtuais. As aplicações que utilizam grandes quantidades de memória podem agora usar a maior quantidade possível de memória fornecida pelo sistema sem as restrições de tabelas de 4 níveis de páginas.

(BZ#1623590)

O kernel RHEL8 suporta IBRS aprimorado para futuras CPUs Intel

O kernel do Red Hat Enterprise Linux agora suporta o uso da capacidade aprimorada de Especulação

Restrita Indireta (IBRS) para mitigar a vulnerabilidade do Spectre V2. Quando ativado, o IBRS terá um desempenho melhor que o das Retpolines (default) para mitigar o Spectre V2 e não interferirá com a tecnologia Intel Control-flow Enforcement. Como resultado, a penalidade de desempenho de permitir a mitigação do Spectre V2 será menor nas CPUs futuras da Intel.

(BZ#1614144)

bpftool para inspeção e manipulação de programas e mapas baseados em eBPF adicionado

O utilitário **bpftool** que serve para inspeção e manipulação simples de programas e mapas baseados na Filtragem de Pacotes Berkeley estendida (eBPF) foi adicionado ao kernel do Linux. **bpftool** é uma parte da árvore de fontes do kernel, e é fornecido pelo pacote **bpftool**, que está incluído como um sub-pacote do pacote **kernel**.

(BZ#1559607)

As fontes do kernel-rt foram atualizadas

As fontes de **kernel-rt** foram atualizadas para utilizar a última árvore de fontes do kernel RHEL. A última árvore de fontes do kernel está agora usando o conjunto de correções em tempo real upstream v4.18, que fornece uma série de correções de bugs e melhorias em relação à versão anterior.

(BZ#1592977)

5.1.4. Gestão de software

YUM melhoria do desempenho e suporte para conteúdo modular

No Red Hat Enterprise Linux 8, a instalação do software é assegurada pela nova versão da ferramenta **YUM**, que é baseada na tecnologia **DNF** (**YUM v4**).

YUM v4 tem as seguintes vantagens em relação ao anterior **YUM v3** utilizado no RHEL 7:

- Aumento do desempenho
- Suporte para conteúdo modular
- API estável e bem projetada para integração com ferramentas

Para informações detalhadas sobre as diferenças entre a nova ferramenta **YUM v4** e a versão anterior **YUM v3** da RHEL 7, veja Mudanças no DNF CLI em comparação com o YUM.

YUM v4 é compatível com **YUM v3** ao usar da linha de comando, editar ou criar arquivos de configuração.

Para instalar o software, você pode usar o comando **yum** e suas opções particulares da mesma forma que no RHEL 7.

Plug-ins e utilitários selecionados foram portados para o novo DNF back end, e podem ser instalados com os mesmos nomes que no RHEL 7. Eles também fornecem links simbólicos de compatibilidade, para que os binários, arquivos de configuração e diretórios possam ser encontrados nos locais habituais.

Note que o API Python API legado fornecido por **YUM v3** não está mais disponível. Os usuários são aconselhados a migrar seus plug-ins e scripts para a nova API fornecida por **YUM v4** (DNF Python API), que é estável e totalmente suportada. O DNF Python API está disponível no DNF API Reference.

As APIs Libdnf e Hawkey (ambas C e Python) são instáveis, e provavelmente mudarão durante o ciclo de vida do Red Hat Enterprise Linux 8.

Para mais detalhes sobre mudanças na disponibilidade de pacotes e ferramentas **YUM**, veja Considerações sobre a adoção do RHEL 8 .

Algumas das características do **YUM v3** podem ter um comportamento diferente em **YUM v4**. Se qualquer mudança desse tipo impactar negativamente seus fluxos de trabalho, favor abrir um caso com o Suporte Red Hat, conforme descrito em Como abro e administro um caso de suporte no Portal do Cliente?

(BZ#1581198)

Características RPM notáveis no RHEL 8

O Red Hat Enterprise Linux 8 é distribuído com RPM 4.14. Esta versão introduz muitas melhorias em relação ao RPM 4.11, que está disponível no RHEL 7. As características mais notáveis incluem:

- Os pacotes **de debuginfo** podem ser instalados em paralelo
- Apoio para dependências fracas
- Apoio para dependências ricas ou booleanas
- Suporte para arquivos de embalagem acima de 4 GB de tamanho
- Suporte para acionadores de arquivos

Além disso, as mudanças mais notáveis incluem:

- Mais estrito para os parcerios de especificações
- Assinatura simplificada verificando a saída em modo não-verbose
- Adições e depreciações em macros

(BZ#1581990)

RPM agora valida todo o conteúdo do pacote antes de iniciar uma instalação

No Red Hat Enterprise Linux 7, o utilitário **RPM** verificou o conteúdo da carga útil de arquivos individuais enquanto desempacotava. No entanto, isto é insuficiente por múltiplas razões:

- Se a carga útil for danificada, ela só é notada após a execução de ações de script, que são irreversíveis.
- Se a carga útil for danificada, a atualização de um pacote é abortada após a substituição de alguns arquivos da versão anterior, o que quebra uma instalação em funcionamento.
- Os hashes em arquivos individuais são feitos em dados não compactados, o que torna **RPM** vulnerável a vulnerabilidades do descompressor.

No Red Hat Enterprise Linux 8, o pacote inteiro é validado antes da instalação em uma etapa separada, usando o melhor hash disponível.

Os pacotes construídos no Red Hat Enterprise Linux 8 usam um novo hash **SHA-256** na carga útil comprimida. Em pacotes assinados, o hash de carga útil é protegido adicionalmente pela assinatura e, portanto, não pode ser alterado sem quebrar uma assinatura e outros hashes no cabeçalho do pacote. Pacotes mais antigos usam o hash **MD5** do cabeçalho e da carga útil, a menos que seja desativado pela configuração.

A macro **%_pkgverify_level** pode ser usada para permitir adicionalmente a verificação da assinatura antes da instalação ou desativar completamente a verificação da carga útil. Além disso, a macro **%_pkgverify_flags** pode ser usada para limitar quais hashes e assinaturas são permitidos. Por exemplo, é possível desativar o uso do hash **MD5** fraco ao custo de compatibilidade com pacotes mais antigos.

(JIRA:RHELPLAN-10596)

5.1.5. Serviços de infra-estrutura

Mudanças notáveis no perfil sintonizado recomendado no RHEL 8

Com esta atualização, o perfil sintonizado recomendado (reportado pelo comando **recomendado de sintonização-adm**) é agora selecionado com base nas seguintes regras - a primeira regra que combina entra em vigor:

- Se a função **de sim propósito** (relatada pelo comando **syspurpose show**) contém **atômica**, e ao mesmo tempo:
  - se o Tuned estiver rodando sobre metal nu, o perfil **atômico-anfitrião** é selecionado
  - se Tuned estiver funcionando em uma máquina virtual, o perfil do **convidado atômico** é selecionado
- Se o Tuned estiver funcionando em uma máquina virtual, o perfil do **convidado virtual** é selecionado
- Se a função **de sincronização** contiver **desktop** ou **estação de trabalho** e o tipo de chassi (reportado por **dmidecode**) for **Notebook**, **Laptop**, ou **Portable**, então o perfil **balanceado** é selecionado
- Se nenhuma das regras acima corresponder, o perfil de **desempenho de produção** é selecionado

(BZ#1565598)

Os arquivos produzidos por named podem ser escritos no diretório de trabalho

Anteriormente, o daemon **named** armazenava alguns dados no diretório de trabalho, que foi somente leitura no Red Hat Enterprise Linux. Com esta atualização, os caminhos foram alterados para arquivos selecionados em subdiretórios, onde a escrita é permitida. Agora, as permissões default do diretório Unix e SELinux permitem a escrita no diretório. Os arquivos distribuídos dentro do diretório ainda são somente leitura para **named**.

(BZ#1588592)

Os bancos de dados Geolite foram substituídos pelos Geolite2 Databases

Os Geolite Databases que estavam presentes no Red Hat Enterprise Linux 7 foram substituídos pelos Geolite2 Databases no Red Hat Enterprise Linux 8.

Os bancos de dados Geolite foram fornecidos pelo pacote **GeoIP**. Este pacote junto com o banco de dados legado não é mais suportado no upstream.

Os bancos de dados Geolite2 são fornecidos por vários pacotes. O pacote **libmaxminddb** inclui a biblioteca e a ferramenta de linha de comando **mmdblookup**, que permite a busca manual de endereços. O binário **geoipupdate** do pacote antigo **GeoIP** agora é fornecido pelo pacote **geoipupdate**, e é capaz de baixar tanto os bancos de dados antigos quanto os novos bancos de dados Geolite2.

(JIRA:RHELPLAN-6746)

Os logs do CUPS são tratados pelo journald

No RHEL 8, os registros CUPS não são mais armazenados em arquivos específicos dentro do diretório **/var/log/cups**, que foi usado no RHEL 7. No RHEL 8, todos os tipos de logs CUPS são registrados centralmente no daemon do sistema de **diário** junto com logs de outros programas. Para acessar os logs do CUPS, use o comando **journalctl -u cups**. Para mais informações, consulte Trabalhando com os logs do CUPS.

(JIRA:RHELPLAN-12764)

Características notáveis do BIND no RHEL 8

RHEL 8 inclui o BIND (Berkeley Internet Name Domain) na versão 9.11. Esta versão do servidor DNS introduz múltiplas novas características e mudanças de características em comparação com a versão 9.10.

Novas características:

- Um novo método de provisionamento de servidores secundários chamado **Catalog Zones** foi adicionado.
- Os Cookies do Sistema de Nome de Domínio são agora enviados pelo serviço **nomeado** e pelo utilitário de **escavação**.
- O recurso **Response Rate Limiting**agora pode ajudar na mitigação dos ataques de amplificação do DNS.
- O desempenho da zona de política de resposta (RPZ) foi melhorado.
- Um novo formato de arquivo de zona chamado **mapa** foi adicionado. Os dados de zona armazenados neste formato podem ser mapeados diretamente na memória, o que permite que as zonas sejam carregadas significativamente mais rápido.
- Uma nova ferramenta chamada **delv** (busca e validação de entidades de domínio) foi adicionada, com semântica semelhante à escavação para procurar dados DNS e realizar a validação interna de Extensões de Segurança DNS (DNSSEC).
- Um novo comando **mdig** está agora disponível. Este comando é uma versão do comando `dig` que envia várias consultas pipelinadas e depois espera por respostas, em vez de enviar uma consulta e esperar pela resposta antes de enviar a próxima consulta.
- Uma nova opção de **pré-fetch**, que melhora o desempenho do resolvedor recursivo, foi adicionada.
- Uma nova opção de zona de **visualização**, que permite que os dados da zona sejam compartilhados entre as opiniões, foi adicionada. Quando esta opção é utilizada, várias visões podem servir as mesmas zonas com autoridade sem armazenar várias cópias na memória.
- Uma nova opção de **max-zone-ttl**, que impõe o máximo de TTLs para zonas, foi adicionada. Quando uma zona contendo um TTL superior é carregada, a carga falha. As atualizações do DNS dinâmico (DDNS) com TTLs superiores são aceitas, mas a TTL é truncada.
- Novas cotas foram adicionadas para limitar consultas que são enviadas por resolvedores recursivos para servidores autorizados que sofrem ataques de negação de serviço.
- O utilitário **nslookup** agora procura por padrão tanto endereços IPv6 quanto IPv4.

- O serviço **nomeado** agora verifica se outros processos de servidor de nomes estão sendo executados antes de iniciar.

- Ao carregar uma zona assinada, **nomeada** agora, verifica se o tempo de início de uma Assinatura de Registro de Recursos (RSIG) está no futuro e, se estiver, regenera o RRSIG imediatamente.

- As transferências de zonas agora usam tamanhos de mensagem menores para melhorar a compressão de mensagens, o que reduz o uso da rede.

Mudanças nas características:

- O esquema **XML** da versão **3** para o canal de estatísticas, incluindo novas estatísticas e uma árvore XML achatada para uma análise mais rápida, é fornecido pela interface HTTP. O esquema **XML** herdado da versão **2** não é mais suportado.

- O serviço **nomeado** agora ouve tanto em interfaces IPv6 como IPv4 por padrão.

- O serviço **nomeado** não suporta mais o GeoIP. Listas de controle de acesso (ACLs) definidas pela localização presumida do remetente da consulta não estão disponíveis.

(JIRA:RHELPLAN-1820)

5.1.6. Conchas e ferramentas de linha de comando

O usuário ninguém substitui o nfsnobody

No Red Hat Enterprise Linux 7, havia:

- o par de **ninguém** usuário e grupo com o ID de 99, e

- o par de usuários e grupos **nfsnobody** com o ID de 65534, que também é o ID padrão de sobrecarga do kernel.

Ambos foram fundidos no par de **ninguém** usuário e grupo, que usa o ID 65534 no Red Hat Enterprise Linux 8. Novas instalações não criam mais o par **nfsnobody**.

Esta mudança reduz a confusão sobre arquivos que **não** são de propriedade de **ninguém**, mas não têm nada a ver com o NFS.

(BZ#1591969)

Sistemas de controle de versão no RHEL 8

A RHEL 8 fornece os seguintes sistemas de controle de versão:

- **Git 2.18**, um sistema de controle de revisão distribuído com uma arquitetura descentralizada.

- **Mercurial 4.8**, um sistema de controle de versão distribuída leve, projetado para o manuseio eficiente de grandes projetos.

- **Subversion 1.10**, um sistema de controle de versão centralizado.

Observe que o Sistema de Versões Concorrentes (CVS) e o Sistema de Controle de Revisão (RCS), disponíveis no RHEL 7, não são distribuídos com o RHEL 8.

(BZ#1693775)

Mudanças notáveis no Subversion 1.10

Subversion 1.10 introduz uma série de novas características desde a versão 1.7 distribuída no RHEL 7, bem como as seguintes alterações de compatibilidade:

- Devido a incompatibilidades nas bibliotecas do **Subversion** utilizadas para apoiar as ligações linguísticas, as ligações **Python 3** para o **Subversion 1.10** não estão disponíveis. Como conseqüência, as aplicações que requerem encadernações **Python** para o **Subversion** não são suportadas.
- Os repositórios baseados em **Berkeley DB** não são mais suportados. Antes de migrar, faça backup dos repositórios criados com **Subversion 1.7** usando o comando **svnadmin dump**. Após instalar o RHEL 8, restaure os repositórios usando o comando **svnadmin load**.
- As cópias de trabalho existentes verificadas pelo cliente **Subversion 1.7** na RHEL 7 devem ser atualizadas para o novo formato antes de poderem ser usadas a partir do **Subversion 1.10**. Após instalar o RHEL 8, executar o comando de **atualização svn** em cada cópia de trabalho.
- A autenticação Smartcard para acessar os repositórios usando **https://** não é mais suportada.

(BZ#1571415)

Mudanças notáveis no dstat

O RHEL 8 é distribuído com uma nova versão da ferramenta **dstat**. Esta ferramenta agora faz parte do conjunto de ferramentas do Performance Co-Pilot (PCP). O arquivo **/usr/bin/dstat** e o nome do pacote **dstat** é agora fornecido pelo pacote **pcp-system-tools**.

A nova versão do **dstat** introduz as seguintes melhorias em relação ao **dstat** disponível no RHEL 7:

- apoio**python3**
- Análise histórica
- Análise do hospedeiro remoto
- Plugins de arquivo de configuração
- Novas métricas de desempenho

(BZ#1684947)

5.1.7. Linguagens de programação dinâmica, servidores web e de banco de dados

Python 3 é a implementação padrão Python no RHEL 8

O Red Hat Enterprise Linux 8 é distribuído com o **Python 3.6**. O pacote pode não ser instalado por default. Para instalar o **Python 3.6**, use o comando **yum install python3**.

Python 2.7 está disponível no pacote **python2**. No entanto, **Python 2** terá um ciclo de vida mais curto e seu objetivo é facilitar uma transição mais suave para **Python 3** para os clientes.

Nem o pacote **python** padrão nem o executável **/usr/bin/python** não versionado é distribuído com o RHEL 8. Os clientes são aconselhados a usar o **python3** ou **python2** diretamente. Alternativamente, os administradores podem configurar o comando **python** não versionado usando o comando **alternativo**.

Para detalhes, veja Usando o Python no Red Hat Enterprise Linux 8 .

(BZ#1580387)

Os scripts Python devem especificar a versão principal em hashbangs no tempo de construção RPM

No RHEL 8, os scripts Python executáveis devem usar hashbangs (shebangs) especificando explicitamente pelo menos a principal versão Python.

O script **/usr/lib/rpm/redhat/brp-mangle-shebangs** buildroot policy (BRP) é executado automaticamente ao construir qualquer pacote RPM. Este script tenta corrigir hashbangs em todos os arquivos executáveis. Quando o script encontra hashbangs Python ambíguos que não especificam a versão principal do Python, ele gera erros e a compilação do RPM falha. Exemplos de tais hashbangs ambíguos incluem:

- **#! /usr/bin/pithon**
- **#! /usr/bin/env python**

Para modificar hashbangs nos scripts Python causando estes erros de construção em tempo de construção RPM, use o script **pathfix.py** do pacote **platform-python-devel**:

```
pathfix.py -pn -i %{__python3} PATH...
```

Múltiplos *PATH*s podem ser especificados. Se um *PATH* é um diretório, o **pathfix.py** escaneia recursivamente qualquer script Python que corresponda ao padrão **^[a-zA-Z0-9_] ^[a-zA-Z0-9_] py$**, não apenas aqueles com um hashbang ambíguo. Adicione o comando para executar **pathfix.py** à seção **%prep** ou ao final da seção **%install**.

Para mais informações, consulte Manuseio de hashbangs em scripts Python.

(BZ#1583620)

Mudanças notáveis no PHP

O Red Hat Enterprise Linux 8 é distribuído com **PHP 7.2**. Esta versão introduz as seguintes mudanças principais sobre o **PHP 5.4**, que está disponível no RHEL 7:

- **PHP** usa FastCGI Process Manager (FPM) por padrão (seguro para uso com um **httpd** com rosca)
- As variáveis **php_value** e **php-flag** não devem mais ser usadas nos arquivos de configuração **httpd**; em vez disso, elas devem ser definidas na configuração de pool: **/etc/php-fpm.d/*.conf**
- Os erros e avisos de script**PHP** são registrados no arquivo **/var/log/php-fpm/wwww-error.log** em vez de **/var/log/httpd/error.log**
- Ao alterar a variável de configuração PHP **max_execution_time**, a configuração **httpd** **ProxyTimeout** deve ser aumentada para corresponder
- O usuário executando scripts **PHP** agora está configurado na configuração do pool FPM (o arquivo **/etc/php-fpm.d/wwww.conf**; o usuário **apache** é o padrão)
- O serviço **php-fpm** precisa ser reiniciado após uma mudança de configuração ou após a instalação de uma nova extensão
- A extensão do **zip** foi movida do pacote **php-common** para um pacote separado, **php-pecl-zip**

As seguintes extensões foram removidas:

- **aspell**
- **mysql** (note que as extensões **mysqli** e **pdo_mysql** ainda estão disponíveis, fornecidas pelo pacote **php-mysqlnd**)
- **memcache**

(BZ#1580430, BZ#1691688)

Mudanças notáveis em Ruby

A RHEL 8 fornece o **Ruby 2.5**, que introduz inúmeras novas características e melhorias em relação ao **Ruby 2.0.0** disponível no RHEL 7. As mudanças notáveis incluem:

- Foi adicionado um coletor de lixo incremental.
- A sintaxe dos **Refinamentos** foi adicionada.
- Os símbolos são agora coletados de lixo.
- Os níveis de segurança **$SAFE=2** e **$SAFE=3** estão agora obsoletos.
- As classes **Fixnum** e **Bignum** foram unificadas na classe **Integer**.
- O desempenho foi melhorado otimizando a classe **Hash**, melhorando o acesso às variáveis de instância e sendo a classe **Mutex** menor e mais rápida.
- Certas APIs antigas foram depreciadas.
- Bibliotecas agrupadas, tais como **RubyGems**, **Rake**, **RDoc**, **Psych**, **Minitest**, e **test-unit**, foram atualizadas.
- Outras bibliotecas, como **mathn**, **DL**, **ext/tk** e **XMLRPC**, que antes eram distribuídas com **Ruby**, são depreciadas ou não estão mais incluídas.
- O esquema de versões **SemVer** é agora utilizado para as versões **Ruby**.

(BZ#1648843)

Mudanças notáveis em Perl

Perl 5.26, distribuído com o RHEL 8, introduz as seguintes mudanças em relação à versão disponível no RHEL 7:

- O**Unicode 9.0** é agora suportado.
- Novas sondas **op-entry**, **loading-file**, e **load-file SystemTap** são fornecidas.
- O mecanismo de cópia-em-escrita é usado ao atribuir escalares para melhorar o desempenho.
- O **IO::Socket::IP** módulo para manuseio de soquetes IPv4 e IPv6 foi adicionado de forma transparente.
- O módulo **Config::Perl::V** para acessar dados **perl -V** de forma estruturada foi adicionado.

- Um novo pacote **perl-App-cpanminus** foi adicionado, que contém o utilitário **cpanm** para obter, extrair, construir e instalar módulos do repositório Comprehensive Perl Archive Network (CPAN).
- O diretório atual **.** foi removido do caminho de busca do módulo **@INC** por razões de segurança.
- A declaração **do do** agora retorna uma advertência de depreciação quando falha em carregar um arquivo por causa da mudança de comportamento descrita acima.
- A chamada **do subrotine(LIST)** não é mais suportada e resulta em um erro de sintaxe.
- Os hashes são randomizados por padrão agora. A ordem na qual chaves e valores são retornados a partir de um hash muda a cada execução do **perl**. Para desativar a randomização, defina a variável de ambiente **PERL_PERTURB_KEYS** para **0**.
- Não são mais permitidos **{** caracteres em padrões de expressão regulares **{** caracteres não escalonados} literalmente.
- O suporte de escopo léxico para a variável **$_** foi removido.
- A utilização do operador **definido** em uma matriz ou hash resulta em um erro fatal.
- A importação de funções do módulo **UNIVERSAL** resulta em um erro fatal.
- As ferramentas **find2perl**, **s2p**, **a2p**, **c2ph** e **pstruct** foram removidas.
- A instalação de **${^ENCODING}** foi removida. O modo de **codificação** padrão do pragma não é mais suportado. Para escrever o código fonte em outras codificações que não **UTF-8**, use a opção **Filtro** da codificação.
- A embalagem **perl** está agora alinhada com o upstream. A embalagem **perl** também instala módulos principais, enquanto o **/usr/bin/perl** intérprete é fornecido pela embalagem **perl intérprete**. Em versões anteriores, o pacote **perl** incluía apenas um intérprete mínimo, enquanto que o pacote **perl-core** incluía tanto o intérprete quanto os módulos do núcleo.
- O módulo **IO::Socket::SSL** Perl não carrega mais um certificado de autoridade de certificado do arquivo **./certs/my-ca.pem** ou do diretório **./ca**, uma chave privada do servidor do arquivo **./certs/server-key.pem**, um certificado do servidor do arquivo **./certs/server-cert.pem**, uma chave privada do cliente do arquivo **./certs/client-key.pem** e um certificado do cliente do arquivo **./certs/client-cert.pem**. Especifique os caminhos para os arquivos explicitamente em seu lugar.

(BZ#1511131)

Node.js novo na RHEL

Node.js, uma plataforma de desenvolvimento de software para construir aplicações de rede rápidas e escaláveis na linguagem de programação JavaScript, é fornecida pela primeira vez na RHEL. Anteriormente, estava disponível apenas como uma Coleção de Software. A RHEL 8 fornece o **Node.js 10**.

(BZ#1622118)

Mudanças notáveis no SWIG

RHEL 8 inclui o Empacotador Simplificado e o Gerador de Interface (SWIG) versão 3.0, que fornece inúmeras novas características, melhorias e correções de bugs sobre a versão 2.0 distribuída no RHEL 7. Mais notavelmente, o suporte para o padrão C 11 foi implementado. **O SWIG** agora suporta também **Go**

1.6, **PHP 7**, **Octave 4.2**, e **Python 3.5**.

(BZ#1660051)

Mudanças notáveis no Apache httpd

O RHEL 8 é distribuído com o Servidor HTTP Apache 2.4.37. Esta versão introduz as seguintes mudanças sobre o **httpd** disponível no RHEL 7:

- O suporte HTTP/2 agora é fornecido pelo pacote **mod_http2**, que é parte do módulo **httpd**.
- O provisionamento e renovação automatizada de certificados TLS usando o protocolo ACME (Automatic Certificate Management Environment) é agora suportado com o pacote **mod_md** (para uso com provedores de certificados como **Let's Encrypt**)
- O Servidor HTTP Apache agora suporta o carregamento de certificados TLS e chaves privadas de tokens de segurança de hardware diretamente dos módulos **PKCS#11**. Como resultado, uma configuração **mod_ssl** pode agora usar URLs **PKCS#11** para identificar a chave privada TLS e, opcionalmente, o certificado TLS nas diretrizes **SSLCertificateKeyFile** e **SSLCertificateFile**.
- O módulo multi-processamento (MPM) configurado por padrão com o Servidor HTTP Apache mudou de um modelo multi-processo, bifurcado (conhecido como **prefork**) para um modelo multi-tarefa de alto desempenho, **evento**. Quaisquer módulos de terceiros que não sejam thread-safe precisam ser substituídos ou removidos. Para alterar o MPM configurado, edite o arquivo **/etc/httpd/conf.modules.d/00-mpm.conf**. Veja a página de manual **httpd.conf(5)** para maiores informações.

Para maiores informações sobre mudanças no **httpd** e sua utilização, veja Configurando o servidor web Apache HTTP.

(BZ#1632754, BZ#1527084, BZ#1581178)

O servidor web nginx novo na RHEL

RHEL 8 apresenta **o nginx 1.14**, um servidor web e proxy que suporta HTTP e outros protocolos, com foco em alta concorrência, desempenho e baixo uso de memória. **nginx** estava anteriormente disponível apenas como uma Coleção de Software.

O servidor web **nginx** agora suporta o carregamento de chaves privadas TLS a partir de fichas de segurança de hardware diretamente dos módulos **PKCS#11**. Como resultado, uma configuração **nginx** pode usar URLs **PKCS#11** para identificar a chave privada TLS na diretiva **ssl_certificate_key**.

(BZ#1545526)

Servidores de banco de dados no RHEL 8

A RHEL 8 fornece os seguintes servidores de banco de dados:

- **MySQL 8.**0, um servidor de banco de dados SQL multiusuário e multi-tarefa. Ele consiste no daemon servidor **MySQL**, **mysqld** e muitos programas clientes.
- **MariaDB 10.3**, um servidor de banco de dados SQL multiusuário e multi-tarefa. Para todos os fins práticos, **o MariaDB** é binário compatível com o **MySQL**.
- **PostgreSQL 10** e **PostgreSQL 9.6**, um avançado sistema de gerenciamento de banco de dados objeto-relacional (SGBD).

- **Redis 5**, uma loja de valores chave avançada. E freqüentemente referido como um servidor de estrutura de dados porque as chaves podem conter cordas, hashes, listas, conjuntos e conjuntos ordenados. O **Redis** é fornecido pela primeira vez na RHEL.

Note que o servidor de banco de dados NoSQL **MongoDB** não está incluído no RHEL 8.0 porque utiliza a Licença Pública do Lado do Servidor (SSPL).

(BZ#1647908)

Mudanças notáveis no MySQL 8.0

O RHEL 8 é distribuído com o **MySQL 8.0**, que fornece, por exemplo, os seguintes aprimoramentos:

- **O MySQL** agora incorpora um dicionário de dados transacionais, que armazena informações sobre objetos de banco de dados.
- **O MySQL** agora suporta papéis, que são coleções de privilégios.
- O conjunto de caracteres padrão foi alterado do **latin1** para **utf8mb4**.
- Foi adicionado suporte para expressões comuns de tabela, tanto não-recorrentes como recursivas.
- **O MySQL** agora suporta funções de janela, que realizam um cálculo para cada linha a partir de uma consulta, usando linhas relacionadas.
- A**InnoDB** agora suporta as opções **NOWAIT** e **SKIP LOCKED** com declarações de leitura de bloqueio.
- As funções relacionadas ao SIG foram melhoradas.
- A funcionalidade do JSON foi melhorada.
- Os novos pacotes **mariadb-connector-c** fornecem uma biblioteca cliente comum para **MySQL** e **MariaDB**. Esta biblioteca é utilizável com qualquer versão dos servidores de banco de dados **MySQL** e **MariaDB**. Como resultado, o usuário é capaz de conectar um build de uma aplicação a qualquer um dos servidores **MySQL** e **MariaDB** distribuídos com o RHEL 8.

Além disso, o servidor **MySQL 8.0** distribuído com RHEL 8 está configurado para usar **mysql_native_password** como o plug-in padrão de autenticação porque as ferramentas e bibliotecas do cliente no RHEL 8 são incompatíveis com o método **caching_sha2_password**, que é usado por padrão na versão upstream do **MySQL 8.** 0.

Para alterar o plug-in de autenticação padrão para **caching_sha2_password**, edite o arquivo **/etc/my.cnf.d/mysql-default-authentication-plugin.cnf** da seguinte forma:

```
[mysqld]
default_authentication_plugin=caching_sha2_password
```

(BZ#1649891, BZ#1519450, BZ#1631400)

Mudanças notáveis no MariaDB 10.3

O MariaDB 10.3 oferece inúmeras novidades sobre a versão 5.5 distribuída na RHEL 7, como por exemplo:

- Expressões comuns da tabela

- Mesas de sistema-versão
- **PARA** laços
- Colunas invisíveis
- Sequências
- **COLUNA ADICIONAL** Instantânea para **InnoDB**
- Compressão de coluna independente do motor de armazenagem
- Replicação paralela
- Replicação de várias fontes

Além disso, os novos pacotes **mariadb-connector-c** fornecem uma biblioteca cliente comum para **MySQL** e **MariaDB**. Esta biblioteca é utilizável com qualquer versão dos servidores de banco de dados **MySQL** e **MariaDB**. Como resultado, o usuário é capaz de conectar um build de uma aplicação a qualquer um dos servidores **MySQL** e **MariaDB** distribuídos com o RHEL 8.

Outras mudanças notáveis incluem:

- O**MariaDB Galera Cluster**, um aglomerado síncrono multi-mestre, é agora uma parte padrão do **MariaDB**.
- **O InnoDB** é usado como o motor de armazenamento padrão em vez do **XtraDB**.
- O subpacote mariadb-bench foi removido.
- O nível padrão permitido de maturidade do plug-in foi alterado para um nível a menos do que a maturidade do servidor. Como resultado, os plug-ins com um nível de maturidade mais baixo que estavam funcionando anteriormente, não serão mais carregados.

Veja também Usando o MariaDB no Red Hat Enterprise Linux 8 .

(BZ#1637034, BZ#1519450, BZ#1688374)

Mudanças notáveis no PostgreSQL

O RHEL 8.0 fornece duas versões do servidor de banco de dados **PostgreSQL**, distribuído em dois fluxos do módulo **postgresql**: **PostgreSQL 10** (o fluxo padrão) e **PostgreSQL 9.6**. O RHEL 7 inclui a versão 9.2 do **PostgreSQL**.

As mudanças notáveis no **PostgreSQL 9.** 6 são, por exemplo:

- Execução paralela das operações seqüenciais: **varredura**, **união** e **agregação**
- Melhorias para a replicação síncrona
- Melhoria da pesquisa de texto completo permitindo aos usuários pesquisar frases
- O driver da federação de dados **postgres_fdw** agora suporta operações remotas de **junção**, **classificação**, **ATUALIZAÇÃO** e **DELETE**
- Melhorias substanciais de desempenho, especialmente em relação à escalabilidade em servidores com várias CPUs

As principais melhorias no **PostgreSQL 10** incluem:

- Replicação lógica usando a **publicação** e **assinatura de** palavras-chave
- Autenticação mais forte por senha com base no mecanismo **SCRAM-SHA-256**
- Divisória de mesa declarativa
- Paralelismo de consultas melhorado
- Melhorias significativas no desempenho geral
- Melhor monitoramento e controle

Veja também Usando o PostgreSQL no Red Hat Enterprise Linux 8 .

(BZ#1660041)

Mudanças notáveis no Squid

O RHEL 8.0 é distribuído com o **Squid 4.4**, um servidor proxy de cache de alto desempenho para clientes web, que suporta FTP, Gopher e objetos de dados HTTP. Este lançamento fornece inúmeras novas características, melhorias e correções de bugs sobre a versão 3.5 disponível no RHEL 7.

As mudanças notáveis incluem:

- Tamanho da fila de ajuda configurável
- Mudanças nos canais de concorrência de ajuda
- Mudanças no binário do ajudante
- Protocolo de Adaptação de Conteúdo Seguro da Internet (ICAP)
- Melhor suporte para o multiprocessamento simétrico (SMP)
- Melhoria da gestão de processos
- Removido o suporte para SSL
- Lado da Borda Removida Inclui (ESI) analisador personalizado
- Mudanças múltiplas de configuração

(BZ#1656871)

Cache de Verniz novo em RHEL

Verniz Cache, um proxy HTTP reverso de alto desempenho, é fornecido pela primeira vez na RHEL. Anteriormente, estava disponível apenas como uma Coleção de Software. O **Vernish Cache** armazena arquivos ou fragmentos de arquivos em memória que são usados para reduzir o tempo de resposta e o consumo de largura de banda da rede em futuras solicitações equivalentes. O RHEL 8.0 é distribuído com o **Verniz Cache 6.0**.

(BZ#1633338)

5.1.8. Desktop

GNOME Shell, versão 3.28 em RHEL 8

O GNOME Shell, versão 3.28 está disponível no Red Hat Enterprise Linux (RHEL) 8. Entre as melhorias notáveis estão

- Novas funcionalidades das caixas GNOME
- Novo teclado na tela
- Suporte ampliado de dispositivos, integração mais significativa para a interface Thunderbolt 3
- Melhorias para o Software GNOME, dconf-editor e Terminal GNOME

(BZ#1649404)

Wayland é o servidor de exibição padrão

Com o Red Hat Enterprise Linux 8, a sessão GNOME e o Gerenciador de Exibição GNOME (GDM) usam **Wayland** como seu servidor de exibição padrão ao invés do servidor **X.org**, que foi usado com a versão anterior principal da RHEL.

Wayland oferece múltiplas vantagens e melhorias em relação a **X.org**. Mais notavelmente:

- Modelo de segurança mais forte
- Manuseio aperfeiçoado de multi-monitores
- Melhoria da escala da interface do usuário (IU)
- A área de trabalho pode controlar diretamente o manuseio das janelas.

Observe que as seguintes características estão atualmente indisponíveis ou não funcionam como esperado:

- As configurações Multi-GPU não são suportadas sob **Wayland**.
- O motorista binário **NVIDIA** não funciona em **Wayland**.
- A utilidade **xrandr** não funciona sob **Wayland** devido a sua abordagem diferente de manuseio, resoluções, rotações e layout. Note que outras utilidades **X.org** para manipulação da tela também não funcionam sob **Wayland**.
- A gravação na tela, o desktop remoto e a acessibilidade nem sempre funcionam corretamente em **Wayland**.
- Não há um gerente de prancheta disponível.
- **Wayland** ignora as pegadas de teclado emitidas pelas aplicações X11, tais como os visualizadores de máquinas virtuais.
- **Wayland** dentro de máquinas virtuais convidadas (VMs) tem problemas de estabilidade e desempenho, por isso é recomendado o uso da sessão X11 para ambientes virtuais.

Se você atualizar para o RHEL 8 a partir de um sistema RHEL 7 onde você usou a sessão **X.org** GNOME, seu sistema continua a usar **X.org**. O sistema também volta automaticamente para **X.org** quando os seguintes drivers gráficos estão em uso:

- O driver binário NVIDIA

- O motorista de **cirrus**
- O motorista de **mga**
- O motorista de **velocidade**

Você pode desativar o uso do site **Wayland** manualmente:

- Para desativar **Wayland** em **GDM**, defina a opção **WaylandEnable=false** no arquivo **/etc/gdm/custom.conf**.
- Para desativar **Wayland** na sessão GNOME, selecione a opção legado X11 usando o menu da roda dentada na tela de login após digitar seu nome de login.

Para obter mais detalhes em **Wayland**, consulte https://wayland.freedesktop.org/.

(BZ#1589678)

Localização de pacotes RPM que estão em repositórios não habilitados por padrão

Os repositórios adicionais para a área de trabalho não estão habilitados por padrão. A desativação é indicada pela linha **habilitada=0** no arquivo **.repo** correspondente. Se você tentar instalar um pacote de tal repositório usando o PackageKit, o PackageKit mostra uma mensagem de erro anunciando que o aplicativo não está disponível. Para tornar o pacote disponível, substitua a linha **habilitado=0** usada anteriormente no respectivo arquivo **.repo** por **habilitado=1**.

(JIRA:RHELPLAN-2878)

GNOME Sofware para gerenciamento de pacotes

O pacote **gnome-packagekit** que forneceu uma coleção de ferramentas para o gerenciamento de pacotes em ambiente gráfico no Red Hat Enterprise Linux 7 não está mais disponível. No Red Hat Enterprise Linux 8, funcionalidade similar é fornecida pelo utilitário **GNOME Software**, que permite instalar e atualizar aplicativos e extensões gnome-shell. **GNOME Software** é distribuído no pacote **gnome-software**.

(JIRA:RHELPLAN-3001)

Escala fracionária disponível para o GNOME Shell em Wayland

Em uma sessão **GNOME Shell on Wayland**, o recurso de escalonamento fracionário está disponível. O recurso torna possível escalar a GUI por frações, o que melhora a aparência da GUI em escala em determinados displays.

Observe que esta característica é atualmente considerada experimental e, portanto, é desativada por padrão.

Para permitir o escalonamento fracionário, execute o seguinte comando:

```
# gsettings set org.gnome.mutter experimental-features" ["scale-monitor-framebuffer"]}"
```

(BZ#1668883)

5.1.9. Habilitação do hardware

Atualizações de firmware usando fwupd estão disponíveis

O RHEL 8 suporta atualizações de firmware, como cápsula UEFI, Atualização de Firmware do Dispositivo (DFU), e outros, usando o daemon **fwupd**. O daemon permite que o software da sessão atualize o firmware do dispositivo em uma máquina local automaticamente.

Para visualizar e aplicar atualizações, você pode usar:

- Um gerente de software GUI, como o GNOME Software
- A ferramenta de linha de comando **fwupdmgr**

Os arquivos de metadados são automaticamente baixados do portal seguro do Linux Vendor Firmware Service (LVFS), e enviados para o **fwupd** sobre o D-Bus. As atualizações que precisam ser aplicadas são baixadas exibindo notificações do usuário e detalhes de atualização. O usuário deve concordar explicitamente com a ação de atualização do firmware antes que a atualização seja realizada.

Observe que o acesso ao LVFS é desativado por padrão.

Para permitir o acesso ao LVFS, clique no botão deslizante no diálogo de **fontes** no Software GNOME, ou execute o comando **fwupdmgr enable-remote lvfs**. Se você usar **fwupdmgr** para obter a lista de atualizações, você será perguntado se deseja habilitar o LVFS.

Com acesso ao LVFS, você receberá atualizações de firmware diretamente do fornecedor de hardware. Note que tais atualizações não foram verificadas pela Red Hat QA.

(BZ#1504934)

Modo de memória para Optane DC A tecnologia de memória persistente é totalmente suportada

Os dispositivos de armazenamento de memória Intel Optane DC Persistent Memory fornecem tecnologia de memória persistente de classe data center, que pode aumentar significativamente o rendimento das transações.

Para utilizar a tecnologia do Modo Memória, seu sistema não requer nenhum condutor especial ou certificação específica. O Modo Memória é transparente para o sistema operacional.

(BZ#1718422)

5.1.10. Gestão da Identidade

Novas verificações de sintaxe de senha no Directory Server

Esta melhoria acrescenta novas verificações de sintaxe de senha ao Directory Server. Os administradores podem agora, por exemplo, ativar verificações de dicionário, permitir ou negar usando seqüências de caracteres e palíndromos. Como resultado, se ativada, a verificação da sintaxe da política de senhas no Servidor de Diretório reforça as senhas mais seguras.

(BZ#1334254)

O Servidor de Diretório agora oferece suporte melhorado ao registro de operações internas

Várias operações no Directory Server, iniciadas pelo servidor e pelos clientes, causam operações adicionais em segundo plano. Anteriormente, o servidor só registrava para operações internas a palavra-chave **Internal** connection, e a identificação da operação era sempre definida como **-1**. Com esta melhoria, o Servidor de Diretório registra a conexão real e o ID da operação. Agora você pode rastrear a operação interna até a operação do servidor ou do cliente que causou esta operação.

(BZ#1358706)

A biblioteca tomcatjss suporta a verificação OCSP usando o respondedor da extensão AIA

Com este aperfeiçoamento, a biblioteca **tomcatjss** suporta a verificação do Protocolo de Status de Certificado Online (OCSP) usando o respondedor da extensão Authority Information Access (AIA) de um certificado. Como resultado, os administradores do Sistema de Certificado da Red Hat podem agora configurar a verificação OCSP que usa a URL da extensão AIA.

(BZ#1636564)

Os comandos pki subsystem-cert-find e pki subsystem-cert-show agora mostram o número de série dos certificados

Com este aperfeiçoamento, os comandos **pki subsystem-cert-find** e **pki subsystem-cert-show** em Certificate System mostram o número de série dos certificados em sua saída. O número de série é uma informação importante e muitas vezes exigida por vários outros comandos. Como resultado, a identificação do número de série de um certificado é agora mais fácil.

(BZ#1566360)

Os comandos do usuário pki e do grupo pki foram depreciados no Sistema de Certificado

Com esta atualização, os novos comandos **pki *<subsystem>*-user** e **pki *<subsystem>*-groups** substituem os comandos **pki user** e **pki group** em Certificate System. Os comandos substituídos ainda funcionam, mas eles exibem uma mensagem de que o comando é depreciado e se referem aos novos comandos.

(BZ#1394069)

Sistema de certificados agora suporta a renovação off-line de certificados de sistema

Com este aprimoramento, os administradores podem usar o recurso de renovação off-line para renovar certificados de sistema configurados no Sistema de Certificados. Quando um certificado de sistema expira, o Sistema de Certificados não inicia. Como resultado do aprimoramento, os administradores não precisam mais de soluções para substituir um certificado de sistema expirado.

(BZ#1669257)

O Sistema de Certificado pode agora criar CSRs com extensão SKI para assinatura externa da CA

Com este aperfeiçoamento, o Sistema de Certificado suporta a criação de um pedido de assinatura de certificado (CSR) com a extensão Subject Key Identifier (SKI) para assinatura de autoridade de certificado externa (CA). Certas ACs exigem esta extensão com um valor particular ou derivada da chave pública da AC. Como resultado, os administradores podem agora usar o parâmetro **pki_req_ski** no arquivo de configuração passado para o utilitário **pkispawn** para criar um CSR com extensão SKI.

(BZ#1656856)

O SSSD não usa mais o valor fallback_homedir da seção [nss] como fallback para domínios AD

Antes do RHEL 7.7, o parâmetro **fallback_homedir** do SSSD em um provedor do Active Directory (AD) não tinha valor padrão. Se **fallback_homedir** não foi definido, o SSSD usou em vez disso o valor do mesmo parâmetro da seção **[nss]** no arquivo **/etc/sssd/sssd.conf**. Para aumentar a segurança, o SSSD no RHEL 7.7 introduziu um valor padrão para **fallback_homedir**. Como conseqüência, o SSSD não cai

mais para o valor definido na seção **[nss]**. Se você quiser usar um valor diferente do padrão para o parâmetro **fallback_homedir** em um domínio AD, você deve defini-lo manualmente na seção do domínio.

(BZ#1652719)

O SSSD agora permite selecionar um dos múltiplos dispositivos de autenticação Smartcard

Por padrão, o System Security Services Daemon (SSSD) tenta detectar automaticamente um dispositivo para autenticação Smartcard. Se houver vários dispositivos conectados, o SSSD seleciona o primeiro que ele detecta. Consequentemente, não é possível selecionar um determinado dispositivo, o que às vezes leva a falhas.

Com esta atualização, você pode configurar uma nova opção **p11_uri** para a seção **[pam]** do arquivo de configuração do **sssd.conf**. Esta opção permite definir qual dispositivo é usado para autenticação Smartcard.

Por exemplo, para selecionar um leitor com o slot id **2** detectado pelo módulo OpenSC PKCS#11, adicionar:

```
p11_uri = biblioteca-descrição=estrutura do smartcard OpenSC;slot-id=2
```

para a seção **[pam]** da **sssd.conf.**

Para detalhes, consulte a página **man sssd.conf**.

(BZ#1620123)

Os usuários locais são armazenados em cache pelo SSSD e servidos através do módulo nss_sss

No RHEL 8, o System Security Services Daemon (SSSD) atende usuários e grupos dos arquivos **/etc/passwd** e **/etc/groups** por padrão. O módulo **sss** nsswitch precede os arquivos do arquivo **/etc/nsswitch.conf**.

A vantagem de servir os usuários locais através do SSSD é que o módulo **nss_sss** tem um rápido **cache com memória mapeada** que acelera a busca do Name Service Switch (NSS) em comparação com o acesso ao disco e a abertura dos arquivos em cada solicitação NSS. Anteriormente, o daemon de cache do Name Service**(nscd**) ajudava a acelerar o processo de acesso ao disco. Entretanto, usar **o nscd** em paralelo com o SSSD é incômodo, pois tanto o SSSD quanto o **nscd** usam seu próprio cache independente. Consequentemente, o uso do **nscd** em configurações onde o SSSD também serve usuários de um domínio remoto, por exemplo LDAP ou Active Directory, pode causar um comportamento imprevisível.

Com esta atualização, a resolução dos usuários e grupos locais é mais rápida no RHEL 8. Note que o usuário **root** nunca é tratado pelo SSSD, portanto, a resolução **root** não pode ser impactada por um erro potencial no SSSD. Note também que se o SSSD não estiver rodando, o módulo **nss_sss** lida com a situação graciosamente, caindo de volta aos **nss_files** para evitar problemas. Não é necessário configurar o SSSD de forma alguma, o domínio de arquivos é adicionado automaticamente.

(JIRA:RHELPLAN-10439)

A KCM substitui a KEYRING como o armazenamento padrão de credenciais

No RHEL 8, o armazenamento padrão do cache de credenciais é o Kerberos Credential Manager (KCM), que é apoiado pelo deamon **sssd-kcm**. O KCM supera as limitações do KEYRING usado anteriormente, tais como sua dificuldade de uso em ambientes de contêineres por não ser espaçado por nomes, e para

visualizar e gerenciar cotas.

Com esta atualização, o RHEL 8 contém um cache de credenciais que é mais adequado para ambientes de contêineres e que fornece uma base para a construção de mais recursos em lançamentos futuros.

(JIRA:RHELPLAN-10440)

Os usuários do Active Directory agora podem administrar a Gestão de Identidade

Com esta atualização, a RHEL 8 permite adicionar uma substituição de ID de usuário para um usuário do Active Directory (AD) como membro de um grupo de Gerenciamento de Identidade (IdM). Uma substituição de ID é um registro que descreve como deve ser um usuário específico de AD ou propriedades de grupo dentro de uma visualização de ID específica, neste caso, a Visualização de Confiança Padrão. Como conseqüência da atualização, o servidor LDAP do IdM é capaz de aplicar regras de controle de acesso para o grupo IdM ao usuário AD.

Os usuários AD agora são capazes de usar os recursos de autoatendimento da IdM UI, por exemplo, para carregar suas chaves SSH, ou alterar seus dados pessoais. Um administrador de AD é capaz de administrar completamente o IdM sem ter duas contas e senhas diferentes. Note que atualmente, recursos selecionados no IdM podem ainda não estar disponíveis para os usuários AD.

(JIRA:RHELPLAN-10442)

sssctl imprime um relatório de regras HBAC para um domínio IdM

Com esta atualização, a utilidade **sssctl** do System Security Services Daemon (SSSD) pode imprimir um relatório de controle de acesso para um domínio de Gerenciamento de Identidade (IdM). Este recurso atende à necessidade de certos ambientes de ver, por razões regulamentares, uma lista de usuários e grupos que podem acessar uma máquina cliente específica. Executando o **sssctl access-report domain_name** em um cliente IdM imprime o subconjunto de regras de controle de acesso baseado em host (HBAC) no domínio IdM que se aplicam à máquina cliente.

Note que nenhum outro fornecedor além da IdM suporta esta característica.

(JIRA:RHELPLAN-10443)

Os pacotes de Gerenciamento de Identidade estão disponíveis como um módulo

No RHEL 8, os pacotes necessários para instalação de um servidor e cliente de Gerenciamento de Identidade (IdM) são enviados como um módulo. O fluxo **cliente** é o fluxo padrão do módulo **idm** e você pode baixar os pacotes necessários para a instalação do cliente sem habilitar o fluxo.

O fluxo do módulo do servidor IdM é chamado de fluxo **DL1**. O fluxo contém vários perfis correspondentes a diferentes tipos de servidores IdM: servidor, dns, adtrust, cliente e padrão. Para baixar os pacotes em um perfil específico do fluxo **DL1**:

1. Habilite o fluxo.
2. Mudar para as RPMs entregues através do fluxo.
3. Execute o comando **yum module install idm:DL1/profile_name**.

Para mudar para um novo fluxo de módulos uma vez que você já tenha habilitado um fluxo específico e feito o download de pacotes dele:

1. Remova todo o conteúdo instalado relevante e desative o fluxo do módulo atual.
2. Habilitar o novo fluxo de módulos.

(JIRA:RHELPLAN-10438)

Adicionada solução de gravação de sessão para RHEL 8

Uma solução de gravação de sessão foi adicionada ao Red Hat Enterprise Linux 8 (RHEL 8). Um novo pacote **tlog** e seu leitor de sessão associado ao console web permitem a gravação e reprodução das sessões do terminal do usuário. A gravação pode ser configurada por usuário ou grupo de usuários através do serviço System Security Services Daemon (SSSD). Todas as entradas e saídas do terminal são capturadas e armazenadas em um formato de texto em um diário do sistema. A entrada é inativa por padrão por razões de segurança para não interceptar senhas brutas e outras informações sensíveis.

A solução pode ser usada para auditoria de sessões de usuários em sistemas sensíveis à segurança. No caso de uma quebra de segurança, as sessões gravadas podem ser revisadas como parte de uma análise forense. Os administradores do sistema agora são capazes de configurar a gravação da sessão localmente e visualizar o resultado da interface do console web RHEL 8 ou da interface da linha de comando usando o utilitário **tlog-play**.

(JIRA:RHELPLAN-1473)

authselect simplifica a configuração da autenticação do usuário

Esta atualização introduz o utilitário **authselect** que simplifica a configuração da autenticação do usuário nos hosts RHEL 8, substituindo o utilitário **authconfig**. o **authselect** vem com uma abordagem mais segura ao gerenciamento de pilha do PAM que torna as mudanças de configuração do PAM mais simples para os administradores do sistema. o **authselect** pode ser usado para configurar métodos de autenticação como senhas, certificados, cartões inteligentes e impressão digital. Note que **o authselect** não configura os serviços necessários para entrar em domínios remotos. Esta tarefa é realizada por ferramentas especializadas, como **realmd** ou **ipa-client-install**.

(JIRA:RHELPLAN-10445)

O SSSD agora impõe por padrão os GPO AD

A configuração padrão para a opção SSSD **ad_gpo_access_control** está agora **fazendo cumprir**. No RHEL 8, o SSSD aplica por padrão as regras de controle de acesso baseadas nos Objetos de Políticas de Grupos de Diretórios Ativos (GPOs).

A Red Hat recomenda garantir que os GPOs sejam configurados corretamente no Active Directory antes de atualizar de RHEL 7 para RHEL 8. Se você não quiser reforçar os GPOs, altere o valor da opção **ad_gpo_access_control** no arquivo /**etc/sssd/sssd.conf** para **permissivo**.

(JIRA:RHELPLAN-51289)

5.1.11. Compiladores e ferramentas de desenvolvimento

Impulso atualizado para a versão 1.66

A biblioteca **Boost** C foi atualizada para a versão upstream 1.66. A versão de **Boost** incluída no Red Hat Enterprise Linux 7 é a 1.53. Para detalhes, veja os changelogs upstream: https://www.boost.org/users/history/

Esta atualização introduz as seguintes mudanças que quebram a compatibilidade com as versões anteriores:

- A função **bs_set_hook()**, a função **splay_set_hook()** dos recipientes splay, e o **bool splay = verdadeiro** parâmetro extra na função **splaytree_algorithms()** na biblioteca **Intrusive** foram removidos.

- Comentários ou concatenação de strings nos arquivos JSON não são mais suportados pelo analisador na biblioteca **Property Tree**.
- Algumas distribuições e funções especiais da biblioteca **Math** foram fixadas para se comportar como documentado e levantar um **overflow_error** em vez de retornar o valor finito máximo.
- Alguns cabeçalhos da biblioteca **Math** foram movidos para o diretório **libs/math/include_private**.
- O comportamento das funções **basic_regex<>::mark_count()** e **basic_regex<>::subexpression(n**) da biblioteca **Regex** foi alterado para corresponder à sua documentação.
- O uso de modelos variádicos na biblioteca **Variant** pode quebrar as funções de metaprogramação.
- O **impulso::python::numeric** API foi removido. Os usuários podem usar **boost::python::numpy** em seu lugar.
- As operações aritméticas sobre indicações de tipos não-objetos não são mais fornecidas na biblioteca atômica.

(BZ#1494495)

Suporte Unicode 11.0.0

A biblioteca do núcleo C do Red Hat Enterprise Linux, **glibc**, foi atualizada para suportar a versão padrão Unicode 11.0.0. Como resultado, todas as APIs de caracteres amplos e multi-byte, incluindo transliteração e conversão entre conjuntos de caracteres, fornecem informações precisas e corretas em conformidade com este padrão.

(BZ#1512004)

O pacote de impulso é agora independente da Python

Com esta atualização, a instalação do pacote **boost** não mais instala a biblioteca **Boost.Python** como uma dependência. Para usar o **Boost.Python**, você precisa instalar explicitamente os pacotes **boost-python3** ou **boost-python3-devel**.

(BZ#1616244)

Um novo pacote compat-libgfortran-48 disponível

Para compatibilidade com as aplicações Red Hat Enterprise Linux 6 e 7 usando a biblioteca Fortran, um novo pacote de compatibilidade **compat-libgfortran-48** está agora disponível, que fornece a biblioteca **libgfortran.so.3**.

(BZ#1607227)

Suporte de retpolina no GCC

Esta atualização adiciona suporte para retpolinas ao GCC. Uma retpolina é uma construção de software usada pelo kernel para reduzir a sobrecarga dos ataques da Variante 2 do Spectre mitigadora descrita no CVE-2017-5715.

(BZ#1535774)

Suporte aprimorado para a arquitetura ARM de 64 bits em componentes da cadeia de ferramentas

Os componentes da cadeia de ferramentas, **GCC** e **binutils**, agora fornecem suporte estendido para a arquitetura ARM de 64 bits. Por exemplo:

- **GCC** e **binutils** agora suportam Extensão Vetorial Escalável (SVE).
- O suporte para o tipo de dados **FP16**, fornecido pela ARM v8.2, foi adicionado ao **GCC**. O tipo de dados **FP16** melhora o desempenho de certos algoritmos.
- As ferramentas do **binutils** agora suportam a definição da arquitetura ARM v8.3, incluindo a Autenticação do Ponteiro. O recurso de Autenticação do Ponteiro impede que o código malicioso corrompa a execução normal de um programa ou do kernel através da criação de seus próprios ponteiros de funções. Como resultado, somente endereços confiáveis são usados quando se ramifica para diferentes lugares no código, o que melhora a segurança.

(BZ#1504980, BZ#1550501, BZ#1504995, BZ#1504993, BZ#1504994)

Otimizações para a glibc para sistemas IBM POWER

Esta atualização fornece uma nova versão da **glibc** que é otimizada tanto para as arquiteturas IBM POWER 8 como para IBM POWER 9. Como resultado, os sistemas IBM POWER 8 e IBM POWER 9 agora mudam automaticamente para a variante adequada e otimizada **da glibc** em tempo de execução.

(BZ#1376834)

Biblioteca GNU C atualizada para a versão 2.28

O Red Hat Enterprise Linux 8 inclui a versão 2.28 da Biblioteca C GNU (glibc). Melhorias notáveis incluem:

- Características de endurecimento da segurança:
  - Os arquivos binários seguros marcados com a bandeira **AT_SECURE** ignoram a variável de ambiente **LD_LIBRARY_PATH**.
  - Os backtraces não são mais impressos por falhas na verificação de pilha para acelerar o desligamento e evitar a execução de mais código em um ambiente comprometido.
- Melhorias de desempenho:
  - O desempenho da função **malloc(**) foi melhorado com um cache local de rosca.
  - Adição da variável de ambiente **GLIBC_TUNABLES** para alterar as características de desempenho da biblioteca.
  - A implementação de semáforos de rosca foi melhorada e novas funções **pthread_rwlock_xxx()** escaláveis foram adicionadas.
  - O desempenho da biblioteca matemática foi melhorado.
- O suporte para Unicode 11.0.0 foi adicionado.
- Foi adicionado suporte melhorado para números de ponto flutuante de 128 bits, conforme definido pelas normas ISO/IEC/IEEE 60559:2011, IEEE 754-2008 e ISO/IEC TS 18661-3:2015.
- Melhorias no resolvedor de stub do Domain Name Service (DNS) relacionadas com o arquivo de configuração **/etc/resolv.conf**:

    - A configuração é automaticamente recarregada quando o arquivo é alterado.
    - Foi adicionado suporte para um número arbitrário de domínios de busca.
    - Foi adicionada uma seleção aleatória adequada para a opção de **rotação**.
- Novas características para o desenvolvimento foram adicionadas, inclusive:
    - Funções de invólucro do Linux para as chamadas de kernel **préadv2** e **pwritev2**
    - Novas funções incluindo **reallocarray()** e **explicit_bzero()**
    - Novas bandeiras para a função **posix_spawnattr_setflags()** tais como **POSIX_SPAWN_SETSID**

(BZ#1512010, BZ#1504125, BZ#506398)

CMake disponível em RHEL

O CMake build system versão 3.11 está disponível no Red Hat Enterprise Linux 8 como o pacote **cmake**.

(BZ#1590139, BZ#1502802)

fazer a versão 4.2.1

O Red Hat Enterprise Linux 8 é distribuído com a ferramenta **make** build versão 4.2.1. Mudanças notáveis incluem:

- Quando uma receita falha, o nome do makefile e o número da linha da receita são mostrados.
- A opção **--trace** foi adicionada para permitir o rastreamento de alvos. Quando esta opção é utilizada, toda receita é impressa antes da invocação mesmo que fosse suprimida, junto com o nome do arquivo e número da linha onde esta receita está localizada, e também com os pré-requisitos que fazem com que ela seja invocada.
- A mistura de regras explícitas e implícitas não faz mais com que **a** execução seja encerrada. Ao invés disso, um aviso é impresso. Note que esta sintaxe é depreciada e pode ser completamente removida no futuro.
- A função **$(file ...)** foi adicionada para escrever texto em um arquivo. Quando chamada sem um argumento de texto, ela só abre e fecha imediatamente o arquivo.
- Uma nova opção, **--output-sync** ou **-O**, faz com que uma saída de múltiplos trabalhos seja agrupada por trabalho e permite uma depuração mais fácil de construções paralelas.
- A opção **--debug** agora aceita também a bandeira **n** (nenhuma) para desativar todas as configurações de depuração atualmente habilitadas.
- O operador **!=** shell assignment operator foi adicionado como uma alternativa à função **$(shell ...**) para aumentar a compatibilidade com os arquivos BSD. Para mais detalhes e diferenças entre o operador e a função, veja o manual do GNU make.
  Note que, como conseqüência, variáveis com um nome terminando em ponto de exclamação e imediatamente seguidas por atribuição, como **variável!=valor**, são agora interpretadas como a nova sintaxe. Para restaurar o comportamento anterior, acrescente um espaço após o ponto de exclamação, tal como **variável! = valor**.
- O **::=** operador de atribuição definido pelo padrão POSIX foi adicionado.

- Quando a variável **.POSIX** for especificada **,** observe os requisitos padrão do POSIX para lidar com a contrabarra e a nova linha. Neste modo, qualquer espaço de fuga antes da contrabarra é preservado, e cada contrabarra seguida por uma nova linha e caracteres de espaço branco é convertida em um único caractere de espaço.

- O comportamento das variáveis **MAKEFLAGS** e **MFLAGS** está agora mais precisamente definido.

- Uma nova variável, **GNUMAKEFLAGS**, é analisada para **fazer** bandeiras de forma idêntica a **MAKEFLAGS**. Como conseqüência, as bandeiras **específicas do fabricante** GNU podem ser armazenadas fora de **MAKEFLAGS** e a portabilidade dos makefiles é aumentada.

- Uma nova variável, **MAKE_HOST**, contendo a arquitetura anfitriã, foi adicionada.

- As novas variáveis, **MAKE_TERMOUT** e **MAKE_TERMERR**, indicam se **a marca** está escrevendo saída padrão e erro em um terminal.

- A definição das opções **-r** e **-R** na variável **MAKEFLAGS** dentro de um makefile agora funciona corretamente e remove todas as regras e variáveis incorporadas, respectivamente.

- A configuração **.RECIPEPREFIX** é agora lembrada por receita. Além disso, as variáveis expandidas nessa receita também utilizam essa configuração de prefixo de receita.

- A configuração **.RECIPEPREFIX** e todas as variáveis específicas do alvo são exibidas na saída da opção **-p** como em um makefile, ao invés de como comentários.

(BZ#1641015)

SystemTap versão 4.0

O Red Hat Enterprise Linux 8 é distribuído com a ferramenta de instrumentação **SystemTap** versão 4.0. As melhorias notáveis incluem:

- O backend ampliado do Berkeley Packet Filter (eBPF) foi melhorado, especialmente as cordas e funções. Para utilizar este backend, comece **SystemTap** com a opção **--runtime=bpf**.

- Um novo serviço de rede de exportação para uso com o sistema de monitoramento Prometheus foi adicionado.

- A implementação da sonda de chamada do sistema foi melhorada para usar os pontos de rastreamento do núcleo, se necessário.

(BZ#1641032)

Melhorias na versão 2.30 do binutils

O Red Hat Enterprise Linux 8 inclui a versão 2.30 do pacote **binutils**. Melhorias notáveis incluem:

- O suporte para novas extensões da arquitetura IBM Z foi melhorado.

Linkers:

- O linker agora coloca os dados de código e somente leitura em segmentos separados por padrão. Como resultado, os arquivos executáveis criados são maiores e mais seguros para executar, pois o carregador dinâmico pode desativar a execução de qualquer página de memória que contenha dados somente leitura.

- Foi adicionado suporte às notas de propriedade GNU que fornecem dicas para o carregador dinâmico sobre o arquivo binário.
- Anteriormente, o linker gerava código executável inválido para a tecnologia Intel Indirect Branch Tracking (IBT). Como conseqüência, os arquivos executáveis gerados não podiam ser iniciados. Este erro foi corrigido.
- Anteriormente, o **gold** linker fundia notas de propriedade de forma imprópria. Como conseqüência, características erradas de hardware poderiam ser habilitadas no código gerado, e o código poderia terminar inesperadamente. Este erro foi corrigido.
- Anteriormente, o linker de **ouro** criava seções de notas com bytes de estofamento no final para alcançar o alinhamento de acordo com a arquitetura. Como o carregador dinâmico não esperava o acolchoamento, ele podia terminar inesperadamente o programa que estava carregando. Este bug foi corrigido.

Outras ferramentas:

- As ferramentas **readelf** e **objdump** agora têm opções para seguir links em arquivos de informação de depuração separados e exibir informações neles também.
- A nova opção **--inlines** amplia a opção existente **--line-numbers** da ferramenta **objdump** para exibir informações de aninhamento para funções inlined.
- A ferramenta **nm** ganhou uma nova opção **- com cordões de versão** para exibir informações da versão de um símbolo após seu nome, se presente.
- O suporte para a arquitetura ARMv8-R e processadores Cortex-R52, Cortex-M23 e Cortex-M33 foi adicionado à montadora.

(BZ#1641004, BZ#1637072, BZ#1501420, BZ#1504114, BZ#1614908, BZ#1614920)

Performance Co-Pilot versão 4.3.0

O Red Hat Enterprise Linux 8 é distribuído com **Performance Co-Pilot** (PCP) versão 4.3.0. As melhorias notáveis incluem:

- A ferramenta **pcp-dstat** agora inclui análise histórica e saída no formato Comma-separated Values (CSV).
- Os utilitários de registro podem usar etiquetas métricas e registros de texto de ajuda.
- A ferramenta **pmdaperfevent** agora informa os números de CPU corretos nos níveis mais baixos de SMT (Simultaneous Multi Threading).
- A ferramenta **pmdapostgresql** agora suporta **Postgres** série 10.x.
- A ferramenta **pmdaredis** agora suporta **Redis** série 5.x.
- A ferramenta **pmdabcc** foi aperfeiçoada com filtragem dinâmica de processos e syscalls, ucalls e ustat por processo.
- A ferramenta **pmdammv** agora exporta etiquetas métricas, e a versão de formato é aumentada para 3.
- A ferramenta **pmdagfs2** suporta métricas adicionais de glock e porta glock.
- Várias correções foram feitas na política da SELinux.

(BZ#1641034)

Chaves de proteção de memória

Esta atualização permite características de hardware que permitem mudanças na bandeira de proteção por página. Os novos invólucros de chamada do sistema **glibc** foram adicionados para as funções **pkey_alloc()**, **pkey_free()**, e **pkey_mprotect()**. Além disso, as funções **pkey_set(**) e **pkey_get(**) foram adicionadas para permitir o acesso aos sinalizadores de proteção por fio.

(BZ#1304448)

O GCC agora não cumpre a norma z13 na IBM Z

Com esta atualização, por padrão o GCC na arquitetura IBM Z constrói código para o processador z13, e o código é sintonizado para o processador z14. Isto é equivalente ao uso das opções **-march=z13** e **-mtune=z14**. Os usuários podem anular este padrão usando explicitamente as opções para a arquitetura e ajuste do alvo.

(BZ#1571124)

elfutils atualizado para a versão 0.174

No Red Hat Enterprise Linux 8, o pacote **elfutils** está disponível na versão 0.174. Mudanças notáveis incluem:

- Anteriormente, a ferramenta **eu-readelf** podia mostrar uma variável com um valor negativo como se tivesse um grande valor não assinado, ou mostrar um grande valor não assinado como um valor negativo. Isto foi corrigido e agora a ferramenta **eu-readelf** procura o tamanho e a assinatura de tipos de valores constantes para exibi-los corretamente.

- Uma nova função **dwarf_next_lines()** para leitura de dados **.debug_line** sem CU foi adicionada à biblioteca **libdw**. Esta função pode ser usada como alternativa às funções **dwarf_getsrclines()** e **dwarf_getsrcfiles()**.

- Anteriormente, arquivos com mais de 65280 seções podiam causar erros nas bibliotecas **libelf** e **libdw** e em todas as ferramentas que as utilizam. Este erro foi corrigido. Como resultado, os valores ampliados de **shnum** e **shstrndx** nos cabeçalhos dos arquivos ELF são tratados corretamente.

(BZ#1641007)

Valgrind atualizado para a versão 3.14

O Red Hat Enterprise Linux 8 é distribuído com a ferramenta de análise de código executável Valgrind versão 3.14. Mudanças notáveis incluem:

- Uma nova opção **-- keep-debuginfo** foi adicionada para permitir a retenção de informações de debug para código descarregado. Como resultado, os traços de pilha salvos podem incluir informações de arquivo e linha para código que não está mais presente na memória.

- Supressões baseadas no nome do arquivo fonte e número da linha foram adicionadas.

- A ferramenta **Helgrind** foi ampliada com uma opção **--delta-stacktrace** para especificar o cálculo de traços de pilha de histórico completo. Notavelmente, o uso desta opção junto com **--history-level=full** pode melhorar o desempenho **do Helgrind** em até 25%.

- A taxa falsa positiva na ferramenta **Memcheck** para otimizar o código nas arcitecturas Intel e AMD 64-bit e na arquitetura ARM 64-bit foi reduzida. Note que você pode usar os **controles**

de definição de custos para controlar o manuseio dos controles de definição e melhorar a taxa às custas do desempenho.

- A Valgrind pode agora reconhecer mais instruções da variante little-endian da IBM Power Systems.
- A Valgrind pode agora processar a maioria das instruções inteiras e vetoriais de string do processador IBM Z arquitetura z13.

Para mais informações sobre as novas opções e suas limitações conhecidas, consulte a página do manual **valgrind(1**).

(BZ#1641029, BZ#1501419)

GDB versão 8.2

O Red Hat Enterprise Linux 8 é distribuído com o depurador GDB versão 8.2 As mudanças notáveis incluem:

- O protocolo IPv6 é suportado para depuração remota com GDB e **gdbserver**.
- A depuração sem informações de depuração foi melhorada.
- O preenchimento de símbolos na interface do usuário GDB foi melhorado para oferecer melhores sugestões, utilizando mais construções sintáticas, tais como etiquetas ABI ou namespaces.
- Os comandos podem agora ser executados em segundo plano.
- Os programas de depuração criados na linguagem de programação Rust são agora possíveis.
- A depuração dos idiomas C e C foi melhorada com suporte de analisadores para os operadores **_Alignof** e **alignof**, referências de valor C e matrizes automáticas de comprimento variável C99.
- Os scripts de extensão da GDB podem agora usar a linguagem Guile scripting.
- A interface da linguagem Python para extensões foi melhorada com novas funções API, decoradores de quadros, filtros e desbobinadores. Além disso, os scripts na seção **.debug_gdb_scripts** da configuração GDB são carregados automaticamente.
- GDB agora usa Python versão 3 para executar seus scripts, incluindo impressoras bonitas, decoradores de quadros, filtros e desbobinadores.
- As arquiteturas ARM e ARM de 64 bits foram melhoradas com registro de execução e repetição de processos, incluindo Thumb 32 bits e instruções de chamada do sistema.
- A GDB agora suporta a Extensão Vetorial Escalável (SVE) na arquitetura ARM de 64 bits.
- Foi adicionado suporte para o registro Intel PKU e Intel Processor Trace.
- A funcionalidade de gravação e reprodução foi ampliada para incluir as instruções **rdrand** e **rdseed** nos sistemas baseados na Intel.
- A funcionalidade do GDB na arquitetura IBM Z foi ampliada com suporte para tracepoints e traçadores rápidos, registros vetoriais e ABI, e a chamada ao sistema **Catch**. Além disso, o GDB agora suporta instruções mais recentes da arquitetura.

- A GDB pode agora usar as sondas estáticas SystemTap do espaço do usuário (SDT) na arquitetura ARM de 64 bits.

(BZ#1641022, BZ#1497096, BZ#1505346, BZ#1592332, BZ#1550502)

a localização daglibc para a RHEL é distribuída em múltiplos pacotes

No RHEL 8, os locais e traduções **glibc** não são mais fornecidos pelo pacote único **glibc-comum**. Em vez disso, cada locale e idioma está disponível em um pacote **glibc-langpack-*CODE***. Além disso, na maioria dos casos, nem todos os locales são instalados por padrão, apenas estes selecionados no instalador. Os usuários devem instalar todos os pacotes de locale adicionais que precisam separadamente, ou se desejarem, podem instalar **glibc-all-langpacks** para obter o arquivo de locales contendo todos os locales da **glibc** instalados como antes.

Para mais informações, consulte Utilizando lancheiras.

(BZ#1512009)

GCC versão 8.2

No Red Hat Enterprise Linux 8, o conjunto de ferramentas GCC é baseado na série de lançamentos GCC 8.2. Mudanças notáveis incluem:

- Numerosas otimizações gerais foram adicionadas, tais como análise de alias, melhorias de vetorizadores, dobramento de código idêntico, análise inter-processal, passe de otimização de fusão de lojas, e outras.
- O endereço Sanitizer foi melhorado. O Higienizador de Vazamento e o Higienizador de Comportamento Indefinido foram adicionados.
- As informações de depuração podem agora ser produzidas no formato DWARF5. Esta capacidade é experimental.
- A ferramenta de análise de cobertura de código fonte GCOV foi ampliada com várias melhorias.
- Novos avisos e diagnósticos melhorados foram acrescentados para a detecção estática de mais erros de programação.
- O GCC foi ampliado para fornecer ferramentas para garantir o endurecimento adicional do código gerado. As melhorias relacionadas com a segurança incluem incorporações para verificação de transbordamento, proteção adicional contra choques de pilha, verificação de endereços-alvo de instruções de controle de fluxo, avisos para funções de manipulação de cordas delimitadas e avisos para detectar índices de matriz fora de limites.

As melhorias na arquitetura e no suporte do processador incluem:

- Várias novas opções específicas de arquitetura para a arquitetura Intel AVX-512, várias de suas microarquiteturas e Extensões de Proteção de Software Intel (SGX) foram adicionadas.
- A geração de códigos pode agora ter como alvo as extensões LSE de arquitetura ARM de 64 bits, ARMv8.2-A Extensões de ponto flutuante de 16 bits (FPE) e ARMv8.2-A, ARMv8.3-A, e ARMv8.4-A versões de arquitetura.
- Foi adicionado suporte para os processadores z13 e z14 da arquitetura IBM Z.

As mudanças notáveis relacionadas a idiomas e normas incluem:

- O padrão padrão usado na compilação de código na linguagem C mudou para C17 com extensões GNU.
- O padrão padrão usado na compilação de código na linguagem C mudou para C 14 com extensões GNU.
- A biblioteca de tempo de execução C agora suporta as normas C 11 e C 14.
- O compilador C agora implementa o padrão C 14.
- O suporte ao padrão de linguagem C11 foi melhorado.
- A nova extensão **__auto_ tipo** GNU C fornece um subconjunto da funcionalidade da palavra-chave C 11 **auto** na linguagem C.
- Os nomes dos tipos **_FloatN** e **_FloatNx** especificados pela norma ISO/IEC TS 18661-3:2015 são agora reconhecidos pelo front end C.
- Passar uma classe vazia como argumento agora não ocupa espaço nas arquiteturas Intel 64 e AMD64, como exigido pela plataforma ABI.
- O valor retornado pelo operador C 11 **alignof** foi corrigido para combinar com o operador C **_Alignof** e retornar alinhamento mínimo. Para encontrar o alinhamento preferido, use a extensão GNU **__alignof__**.
- A versão principal da biblioteca **libgfortran** para o código de linguagem Fortran foi alterada para 5.
- O suporte para os idiomas Ada (GNAT), GCC Go e Objective C/C foi removido. Use o conjunto de ferramentas Go para o desenvolvimento do código Go.

(JIRA:RHELPLAN-7437, BZ#1512593, BZ#1512378)

A biblioteca criptográfica Go modo FIPS agora honra as configurações do sistema

Anteriormente, a biblioteca criptográfica padrão Go sempre usava seu modo FIPS, a menos que estivesse explicitamente desativada no momento da construção da aplicação usando a biblioteca. Como conseqüência, os usuários de aplicações baseadas em Go- não podiam controlar se o modo FIPS era usado. Com esta mudança, a biblioteca não passa para o modo FIPS por padrão quando o sistema não está configurado no modo FIPS. Como resultado, os usuários de aplicações baseadas em Go- em sistemas RHEL têm mais controle sobre o uso do modo FIPS da biblioteca criptográfica Go.

(BZ#1633351)

strace atualizado para a versão 4.24

O Red Hat Enterprise Linux 8 é distribuído com a versão 4.24 da ferramenta **strace**. Mudanças notáveis incluem:

- Os recursos de adulteração de chamadas do sistema foram adicionados com a opção **-e inject=**. Isto inclui injeção de erros, valores de retorno, atrasos e sinais.
- A sintaxe de qualificação de chamadas do sistema foi melhorada:
  - A opção **-e trace=/regex** foi adicionada às chamadas do sistema de filtragem com expressões regulares.

    - A pré-condição de um ponto de interrogação para uma qualificação de chamada de sistema na opção **-e trace=** permite que **o strace** continue, mesmo que a qualificação não corresponda a nenhuma chamada de sistema.

    - A designação da personalidade foi acrescentada às qualificações da chamada ao sistema na opção **-e trace**.

- A descodificação da razão de saída do **kvm vcpu** foi adicionada. Para isso, use a opção **-e kvm=vcpu**.

- A biblioteca **libdw** de **elfutils** é agora usada para desenrolar a pilha quando a opção **-k** é usada. Além disso, o desmembramento de símbolos é realizado utilizando a biblioteca de **libreza**.

- Anteriormente, a opção **-r** fazia com que **o Strace** ignorasse a opção **-t**. Isto foi corrigido, e as duas opções agora são independentes.

- A opção **-A** foi adicionada para abrir arquivos de saída no modo anexo.

- A opção **-X** foi adicionada para configurar a formatação da saída **xlat**.

- A decodificação de endereços de soquetes com a opção **-yy** foi melhorada. Além disso, foi adicionada a impressão do número do dispositivo de bloco e caractere no modo **-yyy**.

- Agora é possível rastrear tanto os binários de 64 bits quanto os de 32 bits com uma única ferramenta de **cinta** na arquitetura IBM Z. Como consequência, o pacote separado da **strace32** não existe mais no RHEL 8.

Além disso, a decodificação dos seguintes itens foi adicionada, melhorada ou atualizada:

- protocolos, mensagens e atributos**da netlink**

- **arch_prctl**, **bpf**, **getsockopt**, **io_pgetevent**, **keyctl**, **prctl**, **pkey_alloc**, **pkey_free**, **pkey_mprotect**, **ptrace**, **rseq**, **setsockopt**, **socket**, **statx** e outras chamadas de sistema

- Comandos múltiplos para a chamada ao sistema **ioctl**

- Constantes de vários tipos

- Traçado de caminho para **execução**, **inotify_add_watch**, **inotify_init**, **select**, **symlink**, chamadas de sistema **symlinkat** e chamadas de sistema **mmap** com argumentos indiretos

- Listas de códigos de sinais

(BZ#1641014)

Conjuntos de ferramentas de compilação no RHEL 8

A RHEL 8.0 fornece os seguintes conjuntos de ferramentas de compilação como Fluxos de Aplicação:

- Clang e LLVM Toolset 7.0.1, que fornece a estrutura de infra-estrutura do compilador LLVM, o compilador Clang para os idiomas C e C, o depurador LLDB, e ferramentas relacionadas para análise de código. Veja o documento Utilizando o Clang e o conjunto de ferramentas LLVM .

- Rust Toolset 1.31, que fornece a linguagem de programação Rust **rustc do** compilador, a ferramenta de construção de **carga** e gerente de dependência, o plugin do **fornecedor de carga**, e as bibliotecas necessárias. Veja o documento Using Rust Toolset (Usando o Rust Toolset).

- Go Toolset 1.11.5, que fornece as ferramentas da linguagem de programação Go e bibliotecas. Go é conhecida alternativamente como **golang**. Veja o documento Using Go Toolset.

(BZ#1695698, BZ#1613515, BZ#1613516, BZ#1613518)

Implementações Java e ferramentas Java no RHEL 8

O repositório AppStream RHEL 8 inclui:

- Os pacotes **java-11-openjdk**, que fornecem o OpenJDK 11 Java Runtime Environment e o OpenJDK 11 Java Software Development Kit.
- Os pacotes **java-1.8.0-openjdk**, que fornecem o OpenJDK 8 Java Runtime Environment e o OpenJDK 8 Java Software Development Kit.
- Os pacotes **icedtea-web**, que fornecem uma implementação do Java Web Start.
- O módulo **formiga**, fornecendo uma biblioteca Java e uma ferramenta de linha de comando para compilar, montar, testar e executar aplicações Java. O **Ant** foi atualizado para a versão 1.10.
- O módulo **maven**, fornecendo uma ferramenta de gerenciamento e compreensão de projetos de software. **O maven** estava anteriormente disponível apenas como uma Coleção de Software ou no canal Opcional não suportado.
- O módulo **scala**, fornecendo uma linguagem de programação de propósito geral para a plataforma Java. **O Scala** estava anteriormente disponível apenas como uma Coleção de Software.

Além disso, os pacotes **java-1.8.0-ibm** são distribuídos através do repositório suplementar. Note que os pacotes neste repositório não são suportados pela Red Hat.

(BZ#1699535)

C ABI mudança em std::string e std::list

A Interface Binária de Aplicação (ABI) das classes **std::string** e **std::list** da biblioteca **libstdc** mudou entre a RHEL 7 (GCC 4.8) e a RHEL 8 (GCC 8) para estar em conformidade com o padrão C 11. A biblioteca **libstdc** suporta tanto a antiga como a nova ABI, mas algumas outras bibliotecas do sistema C não suportam. Como conseqüência, as aplicações que se ligam dinamicamente a estas bibliotecas precisarão ser reconstruídas. Isto afeta todos os modos padrão C, incluindo o C 98. Também afeta as aplicações construídas com compiladores Red Hat Developer Toolset para RHEL 7, que mantiveram a antiga ABI para manter a compatibilidade com as bibliotecas do sistema.

(BZ#1704867)

5.1.12. Sistemas de arquivo e armazenamento

Suporte para Integridade de Dados Campo / Extensão de Integridade de Dados (DIF/DIX)

O DIF/DIX é suportado em configurações onde o fornecedor de hardware o qualificou e fornece suporte total para o adaptador host bus (HBA) particular e configuração da matriz de armazenamento no RHEL.

O DIF/DIX não é suportado nas seguintes configurações:

- Não é suportado para uso no dispositivo de inicialização.
- Não é apoiado em convidados virtualizados.

- A Red Hat não suporta o uso da biblioteca de gerenciamento de armazenamento automático (ASMLib) quando o DIF/DIX é ativado.

O DIF/DIX é ativado ou desativado no dispositivo de armazenamento, o que envolve várias camadas até (e inclusive) a aplicação. O método para ativar o DIF nos dispositivos de armazenamento é dependente do dispositivo.

Para maiores informações sobre a característica DIF/DIX, veja O que é DIF/DIX .

(BZ#1649493)

XFS agora suporta extensões de dados compartilhados de cópia-em-escrita

O sistema de arquivo XFS suporta a funcionalidade compartilhada de cópia-em-escrita de dados. Esta funcionalidade permite que dois ou mais arquivos compartilhem um conjunto comum de blocos de dados. Quando um dos arquivos que compartilham blocos comuns muda, o XFS quebra o link para blocos comuns e cria um novo arquivo. Isto é similar à funcionalidade copy-on-write (COW) encontrada em outros sistemas de arquivos.

As extensões de dados compartilhados por meio de cópia-em-escrita são:

Rápido
A criação de cópias compartilhadas não utiliza a E/S em disco.

Eficiente em termos de espaço
Os blocos compartilhados não consomem espaço adicional em disco.

Transparente
Os arquivos que compartilham blocos comuns agem como arquivos regulares.

Os utilitários de espaço do usuário podem usar extensões de dados compartilhadas de cópia-em-escrita para:

- Clonagem eficiente de arquivos, como por exemplo com o comando **cp --reflink**
- Snapshots por arquivo

Esta funcionalidade também é utilizada pelos subsistemas de kernel como Overlayfs e NFS para uma operação mais eficiente.

As extensões de dados compartilhados copy-on-write agora são habilitadas por padrão ao criar um sistema de arquivos XFS, começando com o pacote **xfsprogs** versão **4.17.0-2.el8**.

Observe que os dispositivos de Acesso Direto (DAX) atualmente não suportam XFS com extensões de dados compartilhados de cópia-em-escrita. Para criar um sistema de arquivo XFS sem este recurso, use o seguinte comando:

```
# mkfs.xfs -m reflink=0 block-device
```

O Red Hat Enterprise Linux 7 pode montar sistemas de arquivo XFS com extensões de dados compartilhadas de cópia-em-escrita somente no modo somente-leitura.

(BZ#1494028)

O tamanho máximo do sistema de arquivo XFS é 1024 TiB

O tamanho máximo suportado de um sistema de arquivo XFS foi aumentado de 500 TiB para 1024 TiB.

Sistemas de arquivos maiores que 500 TiB exigem isso:

- o recurso CRC de metadados e o recurso inode btree gratuito estão ambos habilitados no formato de sistema de arquivos, e
- o tamanho do grupo de alocação é de pelo menos 512 GiB.

No RHEL 8, o utilitário **mkfs.xfs** cria sistemas de arquivo que atendem a esses requisitos por padrão.

O cultivo de um sistema de arquivo menor que não atende a estes requisitos para um novo tamanho maior que 500 TiB não é suportado.

(BZ#1563617)

o sistema de arquivoext4 agora suporta o checksum de metadados

Com esta atualização, os metadados **ext4** são protegidos por **checksums**. Isto permite que o sistema de arquivos reconheça os metadados corruptos, o que evita danos e aumenta a resiliência do sistema de arquivos.

(BZ#1695584)

VDO agora suporta todas as arquiteturas

O Virtual Data Optimizer (VDO) está agora disponível em todas as arquiteturas suportadas pela RHEL 8.

Para a lista de arquiteturas suportadas, ver ???.

(BZ#1534087)

O gerenciador de boot BOOM simplifica o processo de criação de entradas de boot

BOOM é um gerenciador de inicialização para sistemas Linux que utilizam carregadores de inicialização que suportam a especificação BootLoader para configuração de entrada de inicialização. Ele permite uma configuração de inicialização flexível e simplifica a criação de entradas de inicialização novas ou modificadas: por exemplo, para inicializar imagens instantâneas do sistema criado usando LVM.

BOOM não modifica a configuração existente do carregador de inicialização, e apenas insere entradas adicionais. A configuração existente é mantida, e qualquer integração de distribuição, como scripts de instalação e atualização do kernel, continua a funcionar como antes.

BOOM tem uma interface simplificada de linha de comando (CLI) e API que facilitam a tarefa de criar entradas de inicialização.

(BZ#1649582)

LUKS2 é agora o formato padrão para encriptar volumes

No RHEL 8, o formato LUKS versão 2 (LUKS2) substitui o formato antigo LUKS (LUKS1). O subsistema **dm-crypt** e a ferramenta **cryptsetup** agora usa o LUKS2 como o formato padrão para volumes criptografados. O LUKS2 fornece volumes criptografados com redundância de metadados e auto-recuperação no caso de um evento de corrupção parcial de metadados.

Devido ao layout flexível interno, LUKS2 é também um capacitador de características futuras. Ele suporta o desbloqueio automático através do token genérico de kernel-keyring construído em **libcryptsetup** que permite aos usuários desbloquear volumes LUKS2 usando uma frase-chave armazenada no serviço de retenção de kernel-keyring.

Outras melhorias notáveis incluem:

- A configuração da chave protegida usando o esquema de cifra de chave embrulhada.
- Integração mais fácil com a Decriptação Baseada em Políticas (Clevis).
- Até 32 ranhuras de chave - LUKS1 fornece apenas 8 ranhuras de chave.

Para mais detalhes, veja as páginas man **cryptsetup(8)** e **cryptsetup-reencrypt(8)**.

(BZ#1564540)

NVMe/FC é totalmente compatível com os adaptadores Broadcom Emulex e Marvell Qlogic Fibre Channel

O tipo de transporte NVMe sobre Fibre Channel (NVMe/FC) agora é totalmente suportado em modo iniciador quando usado com Broadcom Emulex e Marvell Qlogic Fibre Channel 32Gbit adaptadores que apresentam suporte NVMe.

O NVMe sobre canal de fibra é um tipo adicional de transporte de tecido para o protocolo Nonvolatile Memory Express (NVMe), além do protocolo Remote Direct Memory Access (RDMA), que foi introduzido anteriormente no Red Hat Enterprise Linux.

Habilitação de NVMe/FC:

- Para ativar o NVMe/FC no driver **lpfc**, edite o arquivo **/etc/modprobe.d/lpfc.conf** e adicione a seguinte opção:

```
lpfc_enable_fc4_type=3
```

- Para ativar o NVMe/FC no driver **qla2xxx**, edite o arquivo **/etc/modprobe.d/qla2xxx.conf** e adicione a seguinte opção:

```
qla2xxx.ql2xnvmeenable=1
```

Restrições adicionais:

- O Multipath não é suportado com NVMe/FC.
- O agrupamento NVMe não é suportado com NVMe/FC.
- **kdump** não é suportado com NVMe/FC.
- O Booting from Storage Area Network (SAN) NVMe/FC não é suportado.

(BZ#1649497)

Novo ajuste de configuração de scan_lvs

Uma nova configuração do arquivo de configuração **lvm.conf**, **scan_lvs**, foi adicionada e definida como 0 por padrão. O novo comportamento padrão impede o LVM de procurar PVs que possam existir em cima de LVs; ou seja, ele não irá procurar LVs ativos por mais PVs. A configuração padrão também impede a LVM de criar PVs em cima de LVs.

A colocação de PVs em camadas em cima de LVs pode ocorrer por meio de imagens de VM colocadas em cima de LVs, caso em que não é seguro para o host acessar os PVs. Evitar este acesso inseguro é a principal razão para o novo comportamento padrão. Além disso, em ambientes com muitos LVs ativos, a

quantidade de varredura do dispositivo feita pelo LVM pode ser significativamente reduzida.

O comportamento anterior pode ser restaurado alterando esta configuração para 1.

(BZ#1676598)

Nova seção de anulações do arquivo de configuração DM Multipath

O arquivo **/etc/multipath.conf** agora inclui uma seção de **anulações** que permite que você defina um valor de configuração para todos os seus dispositivos. Estes atributos são usados pela DM Multipath para todos os dispositivos, a menos que sejam sobrescritos pelos atributos especificados na seção **multipercursos** do arquivo **/etc/multipath.conf** para caminhos que contenham o dispositivo. Esta funcionalidade substitui o parâmetro **all_devs** da seção de **dispositivos** do arquivo de configuração, que não é mais suportado.

(BZ#1643294)

A instalação e inicialização a partir de dispositivos NVDIMM é agora suportada

Antes desta atualização, os dispositivos NVDIMM (Nonvolatile Dual Inline Memory Module) em qualquer modo eram ignorados pelo instalador.

Com esta atualização, as melhorias do kernel para suportar os dispositivos NVDIMM fornecem capacidades aprimoradas de desempenho do sistema e acesso melhorado ao sistema de arquivos para aplicações de gravação intensiva como banco de dados ou cargas de trabalho analíticas, bem como redução das despesas gerais da CPU.

Esta atualização introduz suporte para:

- O uso de dispositivos NVDIMM para instalação usando o comando **nvdimm** Kickstart e o GUI, tornando possível a instalação e inicialização a partir de dispositivos NVDIMM em modo setor e reconfigurar os dispositivos NVDIMM em modo setor durante a instalação.
- A extensão dos scripts de **Kickstart** para **Anaconda** com comandos para lidar com dispositivos NVDIMM.
- A capacidade do **grub2**, **efibootmgr** e dos componentes do sistema **efivar** de manusear e inicializar a partir de dispositivos NVDIMM.

(BZ#1499442)

A detecção de caminhos marginais em DM Multipath foi melhorada

O serviço **multipathd** agora suporta a melhor detecção de caminhos marginais. Isto ajuda os dispositivos multipath a evitar caminhos que podem falhar repetidamente, e melhora o desempenho. Os caminhos marginais são caminhos com erros de E/S persistentes mas intermitentes.

As seguintes opções no arquivo **/etc/multipath.conf** controlam o comportamento dos caminhos marginais:

- **percurso_muito_fracassado_tempo_ marginal**,
- **marginal_path_err_sample_time**,
- **caminho_marginal**, e
- **marginal_path_err_recheck_gap_time**.

DM Multipath desabilita um caminho e o testa com E/S repetidas para o tempo de amostragem configurado se:

- as opções de **multicaminhos.conf** listadas são definidas,
- um caminho falha duas vezes no tempo configurado, e
- outros caminhos estão disponíveis.

Se o caminho tiver mais do que a taxa de erro configurado durante este teste, a DM Multipath o ignora durante o tempo de intervalo configurado, e então o retesta para ver se está funcionando bem o suficiente para ser restabelecido.

Para mais informações, consulte a página de manual **multipath.conf**.

(BZ#1643550)

Programação de múltiplas filas em dispositivos de bloco

Os dispositivos de bloco agora usam o agendamento de múltiplas filas no Red Hat Enterprise Linux 8. Isto permite que o desempenho da camada de bloco seja bem dimensionado com drives de estado sólido rápido (SSDs) e sistemas multi-core.

Os programadores tradicionais, que estavam disponíveis na RHEL 7 e versões anteriores, foram removidos. A RHEL 8 suporta apenas programadores multi-funções.

(BZ#1647612)

5.1.13. Alta disponibilidade e clusters

Novos comandos pcs para listar os dispositivos de vigilância disponíveis e testar os dispositivos de vigilância

Para configurar a SBD com Pacemaker, é necessário um dispositivo de vigilância funcional. Esta versão suporta o comando **pcs stonith sbd watchdog list** para listar os dispositivos de watchdog disponíveis no nó local, e o comando **pcs stonith sbd watchdog test** comando para testar um dispositivo de watchdog. Para informações sobre a ferramenta de linha de comando **sbd**, veja a página de manual **sbd**(8).

(BZ#1578891)

O comando pcs agora suporta a filtragem de falhas de recursos por uma operação e seu intervalo

Pacemaker agora rastreia falhas de recursos por uma operação de recurso em cima de um nome de recurso, e um nó. O comando **pcs resource failcount show** agora permite filtrar as falhas por um recurso, nó, operação e intervalo. Ele fornece uma opção para exibir falhas agregadas por um recurso e nó ou detalhadas por um recurso, nó, operação e seu intervalo. Além disso, o comando de **limpeza de recursos pcs** agora permite filtrar falhas por um recurso, nó, operação e intervalo.

(BZ#1591308)

Timestamps ativados no registro corosync

O registro **corosync** não continha anteriormente carimbos temporais, o que dificultava sua relação com registros de outros nós e daemons. Com este lançamento, os timestamps estão presentes no log **corosync**.

(BZ#1615420)

Novos formatos para configuração de clusters de pcs, adição de nós de clusters de pcs e remoção de nós de clusters de pcs

No Red Hat Enterprise Linux 8, **os pcs** suportam totalmente Corosync 3, **knet**, e nomes de nós. Os nomes dos nós agora são necessários e substituem os endereços dos nós no papel de identificador do nó. Os endereços dos nós agora são opcionais.

- No comando **pcs host auth**, os endereços dos nós são os endereços padrão para os nomes dos nós.
- Na **configuração do cluster pcs** e no **nó de cluster pcs adicionar** comandos, os endereços dos nós são os endereços padrão para os endereços dos nós especificados no comando **auth do host pcs**.

Com estas mudanças, os formatos dos comandos para configurar um agrupamento, adicionar um nó a um agrupamento e remover um nó de um agrupamento foram alterados. Para obter informações sobre estes novos formatos de comando, veja a exibição de ajuda para a **configuração do cluster pcs**, **adicionar um nó de cluster pcs** e **remover** comandos do **nó de cluster pcs**.

(BZ#1158816)

Novos comandos pcs

O Red Hat Enterprise Linux 8 introduz os seguintes novos comandos.

- RHEL 8 introduz um novo comando, **pcs cluster node add-guest | remove-guest**, que substitui o **pcs cluster remoto-node add | remove** comando no RHEL 7.
- RHEL 8 introduz um novo comando, **pcs quorum unblock**, que substitui o comando **pcs cluster unblock quorum unblock** no RHEL 7.
- O comando de **reposição de recursos pcs** foi removido, pois duplica a funcionalidade do comando de **limpeza de recursos pcs**.
- RHEL 8 introduz novos comandos que substituem o comando **pcs resource [mostrar]** no RHEL 7:
  - O comando **pcs resource [status]** no RHEL 8 substitui o comando **pcs resource [show]** no RHEL 7.
  - O comando **pcs resource config** no RHEL 8 substitui o comando **pcs resource [show] --full** command no RHEL 7.
  - O comando **pcs resource config *resource id*** no RHEL 8 substitui o comando **pcs resource show *resource id*** no RHEL 7.
- RHEL 8 introduz novos comandos que substituem o comando **pcs stonith [show]** no RHEL 7:
  - O comando **pcs stonith [status]** no RHEL 8 substitui o comando **pcs stonith [show]** no RHEL 7.
  - O comando **pcs stonith config** no RHEL 8 substitui o comando **pcs stonith [show] --full** command no RHEL 7.
  - O comando **pcs stonith config *resource id*** no RHEL 8 substitui o comando **pcs stonith show *resource id*** no RHEL 7.

(BZ#1654280)

Pacemaker 2.0.0 em RHEL 8

Os pacotes de **marcapassos** foram atualizados para a versão upstream do Pacemaker 2.0.0, que fornece uma série de correções e melhorias de bugs em relação à versão anterior:

- O registro detalhado do Pacemaker é agora **/var/log/pacemaker/pacemaker.log** por padrão (não diretamente em **/var/log** ou combinado com o registro **corosync** em **/var/log/cluster**).
- Os processos Pacemaker daemon foram renomeados para tornar a leitura dos logs mais intuitiva. Por exemplo, o **daemon** foi renomeado para **"pacemaker-schedulerd"**.
- O suporte para o **padrão** depreciado **de recursos de recursos de origem duv** idosa e as propriedades de cluster de **inadimplência** foram abandonadas. A **pegajosidade dos recursos** e as propriedades **administradas** devem ser definidas nos padrões de recursos. As configurações existentes (embora não as recentemente criadas) com a sintaxe depreciada serão automaticamente atualizadas para usar a sintaxe suportada.
- Para uma lista mais completa de mudanças, veja a atualização do Pacemaker 2.0 no Red Hat Enterprise Linux 8.

É recomendado que os usuários que estão atualizando um cluster existente usando o Red Hat Enterprise Linux 7 ou anterior, executem **o cib-upgrade de cluster pcs** em qualquer nó de cluster antes e depois de atualizar o RHEL em todos os nós de cluster.

(BZ#1543494)

Recursos mestres renomeados para recursos de clonagem promocionais

O Red Hat Enterprise Linux (RHEL) 8 suporta Pacemaker 2.0, no qual um recurso mestre/escravo não é mais um tipo de recurso separado, mas um recurso clone padrão com um meta-atributo **promocional** definido como **verdadeiro**. As seguintes mudanças foram implementadas em apoio a esta atualização:

- Não é mais possível criar recursos mestres com o comando **pcs**. Ao invés disso, é possível criar recursos de clonagem **promocionais**. As palavras-chave e comandos relacionados foram alterados de **master** para **promotable**.
- Todos os recursos mestres existentes são exibidos como recursos de clonagem promocionais.
- Ao gerenciar um cluster RHEL7 na Web UI, os recursos mestre ainda são chamados de master, já que os clusters RHEL7 não suportam clones promovíveis.

(BZ#1542288)

Novos comandos para autenticar os nós em um cluster

O Red Hat Enterprise Linux (RHEL) 8 incorpora as seguintes mudanças nos comandos usados para autenticar os nós em um cluster.

- O novo comando para autenticação é **pcs host auth**. Este comando permite que os usuários especifiquem nomes de host, endereços e portas **pcsd**.
- O comando **auth do cluster pcs auth auth auth auth auth aut** h authenticates only the nodes in a local cluster and does not accept a node list

- Agora é possível especificar um endereço para cada nó. **pcs**/**pcsd** então se comunicará com cada nó usando o endereço especificado. Estes endereços podem ser diferentes daqueles que **a corosync** utiliza internamente.

- O comando **pcs pcsd clear-auth** foi substituído pelos comandos **pcsd deauth** e **pcs host deauth**. Os novos comandos permitem aos usuários deauthenticate um único host, bem como todos os hosts.

- Anteriormente, a autenticação do nó era bidirecional, e executar o comando **auth do cluster pcs** fazia com que todos os nós especificados fossem autenticados uns contra os outros. O comando **pcs host auth**, entretanto, faz com que apenas o host local seja autenticado contra os nós especificados. Isto permite um melhor controle de qual nó é autenticado contra quais outros nós ao executar este comando. Na própria configuração do cluster, e também ao adicionar um nó, **os pcs** sincronizam automaticamente os tokens no cluster, assim todos os nós no cluster ainda são automaticamente autenticados como antes e os nós do cluster podem se comunicar uns com os outros.

Observe que estas mudanças não são retrocompatíveis. Os nós que foram autenticados em um sistema RHEL 7 precisarão ser autenticados novamente.

(BZ#1549535)

Os comandos pcs agora suportam exibição, limpeza e sincronização do histórico das vedações

O daemon da cerca da Pacemaker segue um histórico de todas as ações de cerca tomadas (pendentes, bem sucedidas e fracassadas). Com este lançamento, os comandos **pcs** permitem aos usuários acessar o histórico das cercas das seguintes maneiras:

- O comando de **status pcs** mostra ações de esgrima falhadas e pendentes

- O comando **pcs status --full** mostra todo o histórico da esgrima

- O comando **pcs stonith history** fornece opções para exibir e limpar o histórico da esgrima

- Embora o histórico da esgrima seja sincronizado automaticamente, o comando **pcs stonith history** agora suporta uma opção de **atualização** que permite que o usuário sincronize manualmente o histórico da esgrima, caso seja necessário

(BZ#1620190, BZ#1615891)

5.1.14. Trabalho em rede

nftables substitui o iptables como a estrutura padrão de filtragem de pacotes de rede

A estrutura **nftables** oferece facilidades de classificação de pacotes e é o sucessor designado para as ferramentas **iptables**, **ip6tables**, **arptables** e **ebtables**. Ela oferece inúmeras melhorias em conveniência, características e desempenho em relação às ferramentas de filtragem de pacotes anteriores, mais notadamente:

- tabelas de pesquisa em vez de processamento linear

- uma estrutura única para ambos os protocolos **IPv4** e **IPv6**

- regras todas aplicadas atomicamente em vez de buscar, atualizar e armazenar um conjunto completo de regras

- suporte para depuração e rastreamento no conjunto de regras(**nftrace**) e monitoramento de eventos de rastreamento (na ferramenta **nft**)
- sintaxe mais consistente e compacta, sem extensões específicas de protocolo
- uma API Netlink para aplicações de terceiros

Da mesma forma que as **iptables**, **as nftables** utilizam tabelas para armazenar correntes. As cadeias contêm regras individuais para a realização de ações. A ferramenta **nft** substitui todas as ferramentas das estruturas de filtragem de pacotes anteriores. A biblioteca **libnftables** pode ser usada para interação de baixo nível com o **nftables** Netlink API sobre a biblioteca **libmnl**.

As ferramentas **iptables**, **ip6tables**, **ebtables** e **arptables** são substituídas por ferramentas drop-in baseadas em nftables com o mesmo nome. Enquanto o comportamento externo é idêntico ao de suas contrapartes legadas, internamente eles usam **nftables** com módulos **netfilter** kernel legados através de uma interface de compatibilidade onde for necessário.

O efeito dos módulos sobre o conjunto de regras **nftables** pode ser observado usando o comando **nft list ruleset**. Como estas ferramentas adicionam tabelas, correntes e regras ao conjunto de regras **nftables**, esteja ciente de que as operações do conjunto de regras **nftables**, tais como o comando **nft flush ruleset**, podem afetar os conjuntos de regras instalados usando os comandos herdados anteriormente separados.

Para identificar rapidamente qual variante da ferramenta está presente, as informações da versão foram atualizadas para incluir o nome do back-end. No RHEL 8, a ferramenta **iptables** baseada em nftables imprime a seguinte seqüência de versões:

```
$ iptables --version
iptables v1.8.0 (nf_tables)
```

Para comparação, a seguinte informação da versão é impressa se a ferramenta **iptables** legados estiver presente:

```
$ iptables --version
iptables v1.8.0 (legacy)
```

(BZ#1644030)

Características notáveis do TCP no RHEL 8

O Red Hat Enterprise Linux 8 é distribuído com a pilha de rede TCP versão 4.18, que oferece maior desempenho, melhor escalabilidade e maior estabilidade. Os desempenhos são aumentados especialmente para servidores TCP ocupados com uma alta taxa de conexão de entrada.

Além disso, dois novos algoritmos de congestionamento TCP, **BBR** e **NV**, estão disponíveis, oferecendo menor latência, e melhor rendimento do que o cúbico na maioria dos cenários.

(BZ#1562998)

firewalld usa nftables por padrão

Com esta atualização, o subsistema de filtragem **nftables** é o backend padrão de firewall para o daemon **firewalld**. Para alterar o backend, use a opção **FirewallBackend** no arquivo **/etc/firewalld/firewalld.conf**.

Esta mudança introduz as seguintes diferenças de comportamento ao utilizar **nftables**:

1. as execuções de regras**iptables** sempre ocorrem antes das regras **firewalld**
   - **DROP** em **iptables** significa que um pacote nunca é visto por **firewalld**
   - **ACCEPT** em **iptables** significa que um pacote ainda está sujeito às regras **firewalld**
2. as regras diretas**firewalld** ainda são implementadas através de **iptables** enquanto outros recursos **firewalld** utilizam **nftables**
3. a execução direta das regras ocorre antes da aceitação genérica das conexões estabelecidas pelo **firewalld**

(BZ#1509026)

Notável mudança no wpa_supplicant no RHEL 8

No Red Hat Enterprise Linux (RHEL) 8, o pacote **wpa_supplicant** é construído com **CONFIG_DEBUG_SYSLOG** habilitado. Isto permite ler o log **wpa_supplicant** usando o utilitário **journalctl** em vez de verificar o conteúdo do arquivo **/var/log/wpa_supplicant.log**.

(BZ#1582538)

NetworkManager agora suporta as funções virtuais SR-IOV

No Red Hat Enterprise Linux 8.0, **NetworkManager** permite configurar o número de funções virtuais (VF) para interfaces que suportam virtualização de E/S de raiz única (SR-IOV). Adicionalmente, **NetworkManager** permite configurar alguns atributos das VFs, tais como o endereço MAC, VLAN, a configuração de **verificação de spoof** e bitrates permitidos. Observe que todas as propriedades relacionadas à SR-IOV estão disponíveis na configuração de conexão **sriov**. Para mais detalhes, consulte a página de manual **nm-settings(5)**.

(BZ#1555013)

Os drivers de rede virtual IPVLAN são agora suportados

No Red Hat Enterprise Linux 8.0, o kernel inclui suporte a drivers de rede virtual IPVLAN. Com esta atualização, as placas de interface de rede virtual IPVLAN (NICs) permitem a conectividade de rede para múltiplos containers expondo um único endereço MAC à rede local. Isto permite que um único host tenha muitos containers superando a possível limitação do número de endereços MAC suportados pelo equipamento de rede peer.

(BZ#1261167)

NetworkManager suporta uma correspondência de nomes de interface wildcard para conexões

Anteriormente, era possível restringir uma conexão a uma determinada interface usando apenas uma correspondência exata no nome da interface. Com esta atualização, as conexões têm uma nova propriedade **match.interface-name** que suporta wildcards. Esta atualização permite aos usuários escolher a interface para uma conexão de uma forma mais flexível usando um padrão curinga.

(BZ#1555012)

Melhorias na pilha de rede 4.18

O Red Hat Enterprise Linux 8.0 inclui a pilha de rede atualizada para a versão upstream 4.18, que fornece várias correções e melhorias de bugs. Mudanças notáveis incluem:

- Introduziu novos recursos de descarga, tais como **UDP_GSO**, e, para alguns drivers de dispositivos, **GRO_HW**.
- Melhoria significativa da escalabilidade para o Protocolo de Datagramas de Usuário (UDP).
- Melhorou o código genérico das pesquisas de opinião pública ocupadas.
- Melhoria da escalabilidade para o protocolo IPv6.
- Melhor escalabilidade para o código de roteamento.
- Adicionado um novo algoritmo padrão de programação de fila de transmissão**,fq_codel**, que melhora um atraso na transmissão.
- Melhoria da escalabilidade para alguns algoritmos de programação de filas de transmissão. Por exemplo, o **pfifo_fast** está agora sem lockless.
- Melhor escalabilidade da unidade de remontagem IP pela remoção da rosca do núcleo de coleta de lixo e fragmentos de IP expiram apenas no tempo limite. Como resultado, o uso da CPU sob DoS é muito menor, e a taxa máxima de queda de fragmentos sustentável é limitada pela quantidade de memória configurada para a unidade de remontagem IP.

(BZ#1562987)

Novas ferramentas para converter iptables em nftables

Esta atualização adiciona as ferramentas **iptables-translate** e **ip6tables-translate** para converter as regras **iptables** ou **ip6tables** existentes nas regras equivalentes para **nftables**. Note que algumas extensões carecem de suporte à tradução. Se tal extensão existir, a ferramenta imprime a regra não traduzida prefixada com o sinal **#**. Por exemplo:

```
| % iptables-translate -A INPUT -j CHECKSUM --checksum-fill
| nft # -A INPUT -j CHECKSUM --checksum-fill
```

Além disso, os usuários podem usar as ferramentas **iptables-restore-translate** e **ip6tables-restore-translate** para traduzir um lixão de regras. Note que antes disso, os usuários podem usar os comandos **iptables-save** ou **ip6tables-save** para imprimir um dump das regras atuais. Por exemplo:

```
| % sudo iptables-save >/tmp/iptables.dump
| % iptables-restore-translate -f /tmp/iptables.dump
| # Translated by iptables-restore-translate v1.8.0 on Wed Oct 17 17:00:13 2018
| add table ip nat
| ...
```

(BZ#1564596)

Novos recursos adicionados à VPN usando NetworkManager

No Red Hat Enterprise Linux 8.0, **NetworkManager** fornece os seguintes novos recursos para VPN:

- Suporte para o protocolo Internet Key Exchange versão 2 (IKEv2).
- Acrescentou mais algumas opções **Libreswan**, tais como as opções **direita, esquerdina**, **estreitamento**, **rechave**, **fragmentação**. Para mais detalhes sobre as opções suportadas, veja a página do homem **nm-settings-libreswan**.
- Atualizou as cifras padrão. Isto significa que quando o usuário não especifica as cifras, o plugin

NetworkManager-libreswan permite que a aplicação **Libreswan** escolha a cifra padrão do sistema. A única exceção é quando o usuário seleciona uma configuração de modo agressivo IKEv1. Neste caso, os valores **ike = aes256-sha1;modp1536** e **eps = aes256-sha1** são passados para **Libreswan**.

(BZ#1557035)

Um novo tipo de pedaço de dados, I-DATA, adicionado ao SCTP

Esta atualização acrescenta um novo tipo de pedaço de dados, **I-DATA**, e programadores de fluxo ao Stream Control Transmission Protocol (SCTP). Anteriormente, o SCTP enviava mensagens de usuário na mesma ordem em que eram enviadas por um usuário. Consequentemente, uma grande mensagem de usuário SCTP bloqueou todas as outras mensagens em qualquer stream até que fossem completamente enviadas. Ao utilizar os pedaços de **I-DATA**, o campo Número de Seqüência de Transmissão (TSN) não está sobrecarregado. Como resultado, o SCTP agora pode programar os fluxos de diferentes maneiras, e o **I-DATA** permite a intercalação de mensagens do usuário (RFC 8260). Note que ambos os pares devem suportar o tipo **I-DATA** chunk.

(BZ#1273139)

ONetworkManager suporta a configuração de recursos de descarga de etool

Com este aprimoramento, o **NetworkManager** suporta a configuração de recursos de descarga de **etool**, e os usuários não precisam mais usar scripts de inicialização ou um script de despacho do **NetworkManager**. Como resultado, os usuários podem agora configurar o recurso de descarregar como parte do perfil de conexão usando um dos seguintes métodos:

- Usando a utilidade **nmcli**
- Editando arquivos-chave no diretório **/etc/NetworkManager/system-connections/**
- Editando os arquivos **/etc/sysconfig/network-scripts/ifcfg-***

Note que este recurso não é atualmente suportado em interfaces gráficas e no utilitário **nmtui**.

(BZ#1335409)

Suporte TCP BBR em RHEL 8

Um novo algoritmo de controle de congestionamento TCP, largura de banda de gargalo e tempo de ida e volta (BBR) é agora suportado no Red Hat Enterprise Linux (RHEL) 8. BBR tenta determinar a largura de banda do link de gargalo e o tempo de viagem de ida e volta (RTT). A maioria dos algoritmos de congestionamento é baseada na perda de pacotes (incluindo CUBIC, o algoritmo padrão de controle de congestionamento TCP do Linux), que tem problemas em links de alta produtividade. O BBR não reage diretamente a eventos de perda, ele ajusta a taxa de pacotes TCP para corresponder com a largura de banda disponível. Os usuários do TCP BBR devem mudar para a configuração de fila **fq** em todas as interfaces envolvidas.

Note que os usuários devem usar explicitamente **o fq** e não o **fq_codel**.

Para mais detalhes, consulte a página **tc-fq** man.

(BZ#1515987)

lksctp-tools, versão 1.0.18 em RHEL 8

O pacote **lksctp-tools**, versão 3.28 está disponível no Red Hat Enterprise Linux (RHEL) 8. Melhorias notáveis e correções de bugs incluem:

- Integração com Travis CI e Coverity Scan
- Suporte para a função **sctp_peeloff_flags**
- Indicação de quais características do núcleo estão disponíveis
- Fixa em questões de Coverity Scan

(BZ#1568622)

Colocação do módulo SCTP na lista negra por padrão no RHEL 8

Para aumentar a segurança, um conjunto de módulos do kernel foi movido para o pacote **kernel-modules-extra**. Estes não são instalados por padrão. Como conseqüência, os usuários não root não podem carregar estes componentes, pois eles estão na lista negra por padrão. Para usar um destes módulos do kernel, o administrador do sistema deve instalar **os módulos do kernel-modules-extra** e remover explicitamente a lista negra de módulos. Como resultado, os usuários não root poderão carregar o componente de software automaticamente.

(BZ#1642795)

Mudanças notáveis no driverctl 0,101

O Red Hat Enterprise Linux 8.0 é distribuído com **driverctl** 0.101. Esta versão inclui as seguintes correções de erros:

- As advertências do **shellcheck** foram fixadas.
- O bash-completion é instalado como **driverctl** em vez de **driverctl-bash-completion.sh.**
- A função **load_override** para ônibus não-PCI foi fixada.
- O serviço de **motorista** carrega todas as anulações antes de atingir o alvo do sistema **.básico.alvo.**

(BZ#1648411)

Acrescentou prioridades ricas em regras ao firewalld

A opção **prioritária** foi acrescentada às ricas regras. Isto permite aos usuários definir a ordem de prioridade desejável durante a execução das regras e proporciona um controle mais avançado sobre as regras ricas.

(BZ#1648497)

NVMe sobre RDMA é suportado no RHEL 8

No Red Hat Enterprise Linux (RHEL) 8, o Nonvolatile Memory Express (NVMe) sobre Remote Direct Memory Access (RDMA) suporta Infiniband, RoCEv2, e iWARP somente no modo iniciador.

Note que a Multipath é suportada apenas no modo failover.

Restrições adicionais:

- A Kdump não é suportada com NVMe/RDMA.
- A inicialização a partir do dispositivo NVMe sobre RDMA não é suportada.

(BZ#1680177)

O back end nf_tables não suporta depuração usando o dmesg

O Red Hat Enterprise Linux 8.0 usa o back end **nf_tables** para firewalls que não suportam a depuração do firewall usando a saída do utilitário **dmesg**. Para depurar regras de firewall, use os comandos **xtables-monitor -t** ou **nft monitor trace** para decodificar os eventos de avaliação de regras.

(BZ#1645744)

O Red Hat Enterprise Linux suporta o VRF

O kernel no RHEL 8.0 suporta encaminhamento e encaminhamento virtual (VRF). Os dispositivos VRF, combinados com o conjunto de regras definidas usando o utilitário **ip**, permitem aos administradores criar domínios VRF na pilha da rede Linux. Estes domínios isolam o tráfego na camada 3 e, portanto, o administrador pode criar diferentes tabelas de roteamento e reutilizar os mesmos endereços IP dentro de diferentes domínios VRF em um host.

(BZ#1440031)

iproute, versão 4.18 em RHEL 8

O pacote **iproute** é distribuído com a versão 4.18 no Red Hat Enterprise Linux (RHEL) 8. A mudança mais notável é que a interface marcada como ethX:Y, tal como eth0:1, não é mais suportada. Para contornar este problema, os usuários devem remover o sufixo do alias, que é os dois pontos e o seguinte número antes de entrar no **ip link show**.

(BZ#1589317)

5.1.15. Segurança

Etiqueta SWID da versão RHEL 8.0

Para permitir a identificação das instalações RHEL 8.0 usando o mecanismo ISO/IEC 19770-2:2015, as etiquetas de identificação de software (SWID) são instaladas em arquivos **/usr/lib/swidtag/redhat**. **com/com.redhat.RHEL-8-<arquitetura>.swidtag** e **/usr/lib/swidtag/redhattag.com/com.redhat.RHEL-8.0-<arquitetura>.swidtag**. O diretório pai destas tags também pode ser encontrado seguindo o link simbólico **/etc/swid/swidtags.d/redhat.com**.

A assinatura XML dos arquivos de tags SWID pode ser verificada usando o comando **xmlsec1 verify**, por exemplo:

```
xmlsec1 verify --trusted-pem /etc/pki/swid/CA/redhatcodesignca.cert
/usr/share/redhat.com/com.redhat.RHEL-8-x86_64.swidtag
```

O certificado da autoridade de certificação de assinatura do código também pode ser obtido na página Chaves de Assinatura do Produto no Portal do Cliente.

(BZ#163636338)

As políticas criptográficas de todo o sistema são aplicadas por padrão

Crypto-policies é um componente do Red Hat Enterprise Linux 8, que configura os subsistemas criptográficos centrais, cobrindo os protocolos TLS, IPsec, DNSSEC, Kerberos e SSH. Ele fornece um pequeno conjunto de políticas, que o administrador pode selecionar usando o comando **update-crypto-policies**.

A política de criptografia do sistema **DEFAULT** oferece configurações seguras para os atuais modelos de ameaça. Ela permite os protocolos TLS 1.2 e 1.3, assim como os protocolos IKEv2 e SSH2. As chaves RSA e os parâmetros Diffie-Hellman são aceitos se forem maiores que 2047 bits.

Veja o artigo Segurança consistente por políticas criptográficas no Red Hat Enterprise Linux 8 no Blog da Red Hat e a página de manual **update-crypto-policies(8)** para mais informações.

(BZ#1591620)

OpenSSH rebaseado para a versão 7.8p1

Os pacotes **openssh** foram atualizados para a versão upstream 7.8p1. Mudanças notáveis incluem:

- Removido o suporte para o protocolo **SSH versão 1**.
- Removido o suporte para o código de autenticação de mensagem **hmac-ripemd160**.
- Removido o suporte para as cifras RC4**(arcfour**).
- Removido o suporte para as cifras **Blowfish**.
- Removido o suporte para as cifras **CAST**.
- Alterou o valor padrão da opção **UseDNS** para **não**.
- Algoritmos de chave pública **DSA** desativados por padrão.
- Mudou o tamanho mínimo do módulo para parâmetros **Diffie-Hellman** para 2048 bits.
- Mudança na semântica da opção de configuração **ExposeAuthInfo**.
- A opção **UsePrivilegeSeparation=sandbox** é agora obrigatória e não pode ser desativada.
- Ajuste o tamanho mínimo aceito da chave **RSA** para 1024 bits.

(BZ#1622511)

A geração automática de chaves de servidor OpenSSH é agora tratada pela sshd-keygen@.service

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 **sshd-keygen@.service** instantiated.

Por exemplo, para desativar a criação automática do tipo de chave RSA:

```
# máscara systemctl sshd-keygen@rsa.service
```

Consulte o arquivo /**etc**/**sysconfig**/**sshd** para obter mais informações.

(BZ#1228088)

As chaves ECDSA são suportadas para autenticação SSH

Este lançamento da suíte **OpenSSH** introduz suporte para chaves ECDSA armazenadas em Cartões Smart Card PKCS #11. Como resultado, os usuários podem agora usar ambas as chaves RSA e ECDSA para autenticação SSH.

(BZ#1645038)

a libssh implementa o SSH como um componente criptográfico central

Esta mudança introduz a **libssh** como um componente criptográfico central no Red Hat Enterprise Linux 8. A biblioteca **libssh** implementa o protocolo Secure Shell (SSH).

Note que o lado cliente da **libssh** segue a configuração definida para **OpenSSH** através de políticas de criptografia em todo o sistema, mas a configuração do lado servidor não pode ser alterada através de políticas de criptografia em todo o sistema.

(BZ#1485241)

TLS 1.3 suporte em bibliotecas criptográficas

Esta atualização permite a Segurança da Camada de Transporte (TLS) 1.3 por padrão em todas as principais bibliotecas de criptografia back-end. Isto permite baixa latência em toda a camada de comunicação do sistema operacional e aumenta a privacidade e segurança das aplicações, aproveitando novos algoritmos, como o RSA-PSS ou X25519.

(BZ#1516728)

NSS agora usa SQL por padrão

As bibliotecas dos Network Security Services (NSS) agora usam o formato de arquivo SQL para o banco de dados de confiança por padrão. O formato de arquivo DBM, que era usado como formato padrão de banco de dados em versões anteriores, não suporta acesso simultâneo ao mesmo banco de dados por vários processos e tem sido depreciado no upstream. Como resultado, aplicações que usam o banco de dados de confiança NSS para armazenar chaves, certificados e informações de revogação agora criam bancos de dados no formato SQL por padrão. Tentativas de criar bancos de dados no formato DBM antigo falham. Os bancos de dados DBM existentes são abertos em modo somente leitura, e são automaticamente convertidos para o formato SQL. Note que o NSS suporta o formato de arquivo SQL desde o Red Hat Enterprise Linux 6.

(BZ#1489094)

O suporte PKCS #11 para Cartões Smart Card e HSMs é agora consistente em todo o sistema

Com esta atualização, o uso de cartões inteligentes e Módulos de Segurança de Hardware (HSM) com a interface criptográfica PKCS #11 se torna consistente. Isto significa que o usuário e o administrador podem usar a mesma sintaxe para todas as ferramentas relacionadas no sistema. As melhorias notáveis incluem:

- Suporte para o esquema PKCS #11 Uniform Resource Identifier (URI) que garante uma habilitação simplificada de tokens em servidores RHEL tanto para administradores quanto para escritores de aplicativos.
- Um método de registro em todo o sistema para Cartões Smart Card e HSMs usando o **pkcs11.conf**.
- Suporte consistente para HSMs e cartões inteligentes está disponível nas aplicações NSS, GnuTLS e OpenSSL (através do motor **openssl-pkcs11**).
- O servidor HTTP Apache**(httpd**) agora suporta HSMs sem problemas.

Para mais informações, consulte a página de manual **pkcs11.conf(5)**.

(BZ#1516741)

Firefox agora funciona com drivers PKCS #11 registrados em todo o sistema

O navegador Firefox carrega automaticamente o módulo **p11-kit-proxy** e cada cartão inteligente registrado em todo o sistema **p11-kit** através do arquivo **pkcs11.conf** é detectado automaticamente. Para usar a autenticação do cliente TLS, nenhuma configuração adicional é necessária e as chaves de um smart card são automaticamente usadas quando um servidor as solicita.

(BZ#1595638)

RSA-PSS é agora suportado no OpenSC

Esta atualização adiciona suporte ao esquema de assinatura criptográfica RSA-PSS ao condutor do Cartão Smart Card **OpenSC**. O novo esquema permite um algoritmo criptográfico seguro necessário para o suporte ao TLS 1.3 no software do cliente.

(BZ#1595626)

Mudanças notáveis em Libreswan no RHEL 8

Os pacotes **da libreswan** foram atualizados para a versão upstream 3.27, que fornece muitas correções e melhorias em relação às versões anteriores. As mudanças mais notáveis incluem:

- Suporte para RSA-PSS (RFC 7427) através de **authby=rsa-sha2**, ECDSA (RFC 7427) através de **authby=ecdsa-sha2**, CURVE25519 usando a palavra-chave **dh31**, e CHACHA20-POLY1305 para IKE e ESP através da palavra-chave de criptografia **chacha20_poly1305** foi adicionada para o protocolo IKEv2.

- O suporte para o módulo KLIPS alternativo foi removido de **Libreswan**, uma vez que o KLIPS foi totalmente depreciado a montante.

- Os grupos Diffie-Hellman DH22, DH23 e DH24 não são mais suportados (de acordo com o RFC 8247).

Observe que o **authby=rsasig** foi alterado para usar sempre o método RSA v1.5, e a opção **authby=rsa-sha2** usa o método RSASSA-PSS. A opção **authby=rsa-sha1** não é válida de acordo com o RFC 8247. Esta é a razão pela qual **Libreswan** não suporta mais o SHA-1 com assinaturas digitais.

(BZ#1566574)

Políticas criptográficas de todo o sistema mudam a versão padrão do IKE em Libreswan para IKEv2

A versão padrão do IKE na implementação do Libreswan IPsec foi alterada de IKEv1 (RFC 2409) para IKEv2 (RFC 7296). Os algoritmos padrão IKE e ESP/AH para uso com IPsec foram atualizados para atender às políticas de criptografia do sistema, RFC 8221, e RFC 8247. Os tamanhos de chave de criptografia de 256 bits são agora preferidos em relação aos tamanhos de chave de 128 bits.

As cifras padrão IKE e ESP/AH agora incluem AES-GCM, CHACHA20POLY1305, e AES-CBC para criptografia. Para verificação de integridade, elas fornecem AEAD e SHA-2. Os grupos Diffie-Hellman agora contêm DH19, DH20, DH21, DH14, DH15, DH16, e DH18.

Os seguintes algoritmos foram removidos das políticas padrão IKE e ESP/AH: AES_CTR, 3DES, SHA1, DH2, DH5, DH22, DH23, e DH24. Com exceção do DH22, DH23 e DH24, estes algoritmos podem ser habilitados pela opção **ike=** ou **phase2alg=/esp=/ah=** nos arquivos de configuração IPsec.

Para configurar conexões IPsec VPN que ainda requerem o protocolo IKEv1, adicione o **ikev2=sem** opção aos arquivos de configuração de conexão. Veja a página de manual **ipsec.conf(5)** para mais informações.

(BZ#1645606)

Mudanças relacionadas à versão IKE em Libreswan

Com esta melhoria, Libreswan lida com as configurações de troca de chaves da Internet (IKE) de maneira diferente:

- A versão padrão da troca de chaves da Internet (IKE) foi alterada de 1 para 2.
- As conexões agora podem usar o protocolo IKEv1 ou IKEv2, mas não ambos.
- A interpretação da opção **ikev2** foi alterada:
  - Os valores **insistem** é interpretado como IKEv2 apenas.
  - Os valores **não** e **nunca** são interpretados apenas como IKEv1.
  - Os valores **propostos**, **sim** e, **permissão** não são mais válidos e resultam em um erro, pois não ficou claro quais versões do IKE resultaram desses valores

(BZ#1648776)

Novas características no OpenSCAP em RHEL 8

O conjunto **OpenSCAP** foi atualizado para a versão 1.3.0, que introduz muitas melhorias em relação às versões anteriores. As características mais notáveis incluem:

- API e ABI foram consolidadas - os símbolos atualizados, depreciados e/ou não utilizados foram removidos.
- As sondas não são executadas como processos independentes, mas como roscas dentro do processo **oscap**.
- A interface da linha de comando foi atualizada.
- As amarrações**Python 2** foram substituídas pelas **Python 3** bindings.

(BZ#1614273)

OGuia de Segurança SCAP agora suporta políticas criptográficas de todo o sistema

Os pacotes **scap-security-guide** foram atualizados para usar políticas criptográficas predefinidas em todo o sistema para configurar os subsistemas criptográficos centrais. O conteúdo de segurança que conflitava com as políticas criptográficas de todo o sistema, ou que as ultrapassava, foi removido.

Observe que esta mudança se aplica somente ao conteúdo de segurança no **scap-security-guide**, e você não precisa atualizar o scanner OpenSCAP ou outros componentes SCAP.

(BZ#1618505)

A interface de linha de comando OpenSCAP foi melhorada

O modo verboso está agora disponível em todos os módulos e submódulos **oscap**. A saída da ferramenta melhorou a formatação.

As opções desvalorizadas foram removidas para melhorar a usabilidade da interface da linha de comando.

As seguintes opções não estão mais disponíveis:

- -**-show** in **oscap xccdf gerar relatório** foi completamente removido.
- -**-probe-root** em **avaliação oval oscap** foi removido. Ela pode ser substituída pela configuração da variável de ambiente, **OSCAP_PROBE_ROOT**.
- -**-sce-resultados** em **oscap xccdf eval** foi substituído por **--check-engine-results**
- o submódulo**validate-xml** foi descartado dos módulos CPE, OVAL, e XCCDF. submódulos **validate** podem ser usados para validar o conteúdo SCAP contra esquemas XML e XSD schematrons.
- **o** comando**oscap - oval list-probes** foi removido, a lista de sondas disponíveis pode ser exibida usando **oscap --version**.

OpenSCAP permite avaliar todas as regras em um determinado benchmark XCCDF, independentemente do perfil, usando o **"(all)" --profile '(all)'**.

(BZ#1618484)

Guia de Segurança SCAP O perfil PCI-DSS alinha-se com a versão 3.2.1

Os pacotes **scap-security-guide** fornecem o perfil PCI-DSS (Payment Card Industry Data Security Standard) para o Red Hat Enterprise Linux 8 e este perfil foi atualizado para se alinhar com a última versão do PCI-DSS - 3.2.1.

(BZ#1618528)

O Guia de Segurança SCAP suporta OSPP 4.2

Os pacotes **scap-security-guide** fornecem um rascunho do perfil OSPP (Protection Profile for General Purpose Operating Systems) versão 4.2 para o Red Hat Enterprise Linux 8. Este perfil reflete os controles de configuração obrigatórios identificados no Anexo de Configuração NIAP do Perfil de Proteção para Sistemas Operacionais de Propósito Geral (Protection Profile Version 4.2). O Guia de Segurança SCAP fornece verificações e scripts automatizados que ajudam os usuários a atender os requisitos definidos no OSPP.

(BZ#1618518)

Mudanças notáveis no rsyslog no RHEL 8

Os pacotes **rsyslog** foram atualizados para a versão upstream 8.37.0, que fornece muitas correções de bugs e melhorias em relação às versões anteriores. As mudanças mais notáveis incluem:

- Melhor processamento das mensagens internas **rsyslog**; possibilidade de limitação da taxa; fixação de um possível impasse.
- Limitação de taxas melhorada em geral; o atual *spam source* está agora registrado.
- Melhor manuseio de mensagens superdimensionadas - o usuário pode agora definir como tratá-las tanto no núcleo quanto em certos módulos com ações separadas.
- as bases de regras**mmnormalize** agora podem ser embutidas no arquivo **de configuração** em vez de criar arquivos separados para elas.

- Todas as variáveis de **configuração**, incluindo variáveis no JSON, são agora não sensíveis a maiúsculas e minúsculas.
- Várias melhorias na saída do PostgreSQL.
- Acrescentou a possibilidade de usar variáveis shell para controlar o processamento da **configuração**, como carregamento condicional de arquivos de configuração adicionais, execução de declarações, ou inclusão de um texto na **configuração**. Note que um uso excessivo deste recurso pode tornar muito difícil a depuração de problemas com **rsyslog**.
- Os modos de criação de arquivos de 4 dígitos podem agora ser especificados na **configuração**.
- A entrada do Protocolo de Registro de Eventos Confiáveis (RELP) agora também pode ser vinculada apenas em um endereço especificado.
- O valor padrão da opção **enable.body** da saída de correio está agora alinhado à documentação
- O usuário pode agora especificar códigos de erro de inserção que devem ser ignorados na saída de **MongoDB**.
- A entrada TCP paralela (pTCP) tem agora o backlog configurável para um melhor balanceamento de carga.
- Para evitar a duplicação de registros que poderiam aparecer quando **o journald** rotacionava seus arquivos, a opção **imjournal** foi adicionada. Observe que o uso desta opção pode afetar o desempenho.

Observe que o sistema com **rsyslog** pode ser configurado para proporcionar melhor desempenho conforme descrito no artigo Configuring system logging without journald or with minimized journald use Knowledgebase article.

(BZ#1613880)

Novo módulo rsyslog: omkafka

Para ativar os cenários de armazenamento de dados centralizado **kafka**, agora você pode encaminhar os logs para a infra-estrutura **kafka** usando o novo módulo **omkafka**.

(BZ#1542497)

rsyslog imfile agora suporta symlinks

Com esta atualização, o módulo **imfile rsyslog** oferece melhor desempenho e mais opções de configuração. Isto permite que você utilize o módulo para casos de uso mais complicado de monitoramento de arquivos. Por exemplo, agora você pode usar monitores de arquivos com padrões globais em qualquer lugar ao longo do caminho configurado e rotacionar alvos de links simbólicos com maior produção de dados.

(BZ#1614179)

O formato padrão do arquivo de configuração rsyslog é agora não-legacy

Os arquivos de configuração nos pacotes **rsyslog** agora usam o formato não legado por padrão. O formato legado ainda pode ser usado, no entanto, misturar declarações de configuração atuais e legados tem várias restrições. As configurações realizadas a partir de versões anteriores do RHEL devem ser revisadas. Veja a página de manual **rsyslog.conf(5)** para mais informações.

(BZ#1619645)

Auditoria 3.0 substitui audispd por auditd

Com esta atualização, a funcionalidade do **audispd** foi transferida para o **auditd**. Como resultado, as opções de configuração do **audispd** são agora parte do **auditd.conf**. Além disso, o diretório **plugins.d** foi movido para **/etc/audit**. O status atual do **auditd** e seus plug-ins pode agora ser verificado executando o comando de **estado do serviço auditd**.

(BZ#1616428)

tangd_port_t permite mudanças da porta padrão para Tang

Esta atualização introduz o tipo **tangd_port_t** SELinux que permite a execução do serviço **tangd** como confinado ao modo de aplicação do SELinux. Essa mudança ajuda a simplificar a configuração de um servidor Tang para ouvir em uma porta definida pelo usuário e também preserva o nível de segurança fornecido pelo SELinux em modo de aplicação.

Consulte a seção Configurando o desbloqueio automatizado de volumes criptografados utilizando a decodificação baseada em políticas para obter mais informações.

(BZ#1664345)

Novas booleans SELinux

Esta atualização da política do sistema SELinux introduz as seguintes booleans:

- colord_use_nfs
- mysql_connect_http
- pdns_can_network_connect_db
- ssh_use_tcpd
- sslh_can_bind_any_port
- sslh_can_connect_any_port
- virt_use_pcscd

Para obter uma lista de booleanos incluindo seu significado, e para descobrir se eles estão habilitados ou desabilitados, instale o pacote **selinux-policy-devel** e use:

```
# semanage boolean -l
```

(JIRA:RHELPLAN-10347)

SELinux agora apóia systemd Sem novos privilégios

Esta atualização introduz a capacidade da política **nnp_nosuid_transition** que permite transições de domínio SELinux sob **No New Privileges** (NNP) ou **nosuid** se **a nnp_nosuid_transition** for permitida entre o antigo e o novo contexto. Os pacotes de **políticas selinux** agora contêm uma política para serviços **systemd** que utilizam o recurso de segurança do **NNP**.

A regra a seguir descreve como permitir esta capacidade para um serviço:

```
allow source_domain  target_type:process2 { nnp_transition nosuid_transition };
```

Por exemplo:

```
allow init_t fprintd_t:process2 { nnp_transition nosuid_transition };
```

A política de distribuição agora também contém uma interface macro m4, que pode ser usada nas políticas de segurança SELinux para serviços que utilizam a função **init_nnp_daemon_domain()**.

(BZ#1594111)

Apoio para uma nova verificação de permissão do mapa no syscall do mmap

A permissão do **mapa** SELinux foi adicionada para controlar o acesso mapeado de memória a arquivos, diretórios, soquetes, e assim por diante. Isto permite que a política SELinux impeça o acesso direto à memória a vários objetos do sistema de arquivos e garanta que todo esse acesso seja revalidado.

(BZ#1592244)

A SELinux agora apóia a permissão getrlimit na classe de processo

Esta atualização introduz uma nova verificação de controle de acesso SELinux, **process:getrlimit**, que foi adicionada para a função **prlimit()**. Isto permite aos desenvolvedores de políticas do SELinux controlar quando um processo tenta ler e depois modificar os limites de recursos de outro processo usando a função **process:setrlimit** permission. Note que o SELinux não restringe um processo de manipulação de seus próprios limites de recursos através do **prlimit()**. Veja as páginas man **prlimit(2**) e **getrlimit(2)** para mais informações.

(BZ#1549772)

selinux-policy agora suporta etiquetas VxFS

Esta atualização introduz o suporte para os atributos estendidos de segurança do Veritas File System (VxFS) (xattrs). Isto permite armazenar etiquetas SELinux apropriadas com objetos no sistema de arquivo em vez do tipo genérico vxfs_t. Como resultado, os sistemas com VxFS com suporte total ao SELinux são mais seguros.

(BZ#1483904)

As bandeiras de endurecimento da segurança são aplicadas de forma mais consistente

As bandeiras de endurecimento de segurança de tempo compilar são aplicadas mais consistentemente nos pacotes RPM na distribuição RHEL 8, e o pacote **redhat-rpm-config** agora fornece automaticamente bandeiras de endurecimento de segurança. As bandeiras de tempo de compilação aplicadas também ajudam a atender às exigências dos Critérios Comuns (CC). Os seguintes sinalizadores de endurecimento de segurança são aplicados:

- Para a detecção de erros de estouro de tampão: **D_FORTIFY_SOURCE=2**
- Endurecimento padrão de biblioteca que verifica as arrays C, vetores e cordas: **D_GLIBCXX_ASSERTIONS**
- Para o Stack Smashing Protector (SSP): **fstack-protector-strong**
- Para o endurecimento de exceções: **fexcepções**
- Para Control-Flow Integrity (CFI): **fcf-protection=full** (somente em arquiteturas AMD e Intel 64-bit)
- Para Randomização do Layout do Espaço de Endereços (ASLR): **fPIE** (para executáveis) ou **fPIC** (para bibliotecas)

- Para proteção contra a vulnerabilidade do Stack Clash: **proteção fstack-clash** (exceto ARM)
- Ligar bandeiras para resolver todos os símbolos na inicialização: - **Wl**, **-z,agora**

Veja a página de manual **gcc(1)** para mais informações.

(JIRA:RHELPLAN-2306)

5.1.16. Virtualização

qemu-kvm 2.12 em RHEL 8

O Red Hat Enterprise Linux 8 é distribuído com **qemu-kvm** 2.12. Esta versão corrige bugs múltiplos e adiciona uma série de melhorias sobre a versão 1.5.3, disponível no Red Hat Enterprise Linux 7.

Notavelmente, as seguintes características foram introduzidas:

- Q35 tipo de máquina convidada
- UEFI bota convidado
- NUMA tuning and pinning in the guest
- vCPU hot plug e hot unplug
- rosqueamento de E/S convidado

Note que alguns dos recursos disponíveis no **qemu-kvm** 2.12 não são suportados no Red Hat Enterprise Linux 8. Para informações detalhadas, veja "Suporte de recursos e limitações na virtualização RHEL 8" no Portal do Cliente da Red Hat.

(BZ#1559240)

O tipo de máquina Q35 é agora suportado pela virtualização

A Red Hat Enterprise Linux 8 introduz o suporte para **Q35**, um tipo de máquina mais moderno baseado em PCI Express. Isto proporciona uma variedade de melhorias nas características e desempenho dos dispositivos virtuais, e garante que uma gama mais ampla de dispositivos modernos seja compatível com a virtualização. Além disso, as máquinas virtuais criadas no Red Hat Enterprise Linux 8 estão configuradas para usar **Q35** por default.

Observe também que o tipo de máquina anteriormente padrão **PC** foi depreciado e só deve ser usado quando se virtualiza sistemas operacionais mais antigos que não suportam Q35.

(BZ#1599777)

KVM apóia UMIP no RHEL 8

A virtualização KVM agora suporta o recurso de Prevenção de Instruções de Modo de Usuário (UMIP), que pode ajudar a evitar o acesso das aplicações de espaço do usuário às configurações de todo o sistema. Isto reduz os vetores potenciais para ataques de escalada de privilégios, e assim torna o KVM hipervisor e suas máquinas convidadas mais seguras.

(BZ#1494651)

Informações adicionais nos relatórios de acidentes de hóspedes da KVM

As informações sobre o acidente que o hipervisor KVM gera se um convidado terminar inesperadamente ou ficar sem resposta foram expandidas. Isto facilita o diagnóstico e a correção de problemas em implementações de virtualização KVM.

(BZ#1508139)

NVIDIA vGPU é agora compatível com o console VNC

Ao usar o recurso de GPU virtual NVIDIA (vGPU), agora é possível usar o console VNC para exibir a saída visual do convidado.

(BZ#1497911)

A Ceph é apoiada pela virtualização

Com esta atualização, o armazenamento Ceph é suportado pela virtualização da KVM em todas as arquiteturas de CPU suportadas pela Red Hat.

(BZ#1578855)

Carregador de inicialização interativo para máquinas virtuais KVM na IBM Z

Ao inicializar uma máquina virtual KVM em um host IBM Z, o firmware do carregador de inicialização QEMU pode agora apresentar uma interface de console interativa do sistema operacional convidado. Isto torna possível solucionar problemas de inicialização do SO hóspede sem acesso ao ambiente host.

(BZ#1508137)

IBM z14 ZR1 suportada em máquinas virtuais

O KVM hypervisor agora suporta o modelo de CPU do servidor IBM z14 ZR1. Isto permite utilizar as características desta CPU em máquinas virtuais KVM que rodam em um sistema IBM Z.

(BZ#1592337)

A KVM apóia Telnet 3270 na IBM Z

Ao utilizar o RHEL 8 como um host em um sistema IBM Z, agora é possível conectar-se a máquinas virtuais no host usando clientes **Telnet 3270**.

(BZ#1570029)

Foi adicionado o sandboxing QEMU

No Red Hat Enterprise Linux 8, o emulador QEMU introduz o recurso de sandboxing. O QEMU sandboxing oferece limitações configuráveis ao que os sistemas chamam de QEMU pode realizar, e assim torna as máquinas virtuais mais seguras. Note que este recurso é habilitado e configurado por padrão.

(JIRA:RHELPLAN-10628)

Novos tipos de máquinas para máquinas virtuais KVM em IBM POWER

Múltiplos novos tipos de máquinas rel-pseries foram habilitados para KVM hypervisors rodando em sistemas IBM POWER 8 e IBM POWER 9. Isto torna possível que máquinas virtuais (VMs) hospedadas no RHEL 8 em um sistema IBM POWER utilizem corretamente as características de CPU destes tipos de máquinas. Além disso, isto permite a migração de VMs no IBM POWER para uma versão mais recente do KVM hypervisor.

(BZ#1585651, BZ#1595501)

Os sistemas ARM 64 agora suportam máquinas virtuais com até 384 vCPUs

Ao utilizar o KVM hypervisor em um sistema ARM 64, agora é possível atribuir até 384 CPUs virtuais (vCPUs) a uma única máquina virtual (VM).

Observe que o número de CPUs físicas no host deve ser igual ou maior do que o número de vCPUs anexadas a suas VMs, pois a RHEL 8 não suporta o excesso de comprometimento de vCPUs.

(BZ#1422268)

Conjuntos de instruções GFNI e CLDEMOT habilitados para Intel Xeon SnowRidge

Máquinas virtuais (VMs) rodando em um host RHEL 8 em um sistema Intel Xeon SnowRidge podem agora usar os conjuntos de instruções GFNI e CLDEMOT. Isto pode aumentar significativamente o desempenho de tais VMs em certos cenários.

(BZ#1494705)

IPv6 habilitado para OVMF

O protocolo IPv6 está agora habilitado no Open Virtual Machine Firmware (OVMF). Isto torna possível para as máquinas virtuais que utilizam OVMF tirar proveito de uma variedade de melhorias de inicialização da rede que o IPv6 oferece.

(BZ#1536627)

Um driver de bloco baseado em VFIO para dispositivos NVMe foi adicionado

O emulador QEMU introduz um driver baseado na função virtual I/O (VFIO) para dispositivos de memória não volátil Express (NVMe). O driver se comunica diretamente com os dispositivos NVMe anexados às máquinas virtuais (VMs) e evita o uso da camada do sistema de kernel e seus drivers NVMe. Como resultado, isto melhora o desempenho dos dispositivos NVMe em máquinas virtuais.

(BZ#1519004)

Suporte multicanal para o driver UIO genérico Hyper-V

O RHEL 8 agora suporta o recurso multicanal para o driver de E/S do Hyper-V Generic userspace I/O (UIO). Isto torna possível que as VMs RHEL 8 rodando no Hyper-V hypervisor utilizem o Data Plane Development Kit (DPDK) Netvsc Poll Mode driver (PMD), que melhora as capacidades de rede destas VMs.

Observe, entretanto, que o status da interface Netvsc atualmente é exibido como Down mesmo quando está em execução e utilizável.

(BZ#1650149)

Melhoria do suporte de página enorme

Ao utilizar o RHEL 8 como um host de virtualização, os usuários podem modificar o tamanho das páginas que retornam a memória de uma máquina virtual (VM) para qualquer tamanho que seja suportado pela CPU. Isto pode melhorar significativamente o desempenho da VM.

Para configurar o tamanho das páginas de memória da VM, edite a configuração XML da VM e adicione o elemento <hugepages> à seção <memoryBacking>.

(JIRA:RHELPLAN-14607)

5.1.17. Apoio

sosreport pode relatar programas e mapas baseados em eBPF

A ferramenta **sosreport** foi aperfeiçoada para relatar qualquer programa e mapas de Filtragem de Pacotes Berkeley (eBPF) estendida carregada no Red Hat Enterprise Linux 8.

(BZ#1559836)

5.2. CORREÇÃO DE ERROS

Esta parte descreve os bugs corrigidos no Red Hat Enterprise Linux 8.0 que têm um impacto significativo sobre os usuários.

5.2.1. Desktop

PackageKit agora pode operar em pacotes de rpm

Com esta atualização, o suporte para operar em pacotes de **rpm** foi adicionado em **PackageKit**.

(BZ#1559414)

5.2.2. Infra-estruturas gráficas

QEMU não lida corretamente com entradas de 8 bytes de ggtt

A QEMU ocasionalmente divide uma entrada **ggtt** de 8 bytes em duas escritas consecutivas de 4 bytes. Cada uma dessas escritas parciais pode desencadear uma escrita **ggtt** de host separada. Às vezes, as duas escritas de **ggtt** são combinadas incorretamente. Conseqüentemente, a tradução para um endereço de máquina falha, e ocorre um log de erros.

(BZ#1598776)

5.2.3. Gestão da Identidade

O Enterprise Security Client utiliza a biblioteca opensc para detecção de token

O Red Hat Enterprise Linux 8.0 suporta apenas a biblioteca **opensc** para cartões inteligentes. Com esta atualização, o Enterprise Security Client (ESC) usa **o opensc** para detecção de token ao invés da biblioteca de **coolkey** removida. Como resultado, os aplicativos detectam corretamente os tokens suportados.

(BZ#1538645)

Sistema de certificados agora suporta logs de depuração rotativos

Anteriormente, o Certificate System usava uma estrutura de registro personalizada, que não suportava rotação de registros. Como conseqüência, os logs de depuração como **/var/log/pki/*instance_name*/ca/debug** cresceram indefinidamente. Com esta atualização, o Certificate System usa a estrutura **java.log.util**, que suporta a rotação de logs. Como resultado, você pode configurar a rotação de logs no arquivo **/var/lib/pki/*instance_name*/conf/logging.properties**.

Para maiores informações sobre rotação de logs, veja a documentação do pacote **java.util.logging**.

(BZ#1565073)

Sistema de Certificado não mais registra avisos de operação do SetAllPropertiesRule quando o serviço é iniciado

Anteriormente, o Sistema de Certificado registrava avisos na operação **SetAllPropertiesRule** no arquivo de log /**var**/**log**/**messages** quando o serviço era iniciado. O problema foi resolvido, e os avisos mencionados não são mais registrados.

(BZ#1424966)

O Sistema de Certificado KRA analisa corretamente as respostas de pedidos chave de clientes

Anteriormente, o Sistema de Certificado mudou para uma nova biblioteca JSON. Como conseqüência, a serialização para certos objetos diferia, e o cliente Python key recovery authority (KRA) não conseguiu analisar as respostas ao **Key Request**. O cliente foi modificado para suportar as respostas usando tanto a antiga quanto a nova biblioteca JSON. Como resultado, o cliente Python KRA analisa corretamente as respostas de **Key Request (Pedido** de Chave).

(BZ#1623444)

5.2.4. Compiladores e ferramentas de desenvolvimento

A GCC não produz mais falsos avisos positivos sobre o acesso fora dos limites

Anteriormente, ao compilar com a opção **-O3** nível de otimização, a Coleção de Compiladores GNU (GCC) ocasionalmente retornava um falso aviso positivo sobre um acesso fora dos limites, mesmo que o código compilado não o contivesse. A otimização foi corrigida e o GCC não exibe mais o aviso falso positivo.

(BZ#1246444)

ltrace exibe corretamente grandes estruturas

Anteriormente, a ferramenta **ltrace** não conseguia imprimir corretamente as grandes estruturas retornadas das funções. O manuseio de grandes estruturas em **ltrace** foi melhorado e agora elas são impressas corretamente.

(BZ#1584322)

Função GCC integrada __builtin_clz retorna valores corretos no IBM Z

Anteriormente, a instrução **FLOGR** da arquitetura IBM Z era dobrada incorretamente pelo compilador GCC. Como conseqüência, a função **__builtin_clz** usando esta instrução poderia retornar resultados errados quando o código fosse compilado com a opção **-funroll-loops** GCC. Este erro foi corrigido e a função agora fornece resultados corretos.

(BZ#1652016)

GDB fornece status de saída diferente de zero quando o último comando em modo batch falha

Anteriormente, a GDB sempre saía com status **0** quando em execução em modo batch, independentemente de erros nos comandos. Como conseqüência, não era possível determinar se os comandos eram bem sucedidos. Este comportamento foi alterado e a GDB agora sai com status **1** quando ocorre um erro no último comando. Isto preserva a compatibilidade com o comportamento anterior, onde todos os comandos são executados. Como resultado, agora é possível determinar se a execução em modo batch da GDB é bem sucedida.

(BZ#1491128)

5.2.5. Sistemas de arquivo e armazenamento

Níveis de impressão mais altos não fazem mais com que o iscsiadm termine de forma inesperada

Anteriormente, o utilitário **iscsiadm** terminou inesperadamente quando o usuário especificou um nível de impressão maior que 0 com a opção **--print** ou **-P**. Este problema foi corrigido e todos os níveis de impressão agora funcionam como esperado.

(BZ#1582099)

o multipathd não desabilita mais o caminho quando não consegue obter a WWID de um caminho

Anteriormente, o serviço **multipathd** tratou uma tentativa fracassada de conseguir a WWID de um caminho como se estivesse obtendo uma WWID vazia. Se o serviço **multipathd** falhou em obter o WWID de um caminho, às vezes ele desativou esse caminho.

Com esta atualização, o **multipathd** continua a usar a antiga WWID se ela não conseguir obter a WWID ao verificar se ela mudou.

Como resultado, o **multipathd** não desabilita mais os caminhos quando não consegue obter a WWID, ao verificar se a WWID mudou.

(BZ#1673167)

5.2.6. Alta disponibilidade e clusters

Nova opção /etc/sysconfig/pcsd para rejeitar a renegociação SSL/TLS iniciada pelo cliente

Quando a renegociação do TLS é habilitada no servidor, um cliente tem permissão para enviar um pedido de renegociação, o que inicia um novo aperto de mão. Os requisitos computacionais de um aperto de mão são mais altos em um servidor do que em um cliente. Isto torna o servidor vulnerável a ataques do DoS. Com esta correção, a configuração **PCSD_SSL_OPTIONS** no arquivo de configuração **/etc/sysconfig/pcsd** aceita a opção **OP_NO_RENEGOTIATION** para rejeitar renegociações. Note que o cliente ainda pode abrir múltiplas conexões a um servidor com um aperto de mão realizado em todas elas.

(BZ#1566430)

Um nó de cluster removido não é mais exibido no status de cluster

Anteriormente, quando um nó era removido com o comando **remover nó de cluster pcs**, o nó removido permanecia visível na saída de uma exibição de **status pcs**. Com esta correção, o nó removido não é mais exibido no status do cluster.

(BZ#1595829)

Os agentes de vedação podem agora ser configurados usando nomes de parâmetros mais recentes e preferidos ou nomes de parâmetros depreciados

Um grande número de parâmetros de agentes de vedação foram renomeados enquanto os antigos nomes de parâmetros ainda são suportados como depreciados. Anteriormente, **os pcs** não eram capazes de definir os novos parâmetros, a menos que fossem usados com a opção **--force**. Com esta correção, **pcs** agora suporta os parâmetros renomeados de agente de cerca enquanto mantém o suporte para os parâmetros depreciados.

(BZ#1436217)

O comando pcs agora lê corretamente o status XML de um cluster para exibição

O comando **pcs** executa o utilitário **crm_mon** para obter o status de um cluster em formato XML. O utilitário **crm_mon** imprime XML para saída padrão e avisos para saída de erro padrão. Anteriormente **o pcs** misturava XML e avisos em um único fluxo e depois não era capaz de analisá-lo como XML. Com esta correção, as saídas de erro padrão e de erro são separadas em **pcs** e a leitura do status XML de um cluster funciona como esperado.

(BZ#1578955)

Os usuários não são mais aconselhados a destruir clusters ao criar novos clusters com nós de clusters existentes

Anteriormente, quando um usuário especificava nós de um cluster existente ao executar o comando **de configuração do cluster pcs** ou ao criar um cluster com a interface Web **pcsd**, pcs relatou isso como um erro e sugeriu que o usuário destruísse o cluster nos nós. Como resultado, os usuários destruiriam o cluster nos nós, quebrando o cluster do qual os nós faziam parte, pois os nós restantes ainda considerariam os nós destruídos como parte do cluster. Com esta correção, os usuários são aconselhados a remover os nós de seu aglomerado, informando-os melhor sobre como abordar o problema sem quebrar seus aglomerados.

(BZ#1596050)

os comandospcs não mais pedem credenciais de forma interativa

Quando um usuário não root executa um comando **pcs** que requer permissão de root, o **pcs** se conecta ao daemon **pcsd** em execução local e passa o comando para ele, já que o daemon **pcsd** roda com permissões de root e é capaz de executar o comando. Anteriormente, se o usuário não fosse autenticado no daemon **pcsd** local, **pcs** pedia um nome de usuário e uma senha interativamente. Isto era confuso para o usuário e exigia um tratamento especial em scripts rodando **pcsd**. Com esta correção, se o usuário não for autenticado, então **o pcs** sai com um erro que aconselha o que fazer: Executar **pcs** como root ou autenticar usando o novo comando de **autenticação local do cliente pcs**. Como resultado, os comandos **pcs** não pedem credenciais de forma interativa, melhorando a experiência do usuário.

(BZ#1554310)

O daemon pcsd agora começa com seu certificado SSL padrão auto-gerado quando as políticas criptográficas estão definidas para FUTURO.

Uma configuração **cripto-política** de **FUTURO** requer que as chaves RSA em certificados SSL tenham pelo menos 3072b de comprimento. Anteriormente, o daemon **pcsd** não começaria quando esta política fosse definida, já que gera certificados SSL com uma chave 2048b. Com esta atualização, o tamanho da chave dos certificados SSL auto-gerados **do pcsd** foi aumentado para 3072b e o **pcsd** agora começa com seu certificado SSL auto-gerado padrão.

(BZ#1638852)

O serviço pcsd agora começa quando a rede estiver pronta

Anteriormente, quando um usuário configurava **o pcsd** para ligar-se a um endereço IP específico e o endereço não estava pronto durante a inicialização quando **o pcsd** tentou iniciar, então **o pcsd** não conseguiu iniciar e foi necessária uma intervenção manual para iniciar o **pcsd**. Com esta correção, o **pcsd.service** depende do **network-online.target**. Como resultado, **o pcsd** inicia quando a rede está pronta e é capaz de se ligar a um endereço IP.

(BZ#1640477)

5.2.7. Trabalho em rede

Algoritmos fracos de TLS não são mais permitidos para o trabalho em rede glib-net

Anteriormente, o pacote **glib-networking** não era compatível com a Política Crypto RHEL 8 para todo o sistema. Como conseqüência, as aplicações que utilizam a biblioteca **glib** para redes podem permitir conexões de Transport Layer Security (TLS) usando algoritmos fracos do que o administrador pretendia. Com esta atualização, a política de criptografia de todo o sistema é aplicada, e agora as aplicações que utilizam **a glib** para redes permitem somente conexões TLS que são aceitáveis de acordo com a política.

(BZ#1640534)

5.2.8. Segurança

A política SELinux agora permite que os processos iscsiuio se conectem ao portal de descobertas

Anteriormente, a política SELinux era muito restritiva para os processos **iscsiuio** e estes processos não eram capazes de acessar os dispositivos **/dev/uio*** usando a chamada ao sistema **mmap**. Como conseqüência, a conexão com o portal de descoberta falhou. Esta atualização acrescenta as regras ausentes à política SELinux e os processos de **iscsiuio** funcionam como esperado no cenário descrito.

(BZ#1626446)

5.2.9. Gestão de assinaturas

dnf e yum podem agora acessar os repos, independentemente dos valores do gerenciador de assinaturas

Anteriormente, os comandos **dnf** ou **yum** ignoravam o prefixo **https://** de uma URL adicionada pelo serviço **subscription-manager**. Os comandos **dnf** ou **yum** atualizados não ignoram as URLs inválidas **https://**. Como conseqüência, o **dnf** e **yum** não conseguiram acessar os repos. Para corrigir o problema, uma nova variável de configuração, **proxy_scheme** foi adicionada ao arquivo **/etc/rhsm/rhsm.conf** e o valor pode ser definido tanto para **http** quanto para **https**. Se nenhum valor for especificado, **subscription-manager** define o **http** como padrão, o que é mais comumente usado.

Observe que se o proxy usa **http**, a maioria dos usuários não deve alterar nada na configuração em **/etc/rhsm/rhsm.conf**. Se o proxy usa **https**, os usuários devem atualizar o valor do **proxy_scheme** para **https**. Então, em ambos os casos, os usuários precisam executar o comando **subscription-manager repos --list** ou esperar pelo processo **rhsmcertd** daemon para regenerar o **/etc/yum.repos.d/redhat.repo** adequadamente.

(BZ#1654531)

5.2.10. Virtualização

A montagem de discos efêmeros no Azure agora funciona de forma mais confiável

Anteriormente, a montagem de um disco efêmero em uma máquina virtual (VM) rodando na plataforma Microsoft Azure falhou se a VM foi "parada (desalocada)" e depois começou. Esta atualização garante que os discos de reconexão sejam tratados corretamente nas circunstâncias descritas, o que evita que o problema ocorra.

(BZ#1615599)

5.3. PREVISÕES TECNOLÓGICAS

Esta parte fornece uma lista de todas as Análises de Tecnologia disponíveis no Red Hat Enterprise Linux 8.0.

Para informações sobre o escopo de suporte da Red Hat para recursos de Technology Preview, veja Technology Preview Features Support Scope .

5.3.1. Kernel

eBPF disponível como uma prévia de tecnologia

O recurso **extended Berkeley Packet Filtering (eBPF)**está disponível como uma Technology Preview tanto para rede quanto para rastreamento. **eBPF** permite ao espaço do usuário anexar programas personalizados em uma variedade de pontos (soquetes, pontos de rastreamento, recepção de pacotes) para receber e processar dados. O recurso inclui uma nova chamada de sistema **bpf()**, que suporta a criação de vários tipos de mapas, e também a inserção de vários tipos de programas no kernel. Note que a chamada de sistema **bpf()** só pode ser usada com sucesso por um usuário com a capacidade **CAP_SYS_ADMIN**, tal como um usuário root. Veja a página de manual **bpf**(2) para mais informações.

(BZ#1559616)

OBCC está disponível como uma Pré-visualização Tecnológica

BPF Compiler Collection (BCC) é um kit de ferramentas de espaço do usuário para criar programas eficientes de rastreamento e manipulação de kernel, disponível como Technology Preview no Red Hat Enterprise Linux 8. **BCC** fornece ferramentas para análise de E/S, rede e monitoramento de sistemas operacionais Linux usando o **Berkeley Packet Filtering estendido (eBPF)**.

(BZ#1548302)

Control Group v2 disponível como uma prévia de tecnologia no RHEL 8

o mecanismo**Control Group v2** é um grupo de controle unificado de hierarquia. **Control Group v2** organiza os processos hierarquicamente e distribui os recursos do sistema ao longo da hierarquia de forma controlada e configurável.

Ao contrário da versão anterior, **Control Group v2** tem apenas uma única hierarquia. Esta hierarquia única permite que o kernel Linux o faça:

- Categorizar os processos com base no papel de seu proprietário.
- Eliminar problemas com políticas conflitantes de múltiplas hierarquias.

Control Group v2 suporta numerosos controladores:

- O controlador de CPU regula a distribuição dos ciclos da CPU. Este controlador implementa:
  - Modelos de limite de peso e largura de banda absoluta para a política normal de programação.
  - Modelo absoluto de alocação de largura de banda para política de programação em tempo real.
- O controlador de memória regula a distribuição da memória. Atualmente, os seguintes tipos de utilização de memória são rastreados:
  - Memória do espaço do usuário - cache de páginas e memória anônima.

- Estruturas de dados do núcleo, tais como amolgadelas e inodes.
- Tampões de soquete TCP.

- O controlador de E/S regula a distribuição dos recursos de E/S.
- O controlador Writeback interage tanto com os controladores de Memória como de E/S e é específico para **Control Group v2**.

As informações acima foram baseadas no link: https://www.kernel.org/doc/Documentation/cgroup-v2.txt. Você pode consultar o mesmo link para obter mais informações sobre determinados controladores **Control Group v2**.

(BZ#1401552)

kdump inicial disponível como uma prévia de tecnologia no Red Hat Enterprise Linux 8

O recurso de **kdump precoce** permite que o núcleo e initramas do acidente sejam carregados suficientemente cedo para capturar as informações **vmcore**, mesmo em caso de acidentes precoces. Para mais detalhes sobre **o kdump inicial**, veja o arquivo **/usr/share/doc/kexec-tools/early-kdump-howto.txt**.

(BZ#1520209)

O driver do dispositivo ibmvnic disponível como Technology Preview

Com o Red Hat Enterprise Linux 8.0, o driver IBM Virtual Network Interface Controller (vNIC) para arquiteturas IBM POWER, **ibmvnic**, está disponível como Technology Preview. vNIC é uma tecnologia de rede virtual PowerVM que fornece capacidades empresariais e simplifica o gerenciamento de rede. É uma tecnologia de alto desempenho e eficiente que, quando combinada com o SR-IOV NIC fornece capacidades de controle de largura de banda Qualidade de Serviço (QoS) no nível do NIC virtual. vNIC reduz significativamente a sobrecarga de virtualização, resultando em latências menores e menos recursos de servidor, incluindo CPU e memória, necessários para a virtualização da rede.

(BZ#1524683)

5.3.2. Infra-estruturas gráficas

Console remoto VNC disponível como Technology Preview para a arquitetura ARM de 64 bits

Na arquitetura ARM de 64 bits, o console remoto da Virtual Network Computing (VNC) está disponível como uma pré-visualização tecnológica. Observe que o resto da pilha de gráficos não está atualmente verificada para a arquitetura ARM de 64 bits.

(BZ#1698565)

5.3.3. Habilitação do hardware

O MD RAID1 sensível ao cluster está disponível como uma pré-visualização tecnológica.

O cluster RAID1 não é habilitado por padrão no espaço do kernel. Se você quiser ter uma tentativa com cluster RAID1, você precisa primeiro construir o kernel com cluster RAID1 como um módulo, use os seguintes passos:

1. Digite o comando **make menuconfig**.

2. Enter the **make && make modules && make modules_install && make install** command.
3. Digite o comando de **reinicialização**.

(BZ#1654482)

5.3.4. Gestão da Identidade

DNSSEC disponível como Technology Preview na IdM

Os servidores de Gerenciamento de Identidade (IdM) com DNS integrado agora suportam Extensões de Segurança DNS (DNSSEC), um conjunto de extensões para o DNS que aumentam a segurança do protocolo DNS. As zonas DNS hospedadas nos servidores IdM podem ser automaticamente assinadas usando DNSSEC. As chaves criptográficas são geradas e giradas automaticamente.

Os usuários que decidirem proteger suas zonas DNS com DNSSEC são aconselhados a ler e seguir estes documentos:

- DNSSEC Práticas Operacionais, Versão 2: http://tools.ietf.org/html/rfc6781#section-2
- Guia de implantação do Sistema de Nomes de Domínio Seguros (DNS): http://dx.doi.org/10.6028/NIST.SP.800-81-2
- Considerações sobre o tempo de prorrogação do DNSSEC: http://tools.ietf.org/html/rfc7583

Observe que os servidores IdM com DNS integrado utilizam DNSSEC para validar as respostas DNS obtidas de outros servidores DNS. Isto pode afetar a disponibilidade de zonas DNS que não são configuradas de acordo com as práticas de nomenclatura recomendadas.

(BZ#1664718)

Gerenciamento da Identidade JSON-RPC API disponível como Technology Preview

Um API está disponível para Gerenciamento de Identidade (IdM). Para visualizar o API, o IdM também fornece um navegador API como Technology Preview (Visualização de Tecnologia).

No Red Hat Enterprise Linux 7.3, o IdM API foi melhorado para permitir múltiplas versões de comandos API. Anteriormente, os aprimoramentos podiam mudar o comportamento de um comando de forma incompatível. Os usuários agora são capazes de continuar usando ferramentas e scripts existentes mesmo que a API do IdM mude. Isto permite:

- Administradores para usar versões anteriores ou posteriores do IdM no servidor do que no cliente gestor.
- Desenvolvedores para usar uma versão específica de uma chamada IdM, mesmo que a versão IdM mude no servidor.

Em todos os casos, a comunicação com o servidor é possível, independentemente se um dos lados utiliza, por exemplo, uma versão mais recente que introduz novas opções para um recurso.

Para obter detalhes sobre o uso da API, consulte Utilização da API de Gerenciamento de Identidade para Comunicação com o Servidor IdM (PREVISÃO TECNOLÓGICA).

(BZ#1664719)

5.3.5. Sistemas de arquivo e armazenamento

Adaptadores Aero disponíveis como uma Pré-visualização Tecnológica

Os seguintes adaptadores Aero estão disponíveis como uma Pré-visualização Tecnológica:

- PCI ID 0x1000:0x00e2 e 0x1000:0x00e6, controlado pelo driver **mpt3sas**
- PCI ID 0x1000:Ox10e5 e 0x1000:0x10e6, controlado pelo driver **megaraid_sas**

(BZ#1663281)

Stratis está agora disponível

Stratis é um novo gerente de armazenamento local. Ele fornece sistemas de arquivos gerenciados em cima de pools de armazenamento com características adicionais para o usuário.

Stratis permite realizar mais facilmente tarefas de armazenamento como, por exemplo

- Gerenciar snapshots e provisionamento fino
- Aumentar automaticamente os tamanhos dos sistemas de arquivo conforme necessário
- Manter sistemas de arquivo

Para administrar o armazenamento Stratis, use o utilitário **Stratis**, que se comunica com o serviço de fundo **Stratisd**.

Stratis é fornecido como uma Pré-visualização Tecnológica.

Para mais informações, consulte a documentação do Stratis: Gerenciamento de armazenamento local em camadas com Stratis.

(JIRA:RHELPLAN-1212)

OverlayFS

O OverlayFS é um tipo de sistema de arquivo sindical. Ele permite a sobreposição de um sistema de arquivos sobreposto a outro. As mudanças são registradas no sistema de arquivo superior, enquanto o sistema de arquivo inferior permanece inalterado. Isto permite que vários usuários compartilhem uma imagem do sistema de arquivo, como um container ou um DVD-ROM, onde a imagem base está em uma mídia somente de leitura. Consulte a documentação do kernel do Linux para obter informações adicionais: https://www.kernel.org/doc/Documentation/filesystems/overlayfs.txt.

A OverlayFS continua sendo uma Pré-visualização Tecnológica na maioria das circunstâncias. Como tal, o kernel registra avisos quando esta tecnologia é ativada.

O suporte completo está disponível para OverlayFS quando usado com motores de contêineres suportados(**podman**, **cri-o**, ou **buildah**) sob as seguintes restrições:

- O OverlayFS é suportado para uso apenas como um driver gráfico do motor do contêiner. Seu uso é suportado apenas para conteúdo de COW de contêineres, não para armazenamento persistente. Você deve colocar qualquer armazenamento persistente em volumes não-OverlayFS. Somente a configuração padrão do motor de contêiner pode ser usada; ou seja, um nível de overlay, um nível inferior, e ambos os níveis inferior e superior estão no mesmo sistema de arquivo.
- Atualmente, apenas o XFS é suportado para uso como um sistema de arquivo de camada inferior.

Além disso, as seguintes regras e limitações se aplicam ao uso do OverlayFS:

- O comportamento do kernel ABI e do espaço do usuário do OverlayFS não são considerados estáveis, e podem ver mudanças em futuras atualizações.
- A OverlayFS fornece um conjunto restrito de padrões POSIX. Teste sua aplicação completamente antes de implementá-la com OverlayFS. Os seguintes casos não são compatíveis com o POSIX:
  - Arquivos inferiores abertos com **O_RDONLY** não recebem atualizações **st_atime** quando os arquivos são lidos.
  - Arquivos inferiores abertos com **O_RDONLY**, depois mapeados com **MAP_SHARED** são inconsistentes com modificações subseqüentes.
  - Os valores **st_ino** ou **d_ino** totalmente compatíveis não são ativados por padrão no RHEL 8, mas você pode ativar a conformidade total do POSIX para eles com uma opção de módulo ou opção de montagem.
    Para obter uma numeração inode consistente, use a opção **xino=em** montagem.

    Você também pode usar as opções **redirect_dir=on** e **index=on** para melhorar a conformidade POSIX. Estas duas opções tornam o formato da camada superior incompatível com uma sobreposição sem estas opções. Ou seja, você pode obter resultados inesperados ou erros se criar uma sobreposição com **redirect_dir=on** ou **index=on**, desmontar a sobreposição, e então montar a sobreposição sem estas opções.
- Comandos utilizados com XFS:
  - Os sistemas de arquivo XFS devem ser criados com a opção **-n ftype=1** habilitada para uso como um overlay.
  - Com os rootfs e qualquer sistema de arquivo criado durante a instalação do sistema, defina os parâmetros **--mkfsoptions=-n ftype=1** no kickstart do Anaconda.
  - Ao criar um novo sistema de arquivo após a instalação, execute o comando **# mkfs -t xfs -n ftype=1 /PATH/TO/DEVICE**.
  - Para determinar se um sistema de arquivo existente é elegível para uso como uma sobreposição, execute o comando **# xfs_info /PATH/TO/DEVICE | grep ftype** para ver se a opção **ftype=1** está habilitada.
- As etiquetas de segurança SELinux são habilitadas por padrão em todos os motores de contêineres suportados com OverlayFS.
- Há vários problemas conhecidos associados ao OverlayFS neste lançamento. Para detalhes, veja *Non-standard behavior* na documentação do kernel Linux: https://www.kernel.org/doc/Documentation/filesystems/overlayfs.txt.

(BZ#1690207)

O sistema de arquivo DAX agora está disponível para ext4 e XFS como uma pré-visualização tecnológica

No Red Hat Enterprise Linux 8.0, o sistema de arquivo DAX do Red Hat está disponível como uma Pré-visualização Tecnológica. O DAX fornece um meio para um aplicativo mapear diretamente a memória persistente em seu espaço de endereços. Para usar DAX, um sistema deve ter alguma forma de memória persistente disponível, geralmente na forma de um ou mais NVDIMMs (Non-Volatile Dual In-line Memory Modules), e um sistema de arquivo que suporte DAX deve ser criado no(s) NVDIMM(s).

Além disso, o sistema de arquivo deve ser montado com a opção de montagem **por dax**. Então, um **mmap** de um arquivo no sistema de arquivo montado por eixo resulta em um mapeamento direto do armazenamento no espaço de endereços da aplicação.

(BZ#1627455)

5.3.6. Alta disponibilidade e clusters

Pacemaker **podman** bundles disponíveis como Technology Preview

Os pacotes de contêineres do Pacemaker agora rodam na plataforma de contêineres do **podman**, com o recurso de pacote de contêineres disponível como uma Pré-visualização Tecnológica. Há uma exceção a este recurso que é a Technology Preview: A Red Hat suporta totalmente o uso de pacotes de Pacemaker para o Red Hat Openstack.

(BZ#1619620)

5.3.7. Trabalho em rede

XDP disponível como uma prévia de tecnologia

O recurso eXpress Data Path (XDP), disponível como Technology Preview, fornece um meio de anexar programas Berkeley Packet Filter (eBPF) estendidos para processamento de pacotes de alto desempenho em um ponto inicial no caminho de entrada de dados do kernel, permitindo análise, filtragem e manipulação de pacotes programáveis eficientes.

(BZ#1503672)

eBPF para tc disponível como uma prévia tecnológica

Como uma prévia tecnológica, o subsistema de controle de tráfego (tc) e a ferramenta **tc** podem anexar programas ampliados de Filtragem de Pacotes Berkeley (eBPF) como classificadores de pacotes e ações para as disciplinas de entrada e saída em fila. Isto permite o processamento programável de pacotes dentro do caminho de dados da rede do kernel.

(BZ#1699825)

AF_XDP disponível como uma prévia tecnológica

O soquete**eXpress Data Path(AF_XDP)da família de endereços** foi projetado para o processamento de pacotes de alto desempenho. Ele acompanha **o XDP** e permite o redirecionamento eficiente de pacotes programmaticamente selecionados para aplicações de espaço do usuário para processamento posterior.

(BZ#1633143)

KTLS disponível como uma prévia tecnológica

No Red Hat Enterprise Linux 8, a Kernel Transport Layer Security (KTLS) é fornecida como um Preview de Tecnologia. O KTLS trata os registros TLS usando a criptografia simétrica ou algoritmos de decodificação no kernel para a cifra AES-GCM. O KTLS também fornece a interface para descarregar a criptografia de registros TLS para os Controladores de Interface de Rede (NICs) que suportam esta funcionalidade.

(BZ#1570255)

TIPC disponível como uma prévia tecnológica

O**TIPC (**Transparent Inter Process Communication) é um protocolo especialmente projetado para a comunicação eficiente dentro de clusters de nós frouxamente emparelhados. Ele funciona como um módulo de kernel e fornece uma ferramenta **tipc** no pacote **iproute2** para permitir que os projetistas criem aplicações que possam se comunicar de forma rápida e confiável com outras aplicações, independentemente de sua localização dentro do cluster. Esta característica está disponível como uma Pré-visualização Tecnológica.

(BZ#1581898)

O serviço de solução de sistema está agora disponível como uma Pré-visualização Tecnológica

O serviço **resolvido pelo sistema** fornece resolução de nomes para aplicações locais. O serviço implementa um resolvedor de stub DNS de cache e validação, uma Resolução de nomes Multicast local (LLMNR) e um resolvedor e respondedor DNS Multicast.

Observe que, mesmo que o pacote **systemd** ofereça **solução de sistema**, este serviço é uma Pré-visualização Tecnológica não suportada.

(BZ#1906489)

5.3.8. Funções do Sistema Red Hat Enterprise Linux

O papel post-fix do Sistema RHEL Papéis disponíveis como Previsão Tecnológica

As Funções do Sistema Red Hat Enterprise Linux fornecem uma interface de configuração para os subsistemas do Red Hat Enterprise Linux, o que facilita a configuração do sistema através da inclusão de Funções Ansíveis. Esta interface permite o gerenciamento das configurações do sistema em múltiplas versões do Red Hat Enterprise Linux, bem como a adoção de novos lançamentos principais.

Os pacotes **rhel-system-roles** são distribuídos através do repositório AppStream.

A função **pós-fixa** está disponível como uma Pré-visualização Tecnológica.

Os seguintes papéis são plenamente apoiados:

- **kdump**
- **rede**
- **selinux**
- **timesync**

Para mais informações, consulte o artigo da Base de Conhecimento sobre os Papéis do Sistema RHEL .

(BZ#1812552)

5.3.9. Virtualização

AMD SEV para máquinas virtuais KVM

Como uma prévia de tecnologia, a RHEL 8 introduz o recurso Secure Encrypted Virtualization (SEV) para máquinas host AMD EPYC que utilizam o hipervisor KVM. Se ativada em uma máquina virtual (VM), a SEV criptografa a memória da VM para que o host não possa acessar os dados na VM. Isto aumenta a segurança da VM se o host for infectado com sucesso por malware.

Observe que o número de VMs que podem usar este recurso de cada vez em um único host é determinado pelo hardware do host. Os processadores AMD EPYC atuais suportam até 15 VMs rodando usando SEV.

Observe também que para VMs com SEV configuradas para poder inicializar, você também deve configurar a VM com um limite de memória dura. Para isso, adicione o seguinte à configuração XML da VM:

```
<memtune>
  <hard_limit unit='KiB'>N</hard_limit>
</memtune>
```

O valor recomendado para N é igual ou maior que o RAM 256 MiB do convidado. Por exemplo, se ao convidado for atribuído 2 RAM GiB, N deve ser 2359296 ou maior.

(BZ#1501618, BZ#1501607)

Intel vGPU

Como uma prévia de tecnologia, agora é possível dividir um dispositivo físico Intel GPU em múltiplos dispositivos virtuais chamados de **dispositivos mediados**. Estes dispositivos mediados podem então ser atribuídos a múltiplas máquinas virtuais (VMs) como GPUs virtuais. Como resultado, estas VMs compartilham a performance de uma única GPU física da Intel.

Observe que apenas as GPUs Intel selecionadas são compatíveis com o recurso vGPU. Além disso, atribuir uma GPU física às VMs torna impossível para o host utilizar a GPU, e pode impedir que a saída de exibição gráfica no host funcione.

(BZ#1528684)

Virtualização aninhada agora disponível no IBM POWER 9

Como uma prévia tecnológica, agora é possível utilizar os recursos de virtualização aninhados nas máquinas host RHEL 8 rodando em sistemas IBM POWER 9. A virtualização aninhada permite que as máquinas virtuais KVM (VMs) atuem como hipervisores, o que permite a execução de VMs dentro de VMs.

Note que a virtualização aninhada também continua sendo uma prévia tecnológica nos sistemas AMD64 e Intel 64.

Observe também que para que a virtualização aninhada funcione no IBM POWER 9, o anfitrião, o convidado e os convidados aninhados atualmente precisam todos executar um dos seguintes sistemas operacionais:

- RHEL 8
- RHEL 7 para POWER 9

(BZ#1505999, BZ#1518937)

A virtualização da KVM é utilizável nas máquinas virtuais RHEL 8 Hyper-V

Como uma prévia de tecnologia, a virtualização KVM aninhada pode agora ser usada no hipervisor Microsoft Hyper-V. Como resultado, você pode criar máquinas virtuais em um sistema RHEL 8 para convidados rodando em um host Hyper-V.

Note que atualmente, esta característica só funciona em sistemas Intel. Além disso, a virtualização aninhada não está, em alguns casos, habilitada por padrão no Hyper-V. Para habilitá-la, veja a seguinte documentação da Microsoft:

https://docs.microsoft.com/en-us/virtualization/hyper-v-on-windows/user-guide/nested-virtualization

(BZ#1519039)

5.4. FUNCIONALIDADE DEPRECIADA

Esta parte fornece uma visão geral da funcionalidade que tem sido *deprecated* no Red Hat Enterprise Linux 8.0.

A funcionalidade deprecated continua a ser suportada até o final da vida útil do Red Hat Enterprise Linux 8. A funcionalidade deprecated provavelmente não será suportada em futuros lançamentos importantes deste produto e não é recomendada para novas implementações. Para a lista mais recente de funcionalidades obsoletas dentro de um lançamento principal em particular, consulte a versão mais recente da documentação de lançamento.

Os componentes de hardware obsoletos não são recomendados para novas implementações nos principais lançamentos atuais ou futuros. As atualizações de drivers de hardware são limitadas apenas à segurança e reparos críticos. A Red Hat recomenda a substituição deste hardware tão logo seja razoavelmente viável.

Um pacote pode ser depreciado e não recomendado para uso posterior. Sob certas circunstâncias, uma embalagem pode ser retirada de um produto. A documentação do produto identifica então pacotes mais recentes que oferecem funcionalidade semelhante, idêntica ou mais avançada que a depreciada, e fornece outras recomendações.

Para informações sobre a funcionalidade presente no RHEL 7, mas que foi *removed* no RHEL 8, veja Considerações sobre a adoção do RHEL 8 .

5.4.1. Instalador e criação de imagem

A opção --interativa do comando Kickstart ignorado foi depreciada

O uso da **opção --interativa** em futuros lançamentos do Red Hat Enterprise Linux resultará em um erro fatal de instalação. É recomendado modificar seu arquivo Kickstart para remover a opção.

(BZ#1637872)

Vários comandos e opções de Kickstart foram depreciados

Usando os seguintes comandos e opções nos arquivos Kickstart RHEL 8, será impresso um aviso nos logs.

- **auth** ou **authconfig**
- **dispositivo**
- **deviceprobe**
- **dmraid**
- **instalar**
- **lilo**

- **lilocheck**
- **mouse**
- **multipath**
- **bootloader - atualização**
- **ignorados --interactivos**
- **partição --ativa**
- **reinicialização --kexec**

Onde apenas opções específicas são listadas, o comando base e suas outras opções ainda estão disponíveis e não são depreciadas.

Para mais detalhes e mudanças relacionadas ao Kickstart, consulte a seção Kickstart changes do documento *Considerations in adopting RHEL 8* .

(BZ#1642765)

5.4.2. Sistemas de arquivo e armazenamento

O NFSv3 sobre UDP foi desativado

O servidor NFS não abre mais ou escuta em um soquete do User Datagram Protocol (UDP) por padrão. Esta mudança afeta apenas a versão 3 do NFS porque a versão 4 requer o Protocolo de Controle de Transmissão (TCP).

O NFS sobre o UDP não é mais suportado no RHEL 8.

(BZ#1592011)

O parâmetro da linha de comando do kernel do elevador é depreciado

O parâmetro de linha de comando do kernel **do elevador** foi usado em versões anteriores do RHEL para definir o programador de discos para todos os dispositivos. No RHEL 8, o parâmetro é depreciado.

O kernel Linux a montante removeu o suporte para o parâmetro **elevador**, mas ainda está disponível no RHEL 8 por razões de compatibilidade.

Observe que o kernel seleciona um programador de disco padrão com base no tipo de dispositivo. Esta é tipicamente a configuração ideal. Se você precisar de um agendador diferente, a Red Hat recomenda que você use as regras do **udev** ou o serviço Tuned para configurá-lo. Combine os dispositivos selecionados e troque o agendador somente para esses dispositivos.

Para mais informações, consulte o artigo a seguir: Por que o parâmetro 'elevador=' não funciona mais no RHEL8.

(BZ#1665295)

O módulo VDO Ansible em pacotes VDO

O módulo VDO Ansible é atualmente fornecido pelo pacote **vdo** RPM. Em um lançamento futuro, o módulo VDO Ansible será movido para os pacotes Ansible RPM.

(BZ#1669537)

5.4.3. Trabalho em rede

Os roteiros de rede são depreciados no RHEL 8

Os scripts de rede são depreciados no Red Hat Enterprise Linux 8 e não são mais fornecidos por default. A instalação básica fornece uma nova versão dos scripts **ifup** e **ifdown** que chamam o serviço **NetworkManager** através da ferramenta **nmcli**. No Red Hat Enterprise Linux 8, para rodar os scripts **ifup** e **ifdown**, o NetworkManager deve estar rodando.

Observe que comandos personalizados em /**sbin**/**ifup-local**, **ifdown-pre-local** e **ifdown-local** scripts não são executados.

Se algum destes scripts for necessário, a instalação dos scripts de rede depreciados no sistema ainda é possível com o seguinte comando:

```
~]# yum instalar os roteiros de rede
```

Os scripts **ifup** e **ifdown** se ligam aos scripts de rede legados instalados.

Chamando os scripts da rede legada mostra um aviso sobre sua desvalorização.

(BZ#1647725)

5.4.4. Segurança

A DSA é depreciada no RHEL 8

O Algoritmo de Assinatura Digital (DSA) é considerado depreciado no Red Hat Enterprise Linux 8. Os mecanismos de autenticação que dependem das chaves DSA não funcionam na configuração default. Note que os clientes **OpenSSH** não aceitam chaves de host DSA mesmo no nível da política criptográfica do sistema **LEGACY**.

(BZ#1646541)

SSL2 Cliente Olá foi depreciado no NSS

A versão 1.2 e anterior do protocolo Transport Layer Security**(TLS**) permite iniciar uma negociação com um **Cliente** Mensagem de **Alô** formatada de forma retrocompatível com o protocolo Secure Sockets Layer**(SSL**) versão 2. O suporte a este recurso na biblioteca de Serviços de Segurança de Rede **(NSS**) foi depreciado e está desativado por padrão.

As aplicações que requerem suporte para este recurso precisam usar a nova API **SSL_ENABLE_V2_COMPATIBLE_HELLO** API para habilitá-la. O suporte para este recurso pode ser removido completamente em futuros lançamentos do Red Hat Enterprise Linux 8.

(BZ#1645153)

TLS 1.0 e TLS 1.1 são depreciados

Os protocolos TLS 1.0 e TLS 1.1 estão desabilitados no nível da política criptográfica do sistema **DEFAULT**. Se seu cenário, por exemplo, uma aplicação de videoconferência no navegador Firefox, exigir o uso dos protocolos depreciados, mude a política criptográfica de todo o sistema para o nível **LEGACY**:

```
# update-crypto-policies --set LEGACY
```

Para mais informações, veja o artigo da base de conhecimento Strong crypto defaults no RHEL 8 e a depreciação de algoritmos criptográficos fracos no Portal do Cliente da Red Hat e a página man **update-crypto-policies(8)**.

(BZ#1660839)

5.4.5. Virtualização

Snapshots de máquinas virtuais não são devidamente suportados no RHEL 8

O atual mecanismo de criação de instantâneos de máquinas virtuais (VM) foi depreciado, pois não está funcionando de forma confiável. Como conseqüência, recomenda-se não utilizar instantâneos de VM no RHEL 8.

Observe que um novo mecanismo de instantâneo da VM está em desenvolvimento e será totalmente implementado em um futuro lançamento menor da RHEL 8.

(BZ#1686057)

O tipo de GPU virtual Cirrus VGA foi depreciado

Com uma futura grande atualização do Red Hat Enterprise Linux, o dispositivo GPU **Cirrus VGA** não será mais suportado nas máquinas virtuais KVM. Portanto, a Red Hat recomenda o uso dos dispositivos **stdvga**, **virtio-vga**, ou **qxl** ao invés do Cirrus VGA.

(BZ#1651994)

virt-manager foi depreciado

O aplicativo Virtual Machine Manager, também conhecido como **virt-manager**, foi desativado. O console web RHEL 8, também conhecido como **Cockpit**, tem a intenção de se tornar seu substituto em um lançamento posterior. É, portanto, recomendado que você utilize o console web para gerenciar a virtualização em uma GUI. Entretanto, no Red Hat Enterprise Linux 8.0, alguns recursos podem ser acessíveis apenas a partir de **virt-manager** ou da linha de comando.

(JIRA:RHELPLAN-10304)

5.4.6. Pacotes depreciados

Os seguintes pacotes foram depreciados e provavelmente não serão incluídos em um futuro grande lançamento do Red Hat Enterprise Linux:

- 389-ds-base-legacy-tools
- authd
- custodia
- hostname
- libidn
- net-tools
- textos em rede
- nss-pam-ldapd

- sendmail
- yp-tools
- ypbind
- ypserv

5.5. PROBLEMAS CONHECIDOS

Esta parte descreve os problemas conhecidos no Red Hat Enterprise Linux 8.

5.5.1. O console web

Não é possível fazer logon no console web RHEL com o shell session_recording

Atualmente, os logins no console Web da RHEL falharão para usuários habilitados para registro de registros. O console web RHEL requer a presença de um shell de usuário no diretório **/etc/shells** para permitir um login bem sucedido. Entretanto, se o **tlog-rec-session** for adicionado ao **/etc/shells**, um usuário gravado poderá desativar a gravação alterando o shell do **tlog-rec-session** para outro shell do **/etc/shells**, usando o utilitário "chsh". A Red Hat não recomenda adicionar o **tlog-rec-session** ao **/etc/shells** por este motivo.

(BZ#1631905)

5.5.2. Instalador e criação de imagem

Os comandos auth e authconfig Kickstart requerem o repositório AppStream

O pacote **authselect-compat** é exigido pelos comandos **auth** e **authconfig** Kickstart durante a instalação. Sem este pacote, a instalação falha se o **auth** ou **authconfig** forem usados. Entretanto, por projeto, o pacote **authselect-compat** está disponível apenas no repositório AppStream.

Para contornar este problema, verifique se os repositórios BaseOS e AppStream estão disponíveis para o instalador ou use o comando Kickstart **authselect** durante a instalação.

(BZ#1640697)

Os drivers de vídeo xorg-x11-drv-fbdev, xorg-x11-drv-vesa e xorg-x11-drv-vmware não são instalados por padrão

Estações de trabalho com modelos específicos de placas gráficas NVIDIA e estações de trabalho com unidades de processamento acelerado AMD específicas não exibirão a janela de login gráfico após uma instalação do servidor RHEL 8.0.

Para contornar este problema, realize uma instalação RHEL 8.0 em uma **estação** de trabalho. Se for necessária uma instalação do **servidor** RHEL 8.0 na estação de trabalho, instale manualmente o grupo de pacotes **base-x** após a instalação, executando o comando **yum -y groupinstall base-x**.

Além disso, as máquinas virtuais que dependem da EFI para suporte gráfico, como a Hyper-V, também são afetadas. Se você selecionou o **Servidor com** ambiente base **GUI** no Hyper-V, você pode não conseguir entrar devido a uma tela preta exibida na reinicialização. Para contornar este problema no Hyper-v, habilite o modo multi ou de usuário único usando os seguintes passos:

1. Reinicializar a máquina virtual.

2. Durante o processo de inicialização, selecione o kernel necessário usando as teclas de seta para cima e para baixo em seu teclado.
3. Pressione a tecla **e** em seu teclado para editar a linha de comando do kernel.
4. Adicionar **systemd.unit=multi-user.target** à linha de comando do kernel no GRUB.
5. Pressione **Ctrl-X** para ligar a máquina virtual.
6. Após o login, execute o comando **yum -y groupinstall base-x**.
7. Reinicie a máquina virtual para acessar o modo gráfico.

(BZ#1687489)

A instalação falha ao usar o comando reboot --kexec

A instalação do RHEL 8 falha ao utilizar um arquivo Kickstart que contém o comando **reboot --kexec**. Para evitar o problema, use o comando **reboot** ao invés de **reiniciar --kexec** em seu arquivo Kickstart.

(BZ#1672405)

Copiar o conteúdo do arquivo Binário DVD.iso para uma partição omite os arquivos .treeinfo e .discinfo

Durante a instalação local, ao copiar o conteúdo do DVD.iso binário RHEL 8 para uma partição, o * no comando **cp <path>/* <mounted partition>/dir** não copia os arquivos **.treeinfo** e **.discinfo**. Estes arquivos são necessários para uma instalação bem sucedida. Como resultado, os repositórios BaseOS e AppStream não são carregados, e uma mensagem de log relacionada a depuração no arquivo **anaconda.log** é o único registro do problema.

Para contornar o problema, copie os arquivos **.treeinfo** e **.discinfo** que faltam para a partição.

(BZ#1692746)

A instalação do Anaconda inclui baixos limites de recursos mínimos para a definição dos requisitos

O Anaconda inicia a instalação em sistemas com o mínimo de recursos necessários disponíveis e não fornece aviso prévio de mensagens sobre os recursos necessários para realizar a instalação com sucesso. Como resultado, a instalação pode falhar e os erros de saída não fornecem mensagens claras para uma possível depuração e recuperação. Para contornar este problema, certifique-se de que o sistema tenha as configurações mínimas de recursos necessários para a instalação: 2GB de memória em PPC64(LE) e 1GB em x86_64. Como resultado, deve ser possível realizar uma instalação bem sucedida.

(BZ#1696609)

Os comandos reboot --kexec e inst.kexec não fornecem um estado previsível do sistema

Realizar uma instalação RHEL com o comando de **reinicialização --kexec** Kickstart ou os parâmetros de inicialização do kernel **inst.kexec** não fornecem o mesmo estado previsível do sistema que uma reinicialização completa. Como conseqüência, mudar para o sistema instalado sem reinicialização pode produzir resultados imprevisíveis.

Note que a característica **kexec** é depreciada e será removida em um futuro lançamento do Red Hat Enterprise Linux.

(BZ#1697896)

5.5.3. Kernel

O módulo i40iw não é carregado automaticamente na inicialização

Devido a muitos DNIs i40e não suportarem iWarp e o módulo **i40iw** não suportar totalmente suspensão/resumo, este módulo não é carregado automaticamente por padrão para garantir que a suspensão/resumo funcione corretamente. Para resolver este problema, edite manualmente o arquivo **/lib/udev/rules.d/90-rdma-hw-modules.rules** para permitir o carregamento automático de **i40iw**.

Observe também que se houver outro dispositivo RDMA instalado com um dispositivo i40e na mesma máquina, o dispositivo não i40e RDMA aciona o serviço **rdma**, que carrega todos os módulos de pilha RDMA habilitados, incluindo o módulo **i40iw**.

(BZ#1623712)

O sistema às vezes fica sem resposta quando muitos dispositivos são conectados

Quando o Red Hat Enterprise Linux 8 configura um grande número de dispositivos, um grande número de mensagens do console ocorre no console do sistema. Isto acontece, por exemplo, quando há um grande número de unidades lógicas (LUNs), com múltiplos caminhos para cada LUN. O fluxo de mensagens do console, além de outros trabalhos que o kernel está fazendo, pode fazer com que o cão de guarda do kernel force o pânico do kernel porque o kernel parece estar pendurado.

Como a varredura acontece no início do ciclo de inicialização, o sistema torna-se insensível quando muitos dispositivos são conectados. Isto normalmente ocorre no momento da inicialização.

Se **o kdump** for ativado em sua máquina durante o evento de varredura do dispositivo após a inicialização, o bloqueio rígido resulta na captura de uma imagem **vmcore**.

Para contornar este problema, aumente o temporizador de bloqueio do cão de guarda. Para isso, adicione a opção **watchdog_thresh=*N*** à linha de comando do kernel. Substitua ***N*** com o número de segundos:

- Se você tiver menos de mil dispositivos, use **30**.
- Se você tiver mais de mil dispositivos, use **60**.

Para armazenamento, o número de dispositivos é o número de caminhos para todos os LUNs: geralmente, o número de dispositivos **/dev/sd***.

Após a aplicação da solução, o sistema não se torna mais insensível quando se configura uma grande quantidade de dispositivos.

(BZ#1598448)

A KSM às vezes ignora as políticas de memória NUMA

Quando o recurso de memória compartilhada do kernel (KSM) é ativado com o parâmetro **merge_across_nodes=1**, o KSM ignora as políticas de memória definidas pela função mbind(), e pode fundir páginas de algumas áreas de memória com nós de Acesso Não-Uniforme à Memória (NUMA) que não correspondem às políticas.

Para contornar este problema, desative o KSM ou configure o parâmetro **merge_across_nodes** para **0** se estiver usando a ligação de memória NUMA com QEMU. Como resultado, as políticas de memória NUMA configuradas para a KVM VM funcionarão como esperado.

(BZ#1153521)

O motorista qede pendura o DNI e o torna inutilizável

Devido a um bug, o driver **qede** para as séries 41000 e 45000 QLogic NICs pode fazer com que as operações de atualização do Firmware e debug de coleta de dados falhem e tornar o NIC inutilizável ou em estado suspenso até que o reinício (PCI reset) do host torne o NIC operacional novamente.

Esta questão foi detectada em todos os cenários a seguir:

- ao atualizar o Firmware do NIC usando o driver da caixa de entrada
- ao coletar dados de depuração rodando o comando **ethtool -d ethx**
- rodando o comando **sosreport**, pois inclui **ethtool -d ethx**.
- quando o motorista da caixa de entrada inicia a coleta automática de dados de depuração, tais como timeout IO, timeout de comando da caixa de correio e uma Atenção de Hardware.

Uma futura errata da Red Hat será lançada via Red Hat Bug Advisory (RHBA) para tratar desta questão. Para resolver este problema, crie um caso em https://access.redhat.com/support para solicitar uma correção suportada para o problema até que a RHBA seja lançada.

(BZ#1697310)

Símbolos de árvores Radix foram adicionados às listas de kernel-abi-whitelists

Os seguintes símbolos em árvore radix foram adicionados ao pacote **kernel-abi-whitelists** no Red Hat Enterprise Linux 8:

- **__radix_tree_insert**
- **__radix_tree_next_slot**
- **radix_tree_delete**
- **radix_tree_gang_lookup**
- **radix_tree_gang_lookup_tag**
- **radix_tree_next_chunk**
- **radix_tree_preload**
- **radix_tree_tag_set**

Os símbolos acima não deveriam estar presentes e serão removidos da lista branca RHEL8.

(BZ#1695142)

podman falha no ponto de verificação de um contêiner no RHEL 8

A versão do pacote Checkpoint and Restore In Userspace (CRIU) está desatualizada no Red Hat Enterprise Linux 8. Como conseqüência, o CRIU não suporta o Checkpoint and Restore In Userspace (CRIU) e o utilitário **podman** falha no Checkpoint Contêiner. Ao executar o comando **podman de ponto de verificação de contêineres**, a seguinte mensagem de erro é exibida: 'checkpointing a container requires at least CRIU 31100' (ponto de verificação de um contêiner requer pelo menos CRIU 31100)

(BZ#1689746)

early-kdump e kdump padrão falham se a opção add_dracutmodules =earlykdump for usada no dracut.conf

Atualmente, ocorre uma inconsistência entre a versão do kernel que está sendo instalada para o **kdump inicial** e a versão do kernel para a qual é gerado o **initramfs**. Como conseqüência, a inicialização com o **early-kdump** ativado, o **early-kdump** falha. Além disso, se o **early-kdump** detecta que está sendo incluído em uma imagem padrão do initramfs do **kdump**, ele força uma saída. Portanto, o serviço de **kdump** padrão também falha ao tentar reconstruir **o kdump** initramfs se o **early-kdump** for adicionado como um módulo **de dracut** padrão. Como conseqüência, tanto **o early-kdump** quanto o **kdump** padrão falham. Para contornar este problema, não adicione **add_dracutmodules =earlykdump** ou qualquer configuração equivalente no arquivo **dracut.conf**. Como resultado, o **early-kdump** não é incluído pelo **dracut** por padrão, o que impede que o problema ocorra. Entretanto, se uma imagem do **early-kdump** for necessária, ela tem que ser criada manualmente.

(BZ#1662911)

O núcleo de depuração não inicia no ambiente de captura de falhas no RHEL 8

Devido à natureza exigente de memória do núcleo de depuração, ocorre um problema quando o núcleo de depuração está em uso e um pânico do núcleo é desencadeado. Como conseqüência, o kernel debug não é capaz de inicializar como o kernel de captura, e em seu lugar é gerado um traço de pilha. Para contornar este problema, aumente a memória do kernel debug de acordo. Como resultado, o kernel debug arranca com sucesso no ambiente de captura de falhas.

(BZ#1659609)

A interface de rede é renomeada para kdump -<interface-nomee> quando o fadump é usado

Quando o dump assistido por firmwares**(fadump**) é utilizado para capturar um vmcore e armazená-lo em uma máquina remota usando o protocolo SSH ou NFS, a interface de rede é renomeada para **kdump -<interface-nome_TERNGREGUNA-** se **<interface-nome_TERNGREGUNA-** for genérica, por exemplo, *eth#, ou net#. Este problema ocorre porque os scripts de captura vmcore no disco inicial da RAM**(initrd**) adicionam o prefixo kdump- ao nome da interface de rede para garantir a nomeação persistente. O mesmo **initrd** é usado também para uma inicialização regular, de modo que o nome da interface é alterado também para o kernel de produção.

(BZ#1745507)

5.5.4. Gestão de software

A execução de uma lista de yum sob um usuário não-rootalista causa YUM crash

Ao executar o comando da **lista yum** sob um usuário não-root após o pacote **libdnf** ter sido atualizado, **YUM** pode terminar inesperadamente. Se você acertar este bug, execute o comando **yum list** sob root para resolver o problema. Como resultado, tentativas subsequentes de executar a **yum list** sob um usuário não-root não causam mais o erro **YUM**.

(BZ#1642458)

YUM v4 pula os repositórios indisponíveis por padrão

YUM v4 padrão para a configuração "skip_if_unavailable=True=True=" para todos os repositórios. Como consequência, se o repositório necessário não estiver disponível, os pacotes do repositório não são considerados nas operações de instalação, busca ou atualização. Posteriormente, alguns comandos **yum** e scripts baseados no yum têm sucesso com o código de saída 0, mesmo que não haja repositórios disponíveis.

Atualmente, não há outra alternativa disponível a não ser atualizar o pacote **libdnf**.

(BZ#1679509)

5.5.5. Serviços de infra-estrutura

As utilidades nslookup e host ignoram respostas de servidores de nomes com recursividade não disponíveis

Se mais servidores de nomes estiverem configurados e não houver recorrência disponível para um servidor de nomes, as utilidades **nslookup** e **host** ignoram as respostas de tal servidor de nomes, a menos que seja o último a ser configurado. No caso do último servidor de nomes configurado, a resposta é aceita mesmo sem a bandeira de **recursividade disponível**. Entretanto, se o último servidor de nomes configurado não estiver respondendo ou for inalcançável, a resolução do nome falha.

Trabalhar em torno do problema:

- Garantir que os servidores de nomes configurados sempre respondam com o conjunto de bandeiras de **recursividade disponível**.
- Permitir a recorrência para todos os clientes internos.

Para solucionar o problema, você também pode usar o utilitário de **escavação** para detectar se a recorrência está disponível ou não.

(BZ#1599459)

5.5.6. Conchas e ferramentas de linha de comando

A encadernaçãoPython do pacote net-snmp não está disponível

O conjunto de ferramentas **Net-SNMP** não fornece ligação para o **Python 3**, que é a implementação padrão do **Python** no RHEL 8. Consequentemente, os pacotes **python-net-snmp**, **python2-net-snmp**, ou **python3-net-snmp** não estão disponíveis no RHEL 8.

(BZ#1584510)

systemd em modo de depuração produz mensagens de registro desnecessárias

O **sistema** e o gerente de serviços em modo de depuração produzem mensagens de registro desnecessárias que começam com:

```
"Falha em adicionar regra para chamada de sistema ..."
```

Liste as mensagens executando:

```
journalctl -b _PID=1
```

Estas mensagens de depuração são inofensivas, e você pode ignorá-las com segurança.

Atualmente, não há nenhuma solução disponível.

(BZ#1658691)

ksh com a armadilha KEYBD mishandles caracteres multibyte

A Korn Shell (KSH) é incapaz de lidar corretamente com caracteres multibyte quando a armadilha **KEYBD** está habilitada. Conseqüentemente, quando o usuário entra, por exemplo, caracteres

japoneses, **ksh** exibe uma string incorreta. Para contornar este problema, desabilite a armadilha **KEYBD** no arquivo **/etc/kshrc** comentando a seguinte linha:

```
trap keybd_trap KEYBD
```

Para mais detalhes, veja uma solução de Knowledgebase relacionada.

(BZ#1503922)

5.5.7. Linguagens de programação dinâmica, servidores web e de banco de dados

Os servidores de banco de dados não podem ser instalados em paralelo

Os módulos **mariadb** e **mysql** não podem ser instalados em paralelo no RHEL 8.0 devido a pacotes de RPM conflitantes.

Por projeto, é impossível instalar mais de uma versão (fluxo) do mesmo módulo em paralelo. Por exemplo, você precisa escolher apenas um dos fluxos disponíveis do módulo **postgresql**, seja **10** (padrão) ou **9,6**. A instalação paralela de componentes é possível na Red Hat Software Collections para RHEL 6 e RHEL 7. No RHEL 8, diferentes versões de servidores de banco de dados podem ser usadas em containers.

(BZ#1566048)

Problemas no mod_cgid logging

Se o módulo **mod_cgid** Apache httpd for usado sob um módulo de multiprocessamento de rosca (MPM), que é a situação padrão no RHEL 8, os seguintes problemas de registro ocorrem:

- A saída **stderr** do script CGI não é prefixada com informações de timestamp padrão.
- A saída **stderr** do script CGI não é redirecionada corretamente para um arquivo de log específico para o **VirtualHost**, se configurado.

(BZ#1633224)

O IO::Socket::SSL Perl módulo não suporta TLS 1.3

Novas características do protocolo TLS 1.3, tais como o reinício da sessão ou autenticação pós-venda, foram implementadas na biblioteca RHEL 8 **OpenSSL**, mas não no módulo **Net::SSLeay** Perl, e portanto não estão disponíveis no módulo **IO::Socket::SSL** Perl. Conseqüentemente, a autenticação do certificado do cliente pode falhar e as sessões de restabelecimento podem ser mais lentas do que com o protocolo TLS 1.2.

Para contornar este problema, desabilite o uso do TLS 1.3 definindo a opção **SSL_version** para o valor **!TLSv1_3** ao criar um objeto **IO::Socket::SSL**.

(BZ#1632600)

A documentação Scala gerada é ilegível

Ao gerar documentação usando o comando **scaladoc**, a página HTML resultante é inutilizável devido à falta de recursos JavaScript.

(BZ#1641744)

5.5.8. Desktop

qxl não funciona em VMs baseadas em Wayland

O driver **qxl** não é capaz de fornecer características de ajuste do modo kernel em certos hypervisors. Consequentemente, os gráficos baseados no protocolo Wayland não estão disponíveis para máquinas virtuais (VMs) que usam **qxl**, e a tela de login baseada no Wayland não inicia.

Para contornar o problema, use :

- O servidor de exibição **Xorg** ao invés de **GNOME Shell on Wayland** em VMs baseadas nos gráficos da QuarkXpress Element Library (QXL).

Ou

- O **virtio** driver em vez do **qxl** driver para suas VMs.

(BZ#1641763)

O prompt do console não é exibido ao executar o systemctl isolar multi-usuário.target

Ao executar o **systemctl isola** o comando **multi-user.target** do Terminal GNOME em uma sessão do GNOME Desktop, apenas um cursor é exibido, e não o prompt do console. Para contornar o problema, pressione as teclas **Ctrl Alt F2**. Como resultado, o prompt do console é exibido.

O comportamento se aplica tanto para **GNOME Shell on Wayland** quanto para **X.Org** display server.

(BZ#1678627)

5.5.9. Infra-estruturas gráficas

A área de trabalho em X.Org fica pendurada quando se muda para resoluções de tela baixas

Ao usar a área de trabalho do GNOME com o servidor de exibição **X.Org**, a área de trabalho fica sem resposta se você tentar mudar a resolução da tela para valores baixos. Para contornar o problema, não defina a resolução da tela para um valor inferior a 800 × 600 pixels.

(BZ#1655413)

radeon falha em reiniciar o hardware corretamente

O driver do kernel **radeon** atualmente não reinicia corretamente o hardware no contexto do kexec. Em vez disso, **o radeon** cai, o que faz com que o resto do serviço **kdump** falhe.

Para contornar este problema, faça **uma** lista negra em **kdump** adicionando a seguinte linha ao arquivo **/etc/kdump.conf**:

```
dracut_args --omit-drivers "radeon"
force_rebuild 1
```

Reinicie a máquina e **kdump**. Após iniciar **kdump**, a linha **force_rebuild 1** pode ser removida do arquivo de configuração.

Note que, neste cenário, nenhum gráfico estará disponível durante **kdump**, mas **kdump** funcionará com sucesso.

(BZ#1694705)

5.5.10. Habilitação do hardware

O status de escravo de backup MII não funciona quando se usa o monitor de link ARP

Por padrão, os dispositivos gerenciados pelo driver i40e, fazem a poda de origem, que deixa cair os pacotes que têm o endereço MAC (Media Access Control) de origem que corresponde a um dos filtros de recepção. Como conseqüência, o status de Interface independente de mídia escrava de backup (MII) não funciona quando se usa o monitoramento do Protocolo de Resolução de Endereço (ARP) na ligação de canais. Para contornar este problema, desabilite a poda de fonte através do seguinte comando:

```
# ethtool --set-priv-flags <ethX> desabilitar-source-pruning on
```

Como resultado, o status de escravo de reserva MII funcionará como esperado.

(BZ#1645433)

O cão de guarda da HP NMI em alguns casos não gera um depósito de lixo

O motorista **hpwdt** para o cão de guarda HP NMI às vezes não é capaz de reivindicar uma interrupção não-máscara (NMI) gerada pelo temporizador HPE, porque o NMI foi consumido pelo motorista **perfmon**.

Como conseqüência, **hpwdt** em alguns casos não pode chamar de pânico para gerar um lixão de emergência.

(BZ#1602962)

5.5.11. Gestão da Identidade

O cache de credenciais KCM não é adequado para um grande número de credenciais em um único cache de credenciais

O Gerente de Credenciais Kerberos (KCM) pode lidar com ccache de até 64 kB. Se ele contiver muitas credenciais, as operações Kerberos, tais como **kinit**, falham devido a um limite de código rígido no buffer usado para transferir dados entre o componente **sssd-kcm** e o banco de dados subjacente.

Para contornar este problema, adicione a opção **ccache_storage = memory** na seção **kcm** do arquivo **/etc/sssd/sssd.conf**. Isto instrui o **kcm** a responder a armazenar apenas as caches de credenciais em memória, não de forma persistente. Se você fizer isso, reiniciar o sistema ou **sssd-kcm** limpa os caches de credenciais.

(BZ#1448094)

A mudança /etc/nsswitch.conf requer uma reinicialização manual do sistema

Qualquer alteração no arquivo **/etc/nsswitch.conf**, por exemplo rodando o comando **authselect select profile_id**, requer uma reinicialização do sistema para que todos os processos relevantes usem a versão atualizada do arquivo **/etc/nsswitch.conf**. Se uma reinicialização do sistema não for possível, reinicie o serviço que une seu sistema ao Active Directory, que é o **System Security Services Daemon** (SSSD) ou **winbind**.

(BZ#1657295)

Valores de tempo limite conflitantes impedem que o SSSD se conecte aos servidores

Alguns dos valores padrão de timeout relacionados às operações de failover utilizadas pelo System Security Services Daemon (SSSD) são conflitantes. Consequentemente, o valor de timeout reservado ao SSSD para falar com um único servidor impede que o SSSD tente outros servidores antes da

operação de conexão como um todo. Para contornar o problema, defina o valor do parâmetro **ldap_opt_timeout** maior que o valor do parâmetro **dns_resolver_timeout**, e defina o valor do parâmetro **dns_resolver_timeout** maior que o valor do parâmetro **dns_resolver_op_timeout**.

(BZ#1382750)

O SSSD só pode procurar certificados únicos nas anulações de identificação

Quando várias sobreposições de ID contêm o mesmo certificado, o System Security Services Daemon (SSSD) não consegue resolver consultas para os usuários que correspondem ao certificado. Uma tentativa de procurar esses usuários não devolve nenhum usuário. Observe que procurar usuários usando seu nome de usuário ou UID funciona como esperado.

(BZ#1446101)

O SSSD não lida corretamente com múltiplas regras de correspondência de certificados com a mesma prioridade

Se um determinado certificado corresponde a várias regras de correspondência de certificados com a mesma prioridade, o System Security Services Daemon (SSSD) utiliza apenas uma das regras. Como alternativa, use uma única regra de correspondência de certificado cujo filtro LDAP consiste nos filtros das regras individuais concatenadas com o | (ou) operador. Para exemplos de regras de correspondência de certificados, consulte a página de manual sss-certamp(5).

(BZ#1447945)

SSSD devolve a adesão incorreta ao grupo LDAP para usuários locais

Se o Serviço de Segurança do Sistema Daemon (SSSD) atende usuários de arquivos locais, o provedor de arquivos não inclui membros de grupo de outros domínios. Como conseqüência, se um usuário local é membro de um grupo LDAP, o comando **id local_user** não retorna a filiação do usuário ao grupo LDAP. Para contornar o problema, ou reverta a ordem dos bancos de dados onde o sistema está procurando a filiação em grupo de usuários no arquivo **/etc/nsswitch.conf**, substituindo **arquivos sss** por **arquivos sss**, ou desabilite o domínio de **arquivos** implícito, adicionando

```
enable_files_domain=False
```

para a seção **[sssd]** no arquivo **/etc/sssd/sssd.conf**.

Como resultado, **id local_user** retorna a adesão correta ao grupo LDAP para usuários locais.

(BZ#1652562)

As regras do sudo podem não funcionar com id_provider=ad se as regras do sudo referirem nomes de grupos

Daemon System Security Services (SSSD) não resolve nomes de grupos do Active Directory durante a operação do **initgroups** por causa de uma otimização da comunicação entre AD e SSSD usando um cache. A entrada do cache contém apenas um Identificador de Segurança (SID) e não nomes de grupos até que o grupo seja solicitado por nome ou ID. Portanto, as regras do sudo não correspondem ao grupo AD, a menos que os grupos sejam totalmente resolvidos antes da execução do sudo.

Para contornar este problema, você precisa desativar a otimização: Abra o arquivo **/etc/sssd/sssd.conf** e adicione o **ldap_use_tokengroups =** parâmetro **falso** na seção **[domain/example.com]**.

(BZ#1659457)

As configurações padrão do PAM para o usuário do sistema foram alteradas no RHEL 8, o que pode influenciar o comportamento do SSSD

A pilha de módulos de autenticação Pluggable (PAM) mudou no Red Hat Enterprise Linux 8. Por exemplo, a sessão do usuário **do sistema** agora inicia uma conversa PAM usando o serviço PAM **do usuário do sistema**. Este serviço agora inclui recursivamente o serviço PAM **system-auth**, que pode incluir a interface **pam_sss.so**. Isto significa que o controle de acesso SSSD é sempre chamado.

Esteja ciente da mudança ao projetar regras de controle de acesso para os sistemas RHEL 8. Por exemplo, você pode adicionar o serviço **de usuário do sistema** à lista de serviços permitidos.

Observe que para alguns mecanismos de controle de acesso, tais como IPA HBAC ou AD GPOs, o serviço **de usuário do sistema** foi adicionado à lista de serviços permitidos por padrão e você não precisa tomar nenhuma ação.

(BZ#1669407)

O servidor IdM não funciona em FIPS

Devido a uma implementação incompleta do conector SSL para Tomcat, um servidor de Gerenciamento de Identidade (IdM) com um servidor de certificados instalado não funciona em máquinas com o modo FIPS habilitado.

(BZ#1673296)

Samba nega o acesso ao usar o plug-in de mapeamento de identificação sss

Para usar o Samba como um servidor de arquivos em um host RHEL unido a um domínio Active Directory (AD), o serviço Samba Winbind deve estar rodando mesmo que o SSSD seja usado para gerenciar usuários e grupos do AD. Se você entrar no domínio usando o comando **join --client-software=sssd** ou sem especificar o parâmetro **--client-software** neste comando, **o reino** cria apenas o arquivo **/etc/sssd/sssd.conf**. Quando você executa Samba no membro do domínio com esta configuração e adiciona uma configuração que usa o back end de mapeamento de ID **sss** ao arquivo **/etc/samba/smb.conf** para compartilhar diretórios, mudanças no back end de mapeamento de ID podem causar erros. Consequentemente, o Samba nega acesso aos arquivos em certos casos, mesmo que o usuário ou grupo exista e seja conhecido pelo SSSD.

Se você planeja atualizar de uma versão anterior do RHEL e o parâmetro **ldap_id_mapping** no arquivo **/etc/sssd/sssd.conf** estiver definido para **True**, que é o padrão, não há nenhuma solução de trabalho disponível. Neste caso, não atualize o host para RHEL 8 até que o problema tenha sido resolvido.

Possíveis soluções em outros cenários:

- Para novas instalações, entre no domínio usando o comando **join --client-software=winbind**. Isto configura o sistema para usar Winbind em vez de SSSD para todas as buscas de usuários e grupos. Neste caso, o Samba usa o plug-in de mapeamento de ID do **rid** ou **ad** ID em **/etc/samba/smb.conf**, dependendo se a opção **--automatic-id-mapping** foi configurada para **sim** (padrão) ou **não**. Se você planeja usar SSSD no futuro ou em outros sistemas, usar **--automatic-id-mapping=no** permite uma migração mais fácil, mas requer que você armazene os POSIX UIDs e GIDs no AD para todos os usuários e grupos.

- Ao atualizar de uma versão RHEL anterior, e se o parâmetro **ldap_id_mapping** no arquivo **/etc/sssd/sssd.conf** estiver configurado para **Falso** e o sistema usar os atributos **uidNumber** e **gidNumber** do AD para o mapeamento de ID:

    1. Alterar a **configuração do idmap <domínio> : backend =** entrada **sss** no arquivo **/etc/samba/smb.conf** para **idmap config <domínio> : backend = ad**

2. Use o comando **systemctl status winbind** para reiniciar o Winbind.

(BZ#1657665)

O serviço nuxwdog falha em ambientes HSM e requer a instalação do pacote keyutils em ambientes não-HSM

O serviço **nuxwdog** watchdog foi integrado ao Sistema de Certificação. Como conseqüência, **o nuxwdog** não é mais fornecido como um pacote separado. Para usar o serviço watchdog, instale o pacote **pki-server**.

Observe que o serviço **nuxwdog** tem seguido os problemas conhecidos:

- O serviço **nuxwdog** não funciona se você usar um módulo de armazenamento de hardware (HSM). Para esta questão, não há nenhuma solução disponível.
- Em um ambiente não-HSM, o Red Hat Enterprise Linux 8.0 não instala automaticamente o pacote **keyutils** como uma dependência. Para instalar o pacote manualmente, use o comando **dnf install keyutils**.

(BZ#1652269)

A adição de anulações de ID de usuários AD funciona somente no IdM CLI

Atualmente, a adição de anulações de ID de usuários do Active Directory (AD) a grupos de Gerenciamento de Identidade (IdM) com a finalidade de conceder acesso a funções de gerenciamento falha na IDM Web UI. Para contornar o problema, use antes a interface de linha de comando (CLI) do IdM.

Note que se você instalou o pacote **ipa-idoverride-memberof-plugin** no servidor IdM depois de executar previamente certas operações usando o utilitário **ipa**, a Red Hat recomenda limpar o cache do utilitário **ipa** para forçá-lo a atualizar sua visão sobre os metadados do servidor IdM.

Para isso, remova o conteúdo do diretório **~/.cache/ipa** para o usuário sob o qual o utilitário **ipa** é executado. Por exemplo, para root:

```
# rm -r /root/.cache/ipa
```

(BZ#1651577)

Nenhuma informação sobre os registros DNS necessários exibidos ao permitir o suporte à confiança do AD na IdM

Ao permitir o suporte à confiança do Active Directory (AD) na instalação do Red Hat Enterprise Linux Identity Management (IdM) com gerenciamento DNS externo, nenhuma informação sobre os registros DNS necessários é exibida. A confiança da floresta no AD não é bem sucedida até que os registros DNS requeridos sejam adicionados. Para contornar este problema, execute o comando 'ipa dns-update-system-records --dry-run' para obter uma lista de todos os registros DNS requeridos pelo IdM. Quando o DNS externo para o domínio IdM definir os registros DNS necessários, é possível estabelecer a confiança da floresta no AD.

(BZ#1665051)

5.5.12. Compiladores e ferramentas de desenvolvimento

Funções sintéticas geradas pelo GCC confundem SystemTap

A otimização do GCC pode gerar funções sintéticas para cópias parcialmente simplificadas de outras funções. Ferramentas como SystemTap e GDB não podem distinguir estas funções sintéticas das funções reais. Como conseqüência, o SystemTap pode colocar sondas em ambos os pontos de entrada de funções sintéticas e reais, e assim registrar múltiplos acertos de sondas para uma única chamada de função real.

Para contornar este problema, os scripts do SystemTap devem ser adaptados com medidas como a detecção de recorrência e a supressão de sondas relacionadas a funções parciais simplificadas. Por exemplo, um script

```
sonda kernel.function({\i1}"can_nice}").call {\i}
```

pode tentar evitar o problema descrito a seguir:

```
global in_can_nice%

probe kernel.function("can_nice").call {
  in_can_nice[tid()] ++;
  if (in_can_nice[tid()] > 1) { next }
  /* code for real probe handler */
}

probe kernel.function("can_nice").return {
  in_can_nice[tid()] --;
}
```

Observe que este roteiro de exemplo não leva em conta todos os cenários possíveis, tais como a falta de kretprobes ou kretprobes, ou a verdadeira repetição pretendida.

(BZ#1169184)

A ferramenta ltrace não informa as chamadas de função

Devido a melhorias no endurecimento binário aplicadas a todos os componentes RHEL, a ferramenta **ltrace** não pode mais detectar chamadas de função em arquivos binários provenientes de componentes RHEL. Como conseqüência, a saída **ltrace** está vazia porque não relata nenhuma chamada detectada quando usada em tais arquivos binários. Não há nenhuma solução atualmente disponível.

Como nota, **ltrace** pode relatar corretamente as chamadas em arquivos binários personalizados construídos sem as respectivas bandeiras de endurecimento.

(BZ#1618748, BZ#1655368)

5.5.13. Sistemas de arquivo e armazenamento

Incapaz de descobrir um alvo iSCSI usando o pacote iscsiuio

O Red Hat Enterprise Linux 8 não permite o acesso simultâneo às áreas de registro PCI. Como conseqüência, **não** foi **possível definir** o erro dos **parâmetros de rede do host (err 29)** e a conexão ao portal de descoberta falhou. Para contornar este problema, configure o parâmetro **iomem=relaxado** do kernel na linha de comando do kernel para a descarga do iSCSI. Isto envolve especificamente qualquer descarregamento usando o driver **bnx2i**. Como resultado, a conexão ao portal de descoberta é agora bem sucedida e o pacote **iscsiuio** agora funciona corretamente.

(BZ#1626629)

Os volumes da VDO perdem o conselho de deduplicação depois de se mudarem para uma plataforma diferente -endian

Virtual Data Optimizer (VDO) escreve o cabeçalho do índice do Serviço de Deduplicação Universal (UDS) no formato endian nativo de sua plataforma. O VDO considera o índice UDS corrompido e o sobrescreve com um novo índice em branco se você mover seu volume VDO para uma plataforma que usa um endian diferente.

Como conseqüência, qualquer conselho de deduplicação armazenado no índice UDS antes de ser sobregravado é perdido. A VDO é então incapaz de deduplicar os dados recém-escritos contra os dados que foram armazenados antes de se mover o volume, levando a uma menor economia de espaço.

(BZ#1696492)

A opção de montagem XFS DAX é incompatível com extensões compartilhadas de dados copy-on-write

Um sistema de arquivo XFS formatado com o recurso de cópia compartilhada de dados não é compatível com a opção **-o dax** mount. Como conseqüência, a montagem de tal sistema de arquivo com **-o dax** falha.

Para contornar o problema, formate o sistema de arquivos com a opção **Refink=0** metadados para desabilitar a cópia compartilhada de extensões de dados:

```
# mkfs.xfs -m reflink=0 block-device
```

Como resultado, a montagem do sistema de arquivo com **-o dax** é um sucesso.

Para mais informações, consulte Criando um espaço de nomes de sistema de arquivos DAX em um NVDIMM.

(BZ#1620330)

Alguns drivers SCSI podem às vezes usar uma quantidade excessiva de memória

Alguns motoristas SCSI usam uma quantidade maior de memória do que na RHEL 7. Em certos casos, tais como a criação de vPort em um adaptador de barramento host Fibre Channel (HBA), o uso de memória pode ser excessivo, dependendo da configuração do sistema.

O aumento do uso de memória é causado pela pré-alocação de memória na camada de bloco. Tanto a programação do dispositivo de blocos multifilas (BLK-MQ) quanto a pilha SCSI multifilas (SCSI-MQ) pré-alocam a memória para cada solicitação de E/S no RHEL 8, levando ao aumento do uso de memória.

(BZ#1733278)

5.5.14. Trabalho em rede

nftables não suporta os tipos de conjuntos IP multidimensionais

A estrutura de filtragem de pacotes **nftables** não suporta tipos de conjuntos com concatenações e intervalos. Consequentemente, não é possível usar tipos de conjunto IP multidimensional, como **hash:net, porta**, com **nftables**.

Para contornar este problema, use a estrutura **iptables** com a ferramenta **ipset** se você precisar de tipos de conjuntos IP multidimensionais.

(BZ#1593711)

O alvo TRACE na página man do iptables-extensions(8) não se refere à variante nf_tables

A descrição do alvo **TRACE** na página man do **iptables-extensions(8**) refere-se apenas à variante **compatriota**, mas o Red Hat Enterprise Linux (RHEL) 8.0 usa a variante **nf_tables**. O utilitário **iptables baseado em nftables** no RHEL usa a expressão **meta nftrace** internamente. Portanto, o kernel não imprime eventos **TRACE** no log do kernel, mas os envia para o espaço do usuário. Entretanto, a página man não faz referência ao utilitário de linha de comando **xtables-monitor** para exibir esses eventos.

(BZ#1658734)

RHEL 8 mostra o status de uma ligação 802.3ad como "Churned" depois que uma chave não estava disponível por um período de tempo prolongado

Atualmente, quando você configura um vínculo de rede 802.3ad e o switch está desligado por um longo período de tempo, o Red Hat Enterprise Linux mostra corretamente o status do vínculo como "Churned", mesmo depois que a conexão retorna a um estado de funcionamento. No entanto, este é o comportamento pretendido, já que o status de "rotacionado" visa dizer ao administrador que ocorreu uma interrupção significativa da conexão. Para limpar este status, reinicie a ligação de rede ou reinicialize o host.

(BZ#1708807)

O comando ebtables não suporta broute table

O comando **ebtables baseado em nftables** no Red Hat Enterprise Linux 8.0 não suporta a tabela de **broute**. Conseqüentemente, os usuários não podem usar este recurso.

(BZ#1649790)

O tráfego de rede IPsec falha durante a descarga de IPsec quando o GRO é desativado

Não se espera que a descarga IPsec funcione quando a descarga de recepção genérica (GRO) estiver desativada no dispositivo. Se o descarregamento de IPsec estiver configurado em uma interface de rede e o GRO estiver desabilitado nesse dispositivo, o tráfego de rede IPsec falha.

Para contornar este problema, mantenha o GRO ativado no dispositivo.

(BZ#1649647)

NetworkManager agora usa o plug-in interno DHCP por padrão

NetworkManager suporta os plug-ins DHCP **internos** e **dhclient**. Por default, **NetworkManager** no Red Hat Enterprise Linux (RHEL) 7 usa o **dhclient** e o RHEL 8 o plug-in **interno**. Em certas situações, os plug-ins se comportam de forma diferente. Por exemplo, **o dhclient** pode usar configurações adicionais especificadas no diretório **/etc/dhcp/**.

Se você atualizar de RHEL 7 para RHEL 8 e **NetworkManager** se comportar de forma diferente, adicione a seguinte configuração à seção **[principal]** no arquivo **/etc/NetworkManager/NetworkManager.conf** para usar o plug-in **dhclient**:

```
[main]
dhcp=dhclient
```

(BZ#1571655)

Opções avançadas de VPN baseadas em IPsec não podem ser alteradas utilizando o gnome-control-center

Ao configurar uma conexão **VPN baseada em IPsec** usando a aplicação **gnome-control-center**, o diálogo **Avançado** exibirá apenas a configuração, mas não permitirá fazer nenhuma alteração. Como conseqüência, os usuários não podem alterar nenhuma opção IPsec avançada. Para contornar este problema, use o **editor de nm-conexão** ou ferramentas **nmcli** para realizar a configuração das propriedades avançadas.

(BZ#1697326)

Os arquivos /etc/hosts.allow e /etc/hosts.deny contêm informações imprecisas

O pacote tcp_wrappers é removido no Red Hat Enterprise Linux (RHEL) 8, mas não seus arquivos, /etc/hosts.allow e /etc/hosts.deny. Como conseqüência, estes arquivos contêm informações desatualizadas, o que não é aplicável para o RHEL 8.

Para contornar este problema, use regras de firewall para filtrar o acesso aos serviços. Para filtragem baseada em nomes de usuário e hostnames, use a configuração específica da aplicação.

(BZ#1663556)

A desfragmentação de IP não pode ser sustentável sob sobrecarga de tráfego na rede

No Red Hat Enterprise Linux 8, o fio do núcleo de coleta de lixo foi removido e os fragmentos de IP expiram apenas no tempo limite. Como resultado, o uso de CPU sob Negação de Serviço (DoS) é muito menor, e a taxa máxima de queda de fragmentos sustentável é limitada pela quantidade de memória configurada para a unidade de remontagem de IP. Com as configurações padrão, as cargas de trabalho que requerem tráfego fragmentado na presença de queda de pacotes, reordenamento de pacotes ou muitos fluxos fragmentados simultâneos podem incorrer em regressão de desempenho relevante.

Neste caso, os usuários podem usar o ajuste apropriado do cache de fragmentação IP no diretório **/proc/sys/net/ipv4** definindo a variável **ipfrag_high_thresh** para limitar a quantidade de memória e a variável **ipfrag_time** para manter por segundos um fragmento de IP na memória. Por exemplo, a variável ipfrag_high_thresh,

echo 419430400 > /proc/sys/net/ipv4/ipfrag_high_thresh echo 1 > /proc/sys/net/ipv4/ipfrag_time

O acima se aplica ao tráfego IPv4. Para IPv6, as sintonizações relevantes são: **ip6frag_high_thresh** e **ip6frag_time** no diretório **/proc/sys/net/ipv6/**.

Observe que qualquer carga de trabalho dependente de tráfego fragmentado de alta velocidade pode causar problemas de estabilidade e desempenho, especialmente com quedas de pacotes, e esse tipo de implantações são altamente desencorajadas na produção.

(BZ#1597671)

Mudança do nome da interface de rede no RHEL 8

No Red Hat Enterprise Linux 8, o mesmo esquema consistente de nomenclatura de dispositivos de rede é usado por default como no RHEL 7. Entretanto, alguns drivers do kernel, como **e1000e**, **nfp**, **qede**, **sfc**, **tg3** e **bnxt_en** mudaram seu nome consistente em uma nova instalação do RHEL 8. Entretanto, os nomes são preservados na atualização a partir do RHEL 7.

(BZ#1701968)

5.5.15. Segurança

libselinux-python só está disponível através de seu módulo

O pacote **libselinux-python** contém apenas ligas Python 2 para o desenvolvimento de aplicações SELinux e é usado para compatibilidade retroativa. Por esta razão, **libselinux-python** não está mais disponível nos repositórios padrão RHEL 8 através do comando **dnf install libselinux-python**.

Para contornar este problema, habilite os módulos **libselinux-python** e **python27**, e instale o pacote **libselinux-python** e suas dependências com os seguintes comandos:

```
# dnf module enable libselinux-python
# dnf install libselinux-python
```

Alternativamente, instale **libselinux-python** usando seu perfil de instalação com um único comando:

```
# módulo dnf instalar libselinux-python:2.8/comum
```

Como resultado, você pode instalar **libselinux-python** usando o respectivo módulo.

(BZ#1666328)

a libssh não cumpre com a política de criptografia do sistema

A biblioteca **libssh** não segue as configurações de política criptográfica de todo o sistema. Como conseqüência, o conjunto de algoritmos suportados não é alterado quando o administrador muda o nível de políticas criptográficas usando o comando **update-crypto-policies**.

Para contornar este problema, o conjunto de algoritmos anunciados precisa ser definido individualmente por cada aplicação que utiliza **a libssh**. Como resultado, quando o sistema é definido ao nível da política LEGACY ou FUTURE, as aplicações que utilizam **a libssh** se comportam de forma inconsistente quando comparadas ao **OpenSSH**.

(BZ#1646563)

Certas cordas prioritárias rsyslog não funcionam corretamente

O suporte para a cadeia de prioridade **GnuTLS** para **imtcp** que permite um controle fino sobre a criptografia não está completo. Conseqüentemente, as seguintes cadeias de prioridade não funcionam corretamente no **rsyslog**:

```
NENHUMA: VERS-ALL:-VERS-TLS1.3: MAC-ALL: DHE-RSA: AES-256-GCM: SIGN-RSA-SHA384:
COMP-ALL: GROUP-ALL
```

Para contornar este problema, use apenas cordas de prioridade que funcionem corretamente:

```
NENHUMA: VERS-ALL:-VERS-TLS1.3: MAC-ALL: ECDHE-RSA: AES-128-CBC: SIGN-RSA-SHA1:
COMP-ALL: GROUP-ALL
```

Como resultado, as configurações atuais devem ser limitadas às cordas que funcionam corretamente.

(BZ#1679512)

Efeitos negativos da configuração padrão de registro sobre o desempenho

A configuração padrão do ambiente de registro pode consumir 4 GB de memória ou até mais e os ajustes dos valores limite de taxa são complexos quando o **sistema-journald** está rodando com o **rsyslog**.

Veja os efeitos negativos da configuração de registro padrão da RHEL sobre o desempenho e suas mitigações Artigo da Base de Conhecimento para mais informações.

(JIRA:RHELPLAN-10431)

OpenSCAP rpmverifypackage não funciona corretamente

As chamadas ao sistema **chdir** e **chroot** são chamadas duas vezes pela sonda **rpmverifypackage**. Conseqüentemente, ocorre um erro quando a sonda é utilizada durante uma varredura **OpenSCAP** com conteúdo personalizado de Open Vulnerability and Assessment Language (OVAL).

Para contornar este problema, não use o teste **rpmverifypackage_test** OVAL em seu conteúdo ou use somente o conteúdo do pacote **scap-security-guide** onde o **rpmverifypackage_test** não é usado.

(BZ#1646197)

SCAP Workbench não gera remediações baseadas em resultados a partir de perfis personalizados

O seguinte erro ocorre quando se tenta gerar funções de remediação baseadas em resultados a partir de um perfil personalizado usando a ferramenta **SCAP Workbench**:

> Erro gerando papel de remediação .../remediação.sh: Código de saída da oscap foi 1: [saída truncada]

Para contornar este problema, use o comando **oscap** com a opção **--tailoring-file**.

(BZ#1640715)

Kickstart usa org_fedora_oscap em vez de com_redhat_oscap no RHEL 8

O Kickstart faz referência ao Protocolo de Automação de Conteúdo de Segurança Aberta (OSCAP) Anaconda add-on como **org_fedora_oscap** em vez de **com_redhat_oscap**, o que pode causar confusão. Isto é feito para preservar a retrocompatibilidade com o Red Hat Enterprise Linux 7.

(BZ#1665082)

OpenSCAP rpmverifyfile não funciona

O scanner **OpenSCAP** não muda corretamente o diretório de trabalho atual no modo off-line, e a função **fchdir** não é chamada com os argumentos corretos na sonda **OpenSCAP rpmverifyfile**. Conseqüentemente, o escaneamento de sistemas de arquivos arbitrários usando o comando **oscap-chroot** falha se o **rpmverifyfile_test** for usado em um conteúdo SCAP. Como resultado, o **oscap-chroot** aborta no cenário descrito.

(BZ#163636431)

OpenSCAP não fornece escaneamento offline de máquinas e recipientes virtuais

A refatoração do **OpenSCAP** codebase fez com que certas sondas RPM falhassem na varredura de VM e sistemas de arquivos de containers em modo off-line. Por esse motivo, as seguintes ferramentas foram removidas do pacote **openscap-utils**: **oscap-vm** e **oscap-chroot**. Além disso, o pacote **openscap-containers** foi completamente removido.

(BZ#1618489)

Um serviço de segurança e verificação de conformidade de contêineres não está disponível

No Red Hat Enterprise Linux 7, o utilitário **oscap-docker** pode ser usado para escaneamento de containers Docker baseado em tecnologias atômicas. No Red Hat Enterprise Linux 8, os comandos Docker- e Atomic relacionados a **OpenSCAP** não estão disponíveis. Como resultado, o **oscap-docker** ou um utilitário equivalente para segurança e verificação de conformidade de containers não está disponível no RHEL 8 no momento.

(BZ#1642373)

A biblioteca OpenSSL TLS não detecta se a ficha PKCS#11 suporta a criação de assinaturas RSA ou RSA-PSS brutas

O protocolo **TLS-1.3** requer o suporte para a assinatura do **RSA-PSS**. Se o token **PKCS#11** não suportar assinaturas **RSA** ou **RSA-PSS brutas**, as aplicações servidoras que usam a biblioteca **OpenSSL TLS** não funcionarão com a chave **RSA** se ela estiver na posse do token **PKCS#11**. Como resultado, a comunicação **TLS** falhará.

Para contornar este problema, configure o servidor ou cliente para usar a versão **TLS-1.2** como a versão mais alta do protocolo **TLS** disponível.

(BZ#1681178)

O Apache httpd não inicia se usar uma chave privada RSA armazenada em um dispositivo PKCS#11 e um certificado RSA-PSS

O padrão PKCS#11 não diferencia entre objetos chave RSA e RSA-PSS e usa o tipo **CKKK_RSA** para ambos. Entretanto, OpenSSL usa tipos diferentes para chaves RSA e RSA-PSS. Como consequência, o motor **openssl-pkcs11** não pode determinar que tipo deve ser fornecido ao OpenSSL para objetos chave PKCS#11 RSA. Atualmente, o mecanismo define o tipo de chave como chaves RSA para todos os objetos PKCS#11 **CKKK_RSA**. Quando OpenSSL compara os tipos de uma chave pública RSA-PSS obtida do certificado com o tipo contido em um objeto de chave privada RSA fornecido pelo motor, conclui que os tipos são diferentes. Portanto, o certificado e a chave privada não correspondem. A verificação realizada na função **X509_check_private_key()** OpenSSL retorna um erro neste cenário. O servidor web **httpd** chama esta função em seu processo de inicialização para verificar se o certificado e a chave fornecida coincidem. Como esta verificação sempre falha para um certificado contendo uma chave pública RSA-PSS e uma chave privada RSA armazenada no módulo PKCS#11, **o httpd** falha ao começar a usar esta configuração. Não há nenhuma solução disponível para este problema.

(BZ#1664802)

httpd não inicia se usar uma chave privada ECDSA sem a correspondente chave pública armazenada em um dispositivo PKCS#11

Ao contrário das chaves da RSA, as chaves privadas da ECDSA não contêm necessariamente informações de chave pública. Neste caso, você não pode obter a chave pública de uma chave privada ECDSA. Por este motivo, um dispositivo PKCS#11 armazena informações de chave pública em um objeto separado, seja um objeto de chave pública ou um objeto de certificado. A OpenSSL espera que a estrutura **EVP_PKEY** fornecida por um motor para uma chave privada contenha as informações da chave pública. Ao preencher a estrutura **EVP_PKEY** a ser fornecida à OpenSSL, o mecanismo no pacote **openssl-pkcs11** tenta buscar as informações da chave pública apenas de objetos de chave pública correspondentes e ignora os objetos certificados atuais.

Quando OpenSSL solicita uma chave privada ECDSA do motor, a estrutura **EVP_PKEY** fornecida não contém as informações da chave pública se a chave pública não estiver presente no dispositivo PKCS#11, mesmo quando um certificado correspondente que contenha a chave pública estiver disponível. Como conseqüência, como o servidor web Apache **httpd** chama a função **X509_check_private_key()**, que requer a chave pública, em seu processo de inicialização, **o httpd** não

inicia neste cenário. Para contornar o problema, armazenar tanto a chave privada quanto a pública no dispositivo PKCS#11 ao usar chaves ECDSA. Como resultado, **o httpd** inicia corretamente quando as chaves ECDSA são armazenadas no dispositivo PKCS#11.

(BZ#1664807)

OpenSSH não lida com PKCS #11 URIs para chaves com etiquetas inadequadas corretamente

A suíte OpenSSH pode identificar os pares de chaves por uma etiqueta. A etiqueta pode diferir em chaves privadas e públicas armazenadas em um cartão inteligente. Consequentemente, especificar o PKCS #11 URIs com a parte do objeto (etiqueta da chave) pode impedir que o OpenSSH encontre objetos apropriados no PKCS #11.

Para contornar este problema, especifique PKCS #11 URIs sem a parte do objeto. Como resultado, o OpenSSH é capaz de usar chaves em cartões inteligentes referenciados usando o PKCS #11 URIs.

(BZ#1671262)

A saída de iptables-ebtables não é 100ompatível com ebtables

No RHEL 8, o comando **ebtables** é fornecido pelo pacote **iptables-ebtables**, que contém uma reimplementação da ferramenta **baseada em nftables**. Esta ferramenta tem uma base de código diferente, e sua saída se desvia em aspectos que são negligenciáveis ou escolhas deliberadas de projeto.

Conseqüentemente, ao migrar seus scripts analisando algumas saídas de **ebtables**, ajuste os scripts para refletir o seguinte:

- A formatação do endereço MAC foi alterada para ser fixada em comprimento. Quando necessário, os valores de bytes individuais contêm um zero inicial para manter o formato de dois caracteres por octeto.
- A formatação dos prefixos IPv6 foi alterada para estar em conformidade com a RFC 4291. A parte móvel após o caractere de barra não contém mais uma máscara de rede no formato de endereço IPv6, mas um comprimento de prefixo. Esta alteração se aplica somente a máscaras válidas (contíguas à esquerda), enquanto outras ainda são impressas na formatação antiga.

(BZ#1674536)

curve25519-sha256 não é suportado por padrão no OpenSSH

O algoritmo de troca de chaves SSH **curva25519-sha256** está faltando nas configurações de políticas criptográficas de todo o sistema para o cliente e servidor OpenSSH, apesar de estar em conformidade com o nível padrão de políticas. Como conseqüência, se um cliente ou um servidor usar **a curva25519-sha256** e este algoritmo não for suportado pelo host, a conexão pode falhar.

Para contornar este problema, você pode anular manualmente a configuração das políticas de criptografia de todo o sistema modificando os arquivos **openssh.config** e **opensshshserver.config** no diretório **/etc/crypto-policies/back-ends/back-ends** para o cliente e servidor OpenSSH. Note que esta configuração é sobrescrita com cada mudança das políticas de criptografia de todo o sistema. Veja a página de manual **update-crypto-policies(8)** para mais informações.

(BZ#1678661)

OpenSSL manipula incorretamente as fichas PKCS #11 que não suportam assinaturas RSA ou RSA-PSS brutas

A biblioteca **OpenSSL** não detecta as capacidades relacionadas às chaves de fichas PKCS #11. Conseqüentemente, o estabelecimento de uma conexão TLS falha quando uma assinatura é criada com um token que não suporta assinaturas RSA ou RSA-PSS brutas.

Para contornar o problema, adicione as seguintes linhas após a linha **.include** no final da seção **crypto_policy** no arquivo **/etc/pki/tls/openssl.cnf**:

```
SignatureAlgorithms =
RSA+SHA256:RSA+SHA512:RSA+SHA384:ECDSA+SHA256:ECDSA+SHA512:ECDSA+SHA384
MaxProtocol = TLSv1.2
```

Como resultado, uma conexão TLS pode ser estabelecida no cenário descrito.

(BZ#1685470)

As conexões SSH com sistemas hospedados em VMware não funcionam

A versão atual da suíte **OpenSSH** introduz uma mudança da bandeira padrão de Qualidade de Serviço IP (IPQoS) nos pacotes SSH, que não é tratada corretamente pela plataforma de virtualização VMware. Conseqüentemente, não é possível estabelecer uma conexão SSH com sistemas na VMware.

Para contornar este problema, inclua o **IPQoS=throughput** no arquivo **ssh_config**. Como resultado, as conexões SSH com sistemas hospedados em VMware-hosted funcionam corretamente.

Veja o artigo RHEL 8 rodando na Estação de Trabalho VMWare incapaz de se conectar via SSH a outros hosts da solução Knowledgebase para mais informações.

(BZ#1651763)

5.5.16. Gestão de assinaturas

Nenhuma mensagem é impressa para o sucesso da configuração e desajuste do nível de serviço

Quando o serviço **candlepin** não tem uma funcionalidade 'syspurpose', o gerente de assinatura usa um caminho de código diferente para definir o argumento de **nível de serviço**. Este caminho de código não imprime o resultado da operação. Como conseqüência, nenhuma mensagem é exibida quando o nível de serviço é definido pelo gerente de assinatura. Isto é especialmente problemático quando o conjunto de **nível de serviço** tem um erro de digitação ou não está verdadeiramente disponível.

(BZ#1661414)

os addons syspurpose addons não têm efeito sobre o gerenciador de assinaturas anexar --auto saída.

No Red Hat Enterprise Linux 8, quatro atributos da ferramenta de linha de comando **syspurpose** foram adicionados: **role,use**, **service_level_agreement** e **addons**. Atualmente, apenas **role**, **usage** e **service_level_agreement** afetam a saída da execução do **gerenciador de assinatura anexado --auto** comando. Os usuários que tentarem definir valores para o argumento **dos addons** não observarão qualquer efeito nas assinaturas que são auto-atribuídas.

(BZ#1687900)

5.5.17. Virtualização

Máquinas virtuais ESXi que foram customizadas usando a bota cloud-init e clonada muito lentamente

Atualmente, se o serviço **cloud-init** é utilizado para modificar uma máquina virtual (VM) que roda no VMware ESXi hypervisor para utilizar IP estático e a VM é então clonada, a nova VM clonada em alguns casos leva um tempo muito longo para reiniciar. Isto é causado pela reescrita **em nuvem** do IP estático da VM para DHCP e, em seguida, a busca por uma fonte de dados disponível.

Para contornar este problema, você pode desinstalar **a nuvem -** depois que a VM for inicializada pela primeira vez. Como resultado, as reinicializações subseqüentes não serão desaceleradas.

(BZ#166666961, BZ#1706482)

Permitindo a virtualização aninhada bloqueia a migração ao vivo

Atualmente, o recurso de virtualização aninhado é incompatível com a migração ao vivo. Portanto, permitindo a virtualização aninhada em um host RHEL 8 impede a migração de qualquer máquina virtual (VMs) do host, bem como salvar instantâneos do estado da VM para o disco.

Observe que a virtualização aninhada é atualmente fornecida como uma Visão Tecnológica no RHEL 8, e, portanto, não é suportada. Além disso, a virtualização aninhada é desabilitada por padrão. Se você quiser ativá-la, use os parâmetros do módulo **kvm_intel.nested** ou **kvm_amd.nested.**

(BZ#1689216)

O uso do cloud-init para fornecer máquinas virtuais no Microsoft Azure falha

Atualmente, não é possível usar o utilitário de **nuvem** para fornecer uma máquina virtual RHEL 8 (VM) na plataforma Microsoft Azure. Para contornar este problema, use um dos seguintes métodos:

- Use o pacote **WALinuxAgent** em vez de **cloud-init** para fornecer VMs no Microsoft Azure.
- Adicione a seguinte configuração à seção **[principal]** no arquivo **/etc/NetworkManager/NetworkManager.conf**:

```
[main]
dhcp=dhclient
```

(BZ#1641190)

Geração 2 RHEL 8 máquinas virtuais às vezes não inicializam nos hosts do Hyper-V Server 2016

Ao usar o RHEL 8 como sistema operacional convidado em uma máquina virtual (VM) rodando em um host Microsoft Hyper-V Server 2016, a VM, em alguns casos, falha no boot e retorna ao menu de boot do GRUB. Além disso, o seguinte erro é registrado no registro de eventos do Hyper-V:

```
O sistema operacional convidado informou que falhou com o seguinte código de erro: 0x1E
```

Este erro ocorre devido a um erro de firmware UEFI no host Hyper-V. Para contornar este problema, use o Hyper-V Server 2019 como o host.

(BZ#1583445)

os comandos devirsh iface-* não funcionam de forma consistente

Atualmente, comandos **virsh iface-***, tais como **virsh iface-start** e **virsh iface-destruição**, frequentemente falham devido a dependências de configuração. Portanto, recomenda-se não usar

comandos **virsh iface-*** para configurar e gerenciar as conexões de rede do host. Ao invés disso, use o programa NetworkManager e suas aplicações de gerenciamento relacionadas.

(BZ#1664592)

Extensões de máquinas virtuais Linux para Azure às vezes não funcionam

O RHEL 8 não inclui o pacote **python2** por padrão. Como conseqüência, a execução de extensões de máquinas virtuais Linux para Azure, também conhecidas como **azure-linux-extensions**, em um RHEL 8 VM em alguns casos falha.

Para aumentar a probabilidade de que **azure-linux-extensions** funcionem como esperado, instale o **python2** no RHEL 8 VM manualmente:

```
# yum install python2
```

(BZ#1561132)

5.5.18. Apoio

a ferramenta redhat-support não coleta automaticamente o sosreport do opencase

O comando **redhat-support-tool** não pode criar um arquivo **sosreport**. Para contornar este problema, execute o comando **sosreport** separadamente e depois digite o comando **redhat-support-tool addattachment -c** para carregar o arquivo ou usar a interface web no Portal do Cliente. Como resultado, será criado um caso e o **sosreport** será carregado.

Note que os **findkerneldebugs**, **btextract**, **analisar** comandos de **diagnóstico** não funcionam como esperado e serão corrigidos em lançamentos futuros.

(BZ#1688274)

CAPÍTULO 6. MUDANÇAS NOTÁVEIS NOS RECIPIENTES

Um conjunto de imagens de containers está disponível para o Red Hat Enterprise Linux (RHEL) 8.0. Mudanças notáveis incluem:

- O Docker não está incluído no RHEL 8.0. Para trabalhar com containers, use as ferramentas **podman**, **buildah**, **skopeo**, e **runc**.
  Para informações sobre essas ferramentas e sobre o uso de containers no RHEL 8, consulte Construção, funcionamento e gerenciamento de containers.

- A ferramenta **podman** foi lançada como um recurso totalmente suportado.
  A ferramenta **podman** gerencia cápsulas, imagens de contêineres e contêineres em um único nó. Ela é construída sobre a biblioteca **libpod**, que permite o gerenciamento de containers e grupos de containers, chamados de pods.

  Para saber como utilizar **podman**, consulte Construção, funcionamento e gerenciamento de containers.

- No RHEL 8 GA, as Imagens de Base Universal da Red Hat (UBI) estão disponíveis recentemente. As UBIs substituem algumas das imagens anteriormente fornecidas pela Red Hat, tais como as imagens base padrão e as imagens base RHEL mínimas.
  Ao contrário das imagens mais antigas da Red Hat, as UBIs são livremente redistribuíveis. Isto significa que elas podem ser usadas em qualquer ambiente e compartilhadas em qualquer lugar. Você pode usá-las, mesmo que não seja cliente da Red Hat.

  Para a documentação da UBI, consulte Construção, funcionamento e gerenciamento de contêineres.

- Na RHEL 8 GA, estão disponíveis imagens adicionais de contêineres que fornecem componentes AppStream, para os quais são distribuídas imagens de contêineres com **Red Hat Software Collections** na RHEL 7. Todas estas imagens RHEL 8 são baseadas na imagem base **do ubi8**.

- As imagens dos contêineres ARM para a arquitetura ARM de 64 bits são totalmente suportadas no RHEL 8.

- O contêiner de **ferramentas** de **redhat-suporte** foi removido no RHEL 8. Os **sos** e as ferramentas de **redhat-suporte** são fornecidos no contêiner de **ferramentas de suporte**. Os administradores do sistema também podem usar esta imagem como base para construir a imagem do contêiner de ferramentas do sistema.

- O suporte para recipientes sem raiz está disponível como uma previsão tecnológica no RHEL 8. Recipientes sem raiz são recipientes que são criados e gerenciados por usuários regulares do sistema sem permissões administrativas.

CAPÍTULO 7. INTERNACIONALIZAÇÃO

7.1. RED HAT ENTERPRISE LINUX 8 IDIOMAS INTERNACIONAIS

O Red Hat Enterprise Linux 8 suporta a instalação de múltiplos idiomas e a mudança de idiomas com base em suas exigências.

- Línguas do leste asiático - japonês, coreano, chinês simplificado e chinês tradicional.
- Línguas européias - inglês, alemão, espanhol, francês, italiano, português e russo.

A tabela a seguir lista as fontes e os métodos de entrada fornecidos para vários idiomas principais.

Idioma	Fonte padrão (Font Package)	Métodos de entrada
Inglês	dejavu-sans-fonts	
Francês	dejavu-sans-fonts	
Alemão	dejavu-sans-fonts	
Italiano	dejavu-sans-fonts	
Russo	dejavu-sans-fonts	
Espanhol	dejavu-sans-fonts	
Português	dejavu-sans-fonts	
Chinês simplificado	google-noto-sans-cjk-ttc-fonts, google-noto-serif-cjk-ttc-fonts	ibus-libpinyin, libpinyin
Chinês Tradicional	google-noto-sans-cjk-ttc-fonts, google-noto-serif-cjk-ttc-fonts	ibus-libzhuyin, libzhuyin
Japonês	google-noto-sans-cjk-ttc-fonts, google-noto-serif-cjk-ttc-fonts	ibus-kkc, libkkc
Coreano	google-noto-sans-cjk-ttc-fonts, google-noto-serif-cjk-ttc-fonts	ibus-hangul, libhangu

7.2. MUDANÇAS NOTÁVEIS NA INTERNACIONALIZAÇÃO DA RHEL 8

A RHEL 8 introduz as seguintes mudanças na internacionalização em comparação com a RHEL 7:

- Foi adicionado o suporte para o padrão da indústria de computação **Unicode 11**.
- A internacionalização é distribuída em múltiplos pacotes, o que permite instalações com menor espaço físico.

Para mais informações, veja a localização da **glibc** para RHEL é distribuída em vários pacotes.

- As atualizações do pacote **glibc** para múltiplos locais estão agora sincronizadas com o Common Locale Data Repository (CLDR).

APÊNDICE A. LISTA DE BILHETES POR COMPONENTE

Componente	Ingressos
389-ds-base	BZ#1334254, BZ#1358706
NetworkManager	BZ#1555013, BZ#1555012, BZ#1557035, BZ#1335409, BZ#1571655
PacoteKit	BZ#1559414
WALinuxAgent	BZ#1561132
anaconda	BZ#1499442, BZ#1500792, BZ#1547908, BZ#1612060, BZ#1595415, BZ#1610806, BZ#1533904, BZ#1672405, JIRA:RHELPLAN-1943, BZ#1677411, BZ#1502323, BZ#1696609
auditoria	BZ#1616428
authselect	BZ#1657295
bcc	BZ#1548302
bind	BZ#1588592
boom-boot	BZ#1649582
impulso	BZ#1494495, BZ#1616244
cloud-init	BZ#1615599, BZ#1641190
cmake	BZ#1590139
cockpit	BZ#1619993, BZ#1631905
criu	BZ#1689746
crypto-policies	BZ#1591620, BZ#1645606, BZ#1678661, BZ#1660839
cryptsetup	BZ#1564540
dispositivo-mapper-multipath	BZ#1643550, BZ#1673167
distribuição	BZ#1516728, BZ#1516741, BZ#1566048

Componente	Ingressos
dnf	BZ#1622580, BZ#1647760, BZ#1581191
driverctl	BZ#1648411
edk2	BZ#1536627
esc	BZ#1538645
firewalld	BZ#1509026, BZ#1648497
gcc	BZ#1169184, BZ#1607227, BZ#1535774, BZ#1504980, BZ#1571124, BZ#1246444, JIRA:RHELPLAN-7437, BZ#1652016
gdb	BZ#1491128
gdm	BZ#1589678, BZ#1641763, BZ#1678627
glib-networking	BZ#1640534
glibc	BZ#1512004, BZ#1376834, BZ#1512010, BZ#1304448, BZ#1512009, BZ#1512006, BZ#1514839, BZ#1533608
gnome-control-center	BZ#1697326
go-toolset-1.10-golang	BZ#1633351
grub2	BZ#1583445
httpd	BZ#1633224, BZ#1632754
membro do ipa-idoverride-memberof	BZ#1651577
ipa	BZ#1664718, BZ#1664719, BZ#1665051
iproute	BZ#1640991, BZ#1589317
iptables	BZ#1644030, BZ#1564596, BZ#1646159, BZ#1658734, BZ#1649790, BZ#1674536
iscsi-iniciador-utils	BZ#1626629, BZ#1582099
kernel-rt	BZ#1592977

Componente	Ingressos
kernel	BZ#1598448, BZ#1559607, BZ#1643522, BZ#1485546, BZ#1562998, BZ#1494651, BZ#1485532, BZ#1494028, BZ#1563617, BZ#1485525, BZ#1261167, BZ#1562987, BZ#1273139, BZ#1401552, BZ#1638465, BZ#1598776, BZ#1503672, BZ#1633143, BZ#1596240, BZ#1534870, BZ#1153521, BZ#1515987, BZ#1642795, BZ#1570255, BZ#1645744, BZ#1440031, BZ#1649647, BZ#1422268, BZ#1494705, BZ#1650149, BZ#1655413, BZ#1651806, BZ#1620330, BZ#1665295, BZ#1505999, BZ#1645433, BZ#1663281, BZ#1695142, BZ#1627455, BZ#1581898, BZ#1597671, BZ#1550498, BZ#1658391, BZ#1623590, BZ#1614144, BZ#1519039, BZ#1524683, BZ#1694705
kexec-tools	BZ#1520209, BZ#1662911
kmod-kvdo	BZ#1534087, BZ#1639512, BZ#1696492
ksh	BZ#1503922
libdnf	BZ#1642458, BZ#1679509
libreswan	BZ#1566574, BZ#1648776, BZ#1657854
libssh	BZ#1485241
libvirt	BZ#1528684
lksctp-tools	BZ#1568622
ltrace	BZ#1618748, BZ#1584322
lvm2	BZ#1676598, BZ#1643543, BZ#1643545, BZ#1643547, BZ#1643549, BZ#1643562, BZ#1643576
mariadb	BZ#1637034
mdadm	BZ#1654482
murmurar	BZ#1668883
net-snmp	BZ#1584510

Componente	Ingressos
nfs-utils	BZ#1592011, BZ#1639432
nftables	BZ#1593711
nginx	BZ#1545526
nodejs-10-módulo	BZ#1622118
nss	BZ#1489094, BZ#1645153
nuxwdog	BZ#1652269
openldap	BZ#1570056
opensc	BZ#1595638, BZ#1595626
openscap	BZ#1614273, BZ#1618484, BZ#1646197, BZ#163636431, BZ#1618489, BZ#1642373, BZ#1618464
openssh	BZ#1622511, BZ#1228088, BZ#1645038, BZ#1671262, BZ#1651763
openssl-pkcs11	BZ#1664802, BZ#1664807
openssl	BZ#1685470
oscap-anaconda-addon	BZ#1665082
marcapasso	BZ#1543494
pcs	BZ#1578891, BZ#1591308, BZ#1615420, BZ#1158816, BZ#1542288, BZ#1549535, BZ#1620190, BZ#1566430, BZ#1595829, BZ#1436217, BZ#1578955, BZ#1596050, BZ#1554310, BZ#1638852, BZ#1640477, BZ#1619620
perl-IO-Socket-SSL	BZ#1632600
perl	BZ#1511131
pki-core	BZ#1565073, BZ#1623444, BZ#1566360, BZ#1394069, BZ#1669257, BZ#1656856, BZ#1673296

Componente	Ingressos
módulo postgresql-9.6	BZ#1660041
pykickstart	BZ#1637872, BZ#1612061
python-rtslib	BZ#1666377
qemu-kvm	BZ#1559240, BZ#1508139, BZ#1497911, BZ#1578855, BZ#1651994, BZ#1621817, BZ#1508137, BZ#1592337, BZ#1570029, BZ#1689216, BZ#1585651, BZ#1519004
redhat-release	BZ#163636338
redhat-support-tool	BZ#1688274
rsyslog	BZ#1613880, BZ#1542497, BZ#1614179, BZ#1619645, BZ#1679512, JIRA:RHELPLAN-10431
módulo escala-2,10	BZ#1641744
scap-security-guide	BZ#1618505, BZ#1618528, BZ#1618518
scap-workbench	BZ#1640715
selinux-policy	BZ#1664345, BZ#1594111, BZ#1592244, BZ#1549772, BZ#1483904, BZ#1626446
configuração	BZ#1591969, BZ#1663556
sos	BZ#1559836
lula	BZ#1656871
sssd	BZ#1448094, BZ#1382750, BZ#1446101, BZ#1447945, BZ#1620123, BZ#1652562, BZ#1659457, BZ#1669407, BZ#1657665
gerenciador de assinaturas	BZ#1654531, BZ#1661414
subversão	BZ#1571415
swig-3.0-módulo	BZ#1660051
systemd	BZ#1658691

Componente	Ingressos
tomcatjss	BZ#142424966, BZ#1636564
sintonizado	BZ#1565598
valgrind	BZ#1500481, BZ#1538009
verniz	BZ#1633338
vdo	BZ#1669537
virt-manager	BZ#1599777, BZ#1643609
wpa_supplicant	BZ#1582538, BZ#1537143
xorg-x11-server	BZ#1687489, BZ#1698565

Componente	Ingressos
outros	JIRA:RHELPLAN-10347, BZ#1646563, JIRA:RHELPLAN-2306, BZ#1640697, BZ#1623712, BZ#1649404, BZ#1581198, BZ#1581990, BZ#1649497, BZ#1695584, BZ#1654280, BZ#1643294, BZ#1647612, BZ#1641015, BZ#1641032, BZ#1641004, BZ#1641034, BZ#1647110, BZ#1641007, BZ#1641029, BZ#1641022, JIRA:RHELPLAN-1212, BZ#1649493, BZ#1559616, BZ#1699825, BZ#1646541, BZ#1647725, BZ#1686057, BZ#1582530, BZ#1581496, BZ#1650618, BZ#1650675, BZ#1650701, JIRA:RHELPLAN-10439, JIRA:RHELPLAN-10440, JIRA:RHELPLAN-10442, JIRA:RHELPLAN-10443, JIRA:RHELPLAN-10438, JIRA:RHELPLAN-2878, JIRA:RHELPLAN-10355, JIRA:RHELPLAN-3010, JIRA:RHELPLAN-10352, JIRA:RHELPLAN-10353, JIRA:RHELPLAN-1473, JIRA:RHELPLAN-10445, JIRA:RHELPLAN-3001, JIRA:RHELPLAN-6746, JIRA:RHELPLAN-10354, JIRA:RHELPLAN-2896, JIRA:RHELPLAN-10304, JIRA:RHELPLAN-10628, JIRA:RHELPLAN-10441, JIRA:RHELPLAN-10444, JIRA:RHELPLAN-1842, JIRA:RHELPLAN-10596, JIRA:RHELPLAN-7291, JIRA:RHELPLAN-12764, BZ#1680177, JIRA:RHELPLAN-14607, JIRA:RHELPLAN-1820, BZ#1684947, BZ#1683712, BZ#1659609, BZ#1504934, BZ#1642765, BZ#1641014, BZ#1692746, BZ#1687900, BZ#1690207, BZ#1693775, BZ#1580387, BZ#1583620, BZ#1580430, BZ#1648843, BZ#1647908, BZ#1649891, BZ#1695698, BZ#1697896, BZ#1698613, BZ#1699535, BZ#1701968, BZ#1704867

AGRADECIMENTOS

Obrigado a todos que deram feedback como parte do RHEL 8 Readiness Challenge. Os 3 primeiros ganhadores são:

- Sterling Alexander
- John Pittman
- Jake Hunsaker

APÊNDICE B. HISTÓRICO DE REVISÃO

0.0-6

Thu Jan 28 2021, Lucie Maňásková(lmanasko@redhat.com)

- Atualizado o capítulo Antevisões Tecnológicas.

0.0-5

Qui Dez 10 2020, Lenka Špačková(lspackova@redhat.com)

- Acrescentou informações sobre o manuseio de AD GPOs em SSSD a Novas características (Gerenciamento de Identidade).

0.0-4

Ter 28 de abril de 2020, Lenka Špačková(lspackova@redhat.com)

- Informações atualizadas sobre atualizações no local em Visão Geral.

0.0-3

Thu Mar 12 2020, Lenka Špačková(lspackova@redhat.com)

- Acrescentou o papel que faltava no sistema RHEL de **pós-fixos** às Antevisões Tecnológicas.

0.0-2

Qua 12 Fev 2020, Jaroslav Klech(jklech@redhat.com)

- Forneceu uma versão completa do kernel para os capítulos Arquiteturas e Novas Características.

0.0-1

Ter 30 de julho de 2019, Lucie Maňásková(lmanasko@redhat.com)

- Lançamento das Notas de Lançamento do Red Hat Enterprise Linux 8.0.1.

0.0-0

Ter 07 de maio de 2019, Ioanna Gkioka(igkioka@redhat.com)

- Lançamento das Notas de Lançamento do Red Hat Enterprise Linux 8.0.