



# Red Hat Enterprise Linux 8

## 8.0 note di rilascio (traduzione automatica)

Note di rilascio per Red Hat Enterprise Linux 8.0 (traduzione automatica)



# Red Hat Enterprise Linux 8 8.0 note di rilascio (traduzione automatica)

---

Note di rilascio per Red Hat Enterprise Linux 8.0 (traduzione automatica)

## Nota Legale

Copyright © 2019 Red Hat, Inc.

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

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

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

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

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

Linux<sup>®</sup> is the registered trademark of Linus Torvalds in the United States and other countries.

Java<sup>®</sup> is a registered trademark of Oracle and/or its affiliates.

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

MySQL<sup>®</sup> is a registered trademark of MySQL AB in the United States, the European Union and other countries.

Node.js<sup>®</sup> is an official trademark of Joyent. Red Hat Software Collections is not formally related to or endorsed by the official Joyent Node.js open source or commercial project.

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

All other trademarks are the property of their respective owners.

## Sommario

Le note di rilascio forniscono una copertura di alto livello dei miglioramenti e delle aggiunte che sono state implementate in Red Hat Enterprise Linux 8.0 e documentano i problemi noti in questa release, così come notevoli bug fix, anteprime tecnologiche, funzionalità deprecate, funzionalità rimosse e altri dettagli.

## Indice

<b>FORNIRE UN FEEDBACK SULLA DOCUMENTAZIONE DI RED HAT (TRADUZIONE AUTOMATICA)</b> .....	<b>4</b>
<b>CAPITOLO 1. PANORAMICA (TRADUZIONE AUTOMATICA)</b> .....	<b>5</b>
Distribuzione	5
Gestione del software	5
Web server, database, linguaggi dinamici	5
Desktop	5
Installazione e creazione di immagini	5
Sistemi di file e archiviazione	6
La sicurezza	6
La rete	6
Virtualizzazione	6
Compilatori e strumenti di sviluppo	6
Elevata disponibilità e cluster	7
Risorse aggiuntive	7
<b>CAPITOLO 2. ARCHITETTURE (TRADUZIONE AUTOMATICA)</b> .....	<b>8</b>
<b>CAPITOLO 3. DISTRIBUZIONE DEI CONTENUTI IN RHEL 8 (TRADUZIONE AUTOMATICA)</b> .....	<b>9</b>
3.1. L'INSTALLAZIONE (TRADUZIONE AUTOMATICA)	9
3.2. REPOSITORY (TRADUZIONE AUTOMATICA)	9
3.3. FLUSSI APPLICATIVI (TRADUZIONE AUTOMATICA)	9
<b>CAPITOLO 4. NUOVE CARATTERISTICHE (TRADUZIONE AUTOMATICA)</b> .....	<b>11</b>
4.1. LA CONSOLE WEB (TRADUZIONE AUTOMATICA)	11
4.2. INSTALLAZIONE E CREAZIONE DI IMMAGINI (TRADUZIONE AUTOMATICA)	12
4.3. IL NOCCIOLO (TRADUZIONE AUTOMATICA)	13
4.4. GESTIONE DEL SOFTWARE (TRADUZIONE AUTOMATICA)	15
4.5. SERVIZI INFRASTRUTTURALI (TRADUZIONE AUTOMATICA)	16
4.6. CONCHIGLIE E STRUMENTI A RIGA DI COMANDO (TRADUZIONE AUTOMATICA)	17
4.7. LINGUAGGI DI PROGRAMMAZIONE DINAMICI, SERVER WEB E DATABASE (TRADUZIONE AUTOMATICA)	18
4.8. DESKTOP (TRADUZIONE AUTOMATICA)	24
4.9. GESTIONE DELL'IDENTITÀ (TRADUZIONE AUTOMATICA)	26
4.10. COMPILATORI E STRUMENTI DI SVILUPPO (TRADUZIONE AUTOMATICA)	30
4.11. SISTEMI DI FILE E ARCHIVIAZIONE (TRADUZIONE AUTOMATICA)	38
4.12. ELEVATA DISPONIBILITÀ E CLUSTER (TRADUZIONE AUTOMATICA)	43
4.13. LA RETE (TRADUZIONE AUTOMATICA)	45
4.14. LA SICUREZZA (TRADUZIONE AUTOMATICA)	49
4.15. VIRTUALIZZAZIONE (TRADUZIONE AUTOMATICA)	54
4.16. SUPPORTABILITÀ (TRADUZIONE AUTOMATICA)	56
<b>CAPITOLO 5. ANTEPRIME TECNOLOGICHE (TRADUZIONE AUTOMATICA)</b> .....	<b>57</b>
5.1. IL NOCCIOLO (TRADUZIONE AUTOMATICA)	57
5.2. SISTEMI DI FILE E ARCHIVIAZIONE (TRADUZIONE AUTOMATICA)	58
5.3. ELEVATA DISPONIBILITÀ E CLUSTER (TRADUZIONE AUTOMATICA)	59
5.4. LA SICUREZZA (TRADUZIONE AUTOMATICA)	59
5.5. VIRTUALIZZAZIONE (TRADUZIONE AUTOMATICA)	59
<b>CAPITOLO 6. FUNZIONALITÀ DEPRESSE (TRADUZIONE AUTOMATICA)</b> .....	<b>61</b>
6.1. INSTALLAZIONE E CREAZIONE DI IMMAGINI (TRADUZIONE AUTOMATICA)	61
6.2. SISTEMI DI FILE E ARCHIVIAZIONE (TRADUZIONE AUTOMATICA)	61
6.3. LA RETE (TRADUZIONE AUTOMATICA)	62

6.4. LA SICUREZZA (TRADUZIONE AUTOMATICA)	62
6.5. VIRTUALIZZAZIONE (TRADUZIONE AUTOMATICA)	62
<b>CAPITOLO 7. PROBLEMI NOTI (TRADUZIONE AUTOMATICA) .....</b>	<b>64</b>
7.1. INSTALLAZIONE E CREAZIONE DI IMMAGINI (TRADUZIONE AUTOMATICA)	64
7.2. IL NOCCIOLO (TRADUZIONE AUTOMATICA)	64
7.3. GESTIONE DEL SOFTWARE (TRADUZIONE AUTOMATICA)	65
7.4. SERVIZI INFRASTRUTTURALI (TRADUZIONE AUTOMATICA)	66
7.5. CONCHIGLIE E STRUMENTI A RIGA DI COMANDO (TRADUZIONE AUTOMATICA)	66
7.6. LINGUAGGI DI PROGRAMMAZIONE DINAMICI, SERVER WEB E DATABASE (TRADUZIONE AUTOMATICA)	67
7.7. GESTIONE DELL'IDENTITÀ (TRADUZIONE AUTOMATICA)	67
7.8. COMPILATORI E STRUMENTI DI SVILUPPO (TRADUZIONE AUTOMATICA)	68
7.9. SISTEMI DI FILE E ARCHIVIAZIONE (TRADUZIONE AUTOMATICA)	69
7.10. LA RETE (TRADUZIONE AUTOMATICA)	70
7.11. LA SICUREZZA (TRADUZIONE AUTOMATICA)	70
<b>CAPITOLO 8. NOTEVOLI MODIFICHE AI CONTENITORI (TRADUZIONE AUTOMATICA) .....</b>	<b>72</b>
<b>APPENDICE A. ELENCO DEI BIGLIETTI PER COMPONENTE .....</b>	<b>73</b>



## FORNIRE UN FEEDBACK SULLA DOCUMENTAZIONE DI RED HAT (TRADUZIONE AUTOMATICA)

Apprezziamo il vostro contributo sulla nostra documentazione. Vi prego, fateci sapere come possiamo renderla migliore. Per farlo:

- Per semplici commenti su passaggi specifici, assicurati di visualizzare la documentazione in formato HTML a più pagine. Evidenziare la parte di testo su cui si desidera commentare. Quindi, fare clic sul pop-up **Aggiungi feedback** che appare sotto il testo evidenziato e seguire le istruzioni visualizzate.
- Per inviare feedback più complessi, crea un biglietto Bugzilla:
  1. Vai al sito [Bugzilla](#)web.
  2. Come componente, utilizzare la **documentazione**.
  3. Compila il campo **Descrizione** con i tuoi suggerimenti di miglioramento. Includere un link alla parte o alle parti pertinenti della documentazione.
  4. Fare clic su **Invia bug**.



# CAPITOLO 1. PANORAMICA (TRADUZIONE AUTOMATICA)

Basato su Fedora 28 e sul kernel upstream 4.18, Red Hat Enterprise Linux 8.0 fornisce agli utenti una base stabile, sicura e coerente nelle distribuzioni ibride di cloud ibrido con gli strumenti necessari per supportare i carichi di lavoro tradizionali ed emergenti. I punti salienti dell'uscita includono:

## Distribuzione

- Il contenuto è disponibile attraverso i repository **BaseOS** e Application Stream (**AppStream**).
- Il repository **AppStream** supporta una nuova estensione del tradizionale formato RPM - *moduli*. Ciò consente di avere a disposizione per l'installazione di più versioni maggiori di un componente.

Vedere [Capitolo 3, Distribuzione dei contenuti in RHEL 8 \(traduzione automatica\)](#) per ulteriori informazioni.

## Gestione del software

- Il gestore di pacchetti **YUM** è ora basato sulla tecnologia **DNF** e fornisce supporto per contenuti modulari, maggiori prestazioni e un'API stabile e ben progettata per l'integrazione con gli utensili.

Vedere [Sezione 4.4, «Gestione del software \(traduzione automatica\)»](#) per maggiori dettagli.

## Web server, database, linguaggi dinamici

- **Python 3.6** è l'implementazione Python predefinita in RHEL 8; il supporto limitato per Python 2.7 è fornito. Nessuna versione di Python è installata di default.
- RHEL 8 fornisce i seguenti **server di database**: MariaDB 10.3, MySQL 8.0, PostgreSQL 10, PostgreSQL 9.6 e Redis 4.0.

Vedere [Sezione 4.7, «Linguaggi di programmazione dinamici, server web e database \(traduzione automatica\)»](#) per ulteriori informazioni.

## Desktop

- **GNOME Shell** è stato aggiornato alla versione 3.28.
- La sessione GNOME e lo GNOME Display Manager usano **Wayland** come server di visualizzazione predefinito. È disponibile anche il server **X.Org**, che in RHEL 7 è il server di visualizzazione predefinito.

Vedere [Sezione 4.8, «Desktop \(traduzione automatica\)»](#) per ulteriori informazioni.

## Installazione e creazione di immagini

- Il programma di installazione di **Anaconda** può utilizzare la crittografia del disco **LUKS2** e installare il sistema su dispositivi **NVDIMM**.
- Il nuovo strumento **Composer** consente agli utenti di creare immagini di sistema personalizzate in una varietà di formati, comprese le immagini preparate per la distribuzione su nuvole di vari fornitori. Composer è disponibile come *Anteprima di tecnologia*.
- Installazione da DVD utilizzando Hardware Management Console (**HMC**) e Support Element (**SE**) su **IBM Z**.

Vedere [Sezione 4.2, «Installazione e creazione di immagini \(traduzione automatica\)»](#) per ulteriori dettagli.

## Sistemi di file e archiviazione

- È stato introdotto il responsabile dello stoccaggio locale **Stratis**. Stratis consente di eseguire facilmente attività di archiviazione complesse e di gestire lo stack di archiviazione utilizzando un'interfaccia unificata.
- Il formato LUKS versione 2 (**LUKS2**) sostituisce il precedente formato LUKS (LUKS1). Il **dm-crypt** sottosistema e lo **cryptsetup** strumento ora utilizza LUKS2 come formato predefinito per i volumi cifrati.

Vedere [Sezione 4.11, «Sistemi di file e archiviazione \(traduzione automatica\)»](#) per ulteriori informazioni.

## La sicurezza

- Le **policy crittografiche** a livello di sistema, che configurano i sottosistemi crittografici di base, che coprono i protocolli TLS, IPsec, SSH, DNSSEC e Kerberos, sono applicate di default. Con il nuovo **update-crypto-policies** comando, l'amministratore può facilmente passare da una modalità all'altra: predefinita, legacy, futura e fips.
- Il supporto per le **smart card** e i moduli di sicurezza hardware (**HSM**) con **PKCS #11** è ora coerente in tutto il sistema.

Vedere [Sezione 4.14, «La sicurezza \(traduzione automatica\)»](#) per ulteriori informazioni.

## La rete

- Il **nftables** framework sostituisce **iptables** nel ruolo della funzione predefinita di filtraggio dei pacchetti di rete.
- Il **firewalld** demone ora usa **nftables** come backend predefinito.
- Supporto per driver di rete virtuale **IPVLAN** che consentono la connettività di rete per contenitori multipli.

Vedere [Sezione 4.13, «La rete \(traduzione automatica\)»](#) per ulteriori dettagli.

## Virtualizzazione

- Una più moderna macchina basata su PCI Express (**Q35**) è ora supportata e configurata automaticamente nelle macchine virtuali create in RHEL 8. Questo fornisce una varietà di miglioramenti nelle caratteristiche e nella compatibilità dei dispositivi virtuali.
- Le macchine virtuali possono ora essere create e gestite utilizzando la console web di RHEL 8, nota anche come **Cockpit**.
- L'emulatore **QEMU** introduce la funzione **sandboxing**, che fornisce limitazioni configurabili a ciò che i sistemi chiamati QEMU possono eseguire, rendendo così le macchine virtuali più sicure.

Vedere [Sezione 4.15, «Virtualizzazione \(traduzione automatica\)»](#) per ulteriori informazioni.

## Compilatori e strumenti di sviluppo

- Il compilatore **GCC** basato sulla versione 8.2 porta il supporto per le versioni standard in linguaggio C++ più recenti, migliori ottimizzazioni, nuove tecniche di indurimento del codice, avvisi migliorati e nuove caratteristiche hardware.

- Diversi strumenti per la generazione del codice, la manipolazione e il debug possono ora gestire sperimentalmente il formato **DWARF5** per il debug delle informazioni.
- Il supporto del kernel per il tracciamento **eBPF** è disponibile per alcuni strumenti, come **BCC**, **PCP**, e **SystemTap**.
- Le **glibc** librerie basate sulla versione 2.28 aggiungono il supporto per Unicode 11, chiamate di sistema Linux più recenti, miglioramenti chiave nello stub resolver DNS, ulteriore rafforzamento della sicurezza e miglioramento delle prestazioni.

Vedere [Sezione 4.10, «Compilatori e strumenti di sviluppo \(traduzione automatica\)»](#) per ulteriori dettagli.

## Elevata disponibilità e cluster

- Il gestore delle risorse del cluster **Pacemaker** è stato aggiornato alla versione upstream 2.0.0.0, che fornisce una serie di correzioni e miglioramenti.
- In RHEL 8, il sistema di configurazione dei **pc** supporta completamente Corosync 3 **knete** i nomi dei nodi.

Vedere [Sezione 4.12, «Elevata disponibilità e cluster \(traduzione automatica\)»](#) per ulteriori informazioni.

## Risorse aggiuntive

- Le capacità e i limiti di Red Hat Enterprise Linux 8.0 rispetto ad altre versioni del sistema sono disponibili nell'articolo [Red Hat Enterprise Linux technology capabilities and limits](#) della Knowledgebase .
- Le informazioni relative al ciclo di vita di Red Hat Enterprise Linux sono fornite nel [Red Hat Enterprise Linux Life Cycle](#) documento.
- Il [Package manifest](#) documento fornisce un elenco dei pacchetti per RHEL 8.

## CAPITOLO 2. ARCHITETTURE (TRADUZIONE AUTOMATICA)

Red Hat Enterprise Linux 8.0 è distribuito con il kernel versione 4.18, che fornisce il supporto per le seguenti architetture:

- Architetture AMD e Intel a 64 bit
- L'architettura ARM a 64 bit
- IBM Power Systems, Little Endian
- IBM Z

## CAPITOLO 3. DISTRIBUZIONE DEI CONTENUTI IN RHEL 8 (TRADUZIONE AUTOMATICA)

### 3.1. L'INSTALLAZIONE (TRADUZIONE AUTOMATICA)

Red Hat Enterprise Linux 8 è installato utilizzando immagini ISO. Sono disponibili due tipi di immagini ISO per le architetture AMD64, Intel a 64 bit, ARM a 64 bit, IBM Power Systems e IBM Z:

- DVD binario ISO: Un'immagine completa dell'installazione che contiene i repository BaseOS e AppStream e permette di completare l'installazione senza repository aggiuntivi.
- Stivale ISO: Un'immagine ISO di avvio minima che viene utilizzata per avviare il programma di installazione. Questa opzione richiede l'accesso ai repository BaseOS e AppStream per installare i pacchetti software. I repository sono parte dell'immagine ISO del DVD binario.

Vedere il [Performing a standard RHEL installation](#) documento per istruzioni su come scaricare immagini ISO, creare supporti di installazione e completare un'installazione RHEL. Per le installazioni automatiche Kickstart e altri argomenti avanzati, vedere il [Performing an advanced RHEL installation](#) documento.

### 3.2. REPOSITORY (TRADUZIONE AUTOMATICA)

Red Hat Enterprise Linux 8 è distribuito attraverso due repository:

- BaseOS
- AppStream

Entrambi i repository sono necessari per un'installazione RHEL di base e sono disponibili con tutti gli abbonamenti RHEL.

Il contenuto del repository BaseOS ha lo scopo di fornire l'insieme centrale della funzionalità del sistema operativo sottostante che costituisce la base per tutte le installazioni. Questo contenuto è disponibile nel formato RPM ed è soggetto a condizioni di supporto simili a quelle delle precedenti versioni di RHEL. Per un elenco dei pacchetti distribuiti tramite BaseOS, si veda il file [Package manifest](#).

Il contenuto nel repository Application Stream include applicazioni di spazio utente aggiuntive, lingue di runtime e database a supporto dei vari carichi di lavoro e casi d'uso. Il contenuto in AppStream è disponibile in uno dei due formati - il familiare formato RPM e un'estensione al formato RPM chiamato *moduli*. Per un elenco dei pacchetti disponibili in AppStream, vedere il file [Package manifest](#).

Inoltre, il repository CodeReady Linux Builder è disponibile con tutti gli abbonamenti RHEL. Fornisce pacchetti aggiuntivi per l'uso da parte degli sviluppatori. I pacchetti inclusi nel repository CodeReady Linux Builder non sono supportati per uso di produzione.

Per ulteriori informazioni sui repository di RHEL 8, vedere il file [Package manifest](#).

### 3.3. FLUSSI APPLICATIVI (TRADUZIONE AUTOMATICA)

Red Hat Enterprise Linux 8.0 introduce il concetto di Application Streams. Versioni multiple di componenti dello spazio utente vengono ora fornite e aggiornate con maggiore frequenza rispetto ai pacchetti del sistema operativo principale. Questo fornisce una maggiore flessibilità per personalizzare Red Hat Enterprise Linux senza influire sulla stabilità della piattaforma o su specifiche implementazioni.

I componenti resi disponibili come Application Streams possono essere confezionati come moduli o

pacchetti RPM e sono consegnati attraverso il repository AppStream in RHEL 8. Ogni componente dell'Application Stream ha un determinato ciclo di vita. Per i dettagli, vedere [Red Hat Enterprise Linux Life Cycle](#).

I moduli sono insiemi di pacchetti che rappresentano un'unità logica: un'applicazione, una pila di lingue, un database o un insieme di strumenti. Questi pacchetti sono costruiti, testati e rilasciati insieme.

I flussi di moduli rappresentano le versioni dei componenti del flusso di applicazioni. Ad esempio, due flussi (versioni) del server di database PostgreSQL sono disponibili nel modulo postgresql: PostgreSQL 10 (il flusso predefinito) e PostgreSQL 9.6. Sul sistema può essere installato un solo flusso di moduli. Diverse versioni possono essere utilizzate in contenitori separati.

I comandi dettagliati del modulo sono descritti nel [Using Application Stream](#) documento. Per un elenco dei moduli disponibili in AppStream, vedere il file [Package manifest](#).

## CAPITOLO 4. NUOVE CARATTERISTICHE (TRADUZIONE AUTOMATICA)

Questa parte descrive le nuove funzionalità e i principali miglioramenti introdotti in Red Hat Enterprise Linux 8.

### 4.1. LA CONSOLE WEB (TRADUZIONE AUTOMATICA)



#### NOTA

La pagina Abbonamenti della console web è ora fornita dal nuovo **subscription-manager-cockpit** pacchetto.

#### È stata aggiunta un'interfaccia firewall alla console web

La pagina **Networking** della console web di RHEL 8 include ora una sezione **Firewall**. In questa sezione, gli utenti possono abilitare o disabilitare il firewall, nonché aggiungere, rimuovere e modificare le regole del firewall.

(BZ#1647110)

#### La console web è ora disponibile per impostazione predefinita

I pacchetti per la console web RHEL 8, nota anche come Cockpit, sono ora parte dei repository di default di Red Hat Enterprise Linux, e possono quindi essere installati immediatamente su un sistema RHEL 8 registrato.

Inoltre, su un'installazione non minima di RHEL 8, la console web viene installata automaticamente e le porte firewall richieste dalla console vengono aperte automaticamente. Prima del login è stato aggiunto un messaggio di sistema che fornisce informazioni su come abilitare o accedere alla console web.

(JIRA:RHELPLAN-10355)

#### Migliore integrazione IdM per la console web

Se il sistema è registrato in un dominio IdM (Identity Management), la console web di RHEL 8 ora utilizza di default le risorse IdM gestite centralmente dal dominio. Ciò include i seguenti benefici:

- Gli amministratori del dominio IdM possono utilizzare la console web per gestire la macchina locale.
- Il server web della console passa automaticamente a un certificato rilasciato dall'IdM certificate authority (CA) e accettato dai browser.
- Gli utenti con un biglietto Kerberos nel dominio IdM non devono fornire le credenziali di login per accedere alla console web.
- Gli host SSH noti al dominio IdM sono accessibili alla console web senza aggiungere manualmente una connessione SSH.

Si noti che, affinché l'integrazione di IdM con la console web funzioni correttamente, l'utente deve prima eseguire l'**ipa-adviser** utilità con l'**enable-admins-sudo** opzione nel sistema master IdM.

(JIRA:RHELPLAN-3010)

### La console web è ora compatibile con i browser mobili

Con questo aggiornamento, i menu e le pagine della console web possono essere navigati su varianti di browser mobile. In questo modo è possibile gestire sistemi che utilizzano la web console RHEL 8 da un dispositivo mobile.

(JIRA:RHELPLAN-10352)

### La pagina iniziale della console web mostra ora gli aggiornamenti e gli abbonamenti mancanti

Se un sistema gestito dalla console web di RHEL 8 ha pacchetti obsoleti o un abbonamento scaduto, viene ora visualizzato un avviso sulla pagina iniziale della console web del sistema.

(JIRA:RHELPLAN-10353)

### La console web supporta ora l'iscrizione PBD

Con questo aggiornamento, è possibile utilizzare l'interfaccia della console web RHEL 8 per applicare le regole PBD (Policy-Based Decryption) ai dischi sui sistemi gestiti. Questo utilizza il client di decrittografia Clevis per facilitare una varietà di funzioni di gestione della sicurezza nella console web, come lo sblocco automatico delle partizioni del disco cifrate LUKS.

(JIRA:RHELPLAN-10354)

### Le macchine virtuali possono ora essere gestite tramite la console web

La **Virtual Machines** pagina può ora essere aggiunta all'interfaccia della console web di RHEL 8, che consente all'utente di creare e gestire macchine virtuali basate su libvirt-based.

(JIRA:RHELPLAN-2896)

## 4.2. INSTALLAZIONE E CREAZIONE DI IMMAGINI (TRADUZIONE AUTOMATICA)

### L'installazione di RHEL da un DVD utilizzando SE e HMC è ora pienamente supportata su IBM Z

L'installazione di Red Hat Enterprise Linux 8 su hardware IBM Z da un DVD con **Support Element (SE)** e **Hardware Management Console (HMC)** è ora pienamente supportata. Questa aggiunta semplifica il processo di installazione su IBM Z con **SE** e **HMC**.

Quando si avvia da un DVD binario, l'installatore chiede all'utente di inserire parametri aggiuntivi del kernel. Per impostare il DVD come sorgente di installazione, aggiungere **inst.repo=hmcai** parametri del kernel. L'installatore abilita quindi l'accesso ai file **SE** e **HMC**, recupera le immagini per stage2 dal DVD e fornisce l'accesso ai pacchetti presenti sul DVD per la selezione del software.

La nuova funzione elimina la necessità di una configurazione di rete esterna e amplia le opzioni di installazione.

(BZ#1500792)

### L'installatore ora supporta il formato di crittografia del disco LUKS2

Il programma di installazione di Red Hat Enterprise Linux 8 ora usa di default il formato LUKS2 ma è possibile selezionare una versione LUKS dalla finestra di partizionamento personalizzato di **Anaconda** o utilizzando le nuove opzioni nei **RAID** comandi di Kickstart **autopart**, **logvol**, **part**, e **part** comandi.



LUKS2 fornisce molti miglioramenti e caratteristiche, ad esempio, estende le capacità del formato su disco e fornisce modi flessibili per memorizzare i metadati.

(BZ#1547908)

### L'Anaconda supporta lo scopo del sistema in RHEL 8

In precedenza, **Anaconda** non ha fornito informazioni sullo scopo del sistema al **Subscription Manager**. In Red Hat Enterprise Linux 8.0, è possibile impostare lo scopo del sistema durante l'installazione utilizzando la **System Purpose** finestra di **Anaconda** o il **syspurpose** comando Kickstart. Al termine dell'installazione, **Subscription Manager** utilizza le informazioni sullo scopo del sistema al momento della sottoscrizione del sistema.

(BZ#1612060)

### Pykickstart supporta lo scopo del sistema in RHEL 8

In precedenza, non era possibile per la **pykickstart** biblioteca fornire informazioni sullo scopo del sistema al **Subscription Manager**. In Red Hat Enterprise Linux 8.0, **pykickstart** analizza il nuovo **syspurpose** comando e registra lo scopo del sistema durante l'installazione automatizzata e parzialmente automatizzata. Le informazioni vengono quindi trasmesse ad **Anaconda**, salvate sul sistema appena installato e disponibili per il **Subscription Manager** al momento dell'iscrizione al sistema.

(BZ#1612061)

### Anaconda supporta un nuovo parametro di avvio del kernel in RHEL 8

In precedenza, si poteva specificare solo un repository di base dai parametri di avvio del kernel. In Red Hat Enterprise Linux 8, un nuovo parametro del kernel **inst.addrepo=<name>,<url>**, permette di specificare un repository aggiuntivo durante l'installazione.

Questo parametro ha due valori obbligatori: il nome del repository e l'URL che punta al repository. Per ulteriori informazioni, vedere <https://anaconda-installer.readthedocs.io/en/latest/boot-options.html#inst-addrepo>

(BZ#1595415)

### Anaconda supporta una ISO unificata in RHEL 8

In Red Hat Enterprise Linux 8.0, una ISO unificata carica automaticamente i repository dei sorgenti di installazione di BaseOS e AppStream.

Questa funzione funziona per il primo repository di base che viene caricato durante l'installazione. Per esempio, se si avvia l'installazione senza repository configurato e si ha l'ISO unificata come repository di base nella GUI, o se si avvia l'installazione usando l'**inst.repo=**opzione che punta all'ISO unificata. Come risultato, il repository AppStream è abilitato sotto la sezione **Repository aggiuntivi** della finestra **Installation Source** GUI. Non è possibile rimuovere il repository di AppStream o modificarne le impostazioni, ma è possibile disattivarlo in **Installation Source**. Questa caratteristica non funziona se si avvia l'installazione usando un diverso repository di base e poi lo si cambia nella ISO unificata. Se lo fai, il repository di base viene sostituito. Tuttavia, il repository AppStream non viene sostituito e punta al file originale.

(BZ#1610806)

## 4.3. IL NOCCIOLO (TRADUZIONE AUTOMATICA)

È ora disponibile l'indirizzamento fisico a 52 bit ARM

Con questo aggiornamento è disponibile il supporto per l'indirizzamento fisico (PA) a 52 bit per l'architettura ARM a 64 bit. In questo modo si ottiene uno spazio di indirizzi più ampio rispetto al precedente sistema PA a 48 bit.

(BZ#1643522)

### Il codice IOMMU supporta tabelle di pagina a 5 livelli in RHEL 8

Il codice dell'unità di gestione della memoria I/O (IOMMU) nel kernel Linux è stato aggiornato per supportare tabelle di pagina a 5 livelli in Red Hat Enterprise Linux 8.

(BZ#1485546)

### Supporto per il paging a 5 livelli

È stato aggiunto un nuovo tipo di tabella di pagina **P4d\_tsoftware** nel kernel Linux per supportare il paging a 5 livelli in Red Hat Enterprise Linux 8.

(BZ#1485532)

### La gestione della memoria supporta tabelle di pagina a 5 livelli

Con Red Hat Enterprise Linux 7, il bus di memoria esistente aveva 48/46 bit di capacità di indirizzamento della memoria virtuale/fisica, e il kernel Linux ha implementato 4 livelli di tabelle di pagina per gestire questi indirizzi virtuali ad indirizzi fisici. La linea di indirizzamento del bus fisico colloca la capacità massima della memoria fisica a 64 TB.

Questi limiti sono stati estesi a 57/52 bit di indirizzo di memoria virtuale/fisica con 128 PiB di spazio di indirizzo virtuale e 4 PB di capacità di memoria fisica.

Con l'intervallo di indirizzi esteso, la gestione della memoria in Red Hat Enterprise Linux 8 aggiunge il supporto per l'implementazione di tabelle di pagina a 5 livelli, per essere in grado di gestire l'intervallo di indirizzi esteso.

(BZ#1485525)

### kernel-signing-ca.cer viene spostato in kernel-coreRHEL 8

In tutte le versioni di Red Hat Enterprise Linux 7, la chiave **kernel-signing-ca.cer** pubblica si trovava nel **kernel-doc** pacchetto. Tuttavia, in Red Hat Enterprise Linux 8, **kernel-signing-ca.cer** è stato spostato nel **kernel-core** pacchetto per ogni architettura.

(BZ#1638465)

### bpftool per l'ispezione e la manipolazione di programmi e mappe basati su eBPF aggiunto

L'**bpftool** utilità che serve per l'ispezione e la semplice manipolazione di programmi e mappe basate sul Berkeley Packet Filtering (eBPF) è stata aggiunta al kernel Linux. **bpftool** fa parte dell'albero dei sorgenti del kernel ed è fornita dal pacchetto **bpftool**, che è incluso come sotto-pacchetto del pacchetto **kernel**.

(BZ#1559607)

### Le kernel-rtfonti sono state aggiornate

I **kernel-rt** sorgenti sono stati aggiornati per utilizzare l'ultimo albero dei sorgenti del kernel RHEL. L'ultimo albero dei sorgenti del kernel utilizza ora il set di patch upstream v4.18 realtime, che fornisce una serie di correzioni e miglioramenti rispetto alla versione precedente.

(BZ#1592977)

## 4.4. GESTIONE DEL SOFTWARE (TRADUZIONE AUTOMATICA)

### Miglioramento delle prestazioni YUM e supporto per contenuti modulari

Su Red Hat Enterprise Linux 8, l'installazione del software è garantita dalla nuova versione dello strumento **YUM**, basato sulla tecnologia **DNF**.

**YUM** basato su **DNF** presenta i seguenti vantaggi rispetto al precedente **YUM v3** utilizzato su RHEL 7:

- Aumento delle prestazioni
- Supporto per contenuti modulari
- API stabile e ben progettata per l'integrazione con gli utensili

Per informazioni dettagliate sulle differenze tra il nuovo utensile **YUM** e la precedente versione **YUM v3** di RHEL 7, vedere [http://dnf.readthedocs.io/en/latest/cli\\_vs\\_yum.html](http://dnf.readthedocs.io/en/latest/cli_vs_yum.html).

**YUM** basato su **DNF** è compatibile con **YUM v3** quando si utilizza dalla riga di comando, modificando o creando file di configurazione.

Per l'installazione del software, è possibile utilizzare il **yum** comando e le sue opzioni particolari allo stesso modo di RHEL 7. I pacchetti possono essere installati sotto i nomi precedenti usando **Provides**. I pacchetti forniscono anche collegamenti simbolici di compatibilità, in modo che i file binari, i file di configurazione e le directory possono essere trovati nelle posizioni abituali.

Si noti che le API legacy Python fornite da **YUM v3** e Libdnf C API sono instabili e probabilmente cambieranno durante il ciclo di vita di Red Hat Enterprise Linux 8. Gli utenti sono invitati a migrare i loro plugin e script alla nuova API DNF Python, che è stabile e completamente supportata. L'API DNF Python è disponibile all'indirizzo <https://dnf.readthedocs.io/en/latest/api.html>

Alcune delle caratteristiche di **YUM v3** possono comportarsi diversamente in **YUM** in base al **DNF**. Se tali modifiche hanno un impatto negativo sui flussi di lavoro, si prega di aprire un caso con Red Hat Support, come descritto in [How do I open and manage a support case on the Customer Portal?](#)

(BZ#1581198)

### Notevoli caratteristiche di RPM in RHEL 8

Red Hat Enterprise Linux 8 è distribuito con RPM 4.14. Questa versione introduce molti miglioramenti rispetto a RPM 4.11, disponibile in RHEL 7. Le caratteristiche più importanti includono:

- I **debuginfo** pacchetti possono essere installati in parallelo
- Sostegno alle dipendenze deboli
- Supporto per dipendenze ricche o booleane
- Supporto per file di imballaggio di dimensioni superiori a 4 GB
- Supporto per i trigger di file

Inoltre, i cambiamenti più importanti includono:

- Più severo, più rigoroso, più rigoroso

- Firma semplificata che controlla l'uscita in modalità non inerbose
- Aggiunte e deprezzamento delle macro

(BZ#1581990)

## RPM ora convalida l'intero contenuto del pacchetto prima di iniziare l'installazione

Su Red Hat Enterprise Linux 7, l'utilità **RPM** ha verificato il contenuto del carico utile dei singoli file durante il disimballaggio. Tuttavia, ciò è insufficiente per molteplici ragioni:

- Se il carico utile è danneggiato, viene notato solo dopo aver eseguito azioni di script, che sono irreversibili.
- Se il carico utile è danneggiato, l'aggiornamento di un pacchetto viene interrotto dopo aver sostituito alcuni file della versione precedente, il che interrompe un'installazione funzionante.
- Gli hash sui singoli file vengono eseguiti su dati non compressi, il che rende gli **RPM** vulnerabili alle vulnerabilità dei decompressore.

Su Red Hat Enterprise Linux 8, l'intero pacchetto viene convalidato prima dell'installazione in una fase separata, utilizzando i migliori hash disponibili.

I pacchetti costruiti su Red Hat Enterprise Linux 8 utilizzano un nuovo **SHA256**hash sul carico utile compresso. Sui pacchetti firmati, l'hash payload è ulteriormente protetto dalla firma, e quindi non può essere modificato senza rompere una firma e altri hash sull'intestazione del pacchetto. I pacchetti più vecchi usano l'**MD5**hash dell'header e del payload, a meno che non sia disabilitato dalla configurazione, come la modalità FIPS.

La macro `%_pkgverify_level`` può essere utilizzata per abilitare ulteriormente la verifica della firma prima dell'installazione o disabilitare completamente la verifica del carico utile. Inoltre, la `%_pkgverify_flags`macro può essere utilizzata per limitare gli hash e le firme consentite. Ad esempio, è possibile disabilitare l'uso dell'**MD5**hash debole al costo della compatibilità con i vecchi pacchetti.

(JIRA:RHELPLAN-1499)

## 4.5. SERVIZI INFRASTRUTTURALI (TRADUZIONE AUTOMATICA)

### Notevoli cambiamenti nel profilo sintonizzato consigliato in RHEL 8

Con questo aggiornamento, il profilo sintonizzato raccomandato (riportato dal **tuned-adm recommend**comando) viene selezionato in base alle seguenti regole - la prima regola che corrisponde ha effetto:

- Se il **syspurpose**ruolo (riportato dal **syspurpose show**comando) contiene **atomic**, e allo stesso tempo:
  - se Tuned funziona su metallo nudo, il **atomic-host**profilo viene selezionato
  - se Tuned è in esecuzione in una macchina virtuale, il **atomic-guest**profilo è selezionato
- Se Tuned è in esecuzione in una macchina virtuale, il **virtual-guest**profilo viene selezionato
- Se il **syspurpose**ruolo contiene **desktop workstation**e il tipo di telaio (riportato da **dmidecode**) è, **Laptop**, o **Portable**, allora il **balanced**profilo è selezionato

- Se nessuna delle regole di cui sopra corrisponde, il **throughput-performance** profilo viene selezionato

(BZ#1565598)

### I file prodotti da `named` possono essere scritti nella `directory` di lavoro

In precedenza, il demone **chiamato** demone ha memorizzato alcuni dati nella `directory` di lavoro, che è stata di sola lettura in Red Hat Enterprise Linux. Con questo aggiornamento, i percorsi sono stati modificati per i file selezionati in `sottdirectory`, dove è consentita la scrittura. Ora, i permessi di default delle cartelle Unix e SELinux permettono di scrivere nella cartella. I file distribuiti all'interno della `directory` sono ancora in sola lettura con **nome**.

(BZ#1588592)

### I database di Geolite sono stati sostituiti da quelli di Geolite2

I Database Geolite presenti in Red Hat Enterprise Linux 7 sono stati sostituiti dai Database Geolite2 su Red Hat Enterprise Linux 8.

I database di geolite sono stati forniti dal **GeoIP** pacchetto. Questo pacchetto, insieme al database legacy, non è più supportato a monte.

I database di Geolite2 sono forniti da pacchetti multipli. Il **libmaxminddb** pacchetto include la libreria e lo strumento a riga di **mmdblookup** comando, che consente la ricerca manuale degli indirizzi. Il **geoipupdate** binario del pacchetto legacy **GeoIP** è ora fornito dal **geoipupdate** pacchetto, ed è in grado di scaricare sia i database legacy che i nuovi database Geolite2.

(JIRA:RHELPLAN-6746)

### I log CUPS sono gestiti da `journald`

In RHEL 8, i log di CUPS non sono più memorizzati in file specifici all'interno della `/var/log/cups` directory, che è stata utilizzata in RHEL 7. In RHEL 8, tutti i tipi di log CUPS sono collegati centralmente nel **journald** demone `systemd` insieme ai log di altri programmi. Per accedere ai log del CUPS, utilizzare il **journalctl -u cups** comando. Per ulteriori informazioni, vedere [Lavorare [https://access.redhat.com/documentation/en-us/red\\_hat\\_enterprise\\_linux/8/html-single/deploying\\_different\\_types\\_of\\_servers/working-with-cups-logs\\_configuring-printing](https://access.redhat.com/documentation/en-us/red_hat_enterprise_linux/8/html-single/deploying_different_types_of_servers/working-with-cups-logs_configuring-printing) con i registri CUPS].

(JIRA:RHELPLAN-12764)

## 4.6. CONCHIGLIE E STRUMENTI A RIGA DI COMANDO (TRADUZIONE AUTOMATICA)

### L'`nobody` utente sostituisce `nfsnobody`

In Red Hat Enterprise Linux 7, c'era:

- la coppia **nobody** utente e gruppo con l'ID di 99, e
- l'**nfsnobody** utente e il gruppo si accoppiano con l'ID di 65534, che è anche l'ID predefinito di overflow del kernel.

Entrambi sono stati fusi nella coppia di **nobody** utenti e gruppi, che utilizza l'ID 65534 in Red Hat Enterprise Linux 8. Nuove installazioni non creano più la **nfsnobody** coppia.

Questa modifica riduce la confusione sui file che sono di proprietà di **nobody**NFS ma che non hanno nulla a che fare con NFS.

(BZ#1591969)

## Sistemi di controllo delle versioni in RHEL 8

RHEL 8 fornisce i seguenti sistemi di controllo delle versioni:

- **Git 2.18** un sistema di controllo delle revisioni distribuite con un'architettura decentralizzata.
- **Mercurial 4.8** un sistema di controllo di versione leggero e distribuito, progettato per la gestione efficiente di grandi progetti.
- **Subversion 1.10** un sistema di controllo versione centralizzato. Per ulteriori informazioni, vedere [Notable changes in Subversion 1.10](#).

Si noti che il Concurrent Versions System (CVS), disponibile in RHEL 7, non è distribuito con RHEL 8.

(BZ#1693775)

## Notevoli cambiamenti in Subversion 1.10

**Subversion 1.10** introduce una serie di nuove funzionalità dalla versione 1.7 distribuita in RHEL 7, così come le seguenti modifiche di compatibilità:

- A causa di incompatibilità nelle **Subversion** librerie utilizzate per supportare i collegamenti linguistici, i **Python 3** collegamenti per **Subversion 1.10** i quali non sono disponibili. Di conseguenza, le applicazioni che richiedono **Python** attacchi per **Subversion** le quali non sono supportate.
- I repository basati su **Berkeley DB** di essi non sono più supportati. Prima di migrare, eseguire il backup dei repository creati con **Subversion 1.7** il **svnadmin dump** comando. Dopo aver installato RHEL 8, ripristinare i repository usando il **svnadmin load** comando.
- Le copie di lavoro esistenti controllate dal **Subversion 1.7** cliente in RHEL 7 devono essere aggiornate al nuovo formato prima di poter essere utilizzate da **Subversion 1.10**. Dopo aver installato RHEL 8, eseguire il **svn upgrade** comando in ogni copia di lavoro.
- L'autenticazione Smartcard per l'accesso ai repository che utilizzano **https://** non è più supportata.

(BZ#1571415)

## 4.7. LINGUAGGI DI PROGRAMMAZIONE DINAMICI, SERVER WEB E DATABASE (TRADUZIONE AUTOMATICA)

### Python 3 è l'implementazione predefinita Python in RHEL 8

Red Hat Enterprise Linux 8 è distribuito con **Python 3.6**. Il pacchetto non è installato di default. Per installare **Python 3.6**, utilizzare il **yum install python3** comando.

**Python 2.7** è disponibile nella **python2** confezione. Tuttavia, **Python 2** avrà un ciclo di vita più breve e il suo scopo è quello di facilitare una transizione più agevole **Python 3** per i clienti.

Con RHEL 8 non vengono distribuiti né il pacchetto predefinito **python** né l' **/usr/bin/python** eseguibile non modificato. I clienti sono invitati a utilizzare **python3** o **python2** direttamente. In alternativa, gli amministratori possono configurare il comando unversioned **python** usando il **alternatives** comando.

Per i dettagli, vedere [Using Python in Red Hat Enterprise Linux 8](#) .

(BZ#1580387)

### Gli script Python devono specificare la versione principale negli hashbangs al momento della compilazione di RPM

In RHEL 8, gli script eseguibili Python devono utilizzare hashbangs (shebangs) specificando esplicitamente almeno la versione principale di Python.

Lo script **/usr/lib/rpm/redhat/brp-mangle-shebangs** buildroot policy (BRP) viene eseguito automaticamente quando si costruisce un qualsiasi pacchetto RPM. Questo script tenta di correggere gli hashbang in tutti i file eseguibili. Quando lo script incontra hashbang ambigui che non possono essere risolti con una versione maggiore di Python, genera errori. Esempi di questi hashbang ambigui sono:

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

Per modificare gli hashbangs negli script Python che causano questi errori di compilazione al momento della compilazione di RPM, utilizzare lo **pathfix.py** script del pacchetto **platform-python-devel**:

```
pathfix.py -pn -i %(__python3) PATH ...
```

È possibile specificare più *PATH*. Se un *PATH* è una directory, verifica **pathfix.py** ricorsivamente la presenza di qualsiasi script Python che corrisponda allo schema **^[a-zA-Z0-9\_]+\\.py\$**, non solo quelli con un hashbang ambiguo. Aggiungere questo comando alla **%prep** sezione o alla fine della **%install** sezione.

Per ulteriori informazioni, vedere [Handling hashbangs in Python scripts](#) .

(BZ#1583620)

### Notevoli cambiamenti in PHP

Red Hat Enterprise Linux 8 è distribuito con **PHP 7.2**. Questa versione introduce le seguenti importanti modifiche **PHP 5.4**, disponibili in RHEL 7:

- **PHP** usa FastCGI Process Manager (FPM) per impostazione predefinita (sicuro per l'uso con un threaded **httpd**)
- Le variabili **php\_value** e **php\_flag** le variabili non dovrebbero più essere usate nei file di **httpd** configurazione, ma dovrebbero essere impostate nella configurazione del pool: **/etc/php-fpm.d/\*.conf**
- **PHP** e gli errori di script e gli avvisi vengono registrati nel **/var/log/php-fpm/www-error.log** file invece che nel file **/var/log/httpd/error.log**
- Quando si cambia la variabile di configurazione di PHP **max\_execution\_time**, l'**httpd ProxyTimeout** impostazione dovrebbe essere aumentata per corrispondere a
- L'utente che esegue **PHP** gli script è ora configurato nella configurazione del pool FPM (il **/etc/php-fpm.d/www.conf** file; l'**apache** utente è quello predefinito)

- Il **php-fpm** servizio deve essere riavviato dopo una modifica della configurazione o dopo l'installazione di una nuova estensione

Le seguenti estensioni sono state rimosse:

- **aspell**
- **mysql** (notare che le estensioni **mysqli** e **pdo\_mysql** le estensioni sono ancora disponibili, fornite a **php-mysqlnd** pacchetto)
- **zip**
- **memcache**

(BZ#1580430, [BZ#1691688](#))

### Notevoli cambiamenti in Ruby

RHEL 8 fornisce **Ruby 2.5**, che introduce numerose nuove caratteristiche e miglioramenti rispetto **Ruby 2.0.0** a quelli disponibili in RHEL 7. Notevoli cambiamenti includono:

- È stato aggiunto il sistema di raccolta incrementale dei rifiuti.
- La **Refinements** sintassi è stata aggiunta.
- I simboli sono ora raccolti nella spazzatura.
- I **\$SAFE=2** livelli di **\$SAFE=3** sicurezza sono ormai obsoleti.
- Le classi **Fixnum** e **Bignum** le classi sono state unificate nella **Integer** classe.
- Le prestazioni sono state migliorate ottimizzando la **Hash** classe, migliorando l'accesso alle variabili di istanza e la **Mutex** classe è sempre più piccola e veloce.
- Alcune vecchie API sono state deprecate.
- Le librerie in bundle, come **RubyGems**, **Rake**, **RDoc**, **Psych**, **Minitest**, e **test-unit**, sono state aggiornate.
- Altre librerie, come **mathn**, **DL**, **ext/tk**, e **XMLRPC**, che in precedenza erano distribuite con **Ruby**, sono deprecate o non più incluse.
- Lo schema di **SemVer** versioning è ora utilizzato per la **Ruby** versioning.

(BZ#1648843)

### Notevoli cambiamenti in Perl

**Perl 5.26** distribuito con RHEL 8, introduce le seguenti modifiche rispetto alla versione disponibile in RHEL 7:

- **Unicode 9.0** è ora supportato.
- Nuovi **op-entry**, **loading-file**, e le **loaded-file SystemTap** sonde sono forniti.
- Il meccanismo di copia in scrittura viene utilizzato quando si assegnano gli scalari per migliorare le prestazioni.



- E' stato aggiunto il **IO::Socket::IP** modulo per la gestione trasparente delle prese IPv4 e IPv6.
- E' stato aggiunto il **Config::Perl::V** modulo per accedere **perl -V** ai dati in modo strutturato.
- È stato aggiunto un nuovo **perl-App-cpanminus** pacchetto che contiene l'**cpanm** utilità per ottenere, estrarre, costruire e installare moduli dal repository Comprehensive Perl Archive Network (CPAN).
- La directory `.corrente` è stata rimossa dal percorso di ricerca del **@INC** modulo per motivi di sicurezza.
- L'**do**istruzione ora restituisce un avviso di deprecazione quando non riesce a caricare un file a causa del cambiamento comportamentale descritto sopra.
- La **do subroutine(LIST)** chiamata non è più supportata e comporta un errore di sintassi.
- Gli hash sono randomizzati per impostazione predefinita. L'ordine in cui le chiavi e i valori restituiti da un hash cambia ad ogni **perl** esecuzione. Per disabilitare la randomizzazione, impostare la variabile d'**PERL\_PERTURB\_KEYS** ambiente su **0**.
- Non sono più ammessi caratteri letterali `{` non scappati nei modelli di espressione regolare.
- Il supporto del campo di applicazione lessicale per la **\$\_** variabile è stato rimosso.
- L'utilizzo dell'**defined** operatore su una matrice o un hash si traduce in un errore fatale.
- L'importazione di funzioni dal **UNIVERSAL** modulo provoca un errore fatale.
- I tasti **find2perl**, **s2p**, **a2p**, **c2p**, e **pstruct** gli strumenti sono stati rimossi.
- La **\${^ENCODING}** struttura è stata rimossa. La modalità predefinita del **encoding** pragma non è più supportata. Per scrivere il codice sorgente in una codifica diversa da **UTF-8**, utilizzare l'**Filter** opzione di codifica.
- L'**perl** imballaggio è ora allineato con quello a monte. Il **perl** pacchetto installa anche i moduli principali, mentre l'**/usr/bin/perl** interprete è fornito dal **perl-interpreter** pacchetto. Nelle versioni precedenti, il **perl** pacchetto comprendeva solo un interprete minimo, mentre il **perl-core** pacchetto comprendeva sia l'interprete che i moduli principali.

(BZ#1511131)

### Node.js nuovo in RHEL 8

**Node.js** Per la prima volta in RHEL viene fornita una piattaforma di sviluppo software per la realizzazione di applicazioni di rete veloci e scalabili nel linguaggio di programmazione JavaScript. In precedenza era disponibile solo come raccolta di software. RHEL 8 fornisce **Node.js 10**.

(BZ#1622118)

### Notevoli cambiamenti in SWIG

RHEL 8 include la versione 3.0 di Simplified Wrapper and Interface Generator (SWIG), che fornisce numerose nuove funzionalità, miglioramenti e correzioni di bug rispetto alla versione 2.0 distribuita in RHEL 7. In particolare, è stato attuato lo standard C++11. SWIG ora supporta anche Go 1.6, PHP 7. Ottava 4.2, e Pitone 3.5.

(BZ#1660051)

## Notevoli cambiamenti in Apache httpd

RHEL 8 è distribuito con il server HTTP Apache 2.4.37. Questa versione introduce le seguenti modifiche **httpd** disponibili in RHEL 7:

- Il supporto HTTP/2 è ora fornito dal **mod\_http2** pacchetto, che fa parte del **httpd** modulo.
- Il provisioning automatico dei certificati TLS e il rinnovo utilizzando il protocollo ACME (Automatic Certificate Management Environment) è ora supportato dal **mod\_md** pacchetto (per l'utilizzo con fornitori di certificati come ad esempio **Let's Encrypt**)
- Il server HTTP Apache supporta ora il caricamento di certificati TLS e chiavi private da token di sicurezza hardware direttamente dai **PKCS#11** moduli. Come risultato, una **mod\_ssl** configurazione può ora utilizzare **PKCS#11** URL per identificare la chiave privata TLS e, facoltativamente, il certificato TLS nelle **SSLCertificateFile** direttive **SSLCertificateKeyFile** direttive.
- Il modulo multi-processo (MPM) configurato di default con il server HTTP Apache è passato da un modello multi-processo, biforcuto (noto come **prefork**) ad un modello multi-threaded ad alte prestazioni, **event**. Tutti i moduli di terze parti che non sono thread-safe devono essere sostituiti o rimossi. Per cambiare il MPM configurato, modificare il **/etc/httpd/conf.modules.d/00-mpm.conf** file. Per ulteriori informazioni, vedere la pagina **httpd.conf(5)** dedicata ai dipendenti.

Per ulteriori informazioni su **httpd**, vedere [Setting up the Apache HTTP web server](#).

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

## Il server nginxweb nuovo in RHEL 8

RHEL 8 introduce **nginx 1.14** un server web e proxy che supporta l'HTTP e altri protocolli, con particolare attenzione all'alta concorrenza, alle prestazioni e al basso consumo di memoria. **nginx** era precedentemente disponibile solo come Software Collection.

Il server **nginx**web ora supporta il caricamento di chiavi private TLS da token di sicurezza hardware direttamente dai **PKCS#11** moduli. Come risultato, una **nginx** configurazione può utilizzare **PKCS#11** URL per identificare la chiave privata TLS nella **ssl\_certificate\_key** direttiva.

(BZ#1545526)

## Server di database in RHEL 8

RHEL 8 fornisce i seguenti server di database:

- **MySQL 8.0** un server di database SQL multi-utente e multi-threaded. Si compone del demone del **MySQL** server e **mysqld** di molti programmi client.
- **MariaDB 10.3** un server di database SQL multi-utente e multi-threaded. Per tutti gli scopi pratici, **MariaDB** è binario compatibile con **MySQL**.
- **PostgreSQL 10** e **PostgreSQL 9.6**, un sistema avanzato di gestione di database oggetti-relazionale (DBMS).
- **Redis 5** un negozio avanzato di chiavi-valore. Viene spesso indicato come server di struttura dati perché le chiavi possono contenere stringhe, hash, liste, set, set e set ordinati. **Redis**

Notare che il server di database NoSQL **MongoDB** non è incluso in RHEL 8.0 perché utilizza la Server Side Public License (SSPL).

(BZ#1647908)

### Notevoli cambiamenti in MySQL 8.0

RHEL 8 è distribuito con **MySQL 8.0**, che prevede, ad esempio, i seguenti miglioramenti:

- **MySQL** incorpora ora un dizionario dei dati transazionali, che memorizza informazioni sugli oggetti del database.
- **MySQL** ora supporta i ruoli, che sono raccolte di privilegi.
- Il set di caratteri predefinito è stato cambiato da **latin1** a **utf8mb4**.
- È stato aggiunto il supporto per le espressioni comuni delle tabelle, sia non recursive che recursive.
- **MySQL** supporta ora le funzioni della finestra, che eseguono un calcolo per ogni riga di una query, utilizzando le righe correlate.
- **InnoDB** supporta ora le istruzioni di lettura **NOWAIT** e **SKIP LOCKED** le opzioni con le istruzioni di blocco.
- Sono state migliorate le funzioni relative ai GIS.
- La funzionalità JSON è stata migliorata.
- I nuovi **mariadb-connector-c** pacchetti forniscono una comune libreria client per **MySQL** e **MariaDB**. Questa libreria è utilizzabile con qualsiasi versione del server **MySQL** e del **MariaDB** database. Come risultato, l'utente è in grado di collegare una build di un'applicazione a uno qualsiasi dei **MySQLMariaDB** server distribuiti con RHEL 8.

Inoltre, il **MySQL 8.0** server distribuito con RHEL 8 è configurato per l'utilizzo **mysql\_native\_password** come plug-in di autenticazione predefinito perché gli strumenti client e le librerie in RHEL 8 sono incompatibili con il **caching\_sha2\_password** metodo, che viene utilizzato di default nella versione upstream **MySQL 8.0**.

Per modificare il plug-in di autenticazione predefinito in **caching\_sha2\_password**, modificare il **/etc/my.cnf.d/mysql-default-authentication-plugin.cnf** file come segue:

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

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

### Notevoli cambiamenti in MariaDB 10.3

**MariaDB 10.3** fornisce numerose novità rispetto alla versione 5.5 distribuita in RHEL 7. Alcuni dei cambiamenti più importanti sono:

- **MariaDB Galera Cluster**, un cluster sincrono multi-master, è ora una parte standard di **MariaDB**.
- **InnoDB** è usato come motore di archiviazione predefinito invece di **XtraDB**.
- Espressioni comuni delle tabelle
- Tavoli con versione di sistema

- **FOR** circuiti
- Colonne invisibili
- Sequenze
- Istantanea **ADD COLUMN** per **InnoDB**
- Compressione della colonna indipendente dal motore di stoccaggio
- Replicazione parallela
- Replicazione multi-fonte

Inoltre, i nuovi **mariadb-connector-c** pacchetti forniscono una comune libreria client per **MySQL** e **MariaDB**. Questa libreria è utilizzabile con qualsiasi versione del server **MySQL** e del **MariaDB** database. Come risultato, l'utente è in grado di collegare una build di un'applicazione a uno qualsiasi dei **MySQL** **MariaDB** server distribuiti con RHEL 8.

Vedi anche [Using MariaDB on Red Hat Enterprise Linux 8](#) .

(BZ#1637034, BZ#1519450)

## 4.8. DESKTOP (TRADUZIONE AUTOMATICA)

### GNOME Shell, versione 3.28 in RHEL 8

GNOME Shell, la versione 3.28 è disponibile in Red Hat Enterprise Linux (RHEL) 8. Notevoli miglioramenti includono:

- Nuove caratteristiche dei box GNOME
- Nuova tastiera su schermo
- Supporto esteso dei dispositivi, in particolare l'integrazione per l'interfaccia Thunderbolt 3
- Miglioramenti per il software GNOME, il dconf-editor e il terminale GNOME

(BZ#1649404)

### Wayland è il server di visualizzazione predefinito

Con Red Hat Enterprise Linux 8, la sessione GNOME e lo GNOME Display Manager (GDM) usano **Wayland** come server di visualizzazione predefinito al posto del server **X.org**, che era usato con la precedente versione principale di RHEL.

**Wayland** offre molteplici vantaggi e miglioramenti rispetto a **X.org**. In particolare:

- Modello di sicurezza più forte
- Migliore gestione multi-monitor
- Scalatura migliorata dell'interfaccia utente (UI)
- Il desktop può controllare direttamente la gestione delle finestre.

Si noti che le seguenti funzioni non sono attualmente disponibili o non funzionano come previsto:

- Le configurazioni multi-GPU non sono supportate da **Wayland**.
- Il driver binario **NVIDIA** non funziona sotto **Wayland**.
- L'**xrandr** utilità non funziona sotto **Wayland** a causa del suo diverso approccio alla gestione, risoluzioni, rotazioni e layout. Si noti che anche altre utilità di **X.org** per manipolare lo schermo non funzionano sotto **Wayland**.
- La registrazione dello schermo, il desktop remoto e l'accessibilità non sempre funzionano correttamente sotto **Wayland**.
- Non è disponibile un manager per appunti.
- **Wayland** ignora i grab di tastiera emessi da applicazioni X11, come i visualizzatori di macchine virtuali.
- **Wayland** all'interno delle macchine virtuali guest (VM) ha problemi di stabilità e prestazioni, quindi si raccomanda di utilizzare la sessione X11 per gli ambienti virtuali.

Se si esegue l'aggiornamento a RHEL 8 da un sistema RHEL 7 in cui si è utilizzata la sessione GNOME di **X.org**, il sistema continua ad usare **X.org**. Il sistema ritorna automaticamente a **X.org** anche quando sono in uso i seguenti driver grafici:

- Il driver binario NVIDIA
- Il **cirrus** conducente
- Il **mga** conducente
- Il **aspeed** conducente

È possibile disabilitare manualmente l'uso di **Wayland**:

- Per disabilitare **Wayland** in **GDM**, impostare l'**WaylandEnable=false** opzione nel **/etc/gdm/custom.conf** file.
- Per disabilitare **Wayland** nella sessione di GNOME, selezionare l'opzione legacy X11 usando il menu della ruota dentata nella schermata di login dopo aver inserito il proprio nome di login.

Per maggiori dettagli su **Wayland**, vedere <https://wayland.freedesktop.org/>.

(BZ#1589678)

### Individuazione dei pacchetti RPM che si trovano in repository non abilitati di default

I repository aggiuntivi per il desktop non sono abilitati di default. La disabilitazione è indicata dalla **enabled=0** riga del file corrispondente **.repo**. Se si tenta di installare un pacchetto da tale repository utilizzando PackageKit, PackageKit mostra un messaggio di errore che annuncia che l'applicazione non è disponibile. Per rendere disponibile il pacchetto, sostituire la **enabled=0** riga usata in precedenza nel rispettivo **.repo** file con **enabled=1**.

(JIRA:RHELPLAN-2878)

### GNOME Software per la gestione dei pacchetti

Il **gnome-packagekit** pacchetto che ha fornito una raccolta di strumenti per la gestione dei pacchetti in ambiente grafico su Red Hat Enterprise Linux 7 non è più disponibile. Su Red Hat Enterprise Linux 8, funzionalità simili sono fornite dall'utilità **GNOME Software**, che permette di installare e aggiornare le

applicazioni e le estensioni di gnome-shell. Il software GNOME è distribuito nel **gnome-software** pacchetto.

(JIRA:RHELPLAN-3001)

## 4.9. GESTIONE DELL'IDENTITÀ (TRADUZIONE AUTOMATICA)

### Nuovo controllo della sintassi della password in Directory Server

Questo miglioramento aggiunge nuovi controlli della sintassi delle password a Directory Server. Gli amministratori possono ora, ad esempio, abilitare i controlli dei dizionari, consentire o negare l'uso di sequenze di caratteri e palindromi. Come risultato, se abilitato, il controllo della sintassi dei criteri delle password in Directory Server impone password più sicure.

(BZ#1334254)

### Directory Server fornisce ora un migliore supporto interno per la registrazione delle operazioni di logging

Diverse operazioni in Directory Server, avviate dal server e dai client, causano operazioni aggiuntive in background. In precedenza, il server registrava solo per le operazioni interne la parola chiave di **Internal** connessione e l'ID dell'operazione era sempre impostato su **-1**. Con questo miglioramento, Directory Server registra la connessione reale e l'ID dell'operazione. È ora possibile rintracciare l'operazione interna al server o all'operazione client che ha causato questa operazione.

Per ulteriori dettagli sulla registrazione interna delle operazioni, vedere il link:[https://access.redhat.com/documentation/en-us/red\\_hat\\_directory\\_server/11/html-single/administration\\_guide/#logging\\_interno\\_operazioni](https://access.redhat.com/documentation/en-us/red_hat_directory_server/11/html-single/administration_guide/#logging_interno_operazioni).

(BZ#1358706)

### La tomcatjsslibreria supporta il controllo OCSP utilizzando il responder dell'estensione AIA

Con questo miglioramento, la **tomcatjss**libreria supporta il controllo del protocollo OCSP (Online Certificate Status Protocol) utilizzando il responder dell'estensione di un certificato dell'Authority Information Access (AIA). Come risultato, gli amministratori di Red Hat Certificate System possono ora configurare il controllo OCSP che utilizza l'URL dell'estensione AIA.

(BZ#1636564)

### Directory Server introduce nuove utilità a riga di comando per gestire le istanze

Red Hat Directory Server 11.0 introduce i programmi di **dsctl** utilità **dscreate**, **dsconf**, e **ds**. Queste utilità semplificano la gestione di Directory Server utilizzando la riga di comando. Ad esempio, è ora possibile utilizzare un comando con parametri per configurare una funzione invece di inviare complesse istruzioni LDIF al server.

Di seguito è riportata una panoramica sullo scopo di ogni utilità:

- Utilizzare l'**dscreate** utilità per creare nuove istanze di Directory Server utilizzando la modalità interattiva o un file INF. Si noti che il formato del file INF è diverso da quello del programma di installazione utilizzato nelle versioni precedenti di Directory Server.
- Utilizzare l'**dsconf** utilità per gestire le istanze di Directory Server durante l'esecuzione. Per esempio, usare **dsconf** per:
  - Configurare le impostazioni nella **cn=config** voce

- Configurare i plug-in
- Configurare la replica
- Backup e ripristino di un'istanza
- Utilizzare l'**dsctl** utilità per gestire le istanze di Directory Server mentre sono offline. Per esempio, usare **dsctl** per:
  - Avviare e fermare un'istanza
  - Re-indicizza il database del server
  - Backup e ripristino di un'istanza

Queste utilità sostituiscono gli script Perl e shell contrassegnati come deprecati in Directory Server 10. Gli script sono ancora disponibili nel **389-ds-base-legacy-tools** pacchetto non supportato, tuttavia Red Hat supporta solo la gestione di Directory Server utilizzando le nuove utilità.

Si noti che la configurazione di Directory Server utilizzando istruzioni LDIF è ancora supportata, ma Red Hat consiglia di utilizzare le utilità.

Per ulteriori dettagli sull'utilizzo delle utilità, vedere il file [Red Hat Directory Server 11 Documentation](#) .

([BZ#1693159](#))

### I comandi **pki subsystem-cert-finde pki subsystem-cert-show**e mostrano ora il numero di serie dei certificati

Con questo miglioramento, i comandi **pki subsystem-cert-find** e **pki subsystem-cert-show** nel sistema di certificazione mostrano il numero di serie dei certificati in uscita. Il numero di serie è un'informazione importante e spesso richiesto da molteplici altri comandi. Di conseguenza, l'identificazione del numero di serie di un certificato è ora più semplice.

([BZ#1566360](#))

### I comandi **pki usare pki group** sono stati deprecati in Certificate System

Con questo aggiornamento, i nuovi **pki <subsystem>-user** comandi e **pki <subsystem>-group** comandi sostituiscono i comandi **pki usare pki group** in Sistema Certificato. I comandi sostituiti funzionano ancora, ma visualizzano il messaggio che il comando è deprecato e si riferiscono ai nuovi comandi.

([BZ#1394069](#))

### Certificate System supporta ora il rinnovo offline dei certificati di sistema

Con questo miglioramento, gli amministratori possono utilizzare la funzione di rinnovo offline per rinnovare i certificati di sistema configurati in Certificate System. Quando un certificato di sistema scade, il sistema di certificazione non si avvia. Come risultato del miglioramento, gli amministratori non hanno più bisogno di soluzioni alternative per sostituire un certificato di sistema scaduto.

([BZ#1669257](#))

### Certificate System può ora creare CSR con estensione SKI per la firma di CA esterna

Con questo miglioramento, Certificate System supporta la creazione di una richiesta di firma del certificato (CSR) con l'estensione Subject Key Identifier (SKI) per la firma dell'autorità di certificazione esterna (CA). Alcune autorità competenti richiedono tale estensione o con un valore particolare o

derivato dalla chiave pubblica delle autorità competenti. Come risultato, gli amministratori possono ora utilizzare il **pki\_req\_ski** parametro nel file di configurazione passato all' **pkispawn** utilità per creare una CSR con estensione SKI.

(BZ#1656856)

### Gli utenti locali sono messi in cache da SSSD e serviti attraverso il **nss\_sss** modulo

In RHEL 8, il demone dei servizi di sicurezza di sistema (SSSD) serve utenti e gruppi da **/etc/passwd** /**/etc/groups** file per impostazione predefinita. Il modulo **sss** nsswitch precede i file nel **/etc/nsswitch.conf** file.

Il vantaggio di servire gli utenti locali attraverso SSSD è che il **nss\_sss** modulo ha una velocità **memory-mapped cache** che velocizza le ricerche di Name Service Switch (NSS) rispetto all'accesso al disco e all'apertura dei file su ogni richiesta NSS. In precedenza, il demone Name service cache daemon (**nscd**) ha contribuito ad accelerare il processo di accesso al disco. Tuttavia, l'uso **nscd** in parallelo con SSSD è complicato, in quanto sia SSSD e **nscd** utilizzare la propria cache indipendente. Di conseguenza, l'utilizzo **nscd** in configurazioni in cui SSSD serve anche utenti da un dominio remoto, ad esempio LDAP o Active Directory, può causare un comportamento imprevedibile.

Con questo aggiornamento, la risoluzione degli utenti e dei gruppi locali è più veloce in RHEL 8. Si noti che l'**root** utente non è mai gestito da SSSD, quindi la **root** risoluzione non può essere influenzata da un potenziale bug in SSSD. Si noti anche che se SSSD non è in esecuzione, il **nss\_sss** modulo gestisce la situazione con grazia, ripiegando per **nss\_files** evitare problemi. Non è necessario configurare SSSD in alcun modo, il dominio dei file viene aggiunto automaticamente.

(JIRA:RHELPLAN-10439)

### KCM sostituisce KEYRING come memoria cache delle credenziali di default

In RHEL 8, la cache delle credenziali di default è il Kerberos Credential Manager (KCM) che è supportato dal **sss-kcm** daemon. KCM supera le limitazioni del KEYRING precedentemente utilizzato, come ad esempio la sua difficoltà di utilizzo in ambienti containerizzati perché non ha un percorso dei nomi, e di visualizzare e gestire le quote.

Con questo aggiornamento, RHEL 8 contiene una cache delle credenziali che è più adatta per ambienti containerizzati e che fornisce una base per costruire più funzionalità nelle versioni future.

(JIRA:RHELPLAN-10440)

### Gli utenti di Active Directory possono ora amministrare la gestione delle identità

Con questo aggiornamento, RHEL 8 consente di aggiungere un ID utente per un utente di Active Directory (AD) come membro di un gruppo Identity Management (IdM). Un ID override è un record che descrive l'aspetto di un utente AD specifico o di un gruppo di proprietà all'interno di una specifica vista ID, in questo caso la vista di fiducia predefinita. Come conseguenza dell'aggiornamento, il server IdM LDAP è in grado di applicare all'utente AD le regole di controllo accessi per il gruppo IdM.

Gli utenti AD sono ora in grado di utilizzare le funzioni self service dell'interfaccia utente IdM, ad esempio per caricare le proprie chiavi SSH o modificare i propri dati personali. Un amministratore AD è in grado di amministrare completamente IdM senza avere due diversi account e password. Si noti che attualmente, le funzioni selezionate in IdM potrebbero non essere ancora disponibili per gli utenti AD.

(JIRA:RHELPLAN-10442)

### **sssctl** stampa un rapporto sulle regole HBAC per un dominio IdM

Con questo aggiornamento, l'**sssctl** utilità del System Security Services Daemon (SSSD) può stampare



un report di controllo degli accessi per un dominio IdM (Identity Management). Questa funzione risponde all'esigenza di alcuni ambienti di vedere, per motivi normativi, un elenco di utenti e gruppi che possono accedere a una specifica macchina client. L'esecuzione **sssctl access-report domain\_namesu** un client IdM stampa il sottoinsieme analizzato delle regole HBAC (host-based access control) nel dominio IdM che si applicano alla macchina client.

Si noti che nessun altro fornitore oltre a IdM supporta questa funzione.

(JIRA:RHELPLAN-10443)

### I pacchetti di Identity Management sono disponibili come modulo

In RHEL 8, i pacchetti necessari per l'installazione di un server e di un client di Identity Management (IdM) sono forniti come modulo. Il **client** flusso è il flusso predefinito del **idm** modulo ed è possibile scaricare i pacchetti necessari per installare il client senza abilitare il flusso.

Lo stream del modulo server IdM è chiamato **DL1** stream. Lo stream contiene profili multipli corrispondenti a diversi tipi di server IdM: server, dns, adtrust, client e default. Per scaricare i pacchetti in un profilo specifico del **DL1** flusso: . Attivare il flusso. . . Passare ai RPM trasmessi attraverso il flusso. . Esegui il **yum module install idm:DL1/profile\_name** comando.

(JIRA:RHELPLAN-10438)

### Aggiunta la soluzione di registrazione delle sessioni per RHEL 8

Una soluzione di registrazione della sessione è stata aggiunta a Red Hat Enterprise Linux 8 (RHEL 8). Un nuovo **tlog** pacchetto e il relativo lettore di sessioni della console web consentono di registrare e riprodurre le sessioni del terminale utente. La registrazione può essere configurata per utente o gruppo di utenti tramite il servizio System Security Services Daemon (SSSD). Tutti gli ingressi e le uscite dei terminali vengono catturati e memorizzati in formato testuale in un giornale di sistema. L'input è inattivo di default per motivi di sicurezza per non intercettare password grezze e altre informazioni sensibili.

La soluzione può essere utilizzata per l'auditing delle sessioni utente su sistemi sensibili alla sicurezza. In caso di violazione della sicurezza, le sessioni registrate possono essere esaminate nell'ambito di un'analisi forense. Gli amministratori di sistema sono ora in grado di configurare la registrazione della sessione in locale e visualizzare il risultato dall'interfaccia web console di RHEL 8 o dall'interfaccia a riga di comando utilizzando l'**tlog-play** utilità.

(JIRA:RHELPLAN-1473)

### authselect semplifica la configurazione dell'autenticazione utente

Questo aggiornamento introduce l'utilità che semplifica la configurazione dell'autenticazione utente sugli host RHEL 8, sostituendo l'**authconfig** utilità. **authselect** viene fornito con un approccio più sicuro alla gestione dello stack PAM che semplifica le modifiche alla configurazione PAM per gli amministratori di sistema. **authselect** può essere utilizzato per configurare metodi di autenticazione come password, certificati, smart card e impronte digitali. Si noti che **authselect** non configura i servizi richiesti per unire domini remoti. Questo compito viene eseguito da strumenti specializzati, come **realmd** o **ipa-client-install**.

(JIRA:RHELPLAN-10445)

### SSSD consente ora di selezionare uno dei diversi dispositivi di autenticazione Smartcard

Con questo aggiornamento, è possibile configurare l'URI PKCS#11 per la selezione dei dispositivi di autenticazione Smartcard.

Per impostazione predefinita, SSSD tenta di rilevare automaticamente un dispositivo per l'autenticazione Smartcard. Se sono collegati più dispositivi, SSSD selezionerà il primo trovato e non è possibile selezionare il dispositivo specifico. Può portare a fallimenti.

Pertanto, è ora possibile configurare una nuova **p11\_uri**opzione per la **[pam]**sezione di **sssd.conf**. Questa opzione consente di definire quale dispositivo verrà utilizzato per l'autenticazione Smartcard.

Ad esempio, per selezionare il lettore con lo slot id '2' rilevato dal modulo OpenSC PKCS#11, aggiungere

```
p11_uri = library-description=OpenSC%20smartcard%20framework;slot-id=2
```

alla **[pam]**sezione di **sssd.conf**.

Si prega di vedere **man sssd-conf**per i dettagli.

(BZ#1620123)

## 4.10. COMPILATORI E STRUMENTI DI SVILUPPO (TRADUZIONE AUTOMATICA)

### Boost aggiornato alla versione 1.66

La libreria **Boost** C++ è stata aggiornata alla versione upstream 1.66. La versione di **Boost** inclusa in Red Hat Enterprise Linux 7 è 1.53. Per i dettagli, vedere i changelogs a monte:

<https://www.boost.org/users/history/>

Questo aggiornamento introduce le seguenti modifiche che rompono la compatibilità con le versioni precedenti:

- La **bs\_set\_hook()**funzione, la **splay\_set\_hook()**funzione dai contenitori splay e i parametri **bool splay = true**extra nella **splaytree\_algorithms()**funzione nella libreria **Intrusive** sono stati rimossi.
- I commenti o la concatenazione di stringhe nei file JSON non sono più supportati dall'analizzatore nella libreria **Property Tree**.
- Alcune distribuzioni e funzioni speciali della libreria **Math** sono state sistemate per comportarsi come documentato e aumentare un **overflow\_error**valore massimo finito invece di restituire il valore massimo.
- Alcune intestazioni della libreria **Math** sono state spostate nella directory **libs/math/include\_private**.
- Il comportamento delle funzioni **basic\_regex<>::mark\_count()**e **basic\_regex<>::subexpression(n)**delle funzioni della libreria **Regex** è stato modificato per adattarlo alla loro documentazione.
- L'uso di modelli variabili nella libreria **Variant** può interrompere le funzioni di metaprogrammazione.
- L'**boost::python::numeric**API è stata rimossa. Gli utenti possono utilizzare **boost::python::numpy**invece.
- Le operazioni aritmetiche su puntatori a tipi di non oggetti non sono più disponibili nella biblioteca atomica.

(BZ#1494495)

## Supporto Unicode 11.0.0.0

La libreria Red Hat Enterprise Linux core C, **glibc**, è stata aggiornata per supportare la versione standard Unicode 11.0.0.0. Di conseguenza, tutte le API a caratteri larghi e a più byte, comprese la traslitterazione e la conversione tra set di caratteri, forniscono informazioni accurate e corrette conformi a questo standard.

(BZ#1512004)

## Il boostpacchetto è ora indipendente da Python

Con questo aggiornamento, l'installazione del **boost**pacchetto non installa più la **Boost.Python**libreria come dipendenza. Per poterlo utilizzare **Boost.Python**, è necessario installare esplicitamente i pacchetti **boost-python3**o **boost-python3-devel**i pacchetti.

(BZ#1616244)

## Un nuovo compat-libgfortran-48pacchetto disponibile

Per la compatibilità con le applicazioni Red Hat Enterprise Linux 6 e 7 che utilizzano la libreria Fortran, è ora disponibile un nuovo pacchetto di **compat-libgfortran-48**compatibilità che fornisce la **libgfortran.so.3**libreria.

(BZ#1607227)

## Supporto Retpoline in GCC

Questo aggiornamento aggiunge il supporto per i retpolines a GCC. Una retpoline è un costruito software usato dal kernel per ridurre l'overhead degli attacchi Spectre Variant 2 descritti in CVE-2017-57-5715.

(BZ#1535774)

## Supporto avanzato per l'architettura ARM a 64 bit nei componenti della toolchain

I componenti della toolchain **GCCe binutils**, ora forniscono un supporto esteso per l'architettura ARM a 64 bit. Per esempio:

- **GCC** e **binutils**ora supportano Scalable Vector Extension (SVE).
- Il supporto per il tipo di **FP16**dati, fornito da ARM v8.2, è stato aggiunto a **GCC**. Il tipo di **FP16**dati migliora le prestazioni di alcuni algoritmi.
- Gli strumenti da **binutils**ora supportano la definizione dell'architettura ARM v8.3, inclusa l'autenticazione del puntatore. La funzione Pointer Authentication impedisce al codice dannoso di corrompere la normale esecuzione di un programma o del kernel creando i propri puntatori di funzione. Come risultato, solo gli indirizzi fidati sono usati quando si ramifica in luoghi diversi del codice, il che migliora la sicurezza.

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

## Ottimizzazioni per glibci sistemi IBM POWER

Questo aggiornamento fornisce **glibc**una nuova versione ottimizzata sia per le architetture IBM POWER 8 che IBM POWER 9. Come risultato, i sistemi IBM POWER 8 e IBM POWER 9 ora passano automaticamente alla variante appropriata e ottimizzata **glibc**durante l'esecuzione.

(BZ#1376834)

## GNU C Library aggiornato alla versione 2.28

Red Hat Enterprise Linux 8 include la versione 2.28 della GNU C Library (glibc). Notevoli miglioramenti includono:

- Caratteristiche di tempra di sicurezza:
  - I file binari sicuri contrassegnati con la **AT\_SECURE** bandiera ignorano la variabile d'**LD\_LIBRARY\_PATH** ambiente.
  - I backtrace non vengono più stampati per verificare i guasti della pila per accelerare lo spegnimento ed evitare di eseguire più codice in un ambiente compromesso.
- Miglioramenti delle prestazioni:
  - Le prestazioni della **malloc()** funzione sono state migliorate con una cache locale del thread.
  - Aggiunta della variabile d'**GLIBC\_TUNABLES** ambiente per modificare le caratteristiche di performance della libreria.
  - L'implementazione di semafori di fili è stata migliorata e sono state aggiunte nuove funzioni scalabili **pthread\_rwlock\_xxx()**.
  - Le prestazioni della biblioteca di matematica sono state migliorate.
- È stato aggiunto il supporto per Unicode 11.0.0.0.0.
- È stato aggiunto il supporto migliorato per i numeri in virgola mobile a 128 bit come definito dagli standard ISO/IEC/IEEE 60559:2011, IEEE 754-2008 e ISO/IEC TS 18661-3:2015.
- Miglioramenti al risolutore di stub resolver del Domain Name Service (DNS) relativi al file di **/etc/resolv.conf** configurazione:
  - La configurazione viene ricaricata automaticamente quando il file viene modificato.
  - È stato aggiunto il supporto per un numero arbitrario di domini di ricerca.
  - È stata aggiunta la corretta selezione casuale per l'**rotate** opzione.
- Sono state aggiunte nuove funzionalità per lo sviluppo, tra cui:
  - Funzioni di wrapper Linux per le chiamate al kernel **preadv2** e **pwritev2** al kernel
  - Nuove funzioni tra cui **reallocarray()** e **explicit\_bzero()**
  - Nuove bandiere per la **posix\_spawnattr\_setflags()** funzione come ad esempio **POSIX\_SPAWN\_SETSID**

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

## CMake disponibile in RHEL

Il sistema di compilazione CMake versione 3.11 è disponibile in Red Hat Enterprise Linux 8 come **cmake** pacchetto.

(BZ#1590139, BZ#1502802)

## make versione 4.2.1

Red Hat Enterprise Linux 8 è distribuito con lo strumento di **make** compilazione versione 4.2.1. Notevoli cambiamenti includono:

- Quando una ricetta fallisce, vengono mostrati il nome del makefile e il numero di riga della ricetta.
- L'**--trace** opzione è stata aggiunta per consentire la tracciabilità degli obiettivi. Quando si usa questa opzione, ogni ricetta viene stampata prima dell'invocazione, anche se sarebbe soppressa, insieme al nome del file e al numero di riga in cui si trova la ricetta, e anche con i prerequisiti che ne causano l'invocazione.
- Mescolare regole esplicite e implicite non comporta **make** più la cessazione dell'esecuzione. Invece, viene stampato un avviso. Si noti che questa sintassi è deprecata e potrebbe essere completamente rimossa in futuro.
- È stata aggiunta la **\$(file ...)** funzione per scrivere testo in un file. Quando viene chiamato senza un argomento di testo, apre e chiude immediatamente il file.
- Una nuova opzione, **--output-sync -O**, fa sì che un output di più lavori sia raggruppato per lavoro e consente un più facile debug delle costruzioni parallele.
- L'**--debug** opzione ora accetta anche il flag (nessuno **n**) per disabilitare tutte le impostazioni di debug attualmente abilitate.
- L'operatore di assegnazione delle **!=shell** è stato aggiunto come alternativa alla **\$(shell ...)** funzione per aumentare la compatibilità con i makefile BSD. Per maggiori dettagli e differenze tra l'operatore e la funzione, vedere il manuale GNU make manual.

Si noti che, di conseguenza, le variabili con un nome che termina con un punto esclamativo e immediatamente seguito da un'assegnazione, come ad esempio **variable!=value**, sono ora interpretate come la nuova sintassi. Per ripristinare il comportamento precedente, aggiungere uno spazio dopo il punto esclamativo, ad esempio **variable! =value**.

+

- È stato aggiunto l'operatore di **::=** assegnazione definito dallo standard POSIX.
- Quando la **.POSIX** variabile è specificata, **make** osserva i requisiti standard POSIX per la gestione del backslash e della nuova linea. In questa modalità, qualsiasi spazio d'uscita prima del backslash viene preservato, e ogni backslash seguito da una nuova linea e da caratteri di spazio bianco viene convertito in un singolo carattere di spazio.
- Il comportamento delle variabili **MAKEFLAGS** e **MFLAGS** delle variabili è ora definito con maggiore precisione.
- Una nuova variabile, **GNUMAKEFLAGS**, viene analizzata per le **make** lanterne in modo identico a **MAKEFLAGS**. Di conseguenza, i flag **make** specifici per GNU possono essere memorizzati all'esterno **MAKEFLAGS** e la portabilità dei makefile è aumentata.
- È stata aggiunta una nuova variabile, **MAKE\_HOST** contenente l'architettura host.
- Le nuove variabili, **MAKE\_TERMOUT** e **MAKE\_TERMERR**, indicano se **make** sta scrivendo l'uscita standard e l'errore ad un terminale.
- L'impostazione delle opzioni **-re -R** della **MAKEFLAGS** variabile all'interno di un makefile ora funziona correttamente e rimuove tutte le regole e le variabili incorporate, rispettivamente.

- L'.**RECIPEPREFIX** impostazione è ora ricordata per ogni ricetta. Inoltre, le variabili espresse in quella ricetta utilizzano anche l'impostazione del prefisso della ricetta.
- L'.**RECIPEPREFIX** impostazione e tutte le variabili specifiche del target sono visualizzate nell'output dell'**-p** opzione come in un makefile, invece che come commenti.

(BZ#1641015)

### I programmi Go costruiti con Go Toolset sono conformi a FIPS

La libreria crittografica disponibile in **Go Toolset** è stata modificata per utilizzare la libreria **OpenSSL** versione 1.1.0 se il sistema host è configurato in modalità FIPS. Di conseguenza, i programmi costruiti con questa versione di **Go Toolset** sono conformi a FIPS.

Per fare in modo che i programmi Go utilizzino solo le routine crittografiche standard non certificate, utilizzare l'**-tags no\_openssl** opzione del **go** compilatore in fase di compilazione.

(BZ#1512570)

### SystemTap versione 4.0

Red Hat Enterprise Linux 8 è distribuito con lo strumento di strumentazione **SystemTap** versione 4.0. Notevoli miglioramenti includono:

- Il backend esteso del Berkeley Packet Filter (eBPF) è stato migliorato, in particolare per quanto riguarda le stringhe e le funzioni. Per utilizzare questo backend, avviare **SystemTap** con l'**--runtime=bpf** opzione.
- È stato aggiunto un nuovo servizio di rete di esportazione da utilizzare con il sistema di monitoraggio Prometheus.
- L'implementazione dell'ispezione delle chiamate di sistema è stata migliorata per utilizzare i tracepoint del kernel, se necessario.

(BZ#1641032)

### Miglioramenti nella binutils versione 2.30

Red Hat Enterprise Linux 8 include la versione 2.30 del **binutils** pacchetto. Notevoli miglioramenti includono:

- È stato migliorato il supporto per le nuove estensioni dell'architettura s390x.

Assemblatore:

- Sono stati aggiunti il supporto per il formato di file WebAssembly e la conversione di WebAssembly nel formato di file wasm32 ELF.
- È stato aggiunto il supporto per l'architettura ARMv8-R e i processori Cortex-R52, Cortex-M23 e Cortex-M33.
- È stato aggiunto il supporto per l'architettura RISC-V.

Linkers:

- Il linker ora inserisce di default codice e dati di sola lettura in segmenti separati. Come risultato, i file eseguibili creati sono più grandi e più sicuri da eseguire, perché il caricatore dinamico può disabilitare l'esecuzione di qualsiasi pagina di memoria contenente dati in sola lettura.

- È stato aggiunto il supporto per le note di proprietà GNU che forniscono suggerimenti al caricatore dinamico sul file binario.
- In precedenza, il linker ha generato codice eseguibile non valido per la tecnologia Intel Indirect Branch Tracking (IBT). Di conseguenza, i file eseguibili generati non potevano partire. Questo bug è stato corretto.
- In precedenza, il **goldlinker** ha unito le note sulla proprietà in modo improprio. Di conseguenza, le caratteristiche hardware errate potrebbero essere abilitate nel codice generato e il codice potrebbe terminare inaspettatamente. Questo bug è stato corretto.
- In precedenza, il **goldlinker** creava sezioni di note con byte di riempimento alla fine per ottenere l'allineamento secondo l'architettura. Poiché il caricatore dinamico non si aspettava l'imbottitura, ha ceduto terminare inaspettatamente il programma che stava caricando. Questo bug è stato corretto.

Altri strumenti:

- Gli strumenti **readelf** e **objdump** hanno ora opzioni per seguire i link in file di informazioni di debug separati e visualizzare le informazioni anche in essi.
- La nuova **--inlines** opzione estende l'opzione esistente **--line-numbers** dello **objdump** strumento per visualizzare le informazioni di nesting per le funzioni in linea.
- Lo **nm** strumento ha acquisito una nuova opzione **--with-version-strings** per visualizzare le informazioni sulla versione di un simbolo dopo il suo nome, se presente.

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

### Performance co-pilota versione 4.1.3

Red Hat Enterprise Linux 8 è distribuito con **Performance co-pilot** (pcp) versione 4.1.3. Notevoli miglioramenti includono:

- Lo **pcp-dstat** strumento ora include l'analisi storica e l'output in formato CSV (Comma-separated Values).
- Le utilità di log possono utilizzare etichette metriche e record di testo di aiuto.
- Lo **pmdaperfevent** strumento ora riporta i numeri di CPU corretti ai livelli più bassi di Simultaneous Multi Threading (SMT).
- Lo **pmdapostgresql** strumento supporta ora **Postgres** serie 10.x.
- Lo **pmdaredis** strumento ora supporta **Redis** serie 5.x.
- Lo **pmdabcc** strumento è stato migliorato con il filtraggio dinamico del processo e le chiamate di sistema per processo, ucalls e ustat.
- Lo **pmdammv** strumento ora esporta le etichette metriche e la versione in formato è aumentata a 3.
- L'**pmdagfs2** utensile supporta metriche aggiuntive di glock e glock holder.
- Sono state apportate diverse correzioni alla politica SELinux.

(BZ#1641034)

## Chiavi di protezione della memoria

Questo aggiornamento consente di abilitare funzioni hardware che consentono di cambiare il flag di protezione per ogni pagina di thread. Sono stati aggiunti i nuovi involucri di chiamata di **glibc** sistema per le **pkey\_mprotect()** funzioni **pkey\_alloc()**, **pkey\_free()**, e **pkey\_get()**. Inoltre, sono state aggiunte le funzioni **pkey\_set()** e **pkey\_get()** le funzioni per consentire l'accesso ai flag di protezione per thread.

(BZ#1304448)

## elfutils aggiornato alla versione 0.174

In Red Hat Enterprise Linux 8, il pacchetto **elfutils** è disponibile nella versione 0.174. Notevoli cambiamenti includono:

- In precedenza, lo **eu-readelf** strumento poteva mostrare una variabile con un valore negativo come se avesse un grande valore non firmato, o mostrare un grande valore non firmato come valore negativo. Questo è stato corretto e **eu-readelf** ora cerca le dimensioni e la segnatura dei tipi di valori costanti per visualizzarli correttamente.
- Alla libreria **libdw** è stata aggiunta una nuova funzione **dwarf\_next\_lines()** per la lettura **.debug\_line** di dati privi di CU. Questa funzione può essere utilizzata in alternativa alle funzioni **dwarf\_getsrclines()** e **dwarf\_getsrcfiles()**.
- In precedenza, i file con più di 65280 sezioni potevano causare errori nelle librerie **libelf** e **libdw** e in tutti gli strumenti che li utilizzavano. Questo bug è stato corretto. Come risultato, le **shstrndx** intestazioni dei file ELF vengono gestite correttamente. **shnum**

(BZ#1641007)

## Valgrind aggiornato alla versione 3.14

Red Hat Enterprise Linux 8 è distribuito con lo strumento di analisi del codice eseguibile Valgrind versione 3.14. Notevoli cambiamenti includono:

- È stata aggiunta una nuova **--keep-debuginfo** opzione per consentire la conservazione delle informazioni di debug per il codice scaricato. Come risultato, le tracce di pila salvate possono includere informazioni su file e linee per il codice che non è più presente in memoria.
- Sono state aggiunte soppressioni basate sul nome del file di origine e sul numero di riga.
- Lo **Helgrind** strumento è stato esteso con un'opzione **--delta-stacktrace** per specificare il calcolo delle tracce complete dello stack storico. In particolare, l'utilizzo di questa opzione insieme **--history-level=full** possono migliorare le **Helgrind** prestazioni fino al 25%.
- È stato ridotto il tasso di falsi positivi nello **Memcheck** strumento per l'ottimizzazione del codice sulle architetture Intel e AMD a 64 bit e sull'architettura ARM a 64 bit. Si noti che è possibile utilizzare il **--expensive-definedness-checks** per controllare la gestione dei controlli di definizione e migliorare il tasso a scapito delle prestazioni.
- Valgrind può ora riconoscere più istruzioni della variante little-endian di IBM Power Systems.
- Valgrind è ora in grado di elaborare parzialmente istruzioni intere e vettoriali di stringa del processore IBM Z architecture z13.

Per maggiori informazioni sulle nuove opzioni e sulle limitazioni conosciute, vedere la pagina del **valgrind(1)** manuale.

(BZ#1641029, BZ#1501419)



## GDB versione 8.2

Red Hat Enterprise Linux 8 è distribuito con la versione 8.2 del debugger di GDB:

- Il protocollo IPv6 è supportato per il debug remoto con GDB e **gdbserver**.
- Il debug senza informazioni di debug è stato migliorato.
- Il completamento dei simboli nell'interfaccia utente di GDB è stato migliorato per offrire migliori suggerimenti utilizzando costruzioni sintattiche come i tag ABI o gli spazi dei nomi.
- I comandi possono ora essere eseguiti in background.
- Il debug dei programmi creati nel linguaggio di programmazione Rust è ora possibile.
- Il debug dei linguaggi C e C++ è stato migliorato con il supporto dell'analizzatore **\_Alignof** e degli operatori, riferimenti al valore `rue` C++ e array automatici a lunghezza variabile C99.
- Gli script di estensione GDB possono ora utilizzare il linguaggio di scripting Guile.
- L'interfaccia del linguaggio di scripting Python per le estensioni è stata migliorata con nuove funzioni API, decoratori di cornici, filtri e svolgitori. Inoltre, gli script nella **.debug\_gdb\_scripts** sezione della configurazione di GDB vengono caricati automaticamente.
- GDB ora usa Python versione 3 per eseguire i suoi script, incluse stampanti, decoratori di cornici, filtri e svolgitori.
- Le architetture ARM e ARM a 64 bit sono state migliorate con la registrazione e la riproduzione dell'esecuzione dei processi, comprese le istruzioni per le chiamate di sistema e Thumb 32-bit.
- È stato aggiunto il supporto per il registro MPX Intel e la violazione legata, il registro PKU ed è stato aggiunto il processore Intel Trace.
- Le funzionalità di registrazione e riproduzione sono state estese per includere le **rdrandrdseed** istruzioni sui sistemi basati su Intel.
- La funzionalità di GDB sull'architettura IBM Z è stata estesa con il supporto di tracepoint e fast tracepoints, registri vettoriali e ABI, e la chiamata di sistema **Catch**. Inoltre, GDB supporta ora istruzioni più recenti dell'architettura.
- GDB può ora utilizzare le sonde statiche di spazio utente SystemTap (SDT) sull'architettura ARM a 64 bit.

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

### La localizzazione per RHEL è distribuita in più pacchetti

In RHEL 8, le località e le traduzioni non sono più fornite dal singolo **glibc-common** pacchetto. Invece, ogni locale e lingua è disponibile in un **glibc-langpack-CODE** pacchetto. Inoltre, non tutti i locali sono installati di default, ma solo quelli selezionati nel programma di installazione. Gli utenti devono installare tutti gli altri pacchetti locali di cui hanno bisogno separatamente.

Per ulteriori informazioni sull'uso degli zaini, vedere [Installing and using langpacks](#).

(BZ#1512009)

### strace aggiornato alla versione 4.24

Red Hat Enterprise Linux 8 è distribuito con lo **strace** strumento versione 4.24. Notevoli cambiamenti includono:

- Le funzioni di manomissione delle chiamate di sistema sono state aggiunte con l'-e **inject=**opzione. Ciò include l'iniezione di errori, valori di ritorno, ritardi e segnali.
- La qualificazione delle chiamate di sistema è stata migliorata:
  - È stata aggiunta un'-e **trace=/regex**opzione per filtrare le chiamate di sistema con espressioni regolari.
  - Prependendo un punto interrogativo a una qualifica di chiamata di sistema nell'-e **trace=**opzione permette di **strace**continuare, anche se la qualifica non corrisponde a nessuna chiamata di sistema.
  - Nell'-e **trace**opzione è stata aggiunta la designazione della personalità alle qualifiche per le chiamate di sistema.
- È stata aggiunta la decodifica del motivo di **kvm vcpu**uscita. Per farlo, utilizzare l'-e **kvm=vcpu**opzione.
- La **libdw**libreria è ora usata per lo svolgimento in pila quando si usa l' **-k**opzione. Inoltre, è possibile demangling dei simboli, quando la **libiberty**libreria è installata sul sistema.
- In precedenza, l'-ropzione ha fatto **strace**sì che l'-topzione fosse ignorata. La questione è stata risolta e le due opzioni sono ora indipendenti.
- È stata aggiunta l'opzione [option]`-A per aprire i file di output in modalità append.
- È stata aggiunta l'-Xopzione per configurare **xlat**la formattazione dell'output.
- È stata migliorata la decodifica degli indirizzi delle prese con l'-yyopzione. Inoltre, è stata aggiunta la stampa del numero del dispositivo a blocchi e caratteri in -yymodalità.

Inoltre, è stata aggiunta, migliorata o aggiornata la decodifica dei seguenti elementi:

- **netlink** protocolli, messaggi e attributi
- **arch\_prctl**, **bpf**, **getsockopt**, **io\_pgetevent**, **kern\_features**, **keyctlprctl**, **pkey\_alloc**, **pkey\_free**, **pkey\_mprotect**, **ptrace**, **rseq**, **setsockopt**, **socket**, **statx**e altre chiamate di sistema
- Molti comandi per la chiamata di **ioctl**sistema
- Costanti di vario tipo
- Tracciamento dei percorsi per **execveat**, **inotify\_add\_watch**, **inotify\_init**, **select**, **symlink**, **symlinkat**chiamate di sistema e chiamate di **mmap**sistema con argomenti indiretti
- Chiamate di sistema specifiche per le architetture ARM **\_\_ARM\_NR\_\***
- Elenchi di codici di segnale

(BZ#1641014)

## 4.11. SISTEMI DI FILE E ARCHIVIAZIONE (TRADUZIONE AUTOMATICA)

**YES** ora supporta estensioni di dati condivisi **copy on write**

## XFS ora supporta estensioni di dati condivisi copy-on-write

Il file system XFS supporta la funzionalità di estensione dei dati condivisa copy-on-write. Questa funzione consente a due o più file di condividere un insieme comune di blocchi di dati. Quando uno dei file che condividono blocchi comuni cambia, XFS interrompe il collegamento ai blocchi comuni e crea un nuovo file. Questa è simile alla funzionalità di copia in scrittura (COW) che si trova in altri file system.

Le estensioni dei dati in copia e scrittura condivisa sono:

### Veloce

La creazione di copie condivise non utilizza l'I/O del disco.

### Efficiente dal punto di vista dello spazio

I blocchi condivisi non consumano spazio su disco aggiuntivo.

### Trasparente

I file che condividono blocchi comuni agiscono come file normali.

Le utilità dello spazio utente possono utilizzare estensioni di dati condivisi copy-on-write per:

- Clonazione efficiente dei file, come ad esempio con il **cp --reflink** comando
- Istantanee per file

Questa funzionalità è utilizzata anche da sottosistemi del kernel come Overlayfs e NFS per un funzionamento più efficiente.

Gli estensioni di dati condivisi copy-on-write sono ora abilitati di default quando si crea un file system XFS, a partire dalla versione **4.17.0-2.el8** del **xfsprogs** pacchetto .

Si noti che i dispositivi ad accesso diretto (DAX) attualmente non supportano XFS con estensioni di dati condivisi copy-on-write. Per creare un file system XFS senza questa funzione, utilizzare il seguente comando:

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

Red Hat Enterprise Linux 7 può montare file system XFS con estensioni di dati condivisi copy-on-write solo in modalità di sola lettura.

(BZ#1494028)

## La dimensione massima del file system XFS è 1024 TiB

La dimensione massima supportata di un file system XFS è stata aumentata da 500 TiB a 1024 TiB.

I file system con dimensioni maggiori di 500 TiB lo richiedono:

- la funzione CRC dei metadati e la funzione free inode btree sono entrambe abilitate nel formato file system, e
- la dimensione del gruppo di assegnazione è di almeno 512 GiB.

In RHEL 8, l'**mkfs.xfs** utilità crea file system che soddisfano questi requisiti per impostazione predefinita.

Non è supportata la crescita di un file system più piccolo che non soddisfa questi requisiti fino ad una nuova dimensione superiore a 500 TiB.

(BZ#1563617)

## VDO supporta ora tutte le architetture

Virtual Data Optimizer (VDO) è ora disponibile su tutte le architetture supportate da RHEL 8.

Per l'elenco delle architetture supportate, vedere [Capitolo 2, Architetture \(traduzione automatica\)](#).

(BZ#1534087)

## Il boot manager BOOM semplifica il processo di creazione di voci di boot

BOOM è un boot manager per sistemi Linux che utilizzano boot loader che supportano la specifica BootLoader per la configurazione dell'avvio. Permette una configurazione flessibile dell'avvio e semplifica la creazione di voci di avvio nuove o modificate: ad esempio, per avviare immagini istantanee del sistema create utilizzando LVM.

BOOM non modifica la configurazione del boot loader esistente e inserisce solo voci aggiuntive. La configurazione esistente viene mantenuta e qualsiasi integrazione della distribuzione, come l'installazione del kernel e gli script di aggiornamento, continua a funzionare come prima.

BOOM ha un'interfaccia a riga di comando semplificata (CLI) e API che facilitano il compito di creare voci di avvio.

(BZ#1649582)

## LUKS2 è ora il formato predefinito per la cifratura dei volumi

In RHEL 8, il formato LUKS versione 2 (LUKS2) sostituisce il formato legacy LUKS (LUKS1). Il **dm-crypt** sottosistema e lo **cryptsetup** strumento ora utilizza LUKS2 come formato predefinito per i volumi cifrati. LUKS2 fornisce volumi crittografati con ridondanza dei metadati e recupero automatico in caso di evento di corruzione parziale dei metadati.

Grazie alla flessibilità del layout interno, LUKS2 è anche un abilitatore di funzionalità future. Supporta lo sblocco automatico attraverso il generico kernel-keyring token integrato **libcryptsetup** che permette agli utenti di sbloccare i volumi LUKS2 usando una passphrase memorizzata nel servizio di ritenzione del kernel-keyring.

Altri miglioramenti degni di nota sono:

- L'impostazione della chiave protetta utilizzando lo schema di cifratura a chiave avvolta.
- Integrazione più facile con la decrittazione basata sulle politiche (Clevis).
- Fino a 32 slot per chiavi - LUKS1 fornisce solo 8 slot per chiavi.

Per maggiori dettagli, vedere la pagina **cryptsetup(8)** e **cryptsetup-reencrypt(8)** la pagina dell'uomo.

(BZ#1564540)

## NVMe/FC è completamente supportato dagli adattatori Broadcom Emulex Fibre Channel Adapters

Il tipo di trasporto NVMe over Fibre Channel (NVMe/FC) è ora pienamente supportato in modalità Initiator se utilizzato con adattatori Broadcom Emulex Fibre Channel 32Gbit.

NVMe over Fibre Channel è un ulteriore tipo di trasporto in tessuto per il protocollo Non Volatile Memory Express (NVMe), in aggiunta al protocollo Remote Direct Memory Access (RDMA) precedentemente introdotto in Red Hat Enterprise Linux.

Per abilitare NVMe/FC nel **lpfc** driver, modificare il `/etc/modprobe.d/lpfc.conf` file e aggiungere la seguente opzione:

```
lpfc_enable_fc4_type=3
```

Driver diversi da **lpfc** quelli che rimangono ancora in Technology Preview.

Ulteriori restrizioni:

- Multipath non è supportato con NVMe/FC.
- Il clustering NVMe non è supportato con NVMe/FC.
- Attualmente, Red Hat Enterprise Linux non supporta l'uso contemporaneo di NVMe/FC e SCSI/FC su una porta iniziatore.
- Il pacchetto *kernel-alt* non supporta NVMe/FC.
- **kdump** non è supportato con NVMe/FC.
- L'avvio da Storage Area Network (SAN) NVMe/FC non è supportato.

(BZ#1649497)

### Nuova overridesezione del file di configurazione DM Multipath

Il `/etc/multipath.conf` file include ora una **overridesezione** che consente di impostare un valore di configurazione per tutti i dispositivi. Questi attributi sono utilizzati da DM Multipath per tutti i dispositivi a meno che non vengano sovrascritti dagli attributi specificati nella **multipaths** sezione del `/etc/multipath.conf` file per i percorsi che contengono il dispositivo. Questa funzionalità sostituisce il **all\_devs** parametro della **devices** sezione del file di configurazione, che non è più supportato.

(BZ#1643294)

### L'installazione e l'avvio da dispositivi NVDIMM è ora supportato

Prima di questo aggiornamento, i dispositivi Non Volatile Dual Inline Memory Module (NVDIMM) in qualsiasi modalità sono stati ignorati dall'installatore.

Con questo aggiornamento, i miglioramenti del kernel per supportare i dispositivi NVDIMM forniscono migliori prestazioni del sistema e un migliore accesso al file system per applicazioni ad alta intensità di scrittura come database o carichi di lavoro analitico, oltre a ridurre l'overhead della CPU.

Questo aggiornamento introduce il supporto per:

- L'utilizzo di dispositivi NVDIMM per l'installazione tramite il comando **nvdimm** Kickstart e l'interfaccia grafica, rendendo possibile installare e avviare da dispositivi NVDIMM in modalità settore e riconfigurare i dispositivi NVDIMM in modalità settore durante l'installazione.
- L'estensione degli **Kickstart** script per **Anaconda** con comandi per la gestione dei dispositivi NVDIMM.
- La capacità di **grub2**, **efibootmgr**, e componenti di **efivar** sistema di gestire e avviare da dispositivi NVDIMM.

(BZ#1499442)

### Migliorata la rilevazione dei percorsi marginali in DM Multipath

Il **multipathd** servizio supporta ora una migliore individuazione dei percorsi marginali. Questo aiuta i dispositivi multipath ad evitare percorsi che possono fallire ripetutamente e migliora le prestazioni. I percorsi marginali sono percorsi con errori I/O persistenti ma intermittenti.

Le seguenti opzioni nel comportamento dei percorsi marginali di controllo dei **/etc/multipath.conf** file:

- **marginal\_path\_double\_failed\_time**,
- **marginal\_path\_err\_sample\_time**,
- **marginal\_path\_err\_rate\_threshold**
- **marginal\_path\_err\_recheck\_gap\_time**.

DM Multipath disabilita un percorso e lo testa con I/O ripetuti per il tempo di campionamento configurato se:

- le **multipath.conf** opzioni elencate sono impostate,
- un percorso fallisce due volte nel tempo configurato, e
- altri percorsi sono disponibili.

Se il percorso ha un tasso di errore superiore a quello configurato durante questo test, DM Multipath lo ignora per il gap time configurato, e poi lo riprova per vedere se sta funzionando abbastanza bene da poter essere ripristinato.

Per ulteriori informazioni, vedere la pagina dell'**multipath.conf** uomo.

(BZ#1643550)

### Comportamento predefinito multi-coda

I dispositivi a blocchi ora utilizzano la programmazione multi-coda in Red Hat Enterprise Linux 8. Ciò consente di scalare bene le prestazioni dello strato a blocchi con unità veloci a stato solido (SSD) e sistemi multi-core.

Il driver SCSI Multiqueue (**scsi-mq**) è ora abilitato di default, e il kernel si avvia con l'**scsi\_mod.use\_blk\_mq=Y** opzione. Questo cambiamento è coerente con il kernel Linux a monte.

Device Mapper Multipath (DM Multipath) richiede che il **scsi-mq** driver sia attivo.

(BZ#1647612)

### Stratis è ora disponibile

Stratis è un nuovo gestore di magazzino locale. Fornisce file system gestiti su pool di storage con funzionalità aggiuntive per l'utente.

Stratis consente di eseguire più facilmente attività di archiviazione come:

- Gestire le istantanee e il thin provisioning
- Aumenta automaticamente le dimensioni del file system in base alle necessità
- Manutenzione dei file system

Per amministrare l'archiviazione di Stratis, utilizzare l'**stratis** utilità, che comunica con il servizio in **stratisd** background.

Per ulteriori informazioni, consultare la documentazione di Stratis: [Managing layered local storage with Stratis](#).

(JIRA:RHELPLAN-1212)

## 4.12. ELEVATA DISPONIBILITÀ E CLUSTER (TRADUZIONE AUTOMATICA)

### Nuovi **pcs** comandi per elencare i dispositivi watchdog disponibili e i dispositivi watchdog di prova

Per configurare SBD con Pacemaker è necessario un dispositivo watchdog funzionante. Questa release supporta il **pcs stonith sbd watchdog list** comando per elencare i dispositivi watchdog disponibili sul nodo locale e il **pcs stonith sbd watchdog test** comando per testare un dispositivo watchdog. Per informazioni sullo strumento a riga di **sbd** comando, vedere la **sbd**(8) pagina man.

(BZ#1578891)

### Il **pcs** comando supporta ora il filtraggio dei guasti delle risorse in base a un'operazione e al suo intervallo

Pacemaker ora tiene traccia dei guasti delle risorse per ogni operazione di una risorsa, oltre al nome di una risorsa e di un nodo. Il **pcs resource failcount show** comando consente ora di filtrare i guasti per risorsa, nodo, operazione e intervallo. Fornisce un'opzione per visualizzare i guasti aggregati per risorsa e nodo o dettagliati per risorsa, nodo, operazione e relativo intervallo. Inoltre, il **pcs resource failcount reset** comando consente ora di filtrare i guasti per risorsa, nodo, operazione e intervallo.

(BZ#1591308)

### Timestamps abilitati nel **corosynclog**

In precedenza il **corosynclog** non conteneva timestamps, il che rendeva difficile collegarlo ai log di altri nodi e demoni. Con questa release, i timestamp sono presenti nel **corosynclog**.

(BZ#1615420)

### Nuovi formati per **pcs cluster setup**, **pcs cluster node add** e **pcs cluster node remove** comandi

In Red Hat Enterprise Linux 8, supporta **pcs** pienamente Corosync 3 **knet** e i nomi dei nodi. I nomi di nodo sono ora necessari e sostituiscono gli indirizzi di nodo nel ruolo di identificatore di nodo. Gli indirizzi di nodo sono ora opzionali.

- Nel **pcs host auth** comando, gli indirizzi di nodo predefiniti per i nomi di nodo
- Nei comandi **pcs cluster setup**, **pcs cluster node add**, gli indirizzi di nodo sono predefiniti agli indirizzi di nodo specificati nel **pcs host auth** comando.

Con queste modifiche, sono cambiati i formati dei comandi per impostare un cluster, aggiungere un nodo ad un cluster e rimuovere un nodo da un cluster. Per informazioni su questi nuovi formati di comando, vedere la guida per i **pcs cluster node remove** comandi **pcs cluster setup**, **pcs cluster node add** e i comandi.

(BZ#1158816)

## Pacemaker 2.0.0.0.0 in RHEL 8

I **pacemaker** pacchetti sono stati aggiornati alla versione upstream di Pacemaker 2.0.0.0.0, che fornisce una serie di correzioni e miglioramenti rispetto alla versione precedente:

- Il log di dettaglio del pacemaker è ora **/var/log/pacemaker/pacemaker.log** di default (non direttamente nel log **/var/log** in combinazione con il **corosynclog** sotto **/var/log/cluster**).
- I processi del demone Pacemaker sono stati rinominati per rendere più intuitiva la lettura dei log. Ad esempio, **pengine** è stato rinominato in **pacemaker-schedulerd**.
- Il supporto per le proprietà deprecate **default-resource-stickiness** e **is-managed-default** cluster è stato abbandonato. Le proprietà **resource-stickiness** e **is-managed** dovrebbero invece essere impostate nelle impostazioni predefinite delle risorse. Le configurazioni esistenti (anche se non di nuova creazione) con la sintassi deprecata verranno automaticamente aggiornate per utilizzare la sintassi supportata.
- Un elenco più completo delle modifiche è disponibile sul sito <https://access.redhat.com/articles/3681151>.

Si raccomanda agli utenti che stanno aggiornando un cluster esistente utilizzando Red Hat Enterprise Linux 7 o precedenti, di girare **pcs cluster cib-upgrade** su qualsiasi nodo del cluster prima e dopo l'aggiornamento di RHEL su tutti i nodi del cluster.

(BZ#1543494)

## Master risorse rinominate in risorse clone promovibili risorse clone

Red Hat Enterprise Linux (RHEL) 8 supporta Pacemaker 2.0, in cui una risorsa master/slave non è più un tipo di risorsa separata ma una risorsa clone standard con un **promotable** meta-attributo impostato su **true**. A sostegno di questo aggiornamento sono state apportate le seguenti modifiche:

- Non è più possibile creare risorse master con il **pcs** comando. Invece, è possibile creare risorse di **promotable** clonazione. Le parole chiave e i comandi correlati sono stati cambiati da **master** a **promotable**.
- Tutte le risorse master esistenti sono visualizzate come risorse clone promovibili.
- Quando si gestisce un cluster RHEL7 nell'interfaccia utente Web, le risorse master sono ancora chiamate master, dato che i cluster RHEL7 non supportano i cloni promozionali.

(BZ#1542288)

## Nuovi comandi per l'autenticazione dei nodi in un cluster

Red Hat Enterprise Linux (RHEL) 8 incorpora le seguenti modifiche ai comandi usati per autenticare i nodi in un cluster.

- Il nuovo comando per l'autenticazione è **pcs host auth**. Questo comando permette agli utenti di specificare nomi di host, indirizzi e **pcsd** porte.
- Il **pcs cluster auth** comando autentica solo i nodi in un cluster locale e non accetta una lista di nodi
- E' ora possibile specificare un indirizzo per ogni nodo. **pcs/pcsd** comunicherà con ogni nodo utilizzando l'indirizzo specificato. Questi indirizzi possono essere diversi da quelli **corosync** utilizzati internamente.



- Il **pcs pcsd clear-auth** comando è stato sostituito dai comandi **pcs pcsd deauth** e **pcs host deauth**. I nuovi comandi consentono agli utenti di deautenticare un singolo host e tutti gli host.
- In precedenza, l'autenticazione dei nodi era bidirezionale e l'esecuzione del **pcs cluster auth** comando ha fatto sì che tutti i nodi specificati fossero autenticati l'uno contro l'altro. Il **pcs host auth** comando, tuttavia, fa sì che solo l'host locale sia autenticato rispetto ai nodi specificati. Questo permette un migliore controllo di quale nodo è autenticato rispetto a quali altri nodi quando si esegue questo comando. Nella configurazione del cluster stesso, e anche quando si aggiunge un nodo, **pcs** sincronizza automaticamente i token sul cluster, in modo che tutti i nodi del cluster siano ancora automaticamente autenticati come prima e i nodi del cluster possano comunicare tra loro.

Si noti che queste modifiche non sono compatibili con le versioni precedenti. I nodi che sono stati autenticati su un sistema RHEL 7 dovranno essere autenticati di nuovo.

(BZ#1549535)

### I **pcs** comandi ora supportano la visualizzazione, la pulizia e la sincronizzazione della storia della schermata

Il demone della recinzione del pacemaker traccia la cronologia di tutte le azioni intraprese (in attesa, riuscite e non riuscite). Con questa release, i **pcs** comandi permettono agli utenti di accedere alla storia della schermata nei seguenti modi:

- Il **pcs status** comando mostra le azioni di schermata fallite e in sospeso
- Il **pcs status --full** comando mostra l'intera storia della schermata
- Il **pcs stonith history** comando fornisce opzioni per visualizzare e ripulire la storia della schermata
- Anche se la storia della schermata è sincronizzata automaticamente, il **pcs stonith history** comando ora supporta un' **update** opzione che permette all'utente di sincronizzare manualmente la storia della schermata, se necessario

(BZ#1620190, BZ#1615891)

## 4.13. LA RETE (TRADUZIONE AUTOMATICA)

### **nftables** sostituisce **iptables** come predefinito il framework di rete per il filtraggio dei pacchetti di rete

Il **nftables** framework fornisce strutture di classificazione dei pacchetti ed è il successore designato per il **iptables**, **ip6tables**, **arptables**, e **ebrtables** strumenti. Offre numerosi miglioramenti in termini di praticità, funzionalità e prestazioni rispetto ai precedenti strumenti di filtraggio dei pacchetti, in particolare:

- tabelle di ricerca al posto dell'elaborazione lineare
- un quadro unico sia per i **IPv6** protocolli **IPv4** che per i protocolli
- regole tutte applicate atomicamente invece di recuperare, aggiornare e memorizzare un insieme di regole completo
- supporto per il debug e il tracciamento nelle regole (**nfttrace**) e il monitoraggio degli eventi traccia (nello **nft** strumento)
- sintassi più coerente e compatta, nessuna estensione specifica del protocollo

- un'API Netlink per applicazioni di terze parti

Analogamente a **iptables**, **nftables** utilizzare i tavoli per riporre le catene. Le catene contengono regole individuali per l'esecuzione delle azioni. Lo **nft** strumento sostituisce tutti gli strumenti dei precedenti framework di filtraggio dei pacchetti. La **libnftables** libreria può essere utilizzata per interazioni di basso livello con le API **nftables** Netlink attraverso la **libmnl** libreria.

Gli **arptables** strumenti **iptables**, **ip6tables**, **ebtables**, sono sostituiti da sostituzioni drop-in basate su **nftables** con lo stesso nome. Mentre il comportamento esterno è identico a quello delle loro controparti legacy, internamente usano **nftables** con i moduli del kernel legacy **netfilter** attraverso un'interfaccia di compatibilità dove richiesto.

L'effetto dei moduli sulle **nftables** regole può essere osservato utilizzando il **nft list ruleset** comando. Dal momento che questi strumenti aggiungono tabelle, catene e regole al **nftables** set di regole, si tenga presente che le operazioni di **nftables** set di regole, come il **nft flush ruleset** comando, potrebbero influenzare i set di regole installati utilizzando i comandi legacy precedentemente separati.

Per identificare rapidamente quale variante dello strumento è presente, le informazioni sulla versione sono state aggiornate per includere il nome del back-end. In RHEL 8, lo strumento basato su **nftables** **iptables** stampa la seguente stringa di versione:

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

Per un confronto, vengono stampate le seguenti informazioni sulla versione se è presente uno strumento legacy **iptables**:

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

(BZ#1644030)

## Notevoli caratteristiche TCP in RHEL 8

Red Hat Enterprise Linux 8 è distribuito con lo stack di rete TCP versione 4.16, che fornisce prestazioni più elevate, migliore scalabilità e maggiore stabilità. Le prestazioni sono potenziate soprattutto per i server TCP occupati con un elevato tasso di connessione in ingresso.

Inoltre, sono disponibili due nuovi algoritmi di congestione TCP **BBR** e **NV**, che offrono una latenza inferiore e un throughput migliore del cubico nella maggior parte degli scenari.

(BZ#1562998)

## firewalld utilizza nftables per impostazione predefinita

Con questo aggiornamento, il sottosistema di **nftables** filtraggio è il backend firewall predefinito per il **firewalld** demone. Per modificare il backend, utilizzare l' **FirewallBackend** opzione nel **/etc/firewalld.conf** file.

Questa modifica introduce le seguenti differenze di comportamento nell'uso di **nftables**:

1. **iptables** l'esecuzione delle regole avviene sempre prima **firewalld** delle regole
  - **DROP** in **iptables** mezzi che un pacchetto non è mai visto da **firewalld**
  - **ACCEPT** in **iptables** significa che un pacchetto è ancora soggetto a delle **firewalld** regole

2. **firewalld** sono ancora implementate attraverso regole dirette, **iptables** mentre le altre **firewalld** funzioni utilizzano **nftables**
3. l'esecuzione diretta delle regole avviene prima dell'accettazione **firewalld** generica delle connessioni stabilite

(BZ#1509026)

### Notevole variazione **wpa\_supplicant** in RHEL 8

In Red Hat Enterprise Linux (RHEL) 8, il **wpa\_supplicant** pacchetto è costruito con **CONFIG\_DEBUG\_SYSLOG** abilitato. Questo permette di leggere il **wpa\_supplicant** log usando l'**journalctl** utilità invece di controllare il contenuto del `/var/log/wpa_supplicant.log` file.

(BZ#1582538)

### NetworkManager supporta ora le funzioni virtuali di SR-IOV

In Red Hat Enterprise Linux 8.0, **NetworkManager** consente di configurare il numero di funzioni virtuali (VF) per interfacce che supportano la virtualizzazione I/O a radice singola (SR-IOV). Inoltre, **NetworkManager** permette di configurare alcuni attributi delle VF, come l'indirizzo MAC, la VLAN, l'**spoof checking** impostazione e i bitrate consentiti. Si noti che tutte le proprietà relative a SR-IOV sono disponibili nelle impostazioni di **sriov** connessione. Per maggiori dettagli, vedere la pagina dell' **nm-settings(5)** uomo.

(BZ#1555013)

### I driver di rete virtuale IPVLAN sono ora supportati

In Red Hat Enterprise Linux 8.0, il kernel include il supporto per i driver di rete virtuale IPVLAN. Con questo aggiornamento, le schede di interfaccia di rete virtuale IPVLAN (NIC) consentono la connettività di rete per contenitori multipli che espongono un singolo indirizzo MAC alla rete locale. Questo permette ad un singolo host di avere molti container che superano la possibile limitazione del numero di indirizzi MAC supportati dalle apparecchiature di rete peer.

(BZ#1261167)

### NetworkManager supporta una corrispondenza del nome dell'interfaccia wildcard per le connessioni

In precedenza, era possibile limitare una connessione ad una data interfaccia utilizzando solo una corrispondenza esatta sul nome dell'interfaccia. Con questo aggiornamento, le connessioni hanno una nuova **match.interface-name** proprietà che supporta le wildcard. Questo aggiornamento consente agli utenti di scegliere l'interfaccia per una connessione in modo più flessibile utilizzando un modello wildcard.

(BZ#1555012)

### Miglioramenti nello stack di rete 4.18

Red Hat Enterprise Linux 8.0 include lo stack di rete aggiornato alla versione upstream 4.18, che fornisce diverse correzioni e miglioramenti. Notevoli cambiamenti includono:

- Introdotte nuove funzioni di scarico, come **UDP\_GSO**, e, per alcuni driver di periferica, **GRO\_HW**.
- Migliorata la scalabilità significativa per il protocollo UDP (User Datagram Protocol).
- Migliorato il codice generico per i sondaggi di occupato.

- Scalabilità migliorata per il protocollo IPv6.
- Migliore scalabilità del codice di routing.
- Aggiunto un nuovo algoritmo predefinito di programmazione della coda di trasmissione, **fq\_codel** che migliora il ritardo di trasmissione.
- Migliore scalabilità per alcuni algoritmi di programmazione delle code di trasmissione. Per esempio, ora **pfifo\_fast** è senza serratura.

(BZ#1562987)

### Nuovi strumenti per convertire iptables in nftables

Questo aggiornamento aggiunge gli **iptables-translate** e **ip6tables-translate** strumenti per convertire le regole esistenti **iptables** o **ip6tables** le regole in quelle equivalenti per **nftables**. Si noti che alcune estensioni non supportano la traduzione. Se tale estensione esiste, lo strumento stampa la regola non tradotta preceduta dal **#**segno. Per esempio:

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

Inoltre, gli utenti possono utilizzare gli strumenti **iptables-restore-translate** e **ip6tables-restore-translate** per tradurre un gran numero di regole. Si noti che prima di questo, gli utenti possono utilizzare i comandi **iptables-save** o **ip6tables-save** per stampare un dump delle regole correnti. Per esempio:

```
| % 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)

### Nuove funzionalità aggiunte alla VPN utilizzando NetworkManager

In Red Hat Enterprise Linux 8.0, **NetworkManager** fornisce le seguenti nuove funzionalità per la VPN:

- Supporto per il protocollo Internet Key Exchange versione 2 (IKEv2).
- Aggiunte alcune altre opzioni di **Libreswan**, come le **fragmentation** opzioni **rightid**, **leftcert**, **narrowing**, **rekey**. Per maggiori dettagli sulle opzioni supportate, vedere la pagina **nm-settings-libreswan** man.
- Aggiornati i codici predefiniti. Questo significa che quando l'utente non specifica i cifratori, il plugin **NetworkManager-libreswan** permette all'applicazione **Libreswan** di scegliere il cifrario predefinito del sistema. L'unica eccezione è quando l'utente seleziona una configurazione in modalità aggressiva IKEv1. In questo caso, i valori **ike = aes256-sha1;modp1536** e **eps = aes256-sha1** sono passati al **Libreswan**.

(BZ#1557035)

### Un nuovo tipo di dati chunk, I-DATA, aggiunto a SCTP

Questo aggiornamento aggiunge un nuovo tipo di data chunk, **I-DATA**, e programmatori di flusso al Stream Control Transmission Protocol (SCTP). In precedenza, SCTP inviava messaggi utente nello stesso ordine in cui erano stati inviati da un utente. Di conseguenza, un messaggio utente SCTP di

grandi dimensioni ha bloccato tutti gli altri messaggi in qualsiasi flusso fino al completo invio. Quando si usano i **I-DATA** pezzi, il campo Numero di sequenza di trasmissione (TSN) non è sovraccarico. Come risultato, SCTP ora può pianificare i flussi in diversi modi e **I-DATA** consente l'interscambio di messaggi utente (RFC 8260). Si noti che entrambi i coetanei devono supportare il tipo di **I-DATA** chunk.

(BZ#1273139)

## 4.14. LA SICUREZZA (TRADUZIONE AUTOMATICA)

### SCAP Security Guide Il profilo PCI-DSS si allinea con la versione 3.2.1

Il **SCAP Security Guide** progetto fornisce il profilo PCI-DSS (Payment Card Industry Data Security Standard) per Red Hat Enterprise Linux 8 ed è stato aggiornato per allinearli all'ultima versione PCI-DSS - 3.2.1.

(BZ#1618528)

### OpenSSH basato sulla versione 7.8p1

I **openssh** pacchetti sono stati aggiornati alla versione upstream 7.8p1. Notevoli cambiamenti includono:

- Rimosso il supporto per il **SSH version 1** protocollo.
- Rimosso il supporto per il codice di autenticazione dei **hmac-ripemd160** messaggi.
- Rimosso il supporto per i cifrari RC4 (**arcfour**).
- Rimosso il supporto per **Blowfish** i cifrari.
- Rimosso il supporto per **CAST** i cifrari.
- Modificato il valore predefinito dell'**UseDNS** opzione in **no**.
- Disabilita gli algoritmi a chiave **DSA** pubblica per impostazione predefinita.
- Cambiata la dimensione minima del modulo per i **Diffie-Hellman** parametri a 2048 bit.
- Semantica modificata dell'opzione di **ExposeAuthInfo** configurazione.
- L'**UsePrivilegeSeparation=sandbox** opzione è ora obbligatoria e non può essere disattivata.
- Impostare la dimensione minima della chiave accettata **RSA** a 1024 bit.

(BZ#1622511)

### RSA-PSS è ora supportato in OpenSC

Questo aggiornamento aggiunge il supporto per lo schema di firma crittografica RSA-PSS al driver della **OpenSC** smart card. Il nuovo schema consente un algoritmo crittografico sicuro necessario per il supporto TLS 1.3 nel software client.

(BZ#1595626)

### Notevoli cambiamenti nella rsyslog RHEL 8

I **rsyslog** pacchetti sono stati aggiornati alla versione upstream 8.37.0, che fornisce molte correzioni e miglioramenti rispetto alle versioni precedenti. I cambiamenti più importanti includono:

- Miglioramento dell'elaborazione dei messaggi interni del **rsyslog**; possibilità di limitarne la velocità; eliminazione di possibili blocchi.
- Limitazione del tasso di *spam* in generale; l'effettiva *fonte di spam* è ora registrata.
- Migliore gestione dei messaggi sovradimensionati - l'utente può ora impostare come trattarli sia nel nucleo che in alcuni moduli con azioni separate.
- **mmnormalize** possono ora essere incorporati nel **config**file invece di creare file separati per loro.
- L'utente può ora impostare la stringa di priorità di **GnuTLS imtcp** che consente un controllo a grana fine sulla crittografia.
- Tutte **config**le variabili, comprese le variabili in JSON, ora non sono più sensibili alle maiuscole e minuscole.
- Vari miglioramenti dell'output PostgreSQL.
- Aggiunta la possibilità di utilizzare variabili di shell per controllare **config**l'elaborazione, come il caricamento condizionale di file di configurazione aggiuntivi, l'esecuzione di istruzioni o l'inclusione di un testo in **config**. Si noti che un uso eccessivo di questa funzione può rendere molto difficile il debug dei problemi con **rsyslog**.
- Le modalità di creazione file a 4 cifre possono ora essere specificate in **config**.
- L'affidabile ingresso RELP (Event Logging Protocol) può ora legarsi anche solo ad un indirizzo specificato.
- Il valore predefinito dell'**enable.body**opzione dell'output di posta è ora allineato alla documentazione
- L'utente può ora specificare codici di errore di inserimento che devono essere ignorati nell'output **MongoDB**.
- L'ingresso Parallel TCP (pTCP) ha ora il backlog configurabile per un migliore bilanciamento del carico.

(BZ#1613880)

### Nuovo modulo **rsyslog**: **omkafka**

Per abilitare gli scenari di archiviazione centralizzata dei dati di **kafka**, è ora possibile inoltrare i log all'infrastruttura **kafka** utilizzando il nuovo **omkafka** modulo.

(BZ#1542497)

### **libssh** implementa SSH come componente crittografico di base

Questa modifica introduce **libssh** come componente crittografico di base in Red Hat Enterprise Linux 8. La **libssh** libreria implementa il protocollo Secure SHell (SSH).

Si noti che **libssh** non è conforme alla politica di crittografia a livello di sistema.

(BZ#1485241)

### Il supporto PKCS #11 per smart card e HSM è ora coerente in tutto il sistema

Con questo aggiornamento, l'utilizzo di smart card e moduli di sicurezza hardware (HSM) con interfaccia token crittografica PKCS #11 diventa coerente. Questo significa che l'utente e l'amministratore possono usare la stessa sintassi per tutti gli strumenti correlati nel sistema. Notevoli miglioramenti includono:

- Supporto per lo schema PKCS #11 Uniform Resource Identifier (URI) che garantisce un'abilitazione semplificata dei token sui server RHEL sia per gli amministratori che per gli application writer.
- Un metodo di registrazione a livello di sistema per le smart card e gli HSM che utilizzano il dominio **pkcs11.conf**.
- Il supporto costante per HSM e smart card è disponibile nelle applicazioni NSS, GnuTLS e OpenSSL (attraverso il **openssl-pkcs11** motore).
- Il server HTTP Apache (**httpd**) ora supporta perfettamente gli HSM.

Per ulteriori informazioni, vedere la pagina dell'**pkcs11.conf(5)**uomo.

(BZ#1516741)

### Le politiche crittografiche a livello di sistema sono applicate di default

Crypto-policies è un componente di Red Hat Enterprise Linux 8, che configura i sottosistemi crittografici di base, coprendo i protocolli TLS, IPSec, SSH, DNSSEC e Kerberos. Fornisce un piccolo insieme di criteri che l'amministratore può selezionare utilizzando il **update-crypto-policies** comando.

La politica di crittografia a livello di **DEFAULT** sistema offre impostazioni sicure per gli attuali modelli di minaccia. Permette i protocolli TLS 1.2 e 1.3, così come i protocolli IKEv2 e SSH2. Le chiavi RSA e i parametri Diffie-Hellman sono accettati se superiori a 2047 bit.

Vedi l'[Consistent security by crypto policies in Red Hat Enterprise Linux 8](#) articolo sul Blog di Red Hat e la pagina **update-crypto-policies(8)**uomo per maggiori informazioni.

(BZ#1591620)

### La guida alla sicurezza SCAP supporta OSPP 4.2

**SCAP Security Guide** fornisce una bozza del profilo OSPP (Protection Profile for General Purpose Operating Systems) versione 4.2 per Red Hat Enterprise Linux 8. Questo profilo riflette i controlli di configurazione obbligatori identificati nell'allegato di configurazione NIAP del profilo di protezione per sistemi operativi generici (profilo di protezione versione 4.2). SCAP Security Guide fornisce controlli automatici e script che consentono agli utenti di soddisfare i requisiti definiti nell'OSPP.

(BZ#1618518)

### Migliorata l'interfaccia a riga di comando OpenSCAP

La modalità verbose è ora disponibile in tutti **oscap** moduli e sottomoduli. L'output dello strumento ha migliorato la formattazione.

Le opzioni depresse sono state rimosse per migliorare l'usabilità dell'interfaccia a riga di comando.

Le seguenti opzioni non sono più disponibili:

- **--show oscap xccdf generate report** è stato completamente rimosso.
- **--probe-root oscap oval eval** è stato rimosso. Può essere sostituito impostando la variabile d'ambiente, **OSCAP\_PROBE\_ROOT**.

- **--sce-results** in **oscap xccdf eval** è stato sostituito da **--check-engine-results**
- **validate-xml** è stato eliminato dai moduli CPE, OVAL e XCCDF. I **validate** sottomoduli possono essere usati invece per validare il contenuto SCAP rispetto agli schemi XML e agli schemi XSD.
- **oscap oval list-probes** è stato rimosso, l'elenco delle sonde disponibili può essere visualizzato utilizzando **oscap --version** invece l'apposito comando.

OpenSCAP permette di valutare tutte le regole in un dato benchmark XCCDF indipendentemente dal profilo utilizzando **--profile '(all)'**.

(BZ#1618484)

## Supporto per un nuovo controllo dell'autorizzazione della mappa sulla chiamata di mmapped sistema

Il permesso SELinux **map** è stato aggiunto per controllare l'accesso mappato in memoria a file, directory, socket, e così via. Questo permette alla politica SELinux di impedire l'accesso diretto alla memoria ai vari oggetti del file system e garantire che ogni accesso sia riconvalidato.

(BZ#1592244)

## SELinux ora supporta systemd No New Privileges

Questo aggiornamento introduce la funzionalità di **nnp\_nosuid\_transition** policy che permette la transizione dei domini SELinux sotto (**NNPNo New Privileges**) o **nosuid** se **nnp\_nosuid\_transition** è consentito tra il vecchio e il nuovo contesto. I **selinux-policy** pacchetti ora contengono una policy per i servizi **systemd** che utilizzano la funzione di **NNP** sicurezza.

La seguente regola descrive come consentire questa capacità per un servizio:

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

Per esempio:

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

La politica di distribuzione ora contiene anche l'interfaccia macro **m4**, che può essere utilizzata nelle politiche di sicurezza SELinux per i servizi che utilizzano la **init\_nnp\_daemon\_domain()** funzione.

(BZ#1594111)

## SELinux ora supporta getrlimiti permessi nella process classe

Questo aggiornamento introduce un nuovo controllo di accesso SELinux, **process:getrlimit** che è stato aggiunto per la **prlimit()** funzione. Questo permette agli sviluppatori di politiche SELinux di controllare quando un processo tenta di leggere e poi modificare i limiti delle risorse di un altro processo usando il **process:setrlimit** permesso. Si noti che SELinux non impedisce a un processo di manipolare i limiti delle proprie risorse attraverso **prlimit()**. Per ulteriori informazioni, vedere la pagina **prlimit(2)** e **getrlimit(2)** la pagina degli uomini.

(BZ#1549772)

## Supporto TLS 1.3 nelle librerie crittografiche

Questo aggiornamento abilita di default Transport Layer Security (TLS) 1.3 in tutte le principali librerie di crittografia back-end. Ciò consente una bassa latenza attraverso il livello di comunicazione del sistema



operativo e migliora la privacy e la sicurezza delle applicazioni sfruttando i nuovi algoritmi, come RSA-PSS o X25519.

(BZ#1516728)

### Nuove caratteristiche in OpenSCAP RHEL 8

La **OpenSCAP** suite è stata aggiornata alla versione upstream 1.3.0, che introduce molti miglioramenti rispetto alle versioni precedenti. Le caratteristiche più importanti includono:

- API e ABI sono stati consolidati - i simboli aggiornati, deprecati e/o non utilizzati sono stati rimossi.
- Le sonde non vengono eseguite come processi indipendenti, ma come fili all'interno del **oscap** processo.
- L'interfaccia della riga di comando è stata aggiornata.
- **Python 2** le rilegature sono state sostituite con **Python 3** rilegature.

(BZ#1614273)

### Audit 3.0 sostituisce audispd con auditd

Con questo aggiornamento, la funzionalità di **audispd** è stata spostata in **auditd**. Di conseguenza, le opzioni di **audispd** configurazione sono ora parte di **auditd.conf**. Inoltre, la **plugins.d** directory è stata spostata in **/etc/audit**. Lo stato attuale dei plug-in **auditd** e dei relativi plug-in può ora essere controllato eseguendo il **service auditd state** comando.

(BZ#1616428)

### rsyslogimfile ora supporta i link simbolici

Con questo aggiornamento, il modulo **rsyslogimfile** offre migliori prestazioni e più opzioni di configurazione. Questo permette di utilizzare il modulo per i casi d'uso più complicati del monitoraggio dei file. Ad esempio, è ora possibile utilizzare monitor di file con modelli di globi in qualsiasi punto del percorso configurato e ruotare i target di symlink con una maggiore velocità di trasferimento dati.

(BZ#1614179)

### La generazione automatica OpenSSH delle chiavi del server è ora gestita da sshd-keygen@.service

**OpenSSH** crea automaticamente le chiavi host del server RSA, ECDSA e ED25519 se mancano. Per configurare la creazione della chiave host in RHEL 8, utilizzare il servizio **sshd-keygen@.service** istanziato.

Ad esempio, per disabilitare la creazione automatica del tipo di chiave RSA:

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

Vedere il **/etc/sysconfig/sshd** file per ulteriori informazioni.

(BZ#1228088)

### Il formato predefinito rsyslog del file di configurazione è ora non legacy

I file di configurazione dei **rsyslog** pacchetti ora usano di default il formato non legacy. Il formato legacy

può essere ancora usato, tuttavia, mescolando le istruzioni di configurazione correnti e quelle legacy ha diversi vincoli. Le configurazioni derivanti dalle precedenti versioni di RHEL devono essere riviste. Per ulteriori informazioni, vedere la pagina **rsyslog.conf(5)** dedicata ai dipendenti.

(BZ#1619645)

### Nuovi booleani SELinux

Questo aggiornamento della politica del sistema SELinux introduce le seguenti booleani:

- `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_pcsd`

Per maggiori dettagli, vedere l'output del seguente comando:

```
# semanage boolean -l
```

(JIRA:RHELPLAN-10347)

## 4.15. VIRTUALIZZAZIONE (TRADUZIONE AUTOMATICA)

### KVM supporta il paging a 5 livelli

Con Red Hat Enterprise Linux 8, la virtualizzazione KVM supporta la funzione di paging a 5 livelli, che aumenta significativamente lo spazio di indirizzi fisici e virtuali che i sistemi host e guest possono utilizzare.

(BZ#1485229)

### KVM supporta UMIP in RHEL 8

La virtualizzazione KVM supporta ora la funzione UMIP (User-Mode Instruction Prevention), che può aiutare a impedire alle applicazioni dello spazio utente di accedere alle impostazioni a livello di sistema. Questo riduce i potenziali vettori per gli attacchi di escalation dei privilegi, rendendo così più sicuri l'hypervisor KVM e le sue macchine ospiti.

(BZ#1494651)

### Ulteriori informazioni nei rapporti di crash degli ospiti KVM

Le informazioni sull'arresto anomalo che l'hypervisor KVM genera se un ospite termina inaspettatamente o non risponde sono state espanse. Questo rende più facile diagnosticare e risolvere i problemi nelle implementazioni della virtualizzazione KVM.

(BZ#1508139)

## qemu-kvm 2.12 in RHEL 8

Red Hat Enterprise Linux 8 è distribuito con **2qemu-kvm.12**. Questa versione corregge diversi bug e aggiunge una serie di miglioramenti rispetto alla versione 1.5.3, disponibile in Red Hat Enterprise Linux 7.

In particolare, sono state introdotte le seguenti caratteristiche:

- Q35 tipo di macchina ospite
- UEFI guest boot
- NUMA tuning e pinning nell'ospite
- vCPU hot plug e hot unplug
- filettatura I/O ospite

Si noti che alcune delle funzionalità disponibili in **2qemu-kvm.12** non sono supportate su Red Hat Enterprise Linux 8. Per informazioni dettagliate, vedere "Supporto delle funzioni e limitazioni nella virtualizzazione di RHEL 8" sul portale clienti di Red Hat.

(BZ#1559240)

## NVIDIA vGPU è ora compatibile con la console VNC

Quando si utilizza la funzione NVIDIA virtual GPU (vGPU), è ora possibile utilizzare la console VNC per visualizzare l'output visivo dell'ospite.

(BZ#1497911)

## Ceph è supportato dalla virtualizzazione

Con questo aggiornamento, lo storage Ceph è supportato dalla virtualizzazione KVM su tutte le architetture di CPU supportate da Red Hat.

(BZ#1578855)

## Il tipo di macchina Q35 è ora supportato dalla virtualizzazione

Red hat Enterprise Linux 8 introduce il supporto per **Q35**, un tipo di macchina basata su PCI Express più moderno. Ciò fornisce una varietà di miglioramenti nelle caratteristiche e nelle prestazioni dei dispositivi virtuali e garantisce che una gamma più ampia di dispositivi moderni sia compatibile con la virtualizzazione. Inoltre, le macchine virtuali create in Red Hat Enterprise Linux 8 sono impostate di default su **Q35**.

Si noti inoltre che il tipo di macchina **PC** predefinito in precedenza è diventato obsoleto e dovrebbe essere utilizzato solo quando si virtualizzano sistemi operativi più vecchi che non supportano Q35.

(BZ#1599777)

## Boot loader interattivo per macchine virtuali su IBM Z

Quando si avvia una macchina virtuale (VM) su un host IBM Z, il firmware del boot loader QEMU presenta ora un'interfaccia di console interattiva. In questo modo è possibile risolvere i problemi di avvio del sistema operativo guest senza accesso all'ambiente host.

(BZ#1508137)

## Aggiunto QEMU sandboxing

In Red Hat Enterprise Linux 8, l'emulatore QEMU introduce la funzione sandboxing. Il sandboxing di QEMU fornisce limitazioni configurabili a ciò che il sistema chiama QEMU può eseguire, rendendo così le macchine virtuali più sicure. Si noti che questa funzione è abilitata e configurata per impostazione predefinita.

(JIRA:RHELPLAN-10628)

### **Set di istruzioni GFNI e CLDEMOT abilitati per Intel Xeon SnowRidge**

Le macchine virtuali (VM) in esecuzione in un host RHEL 8 su un sistema Intel Xeon SnowRidge sono ora in grado di utilizzare i set di istruzioni GFNI e CLDEMOT. Ciò può aumentare significativamente le prestazioni di tali macchine virtuali in alcuni scenari.

(BZ#1494705)

### **È stato aggiunto un driver a blocchi basato su VFIO per i dispositivi NVMe**

L'emulatore QEMU introduce un driver basato sulla funzione virtuale I/O (VFIO) per dispositivi NVMe (NVMe) a memoria non volatile. Il driver comunica direttamente con i dispositivi NVMe collegati alle macchine virtuali (VM) ed evita di utilizzare il livello di sistema del kernel e i suoi driver NVMe. Come risultato, questo migliora le prestazioni dei dispositivi NVMe nelle macchine virtuali.

(BZ#1519004)

### **Miglioramento del supporto per le pagine enormi**

Quando si utilizza RHEL 8 come host di virtualizzazione, gli utenti possono modificare le dimensioni delle pagine della memoria posteriore di una macchina virtuale (VM) a qualsiasi dimensione supportata dalla CPU. Questo può migliorare significativamente le prestazioni della macchina virtuale.

Per configurare le dimensioni delle pagine di memoria della macchina virtuale, modificare la configurazione XML della macchina virtuale e aggiungere l'elemento <hugepages> alla sezione <memoryBacking>.

(JIRA:RHELPLAN-14607)

## **4.16. SUPPORTABILITÀ (TRADUZIONE AUTOMATICA)**

### **sosreport può segnalare programmi e mappe basati su eBPF**

Lo strumento **sosreport** è stato migliorato per segnalare qualsiasi programma di Berkeley Packet Filtering (eBPF) esteso caricato e mappe in Red Hat Enterprise Linux 8.

(BZ#1559836)

## CAPITOLO 5. ANTEPRIME TECNOLOGICHE (TRADUZIONE AUTOMATICA)

Questa parte fornisce un elenco di tutte le Antepreme Tecnologiche disponibili in Red Hat Enterprise Linux 8.0.

Per informazioni sull'ambito di supporto di Red Hat per le funzioni di Anteprema tecnologia, vedere [Technology Preview Features Support Scope](#).

### 5.1. IL NOCCIOLO (TRADUZIONE AUTOMATICA)

#### XDP disponibile in anteprema tecnologica

La funzione **eXpress Data Path (XDP)**, disponibile come Technology Preview, fornisce un mezzo per caricare nel kernel i programmi Berkeley Packet Filter (BPF) per l'elaborazione di pacchetti ad alte prestazioni, rendendo programmabile il percorso dati del kernel.

(BZ#1503672)

#### eBPF disponibile in anteprema tecnologica

La funzione **estesa di Berkeley Packet Filtering (eBPF)** è disponibile come Technology Preview sia per il networking che per il tracing. **eBPF** consente allo spazio utente di collegare programmi personalizzati su una varietà di punti (prese, punti di traccia, ricezione dei pacchetti) per ricevere ed elaborare i dati. La caratteristica include una nuova chiamata **bpf()** di sistema, che permette di creare vari tipi di mappe, e anche di inserire vari tipi di programmi nel kernel. Vedere la **bpf(2)** pagina man per ulteriori informazioni.

(BZ#1559616)

#### BCC è disponibile come Anteprema di tecnologia

**BPF Compiler Collection (BCC)** è un kit di strumenti per creare efficienti programmi di kernel tracing e manipolazione che è disponibile come Technology Preview in Red Hat Enterprise Linux 8. **BCC** fornisce strumenti per l'analisi degli I/O, il networking e il monitoraggio dei sistemi operativi Linux utilizzando il **extended Berkeley Packet Filtering (eBPF)**.

(BZ#1548302)

#### Control Group v2 disponibile come Technology Preview in RHEL 8

**Gruppo di controllo** del meccanismo **v2** è un gruppo di controllo gerarchico unificato. **Control Group v2** organizza i processi gerarchicamente e distribuisce le risorse di sistema lungo la gerarchia in modo controllato e configurabile.

A differenza della versione precedente, **Control Group v2** ha un'unica gerarchia. Questa singola gerarchia permette al kernel Linux di:

- Categorizzare i processi in base al ruolo del loro proprietario.
- Eliminare i problemi con politiche conflittuali di gerarchie multiple.

**Control Group v2** supporta numerosi controllori:

- Il controller della CPU regola la distribuzione dei cicli della CPU. Questo controllore implementa:
  - Modelli di limiti di peso e larghezza di banda assoluta per la normale politica di programmazione.

- Modello assoluto di allocazione della larghezza di banda per la politica di pianificazione in tempo reale.
- Il controller di memoria regola la distribuzione della memoria. Attualmente, vengono tracciati i seguenti tipi di utilizzo della memoria:
  - Memoria Userland - cache delle pagine e memoria anonima.
  - Strutture di dati del kernel come dentries e inodes.
  - Tamponi per socket TCP.
- Il controllore I/O regola la distribuzione delle risorse I/O.
- Il controllore di ripristino interagisce sia con i controllori di memoria che con i controllori di I/O ed è specifico per il **gruppo di controllo v2**

Le informazioni di cui sopra si basano sul link: <https://www.kernel.org/doc/Documentation/cgroup-v2.txt>. È possibile fare riferimento allo stesso link per ottenere maggiori informazioni su particolari controllori **Control Group v2**.

(BZ#1401552)

### **early kdump disponibile come Technology Preview in Red Hat Enterprise Linux 8**

La **early kdump** caratteristica permette al crash kernel e initramfs di caricarsi abbastanza presto per catturare le **vmcore** informazioni anche per i primi crash. Per maggiori dettagli su **early kdump**, vedere il **/usr/share/doc/kexec-tools/early-kdump-howto.txt** file.

(BZ#1520209)

## **5.2. SISTEMI DI FILE E ARCHIVIAZIONE (TRADUZIONE AUTOMATICA)**

### **VDO disponibile come tipo di volume logico LVM**

LVM può ora essere utilizzato per creare volumi logici di tipo *Virtual Data Optimizer* (VDO). VDO è un dispositivo virtuale a blocchi con la possibilità di comprimere e deduplicare i dati.

Questa è una funzione di Anteprema tecnologia.

(BZ#1643553)

### **Supporto per l'integrità dei dati Campo/estensione dell'integrità dei dati (DIF/DIX)**

DIF/DIX è un'aggiunta allo standard SCSI. Rimane in Technology Preview per tutti gli HBA e gli array di storage, ad eccezione di quelli specificamente elencati come supportati.

DIF/DIX aumenta la dimensione del blocco disco da 512 byte comunemente usato da 512 a 520 byte, aggiungendo il Data Integrity Field (DIF). Il DIF memorizza un valore di checksum per il blocco dati che viene calcolato dall'Host Bus Adapter (HBA) quando si verifica una scrittura. Il dispositivo di memorizzazione conferma la somma di controllo al ricevimento e memorizza sia i dati che la somma di controllo. Al contrario, quando si verifica una lettura, la somma di controllo può essere verificata dal dispositivo di memorizzazione e dall'HBA ricevente.

(BZ#1649493)

### **NVMe/FC è disponibile come Technology Preview in adattatori Qlogic usando qla2xxx**

Il tipo di trasporto NVMe over Fibre Channel (NVMe/FC) è disponibile come Technology Preview in adattatori Qlogic utilizzando il **qla2xxx** driver.

(BZ#1649922)

### 5.3. ELEVATA DISPONIBILITÀ E CLUSTER (TRADUZIONE AUTOMATICA)

#### podman Fasci di pacemaker disponibili in anteprima tecnologica

I bundle container di Pacemaker ora vengono eseguiti sulla piattaforma **podman** container, con la funzione container bundle disponibile come Technology Preview. C'è un'eccezione a questa caratteristica che è Technology Preview: Red Hat supporta completamente l'uso dei bundle Pacemaker per Red Hat Openstack.

(BZ#1619620)

### 5.4. LA SICUREZZA (TRADUZIONE AUTOMATICA)

#### Tag SWID della release di RHEL 8.0

Per consentire l'identificazione delle installazioni di RHEL 8.0 utilizzando il meccanismo ISO/IEC 19770-2:2015, i tag di identificazione software (SWID) sono installati in file **/usr/lib/swidtag/redhat.com/com.redhat.RHEL-8-<architecture>.swidtag** e **/usr/lib/swidtag/redhat.com/com.redhat.RHEL-8.0-<architecture>.swidtag**. La directory padre di questi tag può essere trovata anche seguendo il link **/etc/swid/swidtags.d/redhat.com** simbolico.

La firma XML dei file tag SWID può essere verificata ad esempio con il **xmlsec1 verify** comando:

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

Il certificato dell'autorità di certificazione della firma del codice può essere ottenuto anche dalla [Product Signing Keys](#) pagina del Portale clienti.

(BZ#1636338)

### 5.5. VIRTUALIZZAZIONE (TRADUZIONE AUTOMATICA)

#### AMD SEV per macchine virtuali KVM

Come Technology Preview, RHEL 8 introduce la funzione SEV (Secure Encrypted Virtualization) per le macchine host AMD EPYC che utilizzano l'hypervisor KVM. Se abilitato su una macchina virtuale (VM), SEV crittografa la memoria della macchina virtuale in modo che l'host non possa accedere ai dati della macchina virtuale. Questo aumenta la sicurezza della macchina virtuale se l'host è stato infettato con successo da malware.

Si noti che il numero di macchine virtuali che possono utilizzare questa funzione alla volta su un singolo host è determinato dall'hardware dell'host. Gli attuali processori AMD EPYC supportano 15 o meno VM in esecuzione utilizzando SEV.

(BZ#1501618, BZ#1501607)

La virtualizzazione nidificata ora disponibile su IBM POWER 9

Come Technology Preview, è ora possibile utilizzare le funzionalità di virtualizzazione nidificate sulle macchine host RHEL 8 che girano su sistemi IBM POWER 9. La virtualizzazione annidata consente alle macchine virtuali KVM (VM) di agire come hypervisor, il che consente di eseguire le VM all'interno delle VM.

Si noti che per far funzionare la virtualizzazione nidificata su IBM POWER 9, l'host, l'ospite e gli ospiti nidificati attualmente hanno tutti bisogno di eseguire uno dei seguenti sistemi operativi:

- RHEL 8
- RHEL 7 per POTENZA 9

(BZ#1505999)



## CAPITOLO 6. FUNZIONALITÀ DEPRESSE (TRADUZIONE AUTOMATICA)

Questa parte fornisce una panoramica delle funzionalità che sono state deprecate in Red Hat Enterprise Linux 8.0.

Le funzionalità deprese continuano ad essere supportate fino alla fine del ciclo di vita di Red Hat Enterprise Linux 8. Le funzionalità deprese non saranno probabilmente supportate nelle future release principali di questo prodotto e non è raccomandata per le nuove distribuzioni. Per l'elenco più recente delle funzionalità deprecate all'interno di una particolare release maggiore, fare riferimento all'ultima versione della documentazione di rilascio.

I componenti hardware deprecati non sono raccomandati per le nuove distribuzioni sulle principali versioni attuali o future. Gli aggiornamenti dei driver hardware sono limitati alla sicurezza e alle sole correzioni critiche. Red Hat raccomanda di sostituire questo hardware non appena ragionevolmente possibile.

Un pacchetto può essere deprecato e non raccomandato per un ulteriore utilizzo. In determinate circostanze, un imballaggio può essere rimosso da un prodotto. La documentazione del prodotto identifica quindi i pacchetti più recenti che offrono funzionalità simili, identiche o più avanzate a quella deprecata, e fornisce ulteriori raccomandazioni.

### 6.1. INSTALLAZIONE E CREAZIONE DI IMMAGINI (TRADUZIONE AUTOMATICA)

#### L'`--interactive` opzione del comando `ignorediskKickstart` è stata deprecata

L'utilizzo delle `--interactive option` future versioni di Red Hat Enterprise Linux si tradurrà in un errore fatale di installazione. Si raccomanda di modificare il file Kickstart per rimuovere l'opzione.

(BZ#1637872)

### 6.2. SISTEMI DI FILE E ARCHIVIAZIONE (TRADUZIONE AUTOMATICA)

#### NFSv3 su UDP è stato disattivato

Per impostazione predefinita, il server NFS non si apre o ascolta più su una presa UDP (User Datagram Protocol). Questa modifica riguarda solo la versione 3 di NFS perché la versione 4 richiede il Transmission Control Protocol (TCP).

NFS su UDP non è più supportato in RHEL 8.

(BZ#1592011)

#### La modalità `target NVMe/FC` è deprecata

La modalità di destinazione del protocollo di trasporto Non Volatile Memory Express over Fibre Channel (NVMe/FC) era precedentemente disponibile come Technology Preview in RHEL 7. In RHEL 8, la modalità `target NVMe/FC` è deprecata.

Abilitando le porte dell'adattatore bus host (HBA) FC in modalità NVMe target mode si ottiene il seguente messaggio di errore:

**Warning: NVMe over FC Target - This driver has not undergone sufficient testing by Red Hat for this release and therefore cannot be used in production systems.**

(BZ#1664838)

## 6.3. LA RETE (TRADUZIONE AUTOMATICA)

### Gli script di rete sono deprecati in RHEL 8

Gli script di rete sono deprecati in Red Hat Enterprise Linux 8 e non sono più forniti di default. L'installazione di base fornisce una nuova versione degli script **ifup** **ifdown** degli script che chiamano il servizio **NetworkManager** attraverso lo strumento **nmcli**. In Red Hat Enterprise Linux 8, per eseguire gli **ifdown** script **ifup** gli script, **NetworkManager** deve essere in esecuzione.

Si noti che i comandi personalizzati in **/sbin/ifup-local**, **ifdown-pre-local** e **ifdown-local** gli script non vengono eseguiti.

Se uno qualsiasi di questi script è richiesto, l'installazione degli script di rete deprecati nel sistema è ancora possibile con il seguente comando:

```
~]# yum install network-scripts
```

Gli script **ifup** **ifdown** gli script si collegano agli script di rete legacy installati.

Il richiamo degli script di rete legacy mostra un avviso sulla loro deprecazione.

(BZ#1647725)

## 6.4. LA SICUREZZA (TRADUZIONE AUTOMATICA)

### DSA è deprecato in Red Hat Enterprise Linux 8

L'algoritmo di firma digitale (DSA) è considerato deprecato in Red Hat Enterprise Linux 8. I meccanismi di autenticazione che dipendono dalle chiavi DSA non funzionano nella configurazione predefinita. Si noti che i **OpenSSH** clienti non accettano le chiavi host DSA nemmeno nella politica LEGACY.

(BZ#1646541)

### SSL2Client Hello è stato deprecato in NSS

Il protocollo Transport Layer Security (TLS) versione 1.2 e precedenti consente di avviare una trattativa con un **Client Hello** messaggio formattato in modo retrocompatibile con il protocollo Secure Sockets Layer Layer (SSL) versione 2. Il supporto per questa funzione della libreria Network Security Services (NSS) è stato deprecato ed è disabilitato per impostazione predefinita.

Le applicazioni che richiedono il supporto per questa funzione devono utilizzare la nuova **SSL\_ENABLE\_V2\_COMPATIBLE\_HELLO** API per attivarla. Il supporto per questa funzione potrebbe essere completamente rimosso nelle future versioni di Red Hat Enterprise Linux 8.

(BZ#1645153)

## 6.5. VIRTUALIZZAZIONE (TRADUZIONE AUTOMATICA)

### Le istantanee delle macchine virtuali non sono supportate correttamente in RHEL 8

L'attuale meccanismo di creazione di istantanee di macchine virtuali (VM) è stato deprecato, in quanto non funziona in modo affidabile. Di conseguenza, si raccomanda di non utilizzare le istantanee VM in RHEL 8.

Si noti che è in fase di sviluppo un nuovo meccanismo di VM snapshot e sarà pienamente implementato in una futura versione minore di RHEL 8.

([BZ#1686057](#))

### **Il tipo di GPU virtuale Cirrus VGA è stato deprecato**

Con un futuro importante aggiornamento di Red Hat Enterprise Linux, il dispositivo GPU **Cirrus VGA** non sarà più supportato nelle macchine virtuali KVM. Red Hat raccomanda pertanto di utilizzare i dispositivi **stdvga**, **virtio-vga** o **qxl** al posto di Cirrus VGA.

(BZ#1651994)

### **virt-manager è stato deprezzato**

L'applicazione Virtual Machine Manager, nota anche come **virt-manager**, è stata deprecata. La console web RHEL 8, nota anche come **Cockpit**, è destinata a diventare la sua sostituzione in una versione successiva. Si raccomanda pertanto di utilizzare la console web per gestire la virtualizzazione in una GUI. Tuttavia, in Red Hat Enterprise Linux 8.0, alcune funzioni possono essere accessibili solo da **virt-manager** o dalla riga di comando.

(JIRA:RHELPLAN-10304)

## CAPITOLO 7. PROBLEMI NOTI (TRADUZIONE AUTOMATICA)

Questa parte descrive problemi noti in Red Hat Enterprise Linux 8.

### 7.1. INSTALLAZIONE E CREAZIONE DI IMMAGINI (TRADUZIONE AUTOMATICA)

#### I comandi `authselect` e `authconfig` Kickstart richiedono il repository AppStream

Il pacchetto `authselect-compat` è richiesto dai comandi `authselect` e `authconfig` Kickstart durante l'installazione. Senza questo pacchetto, l'installazione fallisce se `authselect` e `authconfig` sono utilizzati. Tuttavia, per progettazione, il pacchetto `authselect-compat` è disponibile solo nel repository AppStream.

Per risolvere questo problema, verificare che i repository BaseOS e AppStream siano disponibili per l'installatore o usare il comando `authselect` Kickstart durante l'installazione.

(BZ#1640697)

#### Copiare il contenuto del file immagine di RHEL 8.0 Binary DVD.iso in una partizione omette i file `.treeinfo` e `.discinfo`

Quando si copia il contenuto del file immagine di RHEL 8.0 Binary DVD.iso in una partizione per un'installazione locale, il comando `cp` con `*` nel comando `cp <path>/^* <mounted partition>/dir` non riesce a copiare i file `.treeinfo` e i file `.discinfo` che sono necessari per un'installazione riuscita. Come risultato, i repository BaseOS e AppStream non vengono caricati, e un messaggio di log di debug nel file `anaconda.log` è l'unico record del problema.

Per risolvere il problema, copiare i file mancanti `.treeinfo` e `.discinfo` nella partizione.

(BZ#1692746)

### 7.2. IL NOCCIOLO (TRADUZIONE AUTOMATICA)

#### Il modulo `i40iw` non si carica automaticamente al boot

A causa di molte schede di rete i40e che non supportano iWarp e del modulo `i40iw` che non supporta completamente `suspend/resume`, questo modulo non viene caricato automaticamente di default per garantire che `suspend/resume` funzioni correttamente. Per risolvere questo problema, modificare manualmente il file `/lib/udev/rules.d/90-rdma-hw-modules.rules` per abilitare il caricamento automatico di `i40iw`.

Si noti inoltre che se c'è un altro dispositivo RDMA installato con un dispositivo i40e sulla stessa macchina, il dispositivo RDMA non i40e attiva il servizio `rdma`, che carica tutti i moduli stack RDMA abilitati, compreso il modulo `i40iw`.

(BZ#1623712)

#### Il sistema a volte diventa insensibile quando sono collegati molti dispositivi

Quando Red Hat Enterprise Linux 8 configura un gran numero di dispositivi, un gran numero di messaggi della console di sistema. Questo accade, ad esempio, quando vi è un gran numero di unità logiche (LUN), con percorsi multipli per ogni LUN. La marea di messaggi della console, oltre ad altri lavori che il kernel sta facendo, potrebbe causare al cane da guardia del kernel di forzare il panico del kernel perché il kernel sembra essere appeso.

Poiché la scansione avviene all'inizio del ciclo di avvio, il sistema non risponde quando sono collegati molti dispositivi. Questo di solito si verifica al momento dell'avvio.

Se **kdump** è abilitato sul computer durante l'evento di scansione del dispositivo dopo l'avvio, il blocco dell'hard lockup comporta la cattura di un'**vmcore**immagine.

Per risolvere questo problema, aumentare il timer di blocco del cane da guardia. Per farlo, aggiungere l'**watchdog\_thresh=N**opzione alla riga di comando del kernel. Sostituire **N** con il numero di secondi:

- Se si dispone di meno di mille dispositivi, utilizzare **30**.
- Se si dispone di più di mille dispositivi, utilizzare **60**.

Per la memorizzazione, il numero di dispositivi è il numero di percorsi di tutti i LUN: generalmente, il numero di **/dev/sd\***dispositivi.

Dopo l'applicazione del workaround, il sistema non diventa più insensibile quando si configura un gran numero di dispositivi.

(BZ#1598448)

### Il KSM a volte ignora le polivie di memoria della NUMA

Quando la funzione di memoria condivisa del kernel (KSM) è abilitata con il parametro "merge\_across\_nodes=1", KSM ignora i criteri di memoria impostati dalla funzione mbind() e può unire pagine da alcune aree di memoria a nodi Non-Uniform Memory Access (NUMA) che non corrispondono ai criteri.

Per risolvere questo problema, disabilita KSM o imposta il parametro merge\_across\_nodes a "0" se si usa la memoria NUMA che si lega con QEMU. Come risultato, i criteri di memoria NUMA configurati per la macchina virtuale KVM funzioneranno come previsto.

(BZ#1153521)

## 7.3. GESTIONE DEL SOFTWARE (TRADUZIONE AUTOMATICA)

### L'esecuzione yum list sotto un utente non root causa l'arresto anomalo di YUM

Quando si esegue il **yum list** comando sotto un utente non root dopo che il **libdnf** pacchetto è stato aggiornato, **YUM** può terminare inaspettatamente. Se si colpisce questo bug, eseguire **yum list** sotto root per risolvere il problema. Come risultato, i successivi tentativi di esecuzione **yum list** sotto un utente non root non causano più l'arresto anomalo di **YUM**.

(BZ#1642458)

### La pagina dell'yum(8) uomo menziona erroneamente il yum module profile comando

La pagina del **yum(8)** manuale afferma erroneamente che lo strumento di gestione dei pacchetti **YUM** include il **yum module profile** comando per fornire dettagli sui profili dei moduli. Tuttavia, questo comando non è più disponibile e quando viene utilizzato, **YUM** visualizza un messaggio di errore su un comando non valido. Per i dettagli sui profili dei moduli, utilizzare invece il nuovo **yum module info --profile** comando.

(BZ#1622580)

### yum-plugin-aliases attualmente non disponibile

Il **yum-plugin-aliases** pacchetto, che fornisce il **alias** comando per aggiungere alias yum personalizzati, non è attualmente disponibile. Di conseguenza, attualmente non è possibile utilizzare gli alias.

(BZ#1647760)

### **yum-plugin-changelog** attualmente non disponibile

Il **yum-plugin-changelog** pacchetto, che consente di visualizzare i log delle modifiche ai pacchetti prima e dopo l'aggiornamento dei pacchetti, non è attualmente disponibile.

(BZ#1581191)

## 7.4. SERVIZI INFRASTRUTTURALI (TRADUZIONE AUTOMATICA)

### Tuned non imposta i parametri della riga di comando di avvio del kernel

Lo strumento **Tuned** non supporta la specifica BLS (Boot Loader Specification), che è abilitata per impostazione predefinita. Di conseguenza, **Tuned** non imposta alcuni parametri della riga di comando di avvio del kernel, il che causa alcuni problemi, come la riduzione delle prestazioni o il fatto che i core della CPU non siano isolati. Per risolvere questo problema, disattivare il BLS e riavviare **Tuned**.

1. Installare il pacchetto *sporco*.
2. Rimuovere la seguente riga dal **/etc/default/grub** file:

```
GRUB_ENABLE_BLSCFG=true
```

3. Ri-generare il **grub2.cfg** file eseguendolo per sistemi non-EFI:

```
grub2-mkconfig -o /etc/grub2.cfg
```

o per i sistemi EFI:

```
grub2-mkconfig -o /etc/grub2-efi.cfg
```

4. Riavviare Riavvia **sintonizzato** eseguendo:

```
systemctl restart tuned
```

Come risultato, **Tuned** imposta i parametri di avvio del kernel come previsto.

(BZ#1576435)

## 7.5. CONCHIGLIE E STRUMENTI A RIGA DI COMANDO (TRADUZIONE AUTOMATICA)

### Python la rilegatura del **net-snmp** pacchetto non è disponibile

La **Net-SNMP** suite di strumenti non è vincolante **Python 3**, che è l'implementazione predefinita **Python** in RHEL 8. Di conseguenza, **python-net-snmp**, **python2-net-snmp**, o **python3-net-snmp** pacchetti non sono disponibili in RHEL 8.

(BZ#1584510)

## 7.6. LINGUAGGI DI PROGRAMMAZIONE DINAMICI, SERVER WEB E DATABASE (TRADUZIONE AUTOMATICA)

### I server di database non sono installabili in parallelo

I moduli **mariadb** e **mysql** i moduli non possono essere installati in parallelo in RHEL 8.0 a causa di pacchetti RPM contrastanti.

Per progettazione, è impossibile installare più di una versione (stream) dello stesso modulo in parallelo. Per esempio, è necessario scegliere solo uno dei flussi disponibili dal **postgresql** modulo, o (predefinito) **10** o **9.6**. L'installazione parallela dei componenti è possibile nelle Red Hat Software Collections per RHEL 6 e RHEL 7. In RHEL 8, diverse versioni di server di database possono essere utilizzate in container.

(BZ#1566048)

### Problemi nella `mod_cgid` registrazione

Se il modulo **mod\_cgid** Apache httpd è usato sotto un modulo threaded multi-processing (MPM), che è la situazione di default in RHEL 8, si verificano i seguenti problemi di logging:

- L'**stderr** output dello script CGI non è preceduto da informazioni standard di timestamp.
- L'**stderr** output dello script CGI non è correttamente reindirizzato ad un file di log specifico per il **VirtualHost**, se configurato.

(BZ#1633224)

### Il modulo `IO::Socket::SSL` Perl non supporta TLS 1.3

Nuove funzionalità del protocollo TLS 1.3, come la ripresa della sessione o l'autenticazione post-handshake, sono state implementate nella libreria RHEL 8 **OpenSSL** ma non nel modulo **Net::SSL** e **Leay** Perl, e quindi non sono disponibili nel modulo **IO::Socket::SSL** Perl. Di conseguenza, l'autenticazione del certificato client potrebbe non riuscire e il ripristino delle sessioni potrebbe essere più lento rispetto al protocollo TLS 1.2.

Per ovviare a questo problema, disabilitare l'utilizzo di TLS 1.3 impostando l'**SSL\_version** opzione sul **!TLSv1\_3** valore quando si crea un **IO::Socket::SSL** oggetto.

(BZ#1632600)

## 7.7. GESTIONE DELL'IDENTITÀ (TRADUZIONE AUTOMATICA)

### La cache delle credenziali KCM non è adatta per un gran numero di credenziali in una singola cache delle credenziali

Se la cache delle credenziali contiene troppe credenziali, le operazioni di Kerberos, come **kinit**, falliscono a causa di un limite hardcoded sul buffer utilizzato per trasferire dati tra il componente **sssd-kcm** e il database sottostante.

Per risolvere questo problema, aggiungere l'**ccache\_storage = memory** opzione nella sezione **kcm** del `/etc/sss/sss.conf` file. Questo istruisce il responder **kcm** per memorizzare solo le credenziali cache in memoria, non in modo persistente. In questo caso, riavviare il sistema o **sssd-kcm** cancella le cache delle credenziali. Si noti che KCM può gestire cache di dimensioni fino a 64 kB.

(BZ#1448094)

## Valori di timeout contrastanti impediscono a SSSD di connettersi ai server

Alcuni dei valori di timeout predefiniti relativi alle operazioni di failover utilizzati dal demone System Security Services Daemon (SSSD) sono in conflitto. Di conseguenza, il valore di timeout riservato a SSSD per parlare con un singolo server impedisce a SSSD di provare altri server prima dell'operazione di connessione. Per risolvere il problema, impostare il valore del parametro `ldap_opt_timeout` superiore al valore del `dns_resolver_timeout` parametro e il valore del `dns_resolver_timeout` parametro superiore al valore del `dns_resolver_op_timeout` parametro.

(BZ#1382750)

## L'utilizzo di una smart card per accedere all'IdM web UI non funziona

Quando un utente tenta di accedere all'interfaccia web di Identity Management (IdM) utilizzando un certificato memorizzato sulla propria smart card, il codice di interfaccia D-Bus di System Security Services Daemon (SSSD) utilizza un richiamo errato per cercare l'utente. Di conseguenza, la ricerca va in crash. Per risolvere il problema, utilizzare altri metodi di autenticazione.

(BZ#1642508)

## Il server IdM non funziona in FIPS

A causa di un'implementazione incompleta del connettore SSL per Tomcat, un server IdM con un server di certificazione installato non funziona su macchine con la modalità FIPS abilitata.

(BZ#1673296)

## Il nuxwdog servizio fallisce in ambienti HSM e richiede l'installazione del keyutilspacchetto in ambienti non HSM

Il servizio `nuxwdog` watchdog è stato integrato nel sistema di certificazione. Di conseguenza, non `nuxwdog` è più fornito come pacchetto separato. Per utilizzare il servizio watchdog, installare il `pki-server` pacchetto.

Si noti che il `nuxwdog` servizio ha i seguenti problemi noti:

- Il `nuxwdog` servizio non funziona se si utilizza un modulo di archiviazione hardware (HSM). Per questo numero non sono disponibili soluzioni alternative.
- In un ambiente non HSM, Red Hat Enterprise Linux 8.0 non installa automaticamente il `keyutils` pacchetto come dipendenza. Per installare manualmente il pacchetto, utilizzare il `dnf install keyutils` comando.

(BZ#1652269)

## 7.8. COMPILATORI E STRUMENTI DI SVILUPPO (TRADUZIONE AUTOMATICA)

### Funzioni sintetiche generate da GCC Confuse SystemTap

L'ottimizzazione GCC può generare funzioni sintetiche per copie parzialmente in linea di altre funzioni. Strumenti come SystemTap e GDB non possono distinguere queste funzioni sintetiche da quelle reali. Di conseguenza, SystemTap può posizionare le sonde sia sui punti di ingresso di funzioni sintetiche che reali, e quindi registrare più colpi di sonda per una singola chiamata di funzione reale.



Per ovviare a questo problema, gli script SystemTap devono essere adattati con misure quali il rilevamento della ricorsione e la soppressione delle sonde relative a funzioni parziali in linea. Per esempio, uno script

```
probe kernel.function("can_nice").call { }
```

può cercare di evitare il problema descritto di seguito:

```
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()] --;
}
```

Si noti che questo script di esempio non tiene conto di tutti gli scenari possibili, come i kprobes mancanti o kretprobes, o la vera e propria ricorsione prevista.

(BZ#1169184)

### Lo ltracestrumento non segnala le chiamate di funzione

A causa dei miglioramenti all'indurimento binario applicato a tutti i componenti RHEL, lo **ltrace**strumento non è più in grado di rilevare le chiamate di funzione nei file binari provenienti dai componenti RHEL. Di conseguenza, l'**ltrace**output è vuoto perché non segnala alcuna chiamata rilevata quando viene utilizzato su tali file binari. Al momento non esiste una soluzione alternativa.

Come nota, **ltrace**può segnalare correttamente le chiamate in file binari personalizzati costruiti senza i rispettivi flag di tempra.

(BZ#1618748, BZ#1655368)

## 7.9. SISTEMI DI FILE E ARCHIVIAZIONE (TRADUZIONE AUTOMATICA)

### Impossibile scoprire un target iSCSI utilizzando il **iscsiuio**pacchetto

Red Hat Enterprise Linux 8 non consente l'accesso simultaneo alle aree di registro PCI. Di conseguenza, è stato impostato un **could not set host net params (err 29)**errore e la connessione al portale di ricerca non è riuscita. Per risolvere questo problema, impostare il parametro **iomem=relaxed**del kernel nella riga di comando del kernel per l'iSCSI offload. Si tratta in particolare di eventuali scarichi effettuati con l'**bnx2i**autista. Di conseguenza, il collegamento al portale di ricerca ha avuto successo e il **iscsiuio**pacchetto ora funziona correttamente.

(BZ#1626629)

### L'opzione di montaggio di XFS DAX è incompatibile con gli estensioni di dati condivisi **copy-on-write**

Un file system XFS formattato con la funzione di copia in scrittura dei dati condivisa non è compatibile con l'opzione di **-o dax**montaggio. Di conseguenza, il montaggio di un sistema di file di questo tipo con dei **-o dax**fallimenti.

Per risolvere il problema, formatta il file system con l'opzione **reflink=0** metadati per disabilitare la copia in scrittura dei dati condivisi:

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

Di conseguenza, il montaggio del **-o dax** file system è riuscito.

(BZ#1620330)

## 7.10. LA RETE (TRADUZIONE AUTOMATICA)

### nftables non supporta tipi di set IP multidimensionali

Il framework di filtraggio dei **nftables** pacchetti non supporta tipi di set con concatenazioni e intervalli. Di conseguenza, non è possibile utilizzare tipi di set IP multidimensionali, come ad esempio **hash:net,port**, con **nftables**.

Per risolvere questo problema, utilizzare il **iptables** framework con lo **ipset** strumento se si richiedono tipi di set IP multidimensionali.

(BZ#1593711)

## 7.11. LA SICUREZZA (TRADUZIONE AUTOMATICA)

### OpenSCAPrpmverifypackage non funziona correttamente

Le chiamate di sistema **chdir** **chroot** le chiamate di sistema vengono chiamate due volte dalla **rpmverifypackage** sonda. Di conseguenza, si verifica un errore quando la sonda viene utilizzata durante una scansione **OpenSCAP** con contenuto personalizzato Open Vulnerability and Assessment Language (OVAL).

Per risolvere questo problema, non utilizzare il test **rpmverifypackage\_testOVAL** nel contenuto o utilizzare solo il contenuto del **scap-security-guide** pacchetto dove **rpmverifypackage\_test** non viene utilizzato.

(BZ#1646197)

### libssh non è conforme alla politica di crittografia a livello di sistema

La **libssh** libreria non segue le impostazioni dei criteri di crittografia a livello di sistema. Di conseguenza, l'insieme degli algoritmi supportati non viene modificato quando l'amministratore cambia il livello delle politiche di crittografia utilizzando il **update-crypto-policies** comando.

Per risolvere questo problema, l'insieme di algoritmi pubblicizzati deve essere impostato individualmente da ogni applicazione che utilizza **libssh**. Di conseguenza, quando il sistema è impostato a livello di politica LEGACY o FUTURE, le applicazioni che utilizzano il sistema si **libssh** comportano in modo incoerente rispetto a **OpenSSH**.

(BZ#1646563)

### SCAP Workbench non riesce a generare riparazioni basate sui risultati da profili personalizzati

Il seguente errore si verifica quando si tenta di generare ruoli di correzione basati sui risultati da un profilo personalizzato utilizzando lo strumento **SCAP Workbench**:

Error generating remediation role .../remediation.sh: Exit code of oscap was 1: [output truncated]

Per risolvere questo problema, utilizzare il **oscap** comando con l'**--tailoring-file** opzione.

(BZ#1640715)

### OpenSCAPrpmverifyfile non funziona

Lo scanner **OpenSCAP** non cambia correttamente la directory di lavoro corrente in modalità offline, e la **fchdir** funzione non viene chiamata con gli argomenti corretti nella sonda **OpenSCAPrpmverifyfile**. Di conseguenza, la scansione di file system arbitrari utilizzando il **oscap-chroot** comando fallisce se **rpmverifyfile\_test** viene utilizzato in un contenuto SCAP. Di conseguenza, si **oscap-chroot** interrompe nello scenario descritto.

(BZ#1636431)

### Non è disponibile un'utilità per la sicurezza e la scansione di conformità dei contenitori

In Red Hat Enterprise Linux 7, l'**oscap-docker** utilità può essere utilizzata per la scansione di contenitori Docker basati sulle tecnologie Atomic. In Red Hat Enterprise Linux 8, i comandi **OpenSCAP** relativi a Docker e Atomic non sono disponibili. Di conseguenza, **oscap-docker** un'utilità equivalente per la sicurezza e la scansione di conformità dei contenitori non è attualmente disponibile in RHEL 8.

(BZ#1642373)

## CAPITOLO 8. NOTEVOLI MODIFICHE AI CONTENITORI (TRADUZIONE AUTOMATICA)

Un set di immagini contenitore è disponibile per Red Hat Enterprise Linux (RHEL) 8.0. Notevoli cambiamenti includono:

- Docker non è incluso in RHEL 8.0. Per lavorare con i contenitori, utilizzare gli attrezzi **podman**, **buildah**, **skopeo** e **runc**.  
Per informazioni su questi utensili e sull'utilizzo dei contenitori in RHEL 8, vedere [Building, running, and managing containers](#).
- Lo strumento **podman** è stato rilasciato come funzione pienamente supportata. Lo strumento **podman** gestisce pod, immagini contenitore e contenitori su un singolo nodo. Si basa sulla libreria **libpod**, che permette la gestione di contenitori e gruppi di contenitori, chiamati pods.

Per imparare ad usare **podman**, vedere [Building, running, and managing containers](#).

- In RHEL 8 GA, Red Hat Universal Base Images (UBI) è nuovamente disponibile. Le UBI sostituiscono alcune delle immagini precedentemente fornite da Red Hat, come le immagini standard e le immagini base RHEL minime.  
A differenza delle vecchie immagini di Red Hat, le UBI sono liberamente ridistribuibili. Ciò significa che possono essere utilizzati in qualsiasi ambiente e condivisi ovunque. Puoi usarli anche se non sei un cliente di Red Hat.

Per la documentazione UBI, vedere [Building, running, and managing containers](#).

- In RHEL 8 GA, sono disponibili ulteriori immagini contenitore che forniscono componenti AppStream, per i quali le immagini contenitore sono distribuite con le **Red Hat Software Collections** in RHEL 7. Tutte queste immagini di RHEL 8 sono basate sull'immagine di **ubi8**base.
- Le immagini del contenitore ARM per l'architettura ARM a 64 bit sono pienamente supportate in RHEL 8.
- Il **rhel-tools** contenitore è stato rimosso in RHEL 8. Gli utensili **rhel-tools** sono forniti nel **support-tools** contenitore. Gli amministratori di sistema possono anche utilizzare questa immagine come base per costruire l'immagine del contenitore degli strumenti di sistema.
- Il supporto per contenitori senza radici è disponibile in anteprima tecnologica in RHEL 8. I contenitori Rootless sono contenitori che vengono creati e gestiti da utenti regolari del sistema senza permessi amministrativi.

## APPENDICE A. ELENCO DEI BIGLIETTI PER COMPONENTE

Componente	Biglietti
<b>389-ds-base</b>	BZ#1334254, BZ#1358706, <a href="#">BZ#1693159</a>
<b>NetworkManager</b>	BZ#1555013, BZ#1555012, BZ#1557035
<b>PackageKit</b>	BZ#1559414
<b>anaconda</b>	BZ#1499442, BZ#1500792, BZ#1547908, BZ#1612060, BZ#1595415, BZ#1610806, BZ#1533904
<b>audit</b>	BZ#1616428
<b>bcc</b>	BZ#1548302
<b>bind</b>	BZ#1588592
<b>boom-boot</b>	BZ#1649582
<b>boost</b>	BZ#1494495, BZ#1616244
<b>cloud-init</b>	BZ#1615599
<b>cmake</b>	BZ#1590139
<b>cockpit</b>	BZ#1619993
<b>crypto-policies</b>	BZ#1591620
<b>cryptsetup</b>	BZ#1564540
<b>device-mapper-multipath</b>	BZ#1643550
<b>distribution</b>	BZ#1566048, BZ#1516741, BZ#1516728
<b>dnf</b>	BZ#1622580, BZ#1647760, BZ#1581191
<b>esc</b>	BZ#1538645
<b>firewalld</b>	BZ#1509026
<b>gcc</b>	BZ#1169184, BZ#1607227, BZ#1535774, BZ#1504980, BZ#1246444
<b>gdm</b>	BZ#1589678

Componente	Biglietti
<b>glibc</b>	BZ#1512004, BZ#1376834, BZ#1512010, BZ#1304448, BZ#1512009, BZ#1512006, BZ#1514839, BZ#1533608
<b>go-toolset</b>	BZ#1512570
<b>httpd</b>	BZ#1633224, BZ#1632754
<b>iproute</b>	BZ#1640991
<b>iptables</b>	BZ#1644030, BZ#1564596
<b>iscsi-initiator-utils</b>	BZ#1626629, BZ#1582099
<b>kernel-rt</b>	BZ#1592977
<b>kernel</b>	BZ#1598448, BZ#1643522, BZ#1485546, BZ#1562998, BZ#1485229, BZ#1494651, BZ#1485532, BZ#1494028, BZ#1563617, BZ#1485525, BZ#1261167, BZ#1562987, BZ#1273139, BZ#1559607, BZ#1401552, BZ#1638465, BZ#1598776, BZ#1503672, BZ#1664838, BZ#1596240, BZ#1534870, BZ#1501618, BZ#1153521, BZ#1494705, BZ#1620330, BZ#1505999
<b>kexec-tools</b>	BZ#1520209
<b>kmod-kvdo</b>	BZ#1534087, BZ#1639512
<b>libdnf</b>	BZ#1642458
<b>libreswan</b>	<a href="#">BZ#1657854</a>
<b>libssh</b>	BZ#1485241
<b>ltrace</b>	BZ#1618748, BZ#1584322
<b>lvm2</b>	BZ#1643553, BZ#1643543, BZ#1643545, BZ#1643547, BZ#1643549, BZ#1643562, BZ#1643576
<b>mariadb</b>	BZ#1637034
<b>net-snmp</b>	BZ#1584510
<b>nfs-utils</b>	BZ#1592011, BZ#1639432
<b>nftables</b>	BZ#1593711

Componente	Biglietti
<b>nginx</b>	BZ#1545526
<b>nodejs-10-module</b>	BZ#1622118
<b>nss</b>	BZ#1645153
<b>nuxwdog</b>	<a href="#">BZ#1652269</a>
<b>openldap</b>	BZ#1570056
<b>opensc</b>	BZ#1595626
<b>openscap</b>	BZ#1646197, BZ#1636431, BZ#1642373, BZ#1618484, BZ#1614273, BZ#1618464
<b>openssh</b>	BZ#1622511, BZ#1228088
<b>pacemaker</b>	<a href="#">BZ#1543494</a>
<b>pcs</b>	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
<b>perl-IO-Socket-SSL</b>	BZ#1632600
<b>perl</b>	BZ#1511131
<b>pki-core</b>	BZ#1565073, BZ#1623444, BZ#1566360, BZ#1394069, <a href="#">BZ#1669257</a> , BZ#1656856, <a href="#">BZ#1673296</a>
<b>pykickstart</b>	BZ#1637872, BZ#1612061
<b>qemu-kvm</b>	BZ#1508139, BZ#1559240, BZ#1497911, BZ#1578855, BZ#1651994, BZ#1621817, BZ#1508137, BZ#1519004
<b>redhat-release</b>	BZ#1636338
<b>rsyslog</b>	BZ#1613880, BZ#1542497, BZ#1614179, BZ#1619645
<b>scap-security-guide</b>	BZ#1618528, BZ#1618518
<b>scap-workbench</b>	BZ#1640715
<b>selinux-policy</b>	BZ#1592244, BZ#1594111, BZ#1549772, BZ#1626446

Componente	Biglietti
<b>setup</b>	BZ#1591969
<b>sos</b>	BZ#1559836
<b>sssd</b>	BZ#1448094, BZ#1382750, BZ#1642508, BZ#1620123
<b>subversion</b>	BZ#1571415
<b>swig-3.0-module</b>	BZ#1660051
<b>tomcatjss</b>	BZ#1424966, BZ#1636564
<b>tuned</b>	BZ#1576435, BZ#1565598
<b>valgrind</b>	BZ#1500481, BZ#1538009
<b>virt-manager</b>	BZ#1599777, BZ#1643609
<b>wpa_supplicant</b>	BZ#1582538
altro	BZ#1646563, BZ#1640697, BZ#1623712, BZ#1649404, BZ#1581198, BZ#1581990, BZ#1649497, BZ#1643294, BZ#1647612, BZ#1641015, BZ#1641032, BZ#1641004, BZ#1641034, BZ#1647110, BZ#1641007, BZ#1641029, BZ#1641022, BZ#1649493, BZ#1649922, BZ#1559616, BZ#1646541, BZ#1647725, <a href="#">BZ#1686057</a> , BZ#1582530, BZ#1581496, BZ#1650618, BZ#1650675, <a href="#">BZ#1650701</a> , JIRA:RHELPLAN-10347, 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-1212, JIRA:RHELPLAN-1473, JIRA:RHELPLAN-10445, JIRA:RHELPLAN-1499, 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-12764, JIRA:RHELPLAN-14607, BZ#1641014, BZ#1692746, BZ#1693775, BZ#1580387, BZ#1583620, BZ#1580430, BZ#1648843, BZ#1647908, BZ#1649891