



Red Hat Enterprise Linux 8

8.0 Примечания к выпуску (машинный перевод)

Замечания к выпуску Red Hat Enterprise Linux 8.0 (машинный перевод)

Red Hat Enterprise Linux 8 8.0 Примечания к выпуску (машинный перевод)

Замечания к выпуску Red Hat Enterprise Linux 8.0 (машинный перевод)

Юридическое уведомление

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[®] is the registered trademark of Linus Torvalds in the United States and other countries.

Java[®] is a registered trademark of Oracle and/or its affiliates.

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

MySQL[®] is a registered trademark of MySQL AB in the United States, the European Union and other countries.

Node.js[®] 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[®] 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.

Аннотация

В заметках о выпуске подробно описываются улучшения и дополнения, которые были реализованы в Red Hat Enterprise Linux 8.0, и описываются известные проблемы в этом выпуске, а также заметные исправления ошибок, предварительные обзоры технологий, устаревшие функции, удаленные функции и другие подробности.

Содержание

ПРЕДОСТАВЛЕНИЕ ОТЗЫВА О ДОКУМЕНТАЦИИ RED HAT (МАШИННЫЙ ПЕРЕВОД)	4
ГЛАВА 1. ОБЗОР (МАШИННЫЙ ПЕРЕВОД)	5
распределение	5
Управление программным обеспечением	5
Веб-серверы, базы данных, динамические языки	5
рабочий стол	5
Установщик и создание образа	5
Файловые системы и хранилище	6
Безопасность	6
сетей	6
Виртуализация	6
Компиляторы и инструменты разработки	7
Высокая доступность и кластеры	7
Дополнительные ресурсы	7
ГЛАВА 2. АРХИТЕКТУРЫ (МАШИННЫЙ ПЕРЕВОД)	8
ГЛАВА 3. РАСПРЕДЕЛЕНИЕ КОНТЕНТА В RHEL 8 (МАШИННЫЙ ПЕРЕВОД)	9
3.1. МОНТАЖ (МАШИННЫЙ ПЕРЕВОД)	9
3.2. ХРАНИЛИЩА (МАШИННЫЙ ПЕРЕВОД)	9
3.3. ПОТОКИ ПРИЛОЖЕНИЙ (МАШИННЫЙ ПЕРЕВОД)	10
ГЛАВА 4. НОВЫЕ ВОЗМОЖНОСТИ (МАШИННЫЙ ПЕРЕВОД)	11
4.1. ВЕБ-КОНСОЛЬ (МАШИННЫЙ ПЕРЕВОД)	11
4.2. УСТАНОВЩИК И СОЗДАНИЕ ОБРАЗА (МАШИННЫЙ ПЕРЕВОД)	12
4.3. ЯДРО (МАШИННЫЙ ПЕРЕВОД)	14
4.4. УПРАВЛЕНИЕ ПРОГРАММНЫМ ОБЕСПЕЧЕНИЕМ (МАШИННЫЙ ПЕРЕВОД)	15
4.5. ИНФРАСТРУКТУРНЫЕ УСЛУГИ (МАШИННЫЙ ПЕРЕВОД)	17
4.6. ОБОЛОЧКИ И ИНСТРУМЕНТЫ КОМАНДНОЙ СТРОКИ (МАШИННЫЙ ПЕРЕВОД)	18
4.7. ДИНАМИЧЕСКИЕ ЯЗЫКИ ПРОГРАММИРОВАНИЯ, ВЕБ-СЕРВЕРЫ И СЕРВЕРЫ БАЗ ДАННЫХ (МАШИННЫЙ ПЕРЕВОД)	19
4.8. РАБОЧИЙ СТОЛ (МАШИННЫЙ ПЕРЕВОД)	25
4.9. УПРАВЛЕНИЕ ИДЕНТИФИКАЦИЕЙ (МАШИННЫЙ ПЕРЕВОД)	27
4.10. КОМПИЛЯТОРЫ И ИНСТРУМЕНТЫ РАЗРАБОТКИ (МАШИННЫЙ ПЕРЕВОД)	32
4.11. ФАЙЛОВЫЕ СИСТЕМЫ И ХРАНИЛИЩЕ (МАШИННЫЙ ПЕРЕВОД)	41
4.12. ВЫСОКАЯ ДОСТУПНОСТЬ И КЛАСТЕРЫ (МАШИННЫЙ ПЕРЕВОД)	46
4.13. СЕТЕЙ (МАШИННЫЙ ПЕРЕВОД)	49
4.14. БЕЗОПАСНОСТЬ (МАШИННЫЙ ПЕРЕВОД)	53
4.15. ВИРТУАЛИЗАЦИЯ (МАШИННЫЙ ПЕРЕВОД)	59
4.16. SUPPORTABILITY (МАШИННЫЙ ПЕРЕВОД)	61
ГЛАВА 5. ТЕХНОЛОГИЧЕСКИЕ ПРЕВЬЮ (МАШИННЫЙ ПЕРЕВОД)	62
5.1. ЯДРО (МАШИННЫЙ ПЕРЕВОД)	62
5.2. ФАЙЛОВЫЕ СИСТЕМЫ И ХРАНИЛИЩЕ (МАШИННЫЙ ПЕРЕВОД)	63
5.3. ВЫСОКАЯ ДОСТУПНОСТЬ И КЛАСТЕРЫ (МАШИННЫЙ ПЕРЕВОД)	64
5.4. БЕЗОПАСНОСТЬ (МАШИННЫЙ ПЕРЕВОД)	64
5.5. ВИРТУАЛИЗАЦИЯ (МАШИННЫЙ ПЕРЕВОД)	65
ГЛАВА 6. УСТАРЕВШАЯ ФУНКЦИОНАЛЬНОСТЬ (МАШИННЫЙ ПЕРЕВОД)	66
6.1. УСТАНОВЩИК И СОЗДАНИЕ ОБРАЗА (МАШИННЫЙ ПЕРЕВОД)	66
6.2. ФАЙЛОВЫЕ СИСТЕМЫ И ХРАНИЛИЩЕ (МАШИННЫЙ ПЕРЕВОД)	66
6.3. СЕТЕЙ (МАШИННЫЙ ПЕРЕВОД)	67

6.4. БЕЗОПАСНОСТЬ (МАШИННЫЙ ПЕРЕВОД)	67
6.5. ВИРТУАЛИЗАЦИЯ (МАШИННЫЙ ПЕРЕВОД)	68
ГЛАВА 7. ИЗВЕСТНЫЕ ВОПРОСЫ (МАШИННЫЙ ПЕРЕВОД)	69
7.1. УСТАНОВЩИК И СОЗДАНИЕ ОБРАЗА (МАШИННЫЙ ПЕРЕВОД)	69
7.2. ЯДРО (МАШИННЫЙ ПЕРЕВОД)	69
7.3. УПРАВЛЕНИЕ ПРОГРАММНЫМ ОБЕСПЕЧЕНИЕМ (МАШИННЫЙ ПЕРЕВОД)	70
7.4. ИНФРАСТРУКТУРНЫЕ УСЛУГИ (МАШИННЫЙ ПЕРЕВОД)	71
7.5. ОБОЛОЧКИ И ИНСТРУМЕНТЫ КОМАНДНОЙ СТРОКИ (МАШИННЫЙ ПЕРЕВОД)	72
7.6. ДИНАМИЧЕСКИЕ ЯЗЫКИ ПРОГРАММИРОВАНИЯ, ВЕБ-СЕРВЕРЫ И СЕРВЕРЫ БАЗ ДАННЫХ (МАШИННЫЙ ПЕРЕВОД)	72
7.7. УПРАВЛЕНИЕ ИДЕНТИФИКАЦИЕЙ (МАШИННЫЙ ПЕРЕВОД)	73
7.8. КОМПИЛЯТОРЫ И ИНСТРУМЕНТЫ РАЗРАБОТКИ (МАШИННЫЙ ПЕРЕВОД)	74
7.9. ФАЙЛОВЫЕ СИСТЕМЫ И ХРАНИЛИЩЕ (МАШИННЫЙ ПЕРЕВОД)	75
7.10. СЕТЕЙ (МАШИННЫЙ ПЕРЕВОД)	75
7.11. БЕЗОПАСНОСТЬ (МАШИННЫЙ ПЕРЕВОД)	75
ГЛАВА 8. ЗАМЕТНЫЕ ИЗМЕНЕНИЯ В КОНТЕЙНЕРАХ (МАШИННЫЙ ПЕРЕВОД)	77
ПРИЛОЖЕНИЕ А. СПИСОК БИЛЕТОВ ПО КОМПОНЕНТАМ	78

ПРЕДОСТАВЛЕНИЕ ОТЗЫВА О ДОКУМЕНТАЦИИ RED HAT (МАШИННЫЙ ПЕРЕВОД)

Мы ценим ваш вклад в нашу документацию. Пожалуйста, дайте нам знать, как мы могли бы сделать это лучше. Для этого:

- Для простых комментариев по конкретным отрывкам, убедитесь, что вы просматриваете документацию в многостраничном формате HTML. Выделите ту часть текста, которую вы хотите прокомментировать. Затем нажмите **Добавить отзыв** всплывающее окно, которое появляется под выделенным текстом, и следуйте отображаемым инструкциям.
- Для отправки более сложных отзывов создайте тикет Bugzilla:
 1. Перейти к [Bugzilla](#) Веб-сайт.
 2. В качестве компонента используйте **Документация**
 3. Заполните **Описание** поле с вашим предложением для улучшения. Включите ссылку на соответствующую часть (и) документации.
 4. Нажмите **Отправить ошибку**

ГЛАВА 1. ОБЗОР (МАШИННЫЙ ПЕРЕВОД)

Основанная на Fedora 28 и вышестоящем ядре 4.18, Red Hat Enterprise Linux 8.0 предоставляет пользователям стабильную, безопасную и согласованную основу для развертывания гибридного облака с инструментами, необходимыми для поддержки традиционных и возникающих рабочих нагрузок. Основные моменты выпуска включают в себя:

распределение

- Контент доступен через BaseOS и поток приложений (AppStream) хранилища.
- AppStream репозиторий поддерживает новое расширение традиционного формата RPM - *модули*. Это позволяет нескольким основным версиям компонента быть доступным для установки.

Увидеть [Глава 3, *Распределение контента в RHEL 8 \(машинный перевод\)*](#) для дополнительной информации.

Управление программным обеспечением

- YUM менеджер пакетов теперь основан на DNF технологии и обеспечивает поддержку модульного контента, повышение производительности и хорошо продуманный стабильный API для интеграции с инструментами.

Увидеть [Раздел 4.4, «Управление программным обеспечением \(машинный перевод\)»](#) Больше подробностей.

Веб-серверы, базы данных, динамические языки

- **питон** 3.6 - реализация Python по умолчанию в RHEL 8; ограниченная поддержка Python 2.7. По умолчанию версия Python не установлена.
- RHEL 8 обеспечивает следующее **серверы баз данных**: MariaDB 10.3, MySQL 8.0, PostgreSQL 10, PostgreSQL 9.6 и Redis 4.0.

Увидеть [Раздел 4.7, «Динамические языки программирования, веб-серверы и серверы баз данных \(машинный перевод\)»](#) для дополнительной информации.

рабочий стол

- GNOME Shell был перебазируется до версии 3.28.
- Сеанс GNOME и использование диспетчера отображения GNOME Wayland как их сервер отображения по умолчанию. X.Org сервер, который является сервером отображения по умолчанию в RHEL 7, также доступен.

Увидеть [Раздел 4.8, «рабочий стол \(машинный перевод\)»](#) для дополнительной информации.

Установщик и создание образа

- **анаконда** установщик может использовать LUKS2 шифрование диска и установка системы на NVDIMM устройства.
- Новый **Композитор** Инструмент позволяет пользователям создавать настраиваемые системные образы в различных форматах, включая

изображения, подготовленные для развертывания в облаках различных поставщиков. Композитор доступен как *Предварительный просмотр технологии*

- Установка с DVD с использованием консоли аппаратного обеспечения (HMC) и элемент поддержки (SE) на IBM Z

Увидеть [Раздел 4.2, «Установщик и создание образа \(машинный перевод\)»](#) для дальнейших деталей.

Файловые системы и хранилище

- **Stratis** был введен менеджер локального хранилища. Stratis позволяет легко выполнять сложные задачи хранения и управлять стеком хранения с помощью унифицированного интерфейса.
- ЛУКС версия 2 (LUKS2) заменяет устаревший формат LUKS (LUKS1). **dm-crypt** подсистема и тому **cryptsetup** Инструмент теперь использует LUKS2 в качестве формата по умолчанию для зашифрованных томов.

Увидеть [Раздел 4.11, «Файловые системы и хранилище \(машинный перевод\)»](#) для дополнительной информации.

Безопасность

- Общесистемную **криптографические политики**, который настраивает основные криптографические подсистемы, охватывающие протоколы TLS, IPSec, SSH, DNSSEC и Kerberos, применяются по умолчанию. С новым **update-crypto-policies** Команда, администратор может легко переключаться между режимами: по умолчанию, наследие, будущее и fips.
- Поддержка для **смарт-карты** и аппаратные модули безопасности (HSM) с PKCS # 11 теперь согласован по всей системе.

Увидеть [Раздел 4.14, «Безопасность \(машинный перевод\)»](#) для дополнительной информации.

сетей

- **nftables** рамки заменяет **iptables** в роли средства фильтрации сетевых пакетов по умолчанию.
- **firewalld** демон теперь использует **nftables** в качестве бэкенда по умолчанию.
- Поддержка для **IPVLAN** драйверы виртуальной сети, которые обеспечивают сетевое подключение для нескольких контейнеров.

Увидеть [Раздел 4.13, «сетей \(машинный перевод\)»](#) для дополнительных деталей.

Виртуализация

- Более современный тип машины на основе PCI Express (Q35) теперь поддерживается и автоматически настраивается на виртуальных машинах, созданных в RHEL 8. Это обеспечивает множество улучшений в функциях и совместимости виртуальных устройств.
- Теперь виртуальные машины можно создавать и управлять ими с помощью веб-консоли RHEL 8, также известной как **кокпит**

- **QEMU** эмулятор вводит **песочница** функция, которая обеспечивает настраиваемые ограничения на то, что системные вызовы QEMU могут выполнять, и, таким образом, делает виртуальные машины более безопасными.

Увидеть [Раздел 4.15, «Виртуализация \(машинный перевод\)»](#) для дополнительной информации.

Компиляторы и инструменты разработки

- **НКУ** Компилятор на основе версии 8.2 обеспечивает поддержку более новых стандартных версий языка C ++, улучшенную оптимизацию, новые методы защиты кода, улучшенные предупреждения и новые аппаратные функции.
- Различные инструменты для генерации кода, манипулирования и отладки теперь могут экспериментально обрабатывать **DWARF5** формат отладочной информации.
- Поддержка ядра для **eBPF** трассировка доступна для некоторых инструментов, таких как **BCC**, **PCP**, а также **SystemTap**
- **glibc** В библиотеках на основе версии 2.28 добавлена поддержка Unicode 11, более новые системные вызовы Linux, ключевые улучшения в преобразователе заглушек DNS, дополнительное усиление безопасности и улучшенная производительность.

Увидеть [Раздел 4.10, «Компиляторы и инструменты разработки \(машинный перевод\)»](#) для дополнительных деталей.

Высокая доступность и кластеры

- **электрокардиостимулятор** Диспетчер ресурсов кластера обновлен до версии 2.0.0, которая содержит ряд исправлений и улучшений.
- В RHEL 8 **штук** Конфигурация системы полностью поддерживает Corosync 3, **knet**, и имена узлов.

Увидеть [Раздел 4.12, «Высокая доступность и кластеры \(машинный перевод\)»](#) для дополнительной информации.

Дополнительные ресурсы

- Возможности и ограничения Red Hat Enterprise Linux 8.0 по сравнению с другими версиями системы доступны в статье базы знаний. [Red Hat Enterprise Linux technology capabilities and limits](#)
- Информация о жизненном цикле Red Hat Enterprise Linux приведена в [Red Hat Enterprise Linux Life Cycle](#) документ.
- [Package manifest](#) Документ содержит список пакетов для RHEL 8.

ГЛАВА 2. АРХИТЕКТУРЫ (МАШИННЫЙ ПЕРЕВОД)

Red Hat Enterprise Linux 8.0 распространяется с версией ядра 4.18, которая поддерживает следующие архитектуры:

- 64-битные архитектуры AMD и Intel
- 64-битная архитектура ARM
- IBM Power Systems, Little Endian
- IBM Z

ГЛАВА 3. РАСПРЕДЕЛЕНИЕ КОНТЕНТА В RHEL 8 (МАШИННЫЙ ПЕРЕВОД)

3.1. МОНТАЖ (МАШИННЫЙ ПЕРЕВОД)

Red Hat Enterprise Linux 8 устанавливается с использованием образов ISO. Для архитектур AMD64, 64-битной, 64-битной ARM, IBM Power Systems и IBM Z доступны два типа ISO-образа:

- Двоичный DVD ISO: Полный установочный образ, содержащий репозитории BaseOS и AppStream и позволяющий завершить установку без дополнительных репозиториях.
- Загрузочный ISO: Минимальный загрузочный ISO-образ, который используется для загрузки в программу установки. Эта опция требует доступа к репозиториям BaseOS и AppStream для установки пакетов программного обеспечения. Репозитории являются частью образа Binary DVD ISO.

Увидеть [Performing a standard RHEL installation](#) документ с инструкциями по загрузке образов ISO, созданию установочного носителя и завершению установки RHEL. Для автоматических установок Kickstart и других расширенных тем см. [Performing an advanced RHEL installation](#) документ.

3.2. ХРАНИЛИЩА (МАШИННЫЙ ПЕРЕВОД)

Red Hat Enterprise Linux 8 распространяется через два репозитория:

- BaseOS
- AppStream

Оба репозитория необходимы для базовой установки RHEL и доступны со всеми подписками RHEL.

Содержимое в репозитории BaseOS предназначено для предоставления базового набора базовых функциональных возможностей ОС, которые обеспечивают основу для всех установок. Этот контент доступен в формате RPM, и на него распространяются условия поддержки, аналогичные тем, которые были в предыдущих выпусках RHEL. Список пакетов, распространяемых через BaseOS, см. В [Package manifest](#)

Содержимое в репозитории Application Stream включает дополнительные приложения пользовательского пространства, языки времени выполнения и базы данных для поддержки различных рабочих нагрузок и вариантов использования. Содержимое в AppStream доступно в одном из двух форматов - знакомый формат RPM и расширение формата RPM, называемое *модули* Список пакетов, доступных в AppStream, см. В [Package manifest](#)

Кроме того, репозиторий CodeReady Linux Builder доступен со всеми подписками RHEL. Он предоставляет дополнительные пакеты для использования разработчиками. Пакеты, включенные в репозиторий CodeReady Linux Builder, не поддерживаются для производственного использования.

Для получения дополнительной информации о репозиториях RHEL 8 см. [Package manifest](#)

3.3. ПОТОКИ ПРИЛОЖЕНИЙ (МАШИННЫЙ ПЕРЕВОД)

Red Hat Enterprise Linux 8.0 представляет концепцию потоков приложений. Несколько версий компонентов пользовательского пространства теперь поставляются и обновляются чаще, чем пакеты основной операционной системы. Это обеспечивает большую гибкость в настройке Red Hat Enterprise Linux без ущерба для базовой стабильности платформы или конкретных развертываний.

Компоненты, доступные в виде потоков приложений, могут быть упакованы в виде модулей или пакетов RPM и доставлены через репозиторий AppStream в RHEL 8. Каждый компонент Application Stream имеет определенный жизненный цикл. Подробнее см. [Red Hat Enterprise Linux Life Cycle](#)

Модули - это наборы пакетов, представляющих логическую единицу: приложение, языковой стек, базу данных или набор инструментов. Эти пакеты собраны, протестированы и выпущены вместе.

Модульные потоки представляют версии компонентов Application Stream. Например, два модуля (версии) сервера базы данных PostgreSQL доступны в модуле postgresql: PostgreSQL 10 (поток по умолчанию) и PostgreSQL 9.6. В системе может быть установлен только один поток модулей. Различные версии могут использоваться в отдельных контейнерах.

Подробные команды модуля описаны в [Using Application Stream](#) документ. Список модулей, доступных в AppStream, см. В [Package manifest](#)

ГЛАВА 4. НОВЫЕ ВОЗМОЖНОСТИ (МАШИННЫЙ ПЕРЕВОД)

В этой части описываются новые функции и основные улучшения, представленные в Red Hat Enterprise Linux 8.

4.1. ВЕБ-КОНСОЛЬ (МАШИННЫЙ ПЕРЕВОД)



ПРИМЕЧАНИЕ

Страница подписок веб-консоли теперь предоставляется новым `subscription-manager-cockpit` пакет.

Интерфейс брандмауэра был добавлен в веб-консоль

сетей страница в веб-консоли RHEL 8 теперь содержит **Брандмауэр** раздел. В этом разделе пользователи могут включать или отключать брандмауэр, а также добавлять, удалять и изменять правила брандмауэра.

(BZ#1647110)

Веб-консоль теперь доступна по умолчанию

Пакеты для веб-консоли RHEL 8, также известные как Cockpit, теперь являются частью стандартных репозиториях Red Hat Enterprise Linux и поэтому могут быть немедленно установлены в зарегистрированной системе RHEL 8.

Кроме того, при минимальной установке RHEL 8 веб-консоль устанавливается автоматически, а порты брандмауэра, необходимые для консоли, автоматически открываются. Системное сообщение также было добавлено до входа в систему, в котором содержится информация о том, как включить или получить доступ к веб-консоли.

(JIRA:RHELPLAN-10355)

Лучшая интеграция IdM для веб-консоли

Если ваша система зарегистрирована в домене Identity Management (IdM), веб-консоль RHEL 8 теперь использует ресурсы IdM централизованно управляемого домена по умолчанию. Это включает в себя следующие преимущества:

- Администраторы домена IdM могут использовать веб-консоль для управления локальным компьютером.
- Веб-сервер консоли автоматически переключается на сертификат, выданный центром сертификации IdM (CA) и принятый браузерами.
- Пользователям с билетом Kerberos в домене IdM не нужно предоставлять учетные данные для доступа к веб-консоли.
- Узлы SSH, известные домену IdM, доступны для веб-консоли без добавления соединения SSH вручную.

Обратите внимание, что для правильной работы интеграции IdM с веб-консолью пользователю необходимо сначала запустить `ipa-adviser` утилита с `enable-admins-sudo` опция в основной системе IdM.

(JIRA:RHELPLAN-3010)

Веб-консоль теперь совместима с мобильными браузерами.

В этом обновлении меню и страницы веб-консоли можно перемещать в вариантах мобильного браузера. Это позволяет управлять системами с помощью веб-консоли RHEL 8 с мобильного устройства.

(JIRA:RHELPLAN-10352)

На главной странице веб-консоли теперь отображаются отсутствующие обновления и подписки.

Если в системе, управляемой веб-консолью RHEL 8, имеются устаревшие пакеты или недействительная подписка, теперь на главной странице веб-консоли системы отображается предупреждение.

(JIRA:RHELPLAN-10353)

Веб-консоль теперь поддерживает регистрацию PBD

В этом обновлении вы можете использовать интерфейс веб-консоли RHEL 8 для применения правил расшифровки на основе политик (PBD) к дискам в управляемых системах. При этом используется клиент дешифрования Clevis для упрощения различных функций управления безопасностью в веб-консоли, таких как автоматическая разблокировка зашифрованных разделов LUKS.

(JIRA:RHELPLAN-10354)

Виртуальными машинами теперь можно управлять с помощью веб-консоли

Virtual Machines Теперь можно добавить страницу в интерфейс веб-консоли RHEL 8, который позволяет пользователю создавать виртуальные машины на основе libvirt и управлять ими.

(JIRA:RHELPLAN-2896)

4.2. УСТАНОВЩИК И СОЗДАНИЕ ОБРАЗА (МАШИННЫЙ ПЕРЕВОД)

Установка RHEL с DVD с помощью SE а также HMC теперь полностью поддерживается в IBM Z

Установка Red Hat Enterprise Linux 8 на оборудование IBM Z с DVD-диска с использованием **Элемент поддержки (SE)** а также **Консоль аппаратного обеспечения (HMC)** теперь полностью поддерживается. Это дополнение упрощает процесс установки на IBM Z с SE а также HMC

При загрузке с бинарного DVD установщик предлагает пользователю ввести дополнительные параметры ядра. Чтобы установить DVD в качестве источника установки, добавьте `inst.repo=hmc` к параметрам ядра. Затем установщик включает SE а

также НМС доступ к файлам, извлекает образы для stage2 с DVD и предоставляет доступ к пакетам на DVD для выбора программного обеспечения.

Новая функция устраняет необходимость настройки внешней сети и расширяет возможности установки.

(BZ#1500792)

Установщик теперь поддерживает формат шифрования диска LUKS2

Установщик Red Hat Enterprise Linux 8 теперь по умолчанию использует формат LUKS2, но вы можете выбрать версию LUKS из **Анаконда - x** Пользовательское окно разделения или с помощью новых параметров в Kickstart **autopart**, **logvol**, **part**, а также **RAID** команды.

LUKS2 предоставляет множество улучшений и функций, например, расширяет возможности формата на диске и предоставляет гибкие способы хранения метаданных.

(BZ#1547908)

анаконда поддерживает назначение системы в RHEL 8

Ранее, **анаконда** не предоставил информацию о назначении системы **Менеджер подписки** В Red Hat Enterprise Linux 8.0 вы можете установить целевое назначение системы во время установки, используя **Анаконда - x System Purpose** окно или кикстарт **syspurpose** команда. Когда установка завершится, **Менеджер подписки** использует информацию о назначении системы при подписке на систему.

(BZ#1612060)

Pykickstart поддерживает назначение системы в RHEL 8

Ранее это было невозможно для **pykickstart** библиотека для предоставления информации о назначении системы **Менеджер подписки** В Red Hat Enterprise Linux 8.0 **pykickstart** разбирает новый **syspurpose** Команда и записывает целевое назначение системы во время автоматической и частично автоматизированной установки. Затем информация передается **анаконда**, сохраненные во вновь установленной системе и доступные для **Менеджер подписки** при подписке на систему.

(BZ#1612061)

анаконда поддерживает новый параметр загрузки ядра в RHEL 8

Ранее вы могли указывать только базовый репозиторий из параметров загрузки ядра. В Red Hat Enterprise Linux 8, новый параметр ядра, **inst.addrepo=<name>,<url>**, позволяет указать дополнительный репозиторий во время установки.

Этот параметр имеет два обязательных значения: имя хранилища и URL-адрес, указывающий на хранилище. Для получения дополнительной информации см.

<https://anaconda-installer.readthedocs.io/en/latest/boot-options.html#inst-addrepo>

(BZ#1595415)

анаконда поддерживает унифицированный ISO в RHEL 8

В Red Hat Enterprise Linux 8.0 унифицированный ISO-файл автоматически загружает репозитории исходных файлов установки BaseOS и AppStream.

Эта функция работает для первого базового репозитория, который загружается во

время установки. Например, если вы загружаете установку без настроенного хранилища и используете унифицированный ISO в качестве базового хранилища в графическом интерфейсе, или если вы загружаете установку с помощью **inst.repo=** вариант, который указывает на единый ISO. В результате хранилище AppStream включается под **Дополнительные репозитории** раздел **Источник установки** Окно GUI. Вы не можете удалить хранилище AppStream или изменить его настройки, но вы можете отключить его в **Источник установки** Эта функция не работает, если вы загружаете установку с использованием другого базового репозитория, а затем меняете его на унифицированный ISO. Если вы это сделаете, базовый репозиторий будет заменен. Однако хранилище AppStream не заменяется и указывает на исходный файл.

(BZ#1610806)

4.3. ЯДРО (МАШИННЫЙ ПЕРЕВОД)

ARM 52-битная физическая адресация теперь доступна

В этом обновлении доступна поддержка 52-битной физической адресации (PA) для 64-битной архитектуры ARM. Это обеспечивает большее адресное пространство, чем предыдущий 48-битный PA.

(BZ#1643522)

Код IOMMU поддерживает 5-уровневые таблицы страниц в RHEL 8

Код модуля управления памятью ввода / вывода (IOMMU) в ядре Linux был обновлен для поддержки 5-уровневых таблиц страниц в Red Hat Enterprise Linux 8.

(BZ#1485546)

Поддержка 5-уровневого пейджинга

новый **P4d_t** тип таблицы страниц программного обеспечения добавлен в ядро Linux для поддержки 5-уровневой подкачки в Red Hat Enterprise Linux 8.

(BZ#1485532)

Управление памятью поддерживает 5-уровневые таблицы страниц

В Red Hat Enterprise Linux 7 существующая шина памяти имела 48/46-битную емкость адресации виртуальной / физической памяти, а ядро Linux реализовало 4 уровня таблиц страниц для управления этими виртуальными адресами по физическим адресам. Линия адресации физической шины установила верхний предел емкости физической памяти в 64 ТБ.

Эти ограничения были расширены до 57/52 бит адресации виртуальной / физической памяти с 128 ПБ виртуального адресного пространства и 4 ПБ емкости физической памяти.

Благодаря расширенному диапазону адресов управление памятью в Red Hat Enterprise Linux 8 добавляет поддержку реализации 5-уровневой таблицы страниц, чтобы иметь возможность обрабатывать расширенный диапазон адресов.

(BZ#1485525)

kernel-signing-ca.cer перемещен в kernel-core в RHEL 8

Во всех версиях Red Hat Enterprise Linux 7 **kernel-signing-ca.cer** открытый ключ был расположен в **kernel-doc** пакет. Однако в Red Hat Enterprise Linux 8 **kernel-signing-ca.cer** был перемещен в **kernel-core** пакет для любой архитектуры.

(BZ#1638465)

bpftool для проверки и управления программами и картами на основе eBPF

bpftool в ядро Linux была добавлена утилита, которая служит для проверки и простого манипулирования программами и картами на основе расширенной Berkeley Packet Filtering (eBPF). **bpftool** является частью дерева исходного кода ядра и предоставляется **bpftool** пакет, который входит в качестве подпакета **ядро** пакет.

(BZ#1559607)

kernel-rt источники были обновлены

kernel-rt исходники были обновлены для использования самого последнего дерева исходных текстов ядра RHEL. В последнем дереве исходного кода ядра теперь используется набор исправлений реального времени v4.18, который предоставляет ряд исправлений ошибок и улучшений по сравнению с предыдущей версией.

(BZ#1592977)

4.4. УПРАВЛЕНИЕ ПРОГРАММНЫМ ОБЕСПЕЧЕНИЕМ (МАШИННЫЙ ПЕРЕВОД)

YUM улучшение производительности и поддержка модульного контента

В Red Hat Enterprise Linux 8 установка программного обеспечения обеспечивается новой версией YUM инструмент, который основан на DNF технология.

YUM основанный на DNF имеет следующие преимущества перед предыдущими YUM v3 используется на RHEL 7:

- Увеличение производительности
- Поддержка модульного контента
- Хорошо разработанный стабильный API для интеграции с инструментами

Для получения подробной информации о различиях между новыми YUM инструмент и предыдущая версия YUM v3 от RHEL 7, см. http://dnf.readthedocs.io/en/latest/cli_vs_yum.html

YUM основанный на DNF совместим с YUM v3 при использовании из командной строки редактирование или создание файлов конфигурации.

Для установки программного обеспечения вы можете использовать **yum** Команда и ее конкретные параметры аналогичны командам RHEL 7. Пакеты могут быть установлены под предыдущими именами, используя **Provides** Пакеты также предоставляют символические ссылки совместимости, поэтому двоичные файлы, файлы конфигурации и каталоги можно найти в обычных местах.

Обратите внимание, что устаревший Python API предоставляется YUM v3 и API-интерфейс Libdnf C нестабилен и, вероятно, изменится в течение жизненного цикла Red Hat Enterprise Linux 8. Пользователям рекомендуется перенести свои плагины и скрипты

на новый DNF Python API, который стабилен и полностью поддерживается. API-интерфейс DNF Python доступен по адресу <https://dnf.readthedocs.io/en/latest/api.html>

Некоторые из YUM v3 функции могут вести себя по-разному в YUM основанный на DNF. Если любое такое изменение негативно влияет на ваши рабочие процессы, пожалуйста, откройте дело в службу поддержки Red Hat, как описано в [How do I open and manage a support case on the Customer Portal?](#)

(BZ#1581198)

Известные особенности RPM в RHEL 8

Red Hat Enterprise Linux 8 распространяется с RPM 4.14. Эта версия содержит множество улучшений по сравнению с RPM 4.11, которая доступна в RHEL 7. Наиболее заметные особенности включают в себя:

- **debuginfo** пакеты могут быть установлены параллельно
- Поддержка слабых зависимостей
- Поддержка богатых или логических зависимостей
- Поддержка упаковки файлов размером более 4 ГБ
- Поддержка файловых триггеров

Кроме того, наиболее заметные изменения включают в себя:

- Более строгий анализатор
- Упрощенная проверка подписи в не подробном режиме
- Дополнения и устаревшие в макросах

(BZ#1581990)

RPM теперь проверяет все содержимое пакета перед началом установки

В Red Hat Enterprise Linux 7 RPM утилита проверяла содержимое отдельных файлов при распаковке. Однако этого недостаточно по нескольким причинам:

- Если полезная нагрузка повреждена, она заметна только после выполнения действий сценария, которые необратимы.
- Если полезная нагрузка повреждена, обновление пакета прерывается после замены некоторых файлов предыдущей версии, что нарушает работоспособность установки.
- Хеши на отдельных файлах выполняются на несжатых данных, что делает RPM уязвимы для уязвимостей декомпрессора.

В Red Hat Enterprise Linux 8 весь пакет проверяется перед установкой на отдельном этапе с использованием наилучшего доступного хэша.

Пакеты, основанные на Red Hat Enterprise Linux 8, используют новый **SHA256** хэш сжатой полезной нагрузки. В подписанных пакетах хэш полезной нагрузки дополнительно защищен подписью и, следовательно, не может быть изменен без разрыва подписи и

других хешей в заголовке пакета. Старые пакеты используют **MD5** хэш заголовка и полезной нагрузки, если он не отключен конфигурацией, такой как режим FIPS.

Макрос ``%_pkgverify_level`` может использоваться для дополнительного включения принудительного подтверждения подписи перед установкой или полного отключения проверки полезной нагрузки. В дополнение `%_pkgverify_flags` макрос может использоваться для ограничения того, какие хэши и подписи разрешены. Например, можно отключить использование слабых **MD5** хэш за счет совместимости со старыми пакетами.

(JIRA:RHELPLAN-1499)

4.5. ИНФРАСТРУКТУРНЫЕ УСЛУГИ (МАШИННЫЙ ПЕРЕВОД)

Заметные изменения в рекомендуемом настроенном профиле в RHEL 8

С этим обновлением рекомендуется настроенный профиль (сообщается `tuned-adm recommend` команда) теперь выбирается на основе следующих правил - первое соответствующее правило вступает в силу:

- Если **syspurpose** роль (сообщается `syspurpose show` команда) содержит **atomic**, и в то же время:
 - если Tuned работает на голом металле, то **atomic-host** профиль выбран
 - если Tuned запущен на виртуальной машине, **atomic-guest** профиль выбран
- Если Tuned запущен на виртуальной машине, **virtual-guest** профиль выбран
- Если **syspurpose** роль содержит **desktop** или же **workstation** и тип шасси (сообщается `dmidecode`) является **Notebook**, **Laptop**, или же **Portable**, тогда **balanced** профиль выбран
- Если ни одно из приведенных выше правил не совпадает, **throughput-performance** профиль выбран

(BZ#1565598)

Файлы, созданные названный можно записать в рабочий каталог

Ранее **названный** Демон сохранил некоторые данные в рабочем каталоге, который был доступен только для чтения в Red Hat Enterprise Linux. С этим обновлением пути для выбранных файлов были изменены в подкаталоги, где запись разрешена. Теперь стандартные разрешения Unix и SELinux для каталога позволяют выполнять запись в каталог. Файлы, распространяемые внутри каталога, по-прежнему доступны только для чтения. **названный**

(BZ#1588592)

Базы данных геолита были заменены базами данных геолита2

Базы данных Geolite, которые присутствовали в Red Hat Enterprise Linux 7, были заменены базами данных Geolite2 в Red Hat Enterprise Linux 8.

Базы геолита были предоставлены **GeoIP** пакет. Этот пакет вместе с устаревшей базой данных больше не поддерживается в апстриме.

Базы данных Geolite2 предоставляются несколькими пакетами. **libmaxminddb** пакет

включает в себя библиотеку и **mmdblookup** инструмент командной строки, который позволяет вручную искать адреса. **geoipupdate** двоичный код из наследия **GeoIP** пакет теперь предоставляется **geoipupdate** пакет и способен загружать как устаревшие базы данных, так и новые базы данных Geolite2.

(JIRA:RHELPLAN-6746)

Журналы CUPS обрабатываются journald

В RHEL 8 журналы CUPS больше не хранятся в определенных файлах в `/var/log/cups` каталог, который использовался в RHEL 7. В RHEL 8 все типы журналов CUPS централизованно регистрируются в `systemd`. **journald** демон вместе с логами из других программ. Для доступа к журналам CUPS используйте **journalctl -u cups** команда. Для получения дополнительной информации см. 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 Работа с журналами CUPS.

(JIRA:RHELPLAN-12764)

4.6. ОБОЛОЧКИ И ИНСТРУМЕНТЫ КОМАНДНОЙ СТРОКИ (МАШИННЫЙ ПЕРЕВОД)

nobody пользователь заменяет nfsnobody

В Red Hat Enterprise Linux 7 было:

- **nobody** пара пользователя и группы с идентификатором 99, и
- **nfsnobody** пара пользователя и группы с идентификатором 65534, который также является идентификатором переполнения ядра по умолчанию.

Оба из них были объединены в **nobody** пара пользователь и группа, которая использует 65534 ID в Red Hat Enterprise Linux 8. Новые установки больше не создают **nfsnobody** пара.

Это изменение уменьшает путаницу в отношении файлов, которые принадлежат **nobody** но не имеет ничего общего с NFS.

(BZ#1591969)

Системы контроля версий в RHEL 8

RHEL 8 предоставляет следующие системы контроля версий:

- **Git 2.18** распределенная система контроля версий с децентрализованной архитектурой.
- **Mercurial 4.8** - легкая распределенная система контроля версий, предназначенная для эффективного управления крупными проектами.
- **Subversion 1.10** централизованная система контроля версий. Для получения дополнительной информации см. [Notable changes in Subversion 1.10](#)

Обратите внимание, что система параллельных версий (CVS), доступная в RHEL 7, не распространяется с RHEL 8.

(BZ#1693775)

Заметные изменения в Subversion 1.10

Subversion 1.10 вводит ряд новых функций, начиная с версии 1.7, распространяемой в RHEL 7, а также следующие изменения совместимости:

- Из-за несовместимости в **Subversion** библиотеки, используемые для поддержки языковых привязок, **Python 3** привязки для **Subversion 1.10** недоступны. Как следствие, приложения, которые требуют **Python** привязки для **Subversion** не поддерживаются.
- Репозитории на основе **Berkeley DB** больше не поддерживаются. Перед миграцией сделайте резервную копию репозитория, созданных с помощью **Subversion 1.7** используя **svnadmin dump** команда. После установки RHEL 8 восстановите репозитории, используя **svnadmin load** команда.
- Существующие рабочие копии проверены **Subversion 1.7** клиент в RHEL 7 должен быть обновлен до нового формата, прежде чем его можно будет использовать из **Subversion 1.10**. После установки RHEL 8 запустите **svn upgrade** Команда в каждой рабочей копии.
- Аутентификация с помощью смарт-карты для доступа к репозиториям с использованием **https://** больше не поддерживается.

(BZ#1571415)

4.7. ДИНАМИЧЕСКИЕ ЯЗЫКИ ПРОГРАММИРОВАНИЯ, ВЕБ-СЕРВЕРЫ И СЕРВЕРЫ БАЗ ДАННЫХ (МАШИННЫЙ ПЕРЕВОД)

Python 3 по умолчанию Python реализация в RHEL 8

Red Hat Enterprise Linux 8 распространяется с **Python 3.6**. Пакет не установлен по умолчанию. Установить **Python 3.6**, использовать **yum install python3** команда.

Python 2.7 доступно в **python2** пакет. Тем не менее, **Python 2** будет иметь более короткий жизненный цикл, и его целью является содействие более плавному переходу к **Python 3** для клиентов.

Ни по умолчанию **python** пакет, ни неверсированный **/usr/bin/python** исполняемый файл распространяется с RHEL 8. Клиентам рекомендуется использовать **python3** или же **python2** непосредственно. Кроме того, администраторы могут настроить неверсированные **python** Команда с использованием **alternatives** команда.

Подробнее см. [Using Python in Red Hat Enterprise Linux 8](#)

(BZ#1580387)

Скрипты Python должны указывать основную версию в hashbangs во время сборки RPM

Ожидается, что в RHEL 8 исполняемые скрипты Python будут использовать хэш-банги (шебанги), явно указывающие по крайней мере основную версию Python.

/usr/lib/rpm/redhat/brp-mangle-shebangs Сценарий политики buildroot (BRP) запускается автоматически при сборке любого пакета RPM. Этот скрипт пытается исправить хэш-банг во всех исполняемых файлах. Когда скрипт встречает неоднозначные хэш-банги, которые не могут быть разрешены в основной версии Python, он генерирует ошибки. Примеры таких неоднозначных hashbangs включают в себя:

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

Чтобы изменить hashbangs в скриптах Python, вызывающих эти ошибки сборки во время сборки RPM, используйте **pathfix.py** сценарий из **Платформа-питон-разви** пакет:

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

множественный **ДОРОЖКА** можно указать. Если **ДОРОЖКА** это каталог, **pathfix.py** рекурсивно сканирует любые скрипты Python, соответствующие шаблону **^[a-zA-Z0-9_]+\.[py]\$,** не только с неоднозначным хэшбэнгом. Добавьте эту команду к **%prep** раздел или в конце **%install** раздел.

Для получения дополнительной информации см. [Handling hashbangs in Python scripts](#)

(BZ#1583620)

Заметные изменения в PHP

Red Hat Enterprise Linux 8 распространяется с **PHP 7.2** Эта версия вводит следующие основные изменения **PHP 5.4**, который доступен в RHEL 7:

- **PHP** по умолчанию использует FastCGI Process Manager (FPM) (безопасно для использования с многопоточным **httpd**)
- **php_value** а также **php-flag** переменные больше не должны использоваться в **httpd** конфигурационные файлы; вместо этого они должны быть установлены в конфигурации пула: **/etc/php-fpm.d/*.conf**
- **PHP** ошибки скрипта и предупреждения записываются в **/var/log/php-fpm/www-error.log** файл вместо **/var/log/httpd/error.log**
- При смене PHP **max_execution_time** переменная конфигурации, **httpd ProxyTimeout** настройка должна быть увеличена, чтобы соответствовать
- Пользователь работает **PHP** сценарии теперь настроены в конфигурации пула FPM (**/etc/php-fpm.d/www.conf** файл; **apache** пользователь по умолчанию)
- **php-fpm** служба должна быть перезапущена после изменения конфигурации или после установки нового расширения

Следующие расширения были удалены:

- **aspell**
- **mysql** (обратите внимание, что **mysql** а также **pdo_mysql** расширения все еще доступны, предоставленные **php-mysqlnd** пакет)
- **zip**
- **memcache**

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

Заметные изменения в Ruby

RHEL 8 обеспечивает **Ruby 2.5**, который вводит многочисленные новые функции и улучшения по сравнению с **Ruby 2.0.0** доступно в RHEL 7. Заметные изменения включают в себя:

- Добавлен инкрементный сборщик мусора.
- **Refinements** Синтаксис был добавлен.
- Символы теперь мусор.
- **SAFE=2** а также **SAFE=3** безопасные уровни теперь устарели.
- **Fixnum** а также **Bignum** классы были объединены в **Integer** учебный класс.
- Производительность была улучшена путем оптимизации **Hash** класс, улучшенный доступ к переменным экземпляра и **Mutex** класс меньше и быстрее.
- Некоторые старые API устарели.
- Связанные библиотеки, такие как **RubyGems**, **Rake**, **RDoc**, **Psych**, **Minitest**, а также **test-unit**, были обновлены.
- Другие библиотеки, такие как **mathn**, **DL**, **ext/tk**, а также **XMLRPC**, которые были ранее распространены с **Ruby**, устарели или больше не включены.
- **SemVer** схема управления версиями теперь используется для **Ruby** управления версиями.

(BZ#1648843)

Заметные изменения в Perl

Perl 5.26, распространяемый с RHEL 8, вносит следующие изменения в версию, доступную в RHEL 7:

- **Unicode 9.0** сейчас поддерживается.
- новый **op-entry**, **loading-file**, а также **loaded-file SystemTap** зонды предоставляются.
- Механизм копирования при записи используется при назначении скаляров для повышения производительности.
- **IO::Socket::IP** добавлен модуль для прозрачной обработки сокетов IPv4 и IPv6.
- **Config::Perl::V** модуль для доступа **perl -V** данные в структурированном виде были добавлены.
- Новый **perl-App-cpanminus** пакет был добавлен, который содержит **cpanm** утилита для получения, распаковки, сборки и установки модулей из хранилища всеобъемлющей архивной сети Perl (CPAN).
- Текущий каталог `.` был удален из **@INC** Путь поиска модуля по соображениям безопасности.
- **do** Теперь оператор возвращает предупреждение об устаревании, если не удастся загрузить файл из-за изменения поведения, описанного выше.

- **do subroutine(LIST)** Вызов больше не поддерживается и приводит к синтаксической ошибке.
- Хэши теперь рандомизированы по умолчанию. Порядок, в котором ключи и значения возвращаются из хэша, изменяется на каждом **perl** запуске. Чтобы отключить рандомизацию, установите **PERL_PERTURB_KEYS** переменная окружения **0**
- Unescaped буквальный { символы в шаблонах регулярных выражений больше не допустимы.
- Поддержка лексической области **\$_** переменная была удалена.
- С использованием **defined** Оператор в массиве или хэш приводит к фатальной ошибке.
- Импорт функций из **UNIVERSAL** модуль приводит к фатальной ошибке.
- **find2perl**, **s2p**, **a2p**, **c2ph**, а также **pstruct** инструменты были удалены.
- **\${^ENCODING}** объект был удален. **encoding** Режим прагмы по умолчанию больше не поддерживается. Написать исходный код в другой кодировке, чем **UTF-8**, используйте кодировку **Filter** вариант.
- **perl** упаковка теперь выровнена по потоку. **perl** Пакет устанавливает также основные модули, в то время как **/usr/bin/perl** переводчик предоставляется **perl-interpreter** пакет. В предыдущих выпусках **perl** пакет включал только минимальный переводчик, тогда как **perl-core** В комплект поставки входят как переводчик, так и основные модули.

(BZ#1511131)

Node.js новый в RHEL 8

Node.js Платформа разработки программного обеспечения для создания быстрых и масштабируемых сетевых приложений на языке программирования JavaScript впервые представлена в RHEL. Ранее он был доступен только как коллекция программного обеспечения. RHEL 8 обеспечивает **Node.js 10**

(BZ#1622118)

Заметные изменения в SWIG

RHEL 8 включает в себя версию 3.0 упрощенного упаковщика и интерфейса (SWIG), которая предоставляет множество новых функций, улучшений и исправлений ошибок по сравнению с версией 2.0, распространяемой в RHEL 7. В частности, поддержка стандарта C++ 11 была реализована. SWIG теперь поддерживает также Go 1.6, PHP 7, Octave 4.2 и Python 3.5.

(BZ#1660051)

Заметные изменения в Apache httpd

RHEL 8 распространяется с сервером Apache HTTP 2.4.37. Эта версия вносит следующие изменения в **httpd** доступно в RHEL 7:

- Поддержка HTTP / 2 теперь предоставляется **mod_http2** пакет, который является частью **httpd** модуль.

- Автоматическое предоставление и обновление сертификатов TLS с использованием протокола Автоматической среды управления сертификатами (ACME) теперь поддерживается с **mod_md** пакет (для использования с поставщиками сертификатов, такими как **Let's Encrypt**)
- HTTP-сервер Apache теперь поддерживает загрузку сертификатов TLS и закрытых ключей с аппаратных токенов безопасности непосредственно из **PKCS#11** модули. В результате **mod_ssl** Конфигурация теперь может использовать **PKCS#11** URL-адреса для идентификации закрытого ключа TLS и, необязательно, сертификата TLS в **SSLCertificateKeyFile** а также **SSLCertificateFile** директивы.
- Модуль многопроцессорной обработки (MPM), сконфигурированный по умолчанию с HTTP-сервером Apache, изменился по сравнению с многопроцессорной раздвоенной моделью (известной как **prefork**) высокопроизводительной многопоточной модели, **event** Любые сторонние модули, которые не являются поточно-ориентированными, необходимо заменить или удалить. Чтобы изменить настроенный MPM, отредактируйте **/etc/httpd/conf.modules.d/00-mpm.conf** файл. Увидеть **httpd.conf(5)** Страница man для получения дополнительной информации.

Для получения дополнительной информации о **httpd**, увидеть [Setting up the Apache HTTP web server](#)

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

nginx веб-сервер новый в RHEL 8

RHEL 8 вводит **nginx 1.14**, веб- и прокси-сервер, поддерживающий HTTP и другие протоколы, с акцентом на высокий параллелизм, производительность и низкое использование памяти. **nginx** Ранее был доступен только как коллекция программного обеспечения.

nginx веб-сервер теперь поддерживает загрузку приватных ключей TLS с аппаратных токенов безопасности непосредственно из **PKCS#11** модули. В результате **nginx** Конфигурация может использовать **PKCS#11** URL-адреса для идентификации закрытого ключа TLS в **ssl_certificate_key** директивы.

(BZ#1545526)

Серверы баз данных в RHEL 8

RHEL 8 предоставляет следующие серверы баз данных:

- **MySQL 8.0** многопоточный многопоточный сервер баз данных SQL. Состоит из **MySQL** серверный демон, **mysqld**, и много клиентских программ.
- **MariaDB 10.3** многопоточный многопоточный сервер баз данных SQL. Для всех практических целей, **MariaDB** двоично-совместим с **MySQL**
- **PostgreSQL 10** а также **PostgreSQL 9.6**, усовершенствованная система управления объектно-реляционными базами данных (СУБД).
- **Redis 5**, расширенный магазин значения ключа. Его часто называют сервером структуры данных, поскольку ключи могут содержать строки, хэши, списки, наборы и отсортированные наборы. **Redis** предоставляется впервые в RHEL.

Обратите внимание, что NoSQL **MongoDB** сервер базы данных не включен в RHEL 8.0, поскольку он использует публичную лицензию на стороне сервера (SSPL).

(BZ#1647908)

Заметные изменения в MySQL 8.0

RHEL 8 распространяется с **MySQL 8.0**, который обеспечивает, например, следующие улучшения:

- **MySQL** теперь включает словарь транзакционных данных, в котором хранится информация об объектах базы данных.
- **MySQL** теперь поддерживает роли, которые являются коллекциями привилегий.
- Набор символов по умолчанию был изменен с **latin1** в **utf8mb4**
- Добавлена поддержка общих табличных выражений, как нерекурсивных, так и рекурсивных.
- **MySQL** теперь поддерживает оконные функции, которые выполняют вычисления для каждой строки из запроса, используя связанные строки.
- **InnoDB** теперь поддерживает **NOWAIT** а также **SKIP LOCKED** варианты с блокировкой операторов чтения.
- ГИС-функции были улучшены.
- Функциональность JSON была улучшена.
- Новый **mariadb-connector-c** пакеты предоставляют общую клиентскую библиотеку для **MySQL** а также **MariaDB**. Эта библиотека может использоваться с любой версией **MySQL** а также **MariaDB** серверы баз данных. В результате пользователь может подключить одну сборку приложения к любому из **MySQL** а также **MariaDB** серверы распространяются с RHEL 8.

В дополнение **MySQL 8.0** сервер, распространяемый с помощью RHEL 8, настроен на использование **mysql_native_password** как подключаемый модуль аутентификации по умолчанию, потому что клиентские инструменты и библиотеки в RHEL 8 несовместимы с **caching_sha2_password** метод, который используется по умолчанию в восходящем потоке **MySQL 8.0** версия.

Чтобы изменить подключаемый модуль аутентификации по умолчанию на **caching_sha2_password**, отредактируйте **/etc/my.cnf.d/mysql-default-authentication-plugin.cnf** файл следующим образом:

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

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

Заметные изменения в MariaDB 10.3

MariaDB 10.3 предоставляет множество новых функций по сравнению с версией 5.5, распространяемой в RHEL 7. Некоторые из наиболее заметных изменений:

- **MariaDB Galera Cluster**, синхронный мультимастерный кластер, теперь является стандартной частью **MariaDB**
- **InnoDB** используется как механизм хранения по умолчанию вместо **XtraDB**
- Общие табличные выражения
- Системные версии таблиц
- **FOR** петли
- Невидимые колонны
- Последовательности
- Мгновенный **ADD COLUMN** за **InnoDB**
- Компрессия колонны, не зависящая от системы хранения
- Параллельная репликация
- Репликация из нескольких источников

Кроме того, новый **mariadb-connector-c** пакеты предоставляют общую клиентскую библиотеку для **MySQL** а также **MariaDB** Эта библиотека может использоваться с любой версией **MySQL** а также **MariaDB** серверы баз данных. В результате пользователь может подключить одну сборку приложения к любому из **MySQL** а также **MariaDB** серверы распространяются с RHEL 8.

Смотрите также [Using MariaDB on Red Hat Enterprise Linux 8](#)

(BZ#1637034, BZ#1519450)

4.8. РАБОЧИЙ СТОЛ (МАШИННЫЙ ПЕРЕВОД)

Оболочка GNOME, версия 3.28 в RHEL 8

Оболочка GNOME версии 3.28 доступна в Red Hat Enterprise Linux (RHEL) 8. Заметные улучшения включают в себя:

- Новые возможности GNOME Boxes
- Новая экранная клавиатура
- Расширенная поддержка устройств, наиболее значительная интеграция для интерфейса Thunderbolt 3
- Улучшения для программного обеспечения GNOME, dconf-editor и терминала GNOME

(BZ#1649404)

Wayland сервер отображения по умолчанию

В Red Hat Enterprise Linux 8 используется сеанс GNOME и диспетчер дисплеев GNOME (GDM) **Wayland** в качестве сервера отображения по умолчанию вместо **X.org** сервер, который использовался в предыдущей основной версии RHEL.

Wayland обеспечивает множество преимуществ и улучшений по сравнению с **X.org**. Наиболее заметно:

- Более сильная модель безопасности
- Улучшенная обработка нескольких мониторов
- Улучшено масштабирование пользовательского интерфейса (UI)
- Рабочий стол может напрямую управлять обработкой окон.

Обратите внимание, что следующие функции в настоящее время недоступны или не работают должным образом:

- Настройки нескольких графических процессоров не поддерживаются **Wayland**
- **NVIDIA** бинарный драйвер не работает под **Wayland**
- **xrandr** Утилита не работает под **Wayland** из-за его другого подхода к обработке, разрешениям, поворотам и макету. Обратите внимание, что другие **X.org** утилиты для работы с экраном не работают под **Wayland**, или.
- Запись экрана, удаленный рабочий стол и специальные возможности не всегда работают правильно **Wayland**
- Менеджер буфера обмена недоступен.
- **Wayland** игнорирует захваты клавиатуры, создаваемые приложениями X11, такими как средства просмотра виртуальных машин.
- **Wayland** Внутри гостевых виртуальных машин (ВМ) есть проблемы со стабильностью и производительностью, поэтому рекомендуется использовать сеанс X11 для виртуальных сред.

Если вы обновитесь до RHEL 8 из системы RHEL 7, в которой вы использовали **X.org** Сессия GNOME, ваша система продолжает использовать **X.org**. Система также автоматически возвращается к **X.org** когда используются следующие графические драйверы:

- Двоичный драйвер **NVIDIA**
- **cirrus** Водитель
- **mga** Водитель
- **aspeed** Водитель

Вы можете отключить использование **Wayland** вручную:

- Отключить **Wayland** в **GDM**, установить **WaylandEnable=false** вариант в **/etc/gdm/custom.conf** файл.
- Отключить **Wayland** в сеансе GNOME выберите устаревшую опцию X11, используя меню зубчатого колеса на экране входа в систему после ввода имени пользователя.

Для более подробной информации о **Wayland**, увидеть <https://wayland.freedesktop.org/>

(BZ#1589678)

Обнаружение RPM-пакетов, которые находятся в репозиториях, не включены по умолчанию

Дополнительные репозитории для рабочего стола не включены по умолчанию. Отключение обозначается **enabled=0** строка в соответствующем **.repo** файл. Если вы попытаетесь установить пакет из такого репозитория с помощью **PackageKit**, **PackageKit** отобразит сообщение об ошибке, сообщающее, что приложение недоступно. Чтобы сделать пакет доступным, замените ранее использованный **enabled=0** строка в соответствующем **.repo** файл с **enabled=1**

(JIRA:RHELPLAN-2878)

GNOME Software для управления пакетами

gnome-packagekit Пакет, предоставляющий набор инструментов для управления пакетами в графической среде в Red Hat Enterprise Linux 7, больше не доступен. В Red Hat Enterprise Linux 8 аналогичные функциональные возможности предоставляются **Программное обеспечение GNOME** утилита, которая позволяет устанавливать и обновлять приложения и расширения оболочки **gnome**. **Программное обеспечение GNOME** распространяется в **gnome-software** пакет.

(JIRA:RHELPLAN-3001)

4.9. УПРАВЛЕНИЕ ИДЕНТИФИКАЦИЕЙ (МАШИННЫЙ ПЕРЕВОД)

Проверка нового синтаксиса пароля на сервере каталогов

Это усовершенствование добавляет новые проверки синтаксиса паролей на сервер каталогов. Администраторы теперь могут, например, включить проверку словаря, разрешить или запретить использование последовательностей символов и палиндромов. В результате, если этот параметр включен, проверка синтаксиса политики паролей на сервере каталогов обеспечивает более безопасные пароли.

(BZ#1334254)

Сервер каталогов теперь обеспечивает улучшенную поддержку ведения журнала внутренних операций

Некоторые операции на сервере каталогов, инициированные сервером и клиентами, вызывают дополнительные операции в фоновом режиме. Ранее сервер регистрировал только внутренние операции **Internal** ключевое слово подключения, а идентификатор операции всегда был установлен на **-1** Благодаря этому усовершенствованию **Directory Server** регистрирует реальное соединение и идентификатор операции. Теперь вы можете отследить внутреннюю операцию на сервере или клиентской операции, которая вызвала эту операцию.

Дополнительные сведения о ведении журнала внутренних операций см. По ссылке: https://access.redhat.com/documentation/en-us/red_hat_directory_server/11/html-single/administration_guide/#logging_internal_operations.

(BZ#1358706)

tomcatjss библиотека поддерживает проверку OCSP с использованием респондента из расширения AIA

С этим улучшением **tomcatjss** библиотека поддерживает проверку протокола состояния сертификата в сети (OCSP) с использованием ответчика из расширения сертификата доступа к информации (AIA). В результате администраторы Red Hat Certificate System теперь могут настроить проверку OCSP, которая использует URL-адрес из расширения AIA.

(BZ#1636564)

Сервер каталогов представляет новые утилиты командной строки для управления экземплярами

Red Hat Directory Server 11.0 представляет **dscreate**, **dsconf**, а также **dsctl** коммунальные услуги. Эти утилиты упрощают управление сервером каталогов с помощью командной строки. Например, теперь вы можете использовать команду с параметрами для настройки функции вместо отправки сложных операторов LDIF на сервер.

Ниже приведен обзор назначения каждой утилиты:

- Использовать **dscreate** утилита для создания новых экземпляров сервера каталогов с использованием интерактивного режима или файла INF. Обратите внимание, что формат файла INF отличается от того, который установщик использовал в предыдущих версиях Сервера каталогов.
- Использовать **dsconf** утилита для управления экземплярами сервера каталогов во время выполнения. Например, используйте **dsconf** чтобы:
 - Настройте параметры в **cn=config** запись
 - Настроить плагины
 - Настроить репликацию
 - Резервное копирование и восстановление экземпляра
- Использовать **dsctl** утилита для управления экземплярами сервера каталогов, когда они в автономном режиме. Например, используйте **dsctl** чтобы:
 - Запустить и остановить экземпляр
 - Переиндексировать базу данных сервера
 - Резервное копирование и восстановление экземпляра

Эти утилиты заменяют сценарии Perl и оболочки, помеченные как устаревшие в Directory Server 10. Скрипты все еще доступны в неподдерживаемых **389-ds-base-legacy-tools** пакет, однако Red Hat поддерживает управление сервером каталогов только с помощью новых утилит.

Обратите внимание, что настройка сервера каталогов с помощью инструкций LDIF все еще поддерживается, но Red Hat рекомендует использовать утилиты.

Для получения дополнительной информации об использовании утилит см. [Red Hat Directory Server 11 Documentation](#)

(BZ#1693159)

pki subsystem-cert-find а также pki subsystem-cert-show Команды теперь показывают серийный номер сертификатов

С этим улучшением **pki subsystem-cert-find** а также **pki subsystem-cert-show** Команды в Системе сертификатов показывают серийный номер сертификатов в их выходных данных. Серийный номер является важной частью информации и часто требуется для нескольких других команд. В результате определить серийный номер сертификата теперь проще.

(BZ#1566360)

pki user а также pki group Команды устарели в системе сертификатов

С этим обновлением новый **pki <subsystem>-user** а также **pki <subsystem>-group** команды заменяют **pki user** а также **pki group** Команды в Системе Сертификации. Замененные команды по-прежнему работают, но они отображают сообщение о том, что команда устарела, и ссылаются на новые команды.

(BZ#1394069)

Система сертификатов теперь поддерживает автономное обновление системных сертификатов.

Благодаря этому усовершенствованию администраторы могут использовать функцию автономного обновления для обновления системных сертификатов, настроенных в системе сертификатов. Когда срок действия системного сертификата истекает, система сертификатов не запускается. В результате усовершенствования администраторам больше не нужны обходные пути для замены устаревшего системного сертификата.

(BZ#1669257)

Система сертификатов теперь может создавать CSR с расширением SKI для подписи внешнего СА

Благодаря этому усовершенствованию Система сертификатов поддерживает создание запроса на подпись сертификата (CSR) с расширением Subject Key Identifier (SKI) для подписи внешнего центра сертификации (CA). Некоторым ЦС требуется это расширение либо с определенным значением, либо из открытого ключа ЦС. В результате администраторы теперь могут использовать **pki_req_ski** параметр в файле конфигурации передается в **pkispawn** утилита для создания CSR с расширением SKI.

(BZ#1656856)

Локальные пользователи кэшируются SSSD и обслуживаются через nss_sss модуль

В RHEL 8 демон служб безопасности системы (SSSD) обслуживает пользователей и группы из **/etc/passwd** а также **/etc/groups** файлы по умолчанию. **sss** Модуль **nsswitch** предшествует файлам в **/etc/nsswitch.conf** файл.

Преимущество обслуживания локальных пользователей через SSSD состоит в том, что **nss_sss** модуль имеет быстрый **memory-mapped cache** это ускоряет поиск NSS по сравнению с доступом к диску и открытием файлов при каждом запросе NSS. Ранее демон кэширования службы имен (**nscd**) помог ускорить процесс доступа к диску. Однако, используя **nscd** параллельно с SSSD является громоздким, так как и SSSD, и **nscd** использовать собственное независимое кэширование. Следовательно, используя **nscd** в установках, где SSSD также обслуживает пользователей из удаленного домена, например LDAP или Active Directory, может вызвать непредсказуемое поведение.

С этим обновлением разрешение локальных пользователей и групп в RHEL 8 стало

быстрее. Обратите внимание, что **root** пользователь никогда не обрабатывается SSSD, поэтому **root** на разрешение не может повлиять потенциальная ошибка в SSSD. Также обратите внимание, что если SSSD не работает, **nss_sss** модуль корректно обрабатывает ситуацию, возвращаясь к **nss_files** чтобы избежать проблем. Вам не нужно настраивать SSSD каким-либо образом, домен файлов добавляется автоматически.

(JIRA:RHELPLAN-10439)

KCM заменяет KEYRING в качестве хранилища кэша учетных данных по умолчанию

В RHEL 8 хранилищем кэша учетных данных по умолчанию является диспетчер учетных данных Kerberos (KCM), который поддерживается **sssd-kcm** Daemon. KCM преодолевает ограничения ранее использовавшегося KEYRING, такие как его сложность в использовании в контейнерах, поскольку он не имеет пространства имен, а также для просмотра и управления квотами.

В этом обновлении RHEL 8 содержит кэш учетных данных, который лучше подходит для контейнерных сред и обеспечивает основу для создания дополнительных функций в будущих выпусках.

(JIRA:RHELPLAN-10440)

Пользователи Active Directory теперь могут администрировать Identity Management

В этом обновлении RHEL 8 позволяет добавить переопределение идентификатора пользователя для пользователя Active Directory (AD) в качестве члена группы Identity Management (IdM). Переопределение идентификатора - это запись, описывающая, как должны выглядеть свойства конкретного пользователя или группы AD в определенном представлении идентификатора, в данном случае представлении доверия по умолчанию. В результате обновления сервер LDAP IdM может применять правила контроля доступа для группы IdM к пользователю AD.

Пользователи AD теперь могут использовать функции самообслуживания IdM UI, например, для загрузки своих ключей SSH или изменения своих личных данных. Администратор AD может полностью администрировать IdM, не имея двух разных учетных записей и паролей. Обратите внимание, что в настоящее время выбранные функции в IdM могут быть недоступны для пользователей AD.

(JIRA:RHELPLAN-10442)

ssscil печатает отчет по правилам HBAC для домена IdM

С этим обновлением **ssscil** утилита System Security Services Daemon (SSSD) может распечатать отчет об управлении доступом для домена Identity Management (IdM). Эта функция отвечает требованиям определенных сред для просмотра по нормативным причинам списка пользователей и групп, которые могут получить доступ к конкретному клиентскому компьютеру. Бег **ssscil access-report domain_name** на клиенте IdM печатает проанализированное подмножество правил управления доступом на основе хоста (HBAC) в домене IdM, которые применяются к клиентскому компьютеру.

Обратите внимание, что никакие другие поставщики, кроме IdM, не поддерживают эту функцию.

(JIRA:RHELPLAN-10443)

Пакеты Identity Management доступны в виде модуля

В RHEL 8 пакеты, необходимые для установки сервера и клиента Identity Management (IdM), поставляются в виде модуля. **client** поток является потоком по умолчанию **idm** модуль, и вы можете скачать пакеты, необходимые для установки клиента, не включая поток.

Поток модуля сервера IdM называется **DL1** поток. Поток содержит несколько профилей, соответствующих различным типам серверов IdM: сервер, DNS, Adtrust, клиент и по умолчанию. Для загрузки пакетов в конкретном профиле **DL1** поток: . Включить поток. , Переключитесь на RPM, доставленные через поток. , Запустить **yum module install idm:DL1/profile_name** команда.

(JIRA:RHELPLAN-10438)

Добавлено решение для записи сессии для RHEL 8

Решение для записи сеансов было добавлено в Red Hat Enterprise Linux 8 (RHEL 8). Новый **tlog** Пакет и связанный с ним проигрыватель сеансов веб-консоли позволяют записывать и воспроизводить сеансы пользовательского терминала. Запись может быть настроена для каждого пользователя или группы пользователей с помощью службы System Security Services Daemon (SSSD). Весь ввод и вывод терминала фиксируется и сохраняется в текстовом формате в системном журнале. Ввод неактивен по умолчанию из соображений безопасности, чтобы не перехватывать необработанные пароли и другую конфиденциальную информацию.

Решение может быть использовано для аудита пользовательских сессий в системах, чувствительных к безопасности. В случае нарушения безопасности записанные сеансы могут быть рассмотрены как часть криминалистического анализа. Системные администраторы теперь могут настраивать запись сеанса локально и просматривать результат через интерфейс веб-консоли RHEL 8 или через интерфейс командной строки, используя **tlog-play** полезность.

(JIRA:RHELPLAN-1473)

authselect упрощает настройку аутентификации пользователя

Это обновление представляет **authselect** утилита, которая упрощает настройку аутентификации пользователя на хостах RHEL 8, заменяя **authconfig** полезность. **authselect** поставляется с более безопасным подходом к управлению стеком PAM, который упрощает изменения конфигурации PAM для системных администраторов. **authselect** может использоваться для настройки методов аутентификации, таких как пароли, сертификаты, смарт-карты и отпечатки пальцев. Обратите внимание, что **authselect** не настраивает службы, необходимые для подключения к удаленным доменам. Эта задача выполняется специализированными инструментами, такими как **realmd** или же **ipa-client-install**

(JIRA:RHELPLAN-10445)

SSSD теперь позволяет вам выбрать одно из нескольких устройств аутентификации Smartcard

Это обновление позволяет настроить URI PKCS # 11 для выбора устройств проверки подлинности с помощью смарт-карты.

По умолчанию SSSD пытается автоматически определить устройство для проверки подлинности с помощью смарт-карты. Если подключено несколько устройств, SSSD выберет первое найденное, и вы не сможете выбрать конкретное устройство. Это

может привести к сбоям.

Таким образом, теперь вы можете настроить новый **p11_uri** вариант для **[pam]** раздел **sssd.conf** Эта опция позволяет вам определить, какое устройство будет использоваться для аутентификации по смарт-карте.

Например, чтобы выбрать считыватель с идентификатором слота '2', обнаруженным модулем OpenSC PKCS # 11, добавьте

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

к **[pam]** раздел **sssd.conf**

Посмотри пожалуйста **man sssd-conf** для деталей.

(BZ#1620123)

4.10. КОМПИЛЯТОРЫ И ИНСТРУМЕНТЫ РАЗРАБОТКИ (МАШИННЫЙ ПЕРЕВОД)

Boost обновлен до версии 1.66

Увеличение Библиотека C ++ была обновлена до вышестоящей версии 1.66. Версия **Увеличение** в Red Hat Enterprise Linux 7 включено 1,53. Для получения дополнительной информации см. Вышеприведенные журналы изменений:

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

Это обновление представляет следующие изменения, нарушающие совместимость с предыдущими версиями:

- **bs_set_hook()** функция, **splay_set_hook()** функция от **splay**-контейнеров, и **bool splay = true** дополнительный параметр в **splaytree_algorithms()** функция в **назойливый** Библиотека была удалена.
- Комментарии или конкатенация строк в файлах JSON больше не поддерживаются анализатором в **Дерево свойств** библиотека.
- Некоторые распределения и специальные функции из **математический** библиотека была исправлена, чтобы вести себя как задокументировано и поднять **overflow_error** вместо того, чтобы возвращать максимальное конечное значение.
- Некоторые заголовки из **математический** библиотека была перемещена в каталог **libs/math/include_private**
- Поведение **basic_regex<>::mark_count()** а также **basic_regex<>::subexpression(n)** функции от **Regex** библиотека была изменена, чтобы соответствовать их документации.
- Использование вариационных шаблонов в **Вариант** библиотека может нарушать функции метапрограммирования.
- **boost::python::numeric** API был удален. Пользователи могут использовать **boost::python::numpy** вместо.

- Арифметические операции с указателями на необъектные типы больше не предоставляются в библиотеке Atomic.

(BZ#1494495)

Поддержка Unicode 11.0.0

Библиотека ядра C Red Hat Enterprise Linux, **Glibc**, был обновлен для поддержки стандартной версии Unicode 11.0.0. В результате все API широких и многобайтовых символов, включая транслитерацию и преобразование между наборами символов, предоставляют точную и правильную информацию, соответствующую этому стандарту.

(BZ#1512004)

boost пакет теперь не зависит от Python

С этим обновлением установка **boost** пакет больше не устанавливает **Boost.Python** библиотека как зависимость. Для того, чтобы использовать **Boost.Python**, необходимо явно установить **boost-python3** или же **boost-python3-devel** пакеты.

(BZ#1616244)

Новый compat-libgfortran-48 пакет доступен

Для совместимости с приложениями Red Hat Enterprise Linux 6 и 7, использующими библиотеку Fortran, новый **compat-libgfortran-48** пакет совместимости теперь доступен, который обеспечивает **libgfortran.so.3** библиотека.

(BZ#1607227)

Ретполиновая поддержка в GCC

Это обновление добавляет поддержку **retpolines** для GCC. Retpoline - это программная конструкция, используемая ядром для снижения накладных расходов на смягчение атак Spectre Variant 2, описанных в CVE-2017-5715.

(BZ#1535774)

Расширенная поддержка 64-битной архитектуры ARM в компонентах цепочки инструментов

Компоненты Toolchain, **GCC** а также **binutils**, Теперь предоставьте расширенную поддержку для 64-битной архитектуры ARM. Например:

- **GCC** а также **binutils** теперь поддерживает масштабируемое векторное расширение (SVE).
- Поддержка для **FP16** тип данных, предоставленный ARM v8.2, был добавлен в **GCC FP16** Тип данных улучшает производительность определенных алгоритмов.
- Инструменты из **binutils** теперь поддерживает определение архитектуры ARM v8.3, включая Pointer Authentication. Функция аутентификации указателя предотвращает нарушение нормального выполнения программы или ядра вредоносным кодом, создавая собственные указатели функций. В результате при переходе к разным местам в коде используются только доверенные адреса, что повышает безопасность.

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

Оптимизации для glibc для систем IBM POWER

Это обновление предоставляет новую версию **glibc** это оптимизировано для архитектур IBM POWER 8 и IBM POWER 9. В результате системы IBM POWER 8 и IBM POWER 9 теперь автоматически переключаются на соответствующие оптимизированные **glibc** вариант во время выполнения.

(BZ#1376834)

Библиотека GNU C обновлена до версии 2.28

Red Hat Enterprise Linux 8 включает версию 2.28 библиотеки GNU C (glibc). Заметные улучшения включают в себя:

- Особенности укрепления безопасности:
 - Безопасные двоичные файлы, помеченные **AT_SECURE** флаг игнорировать **LD_LIBRARY_PATH** переменная окружения.
 - Обратные следы больше не печатаются при сбое проверки стека, чтобы ускорить завершение работы и избежать запуска большего количества кода в скомпрометированной среде.
- Улучшения производительности:
 - Производительность **malloc()** функция была улучшена за счет локального кэша потока.
 - Дополнение **GLIBC_TUNABLES** переменная среды для изменения характеристик производительности библиотеки.
 - Реализация потоковых семафоров была улучшена и масштабируема **pthread_rwlock_xxx()** функции были добавлены.
 - Производительность математической библиотеки была улучшена.
- Добавлена поддержка Unicode 11.0.0.
- Добавлена улучшенная поддержка 128-битных чисел с плавающей запятой, как это определено стандартами ISO / IEC / IEEE 60559: 2011, IEEE 754-2008 и ISO / IEC TS 18661-3: 2015.
- Улучшения решателя заглушки службы доменных имен (DNS), связанные с **/etc/resolv.conf** конфигурационный файл:
 - Конфигурация автоматически перезагружается при изменении файла.
 - Добавлена поддержка произвольного количества поисковых доменов.
 - Правильный случайный выбор для **rotate** опция была добавлена.
- Добавлены новые возможности для разработки, в том числе:
 - Функции обёртки Linux для **preadv2** а также **pwritev2** вызовы ядра
 - Новые функции, в том числе **reallocarray()** а также **explicit_bzero()**
 - Новые флаги для **posix_spawnattr_setflags()** функция, такая как **POSIX_SPAWN_SETSID**

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

СMake доступен в RHEL

Система сборки СMake версии 3.11 доступна в Red Hat Enterprise Linux 8 как **cmake** пакет.

(BZ#1590139, BZ#1502802)

make версия 4.2.1

Red Hat Enterprise Linux 8 распространяется с **make** Версия инструмента сборки 4.2.1. Заметные изменения включают в себя:

- В случае неудачи рецепта отображаются имя файла сборки и номер строки рецепта.
- **--trace** добавлена опция для отслеживания целей. Когда используется эта опция, каждый рецепт печатается перед вызовом, даже если он будет подавлен, вместе с именем файла и номером строки, где расположен этот рецепт, а также с предпосылками, вызывающими его вызов.
- Смешивание явных и неявных правил больше не вызывает **make** прекратить исполнение. Вместо этого выводится предупреждение. Обратите внимание, что этот синтаксис устарел и может быть полностью удален в будущем.
- **\$(file ...)** была добавлена функция записи текста в файл. Когда вызывается без текстового аргумента, он только открывает и сразу закрывает файл.
- Новая опция, **--output-sync** или же **-O**, Позволяет сгруппировать выходные данные из нескольких заданий и упростить отладку параллельных сборок.
- **--debug** опция теперь принимает также **n** (Нет) флаг, чтобы отключить все в настоящее время включены параметры отладки.
- **!=** оператор присваивания оболочки был добавлен в качестве альтернативы **\$(shell ...)** функция для повышения совместимости с make-файлами BSD. Более подробную информацию и различия между оператором и функцией см. В руководстве по GNU make.

Обратите внимание, что, как следствие, переменные с именем, оканчивающимся на восклицательный знак и сразу после которого присваивается, например, **variable!=value**, теперь интерпретируются как новый синтаксис. Чтобы восстановить предыдущее поведение, добавьте пробел после восклицательного знака, например **variable! =value**

+

- **::=** добавлен оператор присваивания, определенный стандартом POSIX.
- Когда **.POSIX** переменная указана, **make** соблюдает стандартные требования POSIX для обработки обратной косой черты и новой строки. В этом режиме все завершающие пробелы до обратной косой черты сохраняются, и каждая обратная косая черта, за которой следует новая строка и символы пробела, преобразуется в один пробел.
- Поведение **MAKEFLAGS** а также **MFLAGS** переменные теперь более точно определены.

- Новая переменная, **GNUMAKEFLAGS**, анализируется для **make** флаги идентично **MAKEFLAGS**. Как следствие, GNU **make**-специфичные флаги могут храниться снаружи **MAKEFLAGS** и переносимость **make**-файлов увеличивается.
- Новая переменная, **MAKE_HOST**, содержащий архитектуру хоста.
- Новые переменные, **MAKE_TERMOUT** а также **MAKE_TERMERR**, укажите, **make** записывает стандартный вывод и ошибку в терминал.
- Настройка **-r** а также **-R** варианты в **MAKEFLAGS** переменная внутри **make**-файла теперь работает правильно и удаляет все встроенные правила и переменные, соответственно.
- **.RECIPEPREFIX** настройка теперь запоминается по рецепту. Кроме того, переменные, раскрытые в этом рецепте, также используют эту настройку префикса рецепта.
- **.RECIPEPREFIX** настройки и все переменные, специфичные для цели, отображаются в выходных данных **-p** вариант как будто в **make**-файле, а не как комментарии.

(BZ#1641015)

Go программы, созданные с Go Toolset соответствуют FIPS

Криптографическая библиотека доступна в Go Toolset был изменен, чтобы использовать OpenSSL версия библиотеки 1.1.0, если хост-система настроена в режиме FIPS. Как следствие, программы, созданные с этой версией Go Toolset соответствуют FIPS.

Чтобы программы Go использовали только несертифицированные стандартные криптографические процедуры, используйте **-tags no_openssl** вариант **go** компилятор во время сборки.

(BZ#1512570)

SystemTap версия 4.0

Red Hat Enterprise Linux 8 распространяется с SystemTap инструментальный инструмент версии 4.0. Заметные улучшения включают в себя:

- Был улучшен расширенный бэкэнд Berkeley Packet Filter (eBPF), особенно строки и функции. Чтобы использовать этот бэкэнд, запустите SystemTap с **--runtime=bpf** вариант.
- Добавлена новая служба экспортной сети для использования с системой мониторинга Prometheus.
- Реализация проверки системного вызова была улучшена, чтобы при необходимости использовать точки трассировки ядра.

(BZ#1641032)

Улучшения в binutils версия 2.30

Red Hat Enterprise Linux 8 включает в себя версию 2.30 **binutils** пакет. Заметные улучшения включают в себя:

- Улучшена поддержка новых расширений архитектуры s390x.

Ассемблер:

- Добавлена поддержка формата файлов WebAssembly и преобразование WebAssembly в формат файлов ELF wasm32.
- Добавлена поддержка архитектуры ARMv8-R и процессоров Cortex-R52, Cortex-M23 и Cortex-M33.
- Добавлена поддержка архитектуры RISC-V.

Linkers:

- Компоновщик теперь по умолчанию помещает код и данные только для чтения в отдельные сегменты. В результате созданные исполняемые файлы становятся больше и безопаснее в работе, поскольку динамический загрузчик может отключить выполнение любой страницы памяти, содержащей данные только для чтения.
- Добавлена поддержка примечаний к свойствам GNU, которые предоставляют подсказки динамическому загрузчику о бинарном файле.
- Ранее компоновщик генерировал недопустимый исполняемый код для технологии Intel Indirect Branch Tracking (IBT). Как следствие, сгенерированные исполняемые файлы не могут запуститься. Эта ошибка была исправлена.
- Ранее **gold** компоновщик неправильно слил заметки о недвижимости. Как следствие, в сгенерированном коде могут быть включены неправильные аппаратные функции, и код может неожиданно завершить работу. Эта ошибка была исправлена.
- Ранее **gold** компоновщик создал разделы заметок с байтами заполнения в конце для достижения выравнивания в соответствии с архитектурой. Поскольку динамический загрузчик не ожидал заполнения, он мог неожиданно завершить программу, которую он загружал. Эта ошибка была исправлена.

Другие инструменты:

- **readelf** а также **objdump** У инструментов теперь есть возможность переходить по ссылкам в отдельные файлы информации об отладке и отображать в них информацию.
- Новый **--inlines** опция расширяет существующие **--line-numbers** вариант **objdump** инструмент для отображения информации о вложенности для встроенных функций.
- **nm** инструмент получил новую опцию **--with-version-strings** отображать информацию о версии символа после его имени, если оно есть.

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

Второй пилот Performace версия 4.1.3

Red Hat Enterprise Linux 8 распространяется с **Производительность второго пилота** (psr) версия 4.1.3. Заметные улучшения включают в себя:

- **pcp-dstat** инструмент теперь включает в себя исторический анализ и вывод в формате с разделителями-запятыми (CSV).
- Утилиты журнала могут использовать метрические метки и текстовые записи справки.
- **pmdaperfevent** Теперь инструмент сообщает правильные номера ЦП на нижних уровнях одновременной многопоточности (SMT).
- **pmdapostgresql** инструмент теперь поддерживает Postgres серия 10.x.
- **pmdaredis** инструмент теперь поддерживает Redis серия 5.x.
- **pmdabcc** инструмент был улучшен динамической фильтрацией процессов и системными вызовами, ucalls и ustat для каждого процесса.
- **pmdammv** Инструмент теперь экспортирует метрические метки, а формат версии увеличен до 3.
- **pmdagfs2** Инструмент поддерживает дополнительные метрики Глок и Глокодержатель.
- Внесено несколько исправлений в политику SELinux.

(BZ#1641034)

Ключи защиты памяти

Это обновление включает аппаратные функции, которые позволяют изменять флаг защиты страниц для каждого потока. Новый **glibc** были добавлены оболочки системных вызовов для **pkey_alloc()**, **pkey_free()**, а также **pkey_mprotect()** функции. В дополнение **pkey_set()** а также **pkey_get()** были добавлены функции, позволяющие получить доступ к флагам защиты для каждого потока.

(BZ#1304448)

elfutils обновлено до версии 0.174

В Red Hat Enterprise Linux 8 **elfutils** Пакет доступен в версии 0.174. Заметные изменения включают в себя:

- Ранее **eu-readelf** Инструмент может показывать переменную с отрицательным значением, как если бы она имела большое значение без знака, или показывать большое значение без знака как отрицательное значение. Это было исправлено и **eu-readelf** теперь ищет размер и подпись типов константных значений для правильного их отображения.
- Новая функция **dwarf_next_lines()** для чтения **.debug_line** данные без CU были добавлены в **libdw** библиотека. Эта функция может использоваться как альтернатива **dwarf_getsrclines()** а также **dwarf_getsrcfiles()** функции.
- Ранее файлы с более чем 65280 разделами могли вызывать ошибки в **libelf** а также **libdw** библиотеки и все инструменты, использующие их. Эта ошибка была исправлена. В результате продлен **shnum** а также **shstrndx** значения в заголовках файлов ELF обрабатываются правильно.

(BZ#1641007)

Valgrind обновлен до версии 3.14

Red Hat Enterprise Linux 8 распространяется с инструментом анализа исполняемого кода Valgrind версии 3.14. Заметные изменения включают в себя:

- Новый **--keep-debuginfo** добавлена возможность сохранения отладочной информации для выгруженного кода. В результате сохраненные трассировки стека могут включать информацию о файлах и строках для кода, которого больше нет в памяти.
- Добавлены подавления на основе имени исходного файла и номера строки.
- **Helgrind** инструмент был расширен с опцией **--delta-stacktrace** указать вычисление полных следов стека истории. В частности, используя эту опцию вместе с **--history-level=full** может улучшить **Helgrind** производительность до 25%.
- Ложноположительный показатель в **Memcheck** инструмент для оптимизированного кода на 64-битной архитектуре Intel и AMD, а также на 64-битной архитектуре ARM. Обратите внимание, что вы можете использовать **--expensive-definedness-checks** контролировать обработку проверок определенности и улучшать скорость за счет производительности.
- Valgrind теперь может распознавать больше инструкций в младшей последовательности IBM Power Systems.
- Valgrind теперь может частично обрабатывать целочисленные и строковые векторные инструкции процессора IBM Z архитектура z13.

Для получения дополнительной информации о новых опциях и их известных ограничениях см. **valgrind(1)** страница справочника.

(BZ#1641029, BZ#1501419)

GDB версия 8.2

Red Hat Enterprise Linux 8 распространяется с отладчиком GDB версии 8.2. К заметным изменениям относятся:

- Протокол IPv6 поддерживается для удаленной отладки с помощью GDB и **gdbserver**
- Отладка без отладочной информации была улучшена.
- Завершение символов в пользовательском интерфейсе GDB было улучшено, чтобы предлагать лучшие предложения с использованием более синтаксических конструкций, таких как теги ABI или пространства имен.
- Команды теперь могут выполняться в фоновом режиме.
- Отладка программ, созданных на языке программирования Rust, теперь возможна.
- Отладка языков C и C++ была улучшена благодаря поддержке парсера для **_Alignof** а также **alignof** операторы, ссылки на значения C++ и автоматические массивы C99 переменной длины.
- Сценарии расширения GDB теперь могут использовать язык сценариев Guile.

- Интерфейс языка сценариев Python для расширений был улучшен благодаря новым функциям API, декораторам кадров, фильтрам и разматывателям. Кроме того, скрипты в `.debug_gdb_scripts` Раздел конфигурации GDB загружаются автоматически.
- GDB теперь использует Python версии 3 для запуска своих скриптов, в том числе симпатичных принтеров, декораторов кадров, фильтров и разматывателей.
- Архитектура ARM и 64-битная ARM были улучшены за счет записи и воспроизведения выполнения процесса, включая 32-битные инструкции Thumb и инструкции системного вызова.
- Добавлена поддержка реестра Intel MPX и нарушения привязки, реестра PKU и Intel Processor Trace.
- Функции записи и воспроизведения были расширены, чтобы включить `rdrand` а также `rdseed` инструкции по системам на базе Intel.
- Функциональность GDB в архитектуре IBM Z была расширена за счет поддержки точек трассировки и быстрых точек трассировки, векторных регистров и ABI, а также `Catch` системный вызов. Кроме того, GDB теперь поддерживает более свежие инструкции архитектуры.
- Теперь GDB может использовать статические зонды пространства пользователя (SDT) SystemTap в 64-битной архитектуре ARM.

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

Локализация для RHEL распространяется в нескольких пакетах

В RHEL 8 локали и переводы больше не предоставляются одним `glibc-common` пакет. Вместо этого каждая локаль и язык доступны в `glibc-langpack-CODE` пакет. Кроме того, не все локали установлены по умолчанию, только те, которые выбраны в программе установки. Пользователи должны установить все остальные языковые пакеты, которые им нужны, отдельно.

Для получения дополнительной информации об использовании langpacks см. [Installing and using langpacks](#)

(BZ#1512009)

strace обновлен до версии 4.24

Red Hat Enterprise Linux 8 распространяется с `strace` версия инструмента 4.24. Заметные изменения включают в себя:

- Функции подделки системных вызовов были добавлены с `-e inject=` вариант. Это включает введение ошибок, возвращаемых значений, задержек и сигналов.
- Квалификация системного вызова была улучшена:
 - `-e trace=/regex` добавлена опция для фильтрации системных вызовов с помощью регулярных выражений.
 - Добавление знака вопроса к квалификации системного вызова в `-e trace=` опция позволяет `strace` продолжить, даже если квалификация не соответствует ни одному системному вызову.

- В квалификации системных вызовов добавлено обозначение личности **-e trace** вариант.
- Расшифровка **kvm vcpu** причина выхода была добавлена. Для этого используйте **-e kvm=vcpu** вариант.
- **libdw** библиотека теперь используется для разматывания стека, когда **-k** опция используется. Кроме того, разделение символов возможно, когда **libiberty** библиотека установлена в системе.
- Ранее **-r** вызванный вариант **strace** игнорировать **-t** вариант. Это было исправлено, и эти два варианта теперь независимы.
- [Опция] **`-A** опция была добавлена для открытия выходных файлов в режиме добавления.
- **-X** опция была добавлена для настройки **xlat** форматирование вывода.
- Расшифровка адресов сокетов с помощью **-yy** опция была улучшена. Кроме того, блочная и символьная печать номера устройства в **-yy** режим был добавлен.

Кроме того, было добавлено, улучшено или обновлено декодирование следующих элементов:

- **netlink** протоколы, сообщения и атрибуты
- **arch_prctl, bpf, getsockopt, io_pgetevent, kern_features, keyctl, prctl, pkey_alloc, pkey_free, pkey_mprotect, ptrace, rseq, setsockopt, socket, statx** и другие системные вызовы
- Много команд для **ioctl** системный вызов
- Константы разных типов
- Трассировка пути для **execveat, inotify_add_watch, inotify_init, select, symlink, symlinkat** системные вызовы и **mmap** системные вызовы с косвенными аргументами
- Системные вызовы, специфичные для архитектур ARM **__ARM_NR_***
- Списки сигнальных кодов

(BZ#1641014)

4.11. ФАЙЛОВЫЕ СИСТЕМЫ И ХРАНИЛИЩЕ (МАШИННЫЙ ПЕРЕВОД)

XFS теперь поддерживает общие экстенды данных для копирования при записи

Файловая система XFS поддерживает функцию экстенда общих данных копирования при записи. Эта функция позволяет двум или более файлам совместно использовать общий набор блоков данных. При изменении любого из файлов, совместно использующих общие блоки, XFS разрывает ссылку на общие блоки и создает новый файл. Это похоже на функциональность копирования при записи (COW) в других файловых системах.

Общие экстенды данных для копирования при записи:

Быстро

Создание общих копий не использует дисковый ввод-вывод.

Пространственно-эффективный

Общие блоки не занимают дополнительного дискового пространства.

прозрачный

Общие файлы, разделяющие блоки, действуют как обычные файлы.

Пользовательские утилиты могут использовать общие экстенды данных для копирования при записи для:

- Эффективное клонирование файлов, например, с **cp --reflink** команда
- Снимки файлов

Эта функциональность также используется подсистемами ядра, такими как Overlayfs и NFS, для более эффективной работы.

Общие экстенды данных для копирования при записи теперь включены по умолчанию при создании файловой системы XFS, начиная с **xfsprogs** версия пакета **4.17.0-2.el8**

Обратите внимание, что устройства прямого доступа (DAX) в настоящее время не поддерживают XFS с общими экстендами копирования при записи. Чтобы создать файловую систему XFS без этой функции, используйте следующую команду:

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

Red Hat Enterprise Linux 7 может монтировать файловые системы XFS с общими экстендами копирования при записи только в режиме только для чтения.

(BZ#1494028)

Максимальный размер файловой системы XFS составляет 1024 ТиБ

Максимальный поддерживаемый размер файловой системы XFS был увеличен с 500 ТиБ до 1024 ТиБ.

Файловые системы размером более 500 ТиБ требуют, чтобы:

- функция CRC метаданных и функция бесплатного inode btree включены в формате файловой системы, и
- размер группы размещения составляет не менее 512 ГиБ.

В RHEL 8 **mkfs.xfs** Утилита создает файловые системы, отвечающие этим требованиям по умолчанию.

Увеличение файловой системы меньшего размера, не соответствующей этим требованиям, до нового размера, превышающего 500 ТиБ, не поддерживается.

(BZ#1563617)

VDO теперь поддерживает все архитектуры

Виртуальный оптимизатор данных (VDO) теперь доступен на всех архитектурах, поддерживаемых RHEL 8.

Список поддерживаемых архитектур смотрите [Глава 2, архитектуры \(машинный перевод\)](#)

(BZ#1534087)

Менеджер загрузки BOOM упрощает процесс создания загрузочных записей

BOOM - это менеджер загрузки для систем Linux, которые используют загрузчики, поддерживающие спецификацию BootLoader для конфигурации начальной загрузки. Это позволяет гибко настраивать загрузку и упрощает создание новых или измененных загрузочных записей: например, для загрузки снимков системы, созданных с использованием LVM.

BOOM не изменяет существующую конфигурацию загрузчика, а только добавляет дополнительные записи. Существующая конфигурация сохраняется, и любая интеграция с дистрибутивом, такая как сценарии установки и обновления ядра, продолжает функционировать, как и раньше.

BOOM имеет упрощенный интерфейс командной строки (CLI) и API, которые облегчают задачу создания загрузочных записей.

(BZ#1649582)

LUKS2 теперь формат по умолчанию для шифрования томов

В RHEL 8 формат LUKS версии 2 (LUKS2) заменяет устаревший формат LUKS (LUKS1). **dm-crypt** подсистема и тому **cryptsetup** Инструмент теперь использует LUKS2 в качестве формата по умолчанию для зашифрованных томов. LUKS2 предоставляет зашифрованные тома с избыточностью метаданных и автоматическим восстановлением в случае частичного повреждения метаданных.

Благодаря внутренней гибкой компоновке LUKS2 также обеспечивает возможности будущих функций. Он поддерживает автоматическую разблокировку с помощью встроенного токена ядра-связки ключей. **libcryptsetup** которые позволяют пользователям разблокировать тома LUKS2 с помощью ключевой фразы, хранящейся в службе хранения ключей ядра.

Другие заметные улучшения включают в себя:

- Настройка защищенного ключа с использованием схемы шифрования с завернутым ключом.
- Более простая интеграция с расшифровкой на основе политик (Clevis).
- До 32 ключевых слотов - LUKS1 предоставляет только 8 ключевых слотов.

Для получения более подробной информации см. **cryptsetup(8)** а также **cryptsetup-reencrypt(8)** справочные страницы.

(BZ#1564540)

NVMe / FC полностью поддерживается на адаптерах Broadcom Emulex Fibre Channel

Транспортный тип NVMe по Fibre Channel (NVMe / FC) теперь полностью поддерживается в режиме инициатора при использовании с адаптерами Broadcom Emulex Fibre Channel 32 Гбит.

NVMe через Fibre Channel является дополнительным типом транспортного протокола для протокола энергонезависимой памяти (NVMe) в дополнение к протоколу удаленного прямого доступа к памяти (RDMA), который ранее был представлен в Red Hat Enterprise Linux.

Чтобы включить NVMe / FC в **lpfc** драйвер, отредактируйте **/etc/modprobe.d/lpfc.conf** файл и добавьте следующую опцию:

```
lpfc_enable_fc4_type=3
```

Драйверы, кроме **lpfc** по-прежнему остаются в технологии предварительного просмотра.

Дополнительные ограничения:

- Multipath не поддерживается с NVMe / FC.
- Кластеризация NVMe не поддерживается с NVMe / FC.
- В настоящее время Red Hat Enterprise Linux не поддерживает одновременное использование NVMe / FC и SCSI / FC на порте инициатора.
- *ядро-альт* Пакет не поддерживает NVMe / FC.
- **kdump** не поддерживается с NVMe / FC.
- Загрузка из сети хранения данных (SAN) NVMe / FC не поддерживается.

(BZ#1649497)

новый overrides раздел файла конфигурации DM Multipath

/etc/multipath.conf Теперь файл содержит **overrides** раздел, который позволяет вам установить значение конфигурации для всех ваших устройств. Эти атрибуты используются DM Multipath для всех устройств, если они не перезаписаны атрибутами, указанными в **multipaths** раздел **/etc/multipath.conf** файл для путей, которые содержат устройство. Эта функциональность заменяет **all_devs** параметр **devices** раздел файла конфигурации, который больше не поддерживается.

(BZ#1643294)

Теперь поддерживается установка и загрузка с устройств NVDIMM

До этого обновления установщик игнорировал устройства с энергонезависимым двойным встроенным модулем памяти (NVDIMM) в любом режиме.

В этом обновлении улучшения ядра для поддержки устройств NVDIMM обеспечивают улучшенные возможности производительности системы и расширенный доступ к файловой системе для приложений с интенсивной записью, таких как базы данных или аналитические рабочие нагрузки, а также снижают нагрузку на процессор.

Это обновление представляет поддержку для:

- Использование устройств NVDIMM для установки с использованием **nvdimm** Команда Kickstart и графический интерфейс пользователя, позволяющие устанавливаться и загружаться с устройств NVDIMM в секторном режиме, а также переконфигурировать устройства NVDIMM в секторный режим во время установки.
- Расширение **Kickstart** сценарии для **анаконда** с командами для работы с устройствами NVDIMM.
- Способность **grub2**, **efibootmgr**, а также **efivar** компоненты системы для обработки и загрузки с устройств NVDIMM.

(BZ#1499442)

Обнаружение маргинальных путей в DM Multipath улучшено

multipathd Сервис теперь поддерживает улучшенное обнаружение маргинальных путей. Это помогает многолучевым устройствам избегать путей, которые могут неоднократно выходить из строя, и повышает производительность. Маргинальные пути - это пути с постоянными, но периодически возникающими ошибками ввода / вывода.

Следующие опции в **/etc/multipath.conf** Поведение маргинальных путей управления файлами:

- **marginal_path_double_failed_time**,
- **marginal_path_err_sample_time**,
- **marginal_path_err_rate_threshold**, а также
- **marginal_path_err_recheck_gap_time**.

DM Multipath отключает путь и проверяет его с повторным вводом / выводом в течение настроенного времени выборки, если:

- перечисленные **multipath.conf** варианты установлены,
- путь не проходит дважды в установленное время, и
- другие пути доступны.

Если во время этого тестирования путь превышает заданную частоту ошибок, DM Multipath игнорирует его в течение настроенного промежутка времени, а затем повторно проверяет его, чтобы убедиться, что он работает достаточно хорошо, чтобы его можно было восстановить.

Для получения дополнительной информации см. **multipath.conf** справочная страница.

(BZ#1643550)

Поведение по умолчанию для нескольких очередей

Блочные устройства теперь используют планирование нескольких очередей в Red Hat Enterprise Linux 8. Это позволяет масштабировать производительность на уровне блоков с быстрыми твердотельными накопителями (SSD) и многоядерными системами.

SCSI Multiqueue (**scsi-mq**) драйвер теперь включен по умолчанию, и ядро загружается с **scsi_mod.use_blk_mq=Y** вариант. Это изменение согласуется с вышестоящим ядром Linux.

Device Mapper Multipath (DM Multipath) требует **scsi-mq** Водитель должен быть активным.

(BZ#1647612)

Stratis теперь доступен

Stratis - это новый менеджер локального хранилища. Он предоставляет управляемые файловые системы поверх пулов хранения с дополнительными функциями для пользователя.

Stratis позволяет вам легче выполнять задачи хранения, такие как:

- Управление снимками и тонкой подготовкой
- Автоматически увеличивать размеры файловой системы по мере необходимости
- Поддерживать файловые системы

Для управления хранилищем Stratis используйте **stratis** утилита, которая связывается с **stratisd** Фоновая служба.

Для получения дополнительной информации см. Документацию Stratis: [Managing layered local storage with Stratis](#)

(JIRA:RHELPLAN-1212)

4.12. ВЫСОКАЯ ДОСТУПНОСТЬ И КЛАСТЕРЫ (МАШИННЫЙ ПЕРЕВОД)

новый pcs команды для вывода списка доступных сторожевых устройств и тестирования сторожевых устройств

Для настройки SBD с Pacemaker требуется работающее сторожевое устройство. Этот релиз поддерживает **pcs stonith sbd watchdog list** команда для перечисления доступных сторожевых устройств на локальном узле, и **pcs stonith sbd watchdog test** Команда для проверки сторожевого устройства. Для получения информации о **sbd** инструмент командной строки, см. **sbd(8)** страница справочника.

(BZ#1578891)

pcs теперь команда поддерживает фильтрацию сбоев ресурсов по операции и ее интервалу

Кардиостимулятор теперь отслеживает сбои ресурсов для каждой операции с ресурсом поверх имени ресурса и узла. **pcs resource failcount show** Теперь команда позволяет фильтровать сбои по ресурсу, узлу, операции и интервалу. Он предоставляет возможность отображать сбои, агрегированные по ресурсу и узлу или детализированные по ресурсу, узлу, операции и его интервалу. Кроме того, **pcs resource failcount reset** Теперь команда позволяет фильтровать сбои по ресурсу, узлу, операции и интервалу.

(BZ#1591308)

Временные метки включены в corosync журнал

corosync log ранее не содержал меток времени, что затрудняло связь его с логами других узлов и демонов. В этом выпуске временные метки присутствуют в **corosync** журнал.

(BZ#1615420)

Новые форматы для pcs cluster setup, pcs cluster node add а также pcs cluster node remove команды

В Red Hat Enterprise Linux 8 **pcs** полностью поддерживает Corosync 3, **knet**, и имена узлов. Имена узлов теперь требуются и заменяют адреса узлов в роли идентификатора узла. Адреса узлов теперь необязательны.

- в **pcs host auth** команда, адреса узлов по умолчанию для имен узлов
- в **pcs cluster setup** а также **pcs cluster node add** команды, адреса узлов по умолчанию равны адресам узлов, указанным в **pcs host auth** команда.

С этими изменениями изменились форматы команд для настройки кластера, добавления узла в кластер и удаления узла из кластера. Информацию об этих новых форматах команд смотрите в справке для **pcs cluster setup**, **pcs cluster node add** а также **pcs cluster node remove** команды.

(BZ#1158816)

Кардиостимулятор 2.0.0 в RHEL 8

pacemaker пакеты были обновлены до предыдущей версии Pacemaker 2.0.0, которая содержит ряд исправлений ошибок и улучшений по сравнению с предыдущей версией:

- Подробный журнал Pacemaker теперь **/var/log/pacemaker/pacemaker.log** по умолчанию (не напрямую в **/var/log** или в сочетании с **corosync** войти под **/var/log/cluster**).
- Процессы демона Pacemaker были переименованы, чтобы сделать чтение журналов более интуитивным. Например, **pengine** был переименован в **pacemaker-schedulerd**
- Поддержка устаревших **default-resource-stickiness** а также **is-managed-default** свойства кластера были удалены. **resource-stickiness** а также **is-managed** свойства должны быть установлены в ресурсах по умолчанию вместо этого. Существующие конфигурации (но не вновь созданные) с устаревшим синтаксисом будут автоматически обновлены для использования поддерживаемого синтаксиса.
- Более полный список изменений доступен по адресу <https://access.redhat.com/articles/3681151>

Пользователям, которые обновляют существующий кластер с помощью Red Hat Enterprise Linux 7 или более ранней версии, рекомендуется запустить **pcs cluster cib-upgrade** на любом узле кластера до и после обновления RHEL на всех узлах кластера.

(BZ#1543494)

Основные ресурсы переименованы в ресурсы для продвижения клонов

Red Hat Enterprise Linux (RHEL) 8 поддерживает Pacemaker 2.0, в котором главный / подчиненный ресурс больше не является отдельным типом ресурса, а является стандартным ресурсом-клоном с **promotable** мета-атрибутом установлен в **true** Следующие изменения были внесены в поддержку этого обновления:

- Больше невозможно создавать главные ресурсы с **pcs** команда. Вместо этого можно создать **promotable** клонировать ресурсы. Связанные ключевые слова и команды были изменены с **master** в **promotable**
- Все существующие главные ресурсы отображаются в виде рекламных клонированных ресурсов.
- При управлении кластером RHEL7 в веб-интерфейсе главные ресурсы по-прежнему называются главными, поскольку кластеры RHEL7 не поддерживают продвигаемые клоны.

(BZ#1542288)

Новые команды для аутентификации узлов в кластере

Red Hat Enterprise Linux (RHEL) 8 включает в себя следующие изменения в командах, используемых для аутентификации узлов в кластере.

- Новая команда для аутентификации **pcs host auth** Эта команда позволяет пользователям указывать имена хостов, адреса и **pcsd** порты.
- **pcs cluster auth** Команда аутентифицирует только узлы в локальном кластере и не принимает список узлов
- Теперь можно указать адрес для каждого узла. **pcs/pcsd** Затем свяжется с каждым узлом, используя указанный адрес. Эти адреса могут отличаться от указанных **corosync** использует внутренне.
- **pcs pcsd clear-auth** команда была заменена **pcs pcsd deauth** а также **pcs host deauth** команды. Новые команды позволяют пользователям деаутентифицировать как один хост, так и все хосты.
- Ранее аутентификация узла была двунаправленной и выполняла **pcs cluster auth** команда вызвала аутентификацию всех указанных узлов друг против друга. **pcs host auth** Однако команда вызывает проверку подлинности только локального хоста на указанных узлах. Это позволяет лучше контролировать, какой узел аутентифицируется по сравнению с другими узлами при выполнении этой команды. На самой настройке кластера, а также при добавлении узла, **pcs** автоматически синхронизирует токены в кластере, поэтому все узлы в кластере по-прежнему автоматически аутентифицируются, как и раньше, и узлы кластера могут связываться друг с другом.

Обратите внимание, что эти изменения не являются обратно совместимыми. Узлы, которые были аутентифицированы в системе RHEL 7, необходимо будет снова аутентифицировать.

(BZ#1549535)

pcs теперь команды поддерживают отображение, очистку и синхронизацию истории фехтования

Демон забора Кардиостимулятора отслеживает историю всех предпринятых действий забора (ожидających, успешных и не выполненных). С этим выпуском **pcs** Команды

позволяют пользователям получать доступ к истории фехтования следующими способами:

- **pcs status** команда показывает неудачные и ожидающие действия ограждения
- **pcs status --full** Команда показывает всю историю фехтования
- **pcs stonith history** Команда предоставляет опции для отображения и очистки истории фехтования
- Хотя история ограждения синхронизируется автоматически, **pcs stonith history** Команда теперь поддерживает **update** опция, которая позволяет пользователю вручную синхронизировать историю ограждения, если это необходимо

(BZ#1620190, BZ#1615891)

4.13. СЕТЕЙ (МАШИННЫЙ ПЕРЕВОД)

nftables заменяет **iptables** в качестве основы фильтрации сетевых пакетов по умолчанию

nftables структура обеспечивает средства классификации пакетов и является назначенным преемником **iptables**, **ip6tables**, **arptables**, а также **ebtables** инструменты. Он предлагает множество улучшений в удобстве, функциях и производительности по сравнению с предыдущими инструментами фильтрации пакетов, в частности:

- таблицы поиска вместо линейной обработки
- единая основа для обоих **IPv4** а также **IPv6** протоколы
- правила все применяются атомарно вместо извлечения, обновления и хранения полного набора правил
- поддержка отладки и трассировки в наборе правил (**nftrace**) и отслеживание событий трассировки (в **nft** инструмент)
- более согласованный и компактный синтаксис, без специфичных для протокола расширений
- Netlink API для сторонних приложений

Аналогично **iptables**, **nftables** использовать таблицы для хранения цепей. Цепочки содержат индивидуальные правила выполнения действий. **nft** Инструмент заменяет все инструменты из предыдущих структур фильтрации пакетов. **libnftables** библиотека может быть использована для низкоуровневого взаимодействия с **nftables** Netlink API через **libmnl** библиотека.

iptables, **ip6tables**, **ebtables** а также **arptables** инструменты заменяются одноименными заменами на основе **nftables**. Хотя внешнее поведение идентично их прежним аналогам, внутренне они используют **nftables** с наследством **netfilter** модули ядра через интерфейс совместимости, где это необходимо.

Влияние модулей на **nftables** набор правил можно наблюдать с помощью **nft list ruleset** команда. Поскольку эти инструменты добавляют таблицы, цепочки и правила к **nftables** набор правил, помните, что **nftables** операции с набором правил, такие как **nft flush ruleset** команда, может повлиять на наборы правил, установленные с помощью ранее отдельных устаревших команд.

Чтобы быстро определить, какой вариант инструмента присутствует, информация о версии была обновлена и теперь включает имя сервера. В RHEL 8 основанная на `nftables` инструмент печатает следующую строку версии:

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

Для сравнения выводится следующая информация о версии, если она унаследована `iptables` инструмент присутствует:

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

(BZ#1644030)

Известные функции TCP в RHEL 8

Red Hat Enterprise Linux 8 распространяется с сетевым стеком TCP версии 4.16, который обеспечивает более высокую производительность, лучшую масштабируемость и большую стабильность. Производительность повышена, особенно для занятого TCP-сервера с высокой скоростью входящего соединения.

Кроме того, два новых алгоритма перегрузки TCP, **BBR** а также **NV**, Доступны, предлагая меньшую задержку и лучшую пропускную способность, чем кубическая в большинстве сценариев.

(BZ#1562998)

firewalld использования nftables по умолчанию

С этим обновлением `nftables` Подсистема фильтрации является брандмауэром брандмауэра по умолчанию для `firewalld` демон. Чтобы изменить бэкэнд, используйте `FirewallBackend` вариант в `/etc/firewalld.conf` файл.

Это изменение вводит следующие различия в поведении при использовании `nftables`:

1. `iptables` выполнение правил всегда происходит раньше `firewalld` правила
 - **DROP** в `iptables` означает, что пакет никогда не виден `firewalld`
 - **ACCEPT** в `iptables` означает, что пакет все еще подлежит `firewalld` правила
2. `firewalld` прямые правила все еще реализуются через `iptables` в то время как другие `firewalld` использование функций `nftables`
3. прямое выполнение правил происходит раньше `firewalld` общее принятие установленных связей

(BZ#1509026)

Заметные изменения в wpa_supplicant в RHEL 8

В Red Hat Enterprise Linux (RHEL) 8 `wpa_supplicant` пакет построен с `CONFIG_DEBUG_SYSLOG` включен. Это позволяет читать `wpa_supplicant` войти с помощью `journalctl` утилита вместо проверки содержимого `/var/log/wpa_supplicant.log` файл.

(BZ#1582538)

Сетевой менеджер теперь поддерживает виртуальные функции SR-IOV

В Red Hat Enterprise Linux 8.0 **Сетевой менеджер** позволяет настроить количество виртуальных функций (VF) для интерфейсов, поддерживающих виртуализацию однокорневого ввода-вывода (SR-IOV). Дополнительно, **Сетевой менеджер** позволяет настроить некоторые атрибуты VF, такие как MAC-адрес, VLAN, **snoop checking** установка и допустимые битрейты. Обратите внимание, что все свойства, связанные с SR-IOV, доступны в **sriov** настройка соединения. Для получения более подробной информации см. **nm-settings(5)** справочная страница.

(BZ#1555013)

Драйверы виртуальной сети IPVLAN теперь поддерживаются

В Red Hat Enterprise Linux 8.0 ядро включает поддержку драйверов виртуальной сети IPVLAN. В этом обновлении виртуальные сетевые интерфейсные карты (NIC) IPVLAN обеспечивают сетевое подключение для нескольких контейнеров, открывая один MAC-адрес для локальной сети. Это позволяет одному хосту иметь много контейнеров, преодолевая возможное ограничение на количество MAC-адресов, поддерживаемых одноранговым сетевым оборудованием.

(BZ#1261167)

Сетевой менеджер поддерживает подстановочный знак соответствия интерфейса для соединений

Ранее можно было ограничить соединение с данным интерфейсом, используя только точное совпадение в имени интерфейса. С этим обновлением соединения имеют новый **match.interface-name** свойство, которое поддерживает подстановочные знаки. Это обновление позволяет пользователям более гибко выбирать интерфейс для подключения с использованием шаблона с подстановочными знаками.

(BZ#1555012)

Улучшения в сетевом стеке 4.18

Red Hat Enterprise Linux 8.0 включает в себя сетевой стек, обновленный до версии 4.18 вышестоящей версии, которая содержит несколько исправлений и улучшений. Заметные изменения включают в себя:

- Введены новые функции разгрузки, такие как **UDP_GSO**, и, для некоторых драйверов устройств, **GRO_HW**
- Улучшена значительная масштабируемость для протокола пользовательских дейтаграмм (UDP).
- Улучшен общий код опроса занятости.
- Улучшенная масштабируемость для протокола IPv6.
- Улучшенная масштабируемость для кода маршрутизации.
- Добавлен новый алгоритм планирования очереди передачи по умолчанию, **fq_codel**, что улучшает задержку передачи.

- Улучшенная масштабируемость для некоторых алгоритмов планирования очереди передачи. Например, **pfifo_fast** сейчас без блокировки.

(BZ#1562987)

Новые инструменты для конвертации iptables в nftables

Это обновление добавляет **iptables-translate** а также **ip6tables-translate** инструменты для преобразования существующих **iptables** или же **ip6tables** правила в эквивалентные для **nftables**. Обратите внимание, что в некоторых расширениях отсутствует поддержка перевода. Если такое расширение существует, инструмент печатает непереуведенное правило с префиксом **#** знак. Например:

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

Кроме того, пользователи могут использовать **iptables-restore-translate** а также **ip6tables-restore-translate** инструменты для перевода дампа правил. Обратите внимание, что до этого пользователи могут использовать **iptables-save** или же **ip6tables-save** Команды для печати дампа текущих правил. Например:

```
| % 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)

Новые функции, добавленные в VPN с использованием Сетевой менеджер

В Red Hat Enterprise Linux 8.0 **Сетевой менеджер** предоставляет следующие новые функции для VPN:

- Поддержка протокола Internet Key Exchange версии 2 (IKEv2).
- Добавил еще немного Libreswan варианты, такие как **rightid**, **leftcert**, **narrowing**, **rekey**, **fragmentation** опции. Для получения дополнительной информации о поддерживаемых параметрах см. **nm-settings-libreswan** справочная страница.
- Обновлены шифры по умолчанию. Это означает, что когда пользователь не указывает шифры, **NetworkManager-libreswan** плагин позволяет **Libreswan** Приложение для выбора системного шифра по умолчанию. Единственное исключение - когда пользователь выбирает конфигурацию агрессивного режима IKEv1. В этом случае **ike = aes256-sha1;modp1536** а также **eps = aes256-sha1** значения передаются **Libreswan**

(BZ#1557035)

Новый тип чанка данных, I-DATA, добавлено в SCTP

Это обновление добавляет новый тип чанка данных, **I-DATA**, и планировщики потоков в протокол управления передачей потока (SCTP). Ранее SCTP отправлял пользовательские сообщения в том же порядке, в котором они были отправлены пользователем. Следовательно, большое пользовательское сообщение SCTP блокировало все остальные сообщения в любом потоке до полной отправки. Когда используешь **I-DATA** чанки, поле порядкового номера передачи (TSN) не перегружено. В

результате SCTP теперь может планировать потоки различными способами, и **I-DATA** позволяет чередование сообщений пользователя (RFC 8260). Обратите внимание, что оба пира должны поддерживать **I-DATA** тип чанка.

(BZ#1273139)

4.14. БЕЗОПАСНОСТЬ (МАШИННЫЙ ПЕРЕВОД)

Руководство по безопасности SCAP Профиль PCI-DSS соответствует версии 3.2.1

SCAP Security Guide Проект предоставляет профиль PCI-DSS (Стандарт безопасности данных индустрии платежных карт) для Red Hat Enterprise Linux 8, и он был обновлен в соответствии с последней версией PCI-DSS - 3.2.1.

(BZ#1618528)

OpenSSH перебазируется на версию 7.8p1

openssh пакеты были обновлены до версии 7.8p1. Заметные изменения включают в себя:

- Удалена поддержка **SSH version 1** протокол.
- Удалена поддержка **hmac-ripemd160** код аутентификации сообщения.
- Убрана поддержка RC4 (**arcfour**) шифры.
- Удалена поддержка **Blowfish** шифры.
- Удалена поддержка **CAST** шифры.
- Изменено значение по умолчанию **UseDNS** возможность **no**
- инвалид **DSA** алгоритмы открытых ключей по умолчанию.
- Изменен минимальный размер модуля для **Diffie-Hellman** параметры до 2048 бит.
- Изменилась семантика **ExposeAuthInfo** Вариант конфигурации.
- **UsePrivilegeSeparation=sandbox** опция теперь обязательна и не может быть отключена.
- Установите минимально допустимое **RSA** размер ключа до 1024 бит.

(BZ#1622511)

RSA-PSS теперь поддерживается в OpenSC

Это обновление добавляет поддержку схемы криптографической подписи RSA-PSS к **OpenSC** драйвер смарт-карты. Новая схема обеспечивает безопасный криптографический алгоритм, необходимый для поддержки TLS 1.3 в клиентском программном обеспечении.

(BZ#1595626)

Заметные изменения в rsyslog в RHEL 8

rsyslog пакеты были обновлены до вышестоящей версии 8.37.0, которая содержит множество исправлений ошибок и улучшений по сравнению с предыдущими версиями. Наиболее заметные изменения включают в себя:

- Улучшенная обработка **Rsyslog** внутренние сообщения; возможность ограничения их скорости; исправлен возможный тупик.
- Улучшенное ограничение скорости в целом; Настоящий *источник спама* сейчас зарегистрирован.
- Улучшена обработка негабаритных сообщений - теперь пользователь может настроить обработку их как в ядре, так и в определенных модулях с помощью отдельных действий.
- **mmnormalize** базы правил теперь могут быть встроены в **config** файл вместо создания отдельных файлов для них.
- Теперь пользователь может установить **GnuTLS** приоритетная строка для **imtcp** это позволяет детально контролировать шифрование.
- Все **config** переменные, включая переменные в JSON, теперь не чувствительны к регистру.
- Различные улучшения вывода PostgreSQL.
- Добавлена возможность использовать переменные оболочки для управления **config** обработка, такая как условная загрузка дополнительных файлов конфигурации, выполнение операторов или включение текста в **config** Обратите внимание, что чрезмерное использование этой функции может затруднить отладку **Rsyslog**
- Режимы создания 4-значных файлов теперь можно указывать в **config**
- Входные данные протокола надежной регистрации событий (RELP) теперь могут связываться также только по указанному адресу.
- Значение по умолчанию **enable.body** опция вывода почты теперь приведена в соответствие с документацией
- Теперь пользователь может указать коды ошибок вставки, которые следует игнорировать в **MongoDB** выход.
- Параллельный вход TCP (pTCP) теперь имеет настраиваемое отставание для лучшей балансировки нагрузки.

(BZ#1613880)

новый Rsyslog модуль: omkafka

Включить **Кафка** централизованные сценарии хранения данных, теперь вы можете пересылать журналы на **Кафка** инфраструктура с использованием нового **omkafka** модуль.

(BZ#1542497)

libssh реализует SSH в качестве основного криптографического компонента

Это изменение вводит **libssh** в качестве основного криптографического компонента в Red Hat Enterprise Linux 8. **libssh** библиотека реализует протокол Secure Shell (SSH).

Обратите внимание, что **libssh** не соответствует общесистемной политике шифрования.

(BZ#1485241)

Поддержка PKCS # 11 для смарт-карт и HSM теперь одинакова во всей системе

Благодаря этому обновлению использование смарт-карт и аппаратных модулей безопасности (HSM) с интерфейсом криптографических токенов PKCS # 11 становится согласованным. Это означает, что пользователь и администратор могут использовать один и тот же синтаксис для всех связанных инструментов в системе. Заметные улучшения включают в себя:

- Поддержка схемы универсального идентификатора ресурса (URI) PKCS # 11, которая обеспечивает упрощенное включение токенов на серверах RHEL как для администраторов, так и для разработчиков приложений.
- Общесистемный метод регистрации для смарт-карт и HSM, использующих **pkcs11.conf**
- Последовательная поддержка для HSM и смарт-карт доступна в NSS, GnuTLS и OpenSSL (через **openssl-pkcs11** двигатель) приложения.
- HTTP-сервер Apache (**httpd**) теперь без проблем поддерживает HSM.

Для получения дополнительной информации см. **pkcs11.conf(5)** справочная страница.

(BZ#1516741)

Общесистемные криптографические политики применяются по умолчанию

Крипто-политики - это компонент Red Hat Enterprise Linux 8, который настраивает основные криптографические подсистемы, охватывающие протоколы TLS, IPSec, SSH, DNSSec и Kerberos. Он предоставляет небольшой набор политик, которые администратор может выбрать с помощью **update-crypto-policies** команда.

DEFAULT Общесистемная криптографическая политика предлагает безопасные настройки для текущих моделей угроз. Он поддерживает протоколы TLS 1.2 и 1.3, а также протоколы IKEv2 и SSH2. Ключи RSA и параметры Диффи-Хеллмана принимаются, если они превышают 2047 бит.

Увидеть [Consistent security by crypto policies in Red Hat Enterprise Linux 8](#) статья в блоге Red Hat и **update-crypto-policies(8)** Страница man для получения дополнительной информации.

(BZ#1591620)

Руководство по безопасности SCAP поддерживает OSPP 4.2

SCAP Security Guide содержит черновой вариант профиля профиля OSPP (профиль защиты для операционных систем общего назначения) 4.2 для Red Hat Enterprise Linux 8. Этот профиль отражает обязательные элементы управления конфигурацией, указанные в Приложении по настройке NIAP к профилю защиты для операционных

систем общего назначения (версия профиля защиты 4.2). Руководство по безопасности SCAP предоставляет автоматизированные проверки и сценарии, которые позволяют пользователям выполнять требования, определенные в OSPP.

(BZ#1618518)

Интерфейс командной строки OpenSCAP был улучшен

Подробный режим теперь доступен во всех **oscap** модули и submodule. Вывод инструмента имеет улучшенное форматирование.

Устаревшие параметры были удалены, чтобы улучшить удобство использования интерфейса командной строки.

Следующие опции больше не доступны:

- **--show** в **oscap xccdf generate report** был полностью удален
- **--probe-root** в **oscap oval eval** был удален. Его можно заменить, установив переменную среды, **OSCAP_PROBE_ROOT**
- **--sce-results** в **oscap xccdf eval** был заменен **--check-engine-results**
- **validate-xml** submodule удален из модулей CPE, OVAL и XCCDF. **validate** submodule могут быть использованы вместо этого для проверки содержимого SCAP по схемам XML и схемам XSD.
- **oscap oval list-probes** команда была удалена, список доступных зондов можно отобразить с помощью **oscap --version** вместо.

OpenSCAP позволяет оценить все правила в данном эталонном тесте XCCDF независимо от профиля, используя **--profile '(all)'**

(BZ#1618484)

Поддержка проверки разрешения новой карты на mmap Системный вызов

SELinux **mmap** добавлено разрешение для управления отображением в памяти доступа к файлам, каталогам, сокетам и т. д. Это позволяет политике SELinux предотвращать прямой доступ к памяти для различных объектов файловой системы и обеспечивать повторную проверку каждого такого доступа.

(BZ#1592244)

SELinux теперь поддерживает Systemd No New Privileges

Это обновление представляет **nnp_nosuid_transition** возможность политики, которая позволяет переходы домена SELinux под **No New Privileges** (NNP) или **nosuid** если **nnp_nosuid_transition** допускается между старым и новым контекстами. **selinux-policy** пакеты теперь содержат политику для **Systemd** услуги, которые используют **NNP** функция безопасности.

Следующее правило описывает разрешение этой возможности для службы:

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

Например:

■

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

Политика распространения теперь также содержит макроинтерфейс `m4`, который можно использовать в политиках безопасности SELinux для служб, использующих `init_nnp_daemon_domain()` функция.

(BZ#1594111)

SELinux теперь поддерживает `getrlimit` разрешение в `process` учебный класс

Это обновление представляет новую проверку контроля доступа SELinux, `process:getrlimit`, который был добавлен для `prlimit()` функция. Это позволяет разработчикам политики SELinux контролировать, когда один процесс пытается прочитать, а затем изменить ограничения ресурсов другого процесса, используя `process:setrlimit` разрешение. Обратите внимание, что SELinux не ограничивает процесс от манипулирования собственными ограничениями ресурсов через `prlimit()` Увидеть `prlimit(2)` а также `getrlimit(2)` справочные страницы для получения дополнительной информации.

(BZ#1549772)

Поддержка TLS 1.3 в криптографических библиотеках

Это обновление включает транспортную безопасность уровня TLS 1.3 по умолчанию во всех основных внутренних криптобиблиотеках. Это обеспечивает низкую задержку на уровне связи операционной системы и повышает конфиденциальность и безопасность приложений, используя преимущества новых алгоритмов, таких как RSA-PSS или X25519.

(BZ#1516728)

Новые функции в OpenSCAP в RHEL 8

OpenSCAP Пакет обновлен до версии 1.3.0, которая содержит множество улучшений по сравнению с предыдущими версиями. Наиболее заметные особенности включают в себя:

- API и ABI были объединены - обновленные, устаревшие и / или неиспользуемые символы были удалены.
- Зонды запускаются не как независимые процессы, а как потоки внутри **oscap** процесс.
- Интерфейс командной строки обновлен.
- **Python 2** привязки были заменены **Python 3** привязок.

(BZ#1614273)

Аудит 3.0 заменяет `audispd` с `auditd`

С этим обновлением функционал `audispd` был перемещен в `auditd` В следствии, `audispd` параметры конфигурации теперь являются частью `auditd.conf` В дополнение `plugins.d` каталог был перемещен в `/etc/audit` Текущий статус `auditd` и его плагины теперь можно проверить, запустив `service auditd state` команда.

(BZ#1616428)

Rsyslog imfile теперь поддерживает символические ссылки

С этим обновлением **Rsyslog imfile** Модуль обеспечивает лучшую производительность и больше вариантов конфигурации. Это позволяет использовать модуль для более сложных случаев использования файлов. Например, теперь вы можете использовать файловые мониторы с шаблонами глобусов в любом месте по настроенному пути и поворачивать цели символических ссылок с повышенной пропускной способностью данных.

(BZ#1614179)

Автоматический OpenSSH генерация ключей сервера теперь обрабатывается sshd-keygen@.service

OpenSSH автоматически создает ключи хоста сервера RSA, ECDSA и ED25519, если они отсутствуют. Чтобы настроить создание ключа хоста в RHEL 8, используйте **sshd-keygen@.service** конкретизированный сервис.

Например, чтобы отключить автоматическое создание типа ключа RSA:

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

Увидеть **/etc/sysconfig/sshd** файл для получения дополнительной информации.

(BZ#1228088)

По умолчанию rsyslog формат файла конфигурации теперь не является устаревшим

Файлы конфигурации в **rsyslog** пакеты теперь по умолчанию используют нестандартный формат. Устаревший формат все еще можно использовать, однако, смешивание текущих и устаревших операторов конфигурации имеет несколько ограничений. Конфигурации, перенесенные из предыдущих выпусков RHEL, должны быть пересмотрены. Увидеть **rsyslog.conf(5)** Страница man для получения дополнительной информации.

(BZ#1619645)

Новый булев SELinux

Это обновление системной политики SELinux представляет следующие логические значения:

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

Для получения дополнительной информации см. Вывод следующей команды:

semanage boolean -l

(JIRA:RHELPLAN-10347)

4.15. ВИРТУАЛИЗАЦИЯ (МАШИННЫЙ ПЕРЕВОД)

KVM поддерживает 5-уровневый пейджинг

В Red Hat Enterprise Linux 8 виртуализация KVM поддерживает пятиуровневую функцию подкачки, которая значительно увеличивает физическое и виртуальное адресное пространство, которое могут использовать хост и гостевые системы.

(BZ#1485229)

KVM поддерживает UMIP в RHEL 8

Виртуализация KVM теперь поддерживает функцию предотвращения инструкций в режиме пользователя (UMIP), которая может помочь предотвратить доступ приложений из пользовательского пространства к общесистемным настройкам. Это уменьшает потенциальные векторы для атак на повышение привилегий и, таким образом, делает гипервизор KVM и его гостевые машины более безопасными.

(BZ#1494651)

Дополнительная информация в отчетах о сбоях гостей KVM

Информация о сбое, которую генерирует гипервизор KVM, если гость неожиданно завершает работу или перестает отвечать, была расширена. Это облегчает диагностику и устранение проблем в развертываниях виртуализации KVM.

(BZ#1508139)

qemu-kvm 2,12 в RHEL 8

Red Hat Enterprise Linux 8 распространяется с **qemu-kvm 2.12**. Эта версия исправляет несколько ошибок и добавляет ряд улучшений по сравнению с версией 1.5.3, доступной в Red Hat Enterprise Linux 7.

Примечательно, что были введены следующие функции:

- Тип гостевой машины Q35
- Гостевой загрузчик UEFI
- NUMA тюнинг и пиннинг в гостях
- горячее подключение vCPU и горячее отключение
- гостевой ввод-вывод

Обратите внимание, что некоторые функции доступны в **qemu-kvm 2.12** не поддерживаются в Red Hat Enterprise Linux 8. Для получения подробной информации см. «Поддержка функций и ограничения в виртуализации RHEL 8» на портале для клиентов Red Hat.

(BZ#1559240)

NVIDIA vGPU теперь совместима с консолью VNC

При использовании функции виртуального графического процессора NVIDIA (vGPU) теперь можно использовать консоль VNC для отображения визуального вывода гостя.

(BZ#1497911)

Серв поддерживается виртуализацией

В этом обновлении хранилище Серв поддерживается виртуализацией KVM на всех архитектурах ЦП, поддерживаемых Red Hat.

(BZ#1578855)

Q35 тип машины теперь поддерживается виртуализацией

Red Hat Enterprise Linux 8 представляет поддержку Q35, более современный тип машины на основе PCI Express. Это обеспечивает ряд улучшений в функциях и производительности виртуальных устройств и гарантирует, что более широкий спектр современных устройств совместим с виртуализацией. Кроме того, виртуальные машины, созданные в Red Hat Enterprise Linux 8, настроены на использование Q35 по умолчанию.

Также обратите внимание, что ранее по умолчанию ПК тип машины устарел и должен использоваться только при виртуализации старых операционных систем, не поддерживающих Q35.

(BZ#1599777)

Интерактивный загрузчик для виртуальных машин на IBM Z

При загрузке виртуальной машины (VM) на хосте IBM Z встроенное ПО загрузчика QEMU теперь имеет интерфейс интерактивной консоли. Это позволяет устранять проблемы загрузки гостевой ОС без доступа к среде хоста.

(BZ#1508137)

QEMU песочница была добавлена

В Red Hat Enterprise Linux 8 эмулятор QEMU представляет функцию песочницы. Песочница QEMU предоставляет настраиваемые ограничения на то, что системные вызовы QEMU могут выполнять, и, таким образом, делает виртуальные машины более безопасными. Обратите внимание, что эта функция включена и настроена по умолчанию.

(JIRA:RHELPLAN-10628)

Наборы команд GFNI и CLDEMOT включены для Intel Xeon SnowRidge

Виртуальные машины (VM), работающие на хосте RHEL 8 в системе Intel Xeon SnowRidge, теперь могут использовать наборы команд GFNI и CLDEMOT. Это может значительно повысить производительность таких виртуальных машин в определенных сценариях.

(BZ#1494705)

Добавлен блочный драйвер на основе VFIO для устройств NVMe

Эмулятор QEMU представляет драйвер на основе виртуальных функций ввода / вывода (VFIO) для устройств энергонезависимой памяти (NVMe). Драйвер напрямую связывается с устройствами NVMe, подключенными к виртуальным машинам (VM), и

избегает использования системного уровня ядра и его драйверов NVMe. В результате это повышает производительность устройств NVMe на виртуальных машинах.

(BZ#1519004)

Улучшена поддержка огромных страниц

При использовании RHEL 8 в качестве хоста виртуализации пользователи могут изменять размер страниц, которые поддерживают память виртуальной машины (ВМ), до любого размера, поддерживаемого ЦП. Это может значительно улучшить производительность виртуальной машины.

Чтобы настроить размер страниц памяти виртуальной машины, отредактируйте XML-конфигурацию виртуальной машины и добавьте элемент `<largepages>` в раздел `<memoryBacking>`.

(JIRA:RHELPLAN-14607)

4.16. SUPPORTABILITY (МАШИННЫЙ ПЕРЕВОД)

sosreport может сообщать о программах и картах на основе eBPF

sosreport Усовершенствован инструмент для отчетов о любых загруженных расширенных программах и картах Berkeley Packet Filtering (eBPF) в Red Hat Enterprise Linux 8.

(BZ#1559836)

ГЛАВА 5. ТЕХНОЛОГИЧЕСКИЕ ПРЕВЬЮ (МАШИННЫЙ ПЕРЕВОД)

В этой части представлен список всех предварительных версий технологий, доступных в Red Hat Enterprise Linux 8.0.

Информацию о возможностях поддержки Red Hat для функций предварительного просмотра см. В разделе [Technology Preview Features Support Scope](#)

5.1. ЯДРО (МАШИННЫЙ ПЕРЕВОД)

XDP доступны в качестве предварительного просмотра технологии

eXpress Data Path (XDP) Функция, которая доступна в качестве предварительного просмотра технологии, предоставляет возможность загружать программы Berkeley Packet Filter (BPF) для высокопроизводительной обработки пакетов в ядре, делая программируемый сетевой путь данных ядра.

(BZ#1503672)

eBPF доступны в качестве предварительного просмотра технологии

расширенная фильтрация пакетов Беркли (eBPF) функция доступна в качестве предварительного просмотра технологии для сетей и трассировки. **eBPF** позволяет пользователю присоединять пользовательские программы к различным точкам (сокеты, точки трассировки, прием пакетов) для получения и обработки данных. Функция включает в себя новый системный вызов **bpf()**, что позволяет создавать различные типы карт, а также вставлять различные типы программ в ядро. Увидеть **bpf(2)** справочная страница для получения дополнительной информации.

(BZ#1559616)

BCC доступен в качестве предварительного просмотра технологии

BPF Compiler Collection (BCC) представляет собой набор инструментов пользовательского пространства для создания эффективных программ трассировки и манипулирования ядром, который доступен в виде предварительного просмотра технологии в Red Hat Enterprise Linux 8. **BCC** предоставляет инструменты для анализа ввода-вывода, создания сетей и мониторинга операционных систем Linux с использованием **extended Berkeley Packet Filtering (eBPF)**

(BZ#1548302)

Контрольная группа v2 доступно как предварительный просмотр технологии в RHEL 8

Контрольная группа v2 Механизм представляет собой единую иерархию контрольной группы. **Контрольная группа v2** организует процессы иерархически и распределяет системные ресурсы по иерархии контролируемым и настраиваемым образом.

В отличие от предыдущей версии, **Контрольная группа v2** имеет только одну иерархию. Эта единая иерархия позволяет ядру Linux:

- Распределите процессы по категориям в зависимости от роли их владельца.

- Устраните проблемы с конфликтующими политиками нескольких иерархий.

Контрольная группа v2 поддерживает многочисленные контроллеры:

- Контроллер ЦП регулирует распределение циклов ЦП. Этот контроллер реализует:
 - Модели ограничения веса и абсолютной полосы пропускания для обычной политики планирования.
 - Абсолютная модель распределения полосы пропускания для политики планирования в реальном времени.
- Контроллер памяти регулирует распределение памяти. В настоящее время отслеживаются следующие типы использования памяти:
 - Userland memory - страничный кеш и анонимная память.
 - Структуры данных ядра, такие как дендрии и иномы.
 - Буферы сокетов TCP.
- Контроллер ввода / вывода регулирует распределение ресурсов ввода / вывода.
- Контроллер обратной записи взаимодействует с контроллерами памяти и ввода-вывода и является **Контрольная группа v2** конкретный.

Информация выше была основана на ссылке:

<https://www.kernel.org/doc/Documentation/cgroup-v2.txt> Вы можете обратиться к той же ссылке, чтобы получить больше информации о конкретном **Контрольная группа v2** контроллеры.

(BZ#1401552)

early kdump доступно как предварительный просмотр технологии в Red Hat Enterprise Linux 8

early kdump Функция позволяет загружать ядро и initramfs достаточно рано, чтобы захватить **vmcore** информация даже для ранних сбоях. Для более подробной информации о **early kdump**, см `/usr/share/doc/kexec-tools/early-kdump-howto.txt` файл.

(BZ#1520209)

5.2. ФАЙЛОВЫЕ СИСТЕМЫ И ХРАНИЛИЩЕ (МАШИННЫЙ ПЕРЕВОД)

VDO доступен как тип логического тома LVM

LVM теперь можно использовать для создания логических томов *Оптимизатор виртуальных данных* (VDO) тип. VDO - это виртуальное блочное устройство с возможностью сжатия и дедупликации данных.

Это функция предварительного просмотра технологии.

(BZ#1643553)

Поддержка поля целостности данных / расширения целостности данных (DIF / DIX)

DIF / DIX является дополнением к стандарту SCSI. Он остается в Technology Preview для всех адаптеров HBA и массивов хранения, кроме тех, которые специально указаны как поддерживаемые.

DIF / DIX увеличивает размер обычно используемого 512-байтового дискового блока с 512 до 520 байт, добавляя поле целостности данных (DIF). DIF хранит значение контрольной суммы для блока данных, которое вычисляется адаптером хост-шины (HBA), когда происходит запись. Затем запоминающее устройство подтверждает контрольную сумму при получении и сохраняет как данные, так и контрольную сумму. И наоборот, когда происходит чтение, контрольная сумма может быть проверена устройством хранения и принимающим HBA.

(BZ#1649493)

NVMe / FC доступен в качестве предварительного просмотра технологии в адаптерах Qlogic с использованием qla2xxx

Транспортный тип NVMe по Fibre Channel (NVMe / FC) доступен в качестве предварительного просмотра технологии в адаптерах Qlogic с использованием **qla2xxx** Водитель.

(BZ#1649922)

5.3. ВЫСОКАЯ ДОСТУПНОСТЬ И КЛАСТЕРЫ (МАШИННЫЙ ПЕРЕВОД)

электрокардиостимулятор podman комплекты доступны в качестве предварительного просмотра технологии

Контейнеры Pacemaker теперь работают на **podman** Контейнерная платформа, с функцией контейнерной связки, доступной в качестве предварительного просмотра технологии. Из этой функции есть одно исключение: предварительный просмотр технологий: Red Hat полностью поддерживает использование пакетов Pacemaker для Red Hat Openstack.

(BZ#1619620)

5.4. БЕЗОПАСНОСТЬ (МАШИННЫЙ ПЕРЕВОД)

SWID-тег выпуска RHEL 8.0

Для обеспечения возможности идентификации установок RHEL 8.0 с использованием механизма ISO / IEC 19770-2: 2015 в файлах установлены теги идентификации программного обеспечения (SWID). **/usr/lib/swidtag/redhat.com/com.redhat.RHEL-8-
<architecture>.swidtag** а также **/usr/lib/swidtag/redhat.com/com.redhat.RHEL-8.0-
<architecture>.swidtag** Родительский каталог этих тегов также можно найти, следуя **/etc/swid/swidtags.d/redhat.com** символическая ссылка.

XML-подпись файлов тегов SWID может быть проверена с помощью **xmlsec1 verify** команда, например:

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

-

Сертификат центра сертификации для подписи кода также можно получить в [Product Signing Keys](#) страница на портале для клиентов.

(BZ#1636338)

5.5. ВИРТУАЛИЗАЦИЯ (МАШИННЫЙ ПЕРЕВОД)

AMD SEV для виртуальных машин KVM

В качестве предварительной версии технологии RHEL 8 представляет функцию защищенной зашифрованной виртуализации (SEV) для хост-компьютеров AMD EPYC, использующих гипервизор KVM. При включении на виртуальной машине (VM) SEV шифрует память VM, чтобы хост не мог получить доступ к данным на VM. Это повышает безопасность VM, если хост успешно заражен вредоносным ПО.

Обратите внимание, что количество виртуальных машин, которые могут использовать эту функцию одновременно на одном хосте, определяется оборудованием хоста. Современные процессоры AMD EPYC поддерживают 15 или менее работающих виртуальных машин, использующих SEV.

(BZ#1501618, BZ#1501607)

Вложенная виртуализация теперь доступна в IBM POWER 9

В качестве предварительной версии технологии теперь можно использовать функции вложенной виртуализации на хост-машинах RHEL 8, работающих в системах IBM POWER 9. Вложенная виртуализация позволяет виртуальным машинам (виртуальным машинам) KVM выступать в качестве гипервизоров, что позволяет запускать виртуальные машины внутри виртуальных машин.

Обратите внимание, что для работы вложенной виртуализации в IBM POWER 9 хост, гость и вложенные гости в настоящее время должны использовать одну из следующих ОС:

- RHEL 8
- RHEL 7 для POWER 9

(BZ#1505999)

ГЛАВА 6. УСТАРЕВШАЯ ФУНКЦИОНАЛЬНОСТЬ (МАШИННЫЙ ПЕРЕВОД)

В этой части представлен обзор функциональности, которая устарела в Red Hat Enterprise Linux 8.0.

Устаревшие функциональные возможности продолжают поддерживаться до конца срока службы Red Hat Enterprise Linux 8. Устаревшая функциональность, скорее всего, не будет поддерживаться в будущих основных выпусках этого продукта и не рекомендуется для новых развертываний. Самый последний список устаревших функций в конкретном основном выпуске см. В последней версии документации по выпуску.

Устаревшие аппаратные компоненты не рекомендуются для новых развертываний в текущих или будущих основных выпусках. Обновления драйверов оборудования ограничиваются только безопасностью и критическими исправлениями. Red Hat рекомендует заменить это оборудование как можно скорее.

Пакет может быть устаревшим и не рекомендуется для дальнейшего использования. При определенных обстоятельствах упаковка может быть удалена из продукта. Документация по продукту затем определяет более свежие пакеты, которые предлагают функциональность, аналогичную, идентичную или более продвинутую по сравнению с устаревшей, и предоставляет дальнейшие рекомендации.

6.1. УСТАНОВЩИК И СОЗДАНИЕ ОБРАЗА (МАШИННЫЙ ПЕРЕВОД)

--interactive вариант ignoredisk Команда Kickstart устарела

С использованием **--interactive option** в будущих выпусках Red Hat Enterprise Linux возникнет фатальная ошибка установки. Рекомендуется изменить файл кикстарта, чтобы удалить этот параметр.

(BZ#1637872)

6.2. ФАЙЛОВЫЕ СИСТЕМЫ И ХРАНИЛИЩЕ (МАШИННЫЙ ПЕРЕВОД)

NFSv3 через UDP отключен

Сервер NFS больше не открывается и не прослушивает сокет протокола пользовательских дейтаграмм (UDP) по умолчанию. Это изменение касается только NFS версии 3, поскольку для версии 4 требуется протокол управления передачей (TCP).

NFS через UDP больше не поддерживается в RHEL 8.

(BZ#1592011)

Целевой режим NVMe / FC устарел

Целевой режим транспортного протокола энергонезависимой памяти Express через Fibre Channel (NVMe / FC) ранее был доступен в качестве предварительного просмотра технологии в RHEL 7. В RHEL 8 целевой режим NVMe / FC устарел.

Включение портов адаптера шины хоста FC (HBA) в целевом режиме NVMe приводит к следующему сообщению об ошибке:

```
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. СЕТЕЙ (МАШИННЫЙ ПЕРЕВОД)

Сетевые сценарии устарели в RHEL 8

Сетевые сценарии устарели в Red Hat Enterprise Linux 8 и больше не предоставляются по умолчанию. Базовая установка предоставляет новую версию **ifup** а также **ifdown** сценарии, которые называют **Сетевой менеджер** обслуживание через **nmcli** инструмент. В Red Hat Enterprise Linux 8, чтобы запустить **ifup** и **ifdown** сценарии, NetworkManager должен быть запущен.

Обратите внимание, что пользовательские команды в **/sbin/ifup-local**, **ifdown-pre-local** а также **ifdown-local** скрипты не выполняются.

Если какой-либо из этих сценариев требуется, установка устаревших сетевых сценариев в системе все еще возможна с помощью следующей команды:

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

ifup а также **ifdown** скрипты ссылаются на установленные старые сетевые скрипты.

Вызов устаревших сетевых сценариев показывает предупреждение об их устаревании.

(BZ#1647725)

6.4. БЕЗОПАСНОСТЬ (МАШИННЫЙ ПЕРЕВОД)

DSA устарела в Red Hat Enterprise Linux 8

Алгоритм цифровой подписи (DSA) считается устаревшим в Red Hat Enterprise Linux 8. Механизмы аутентификации, которые зависят от ключей DSA, не работают в конфигурации по умолчанию. Обратите внимание, что **OpenSSH** клиенты не принимают ключи хоста DSA даже в политике LEGACY.

(BZ#1646541)

SSL2 Client Hello устарел в NSS

Безопасность транспортного уровня (TLS) протокол версии 1.2 и более ранние позволяют начать переговоры с **Client Hello** сообщение отформатировано так, чтобы оно было обратно совместимо с уровнем защищенных сокетов (SSL) версия протокола 2. Поддержка этой функции в службах сетевой безопасности (NSS) библиотека устарела и по умолчанию отключена.

Приложения, которым требуется поддержка этой функции, должны использовать новый **SSL_ENABLE_V2_COMPATIBLE_HELLO** API, чтобы включить его. Поддержка этой функции может быть полностью удалена в будущих выпусках Red Hat Enterprise Linux 8.

(BZ#1645153)

6.5. ВИРТУАЛИЗАЦИЯ (МАШИННЫЙ ПЕРЕВОД)

Снимки виртуальной машины не поддерживаются должным образом в RHEL 8

Текущий механизм создания снимков виртуальной машины (VM) устарел, так как он не работает надежно. Как следствие, рекомендуется не использовать моментальные снимки виртуальной машины в RHEL 8.

Обратите внимание, что новый механизм моментальных снимков виртуальной машины находится в стадии разработки и будет полностью реализован в будущем небольшом выпуске RHEL 8.

([BZ#1686057](#))

Cirrus VGA виртуальный тип графического процессора устарел

В будущем крупном обновлении Red Hat Enterprise Linux **Cirrus VGA** Устройство с графическим процессором больше не будет поддерживаться в виртуальных машинах KVM. Поэтому Red Hat рекомендует использовать **stdvga**, **Virtio-VGA**, или же **QXL** устройства вместо Cirrus VGA.

([BZ#1651994](#))

вирт-менеджер устарел

Приложение Virtual Machine Manager, также известное как **вирт-менеджер**, не рекомендуется. Веб-консоль RHEL 8, также известная как **кокпит**, предназначен стать его заменой в последующем выпуске. Поэтому рекомендуется использовать веб-консоль для управления виртуализацией в графическом интерфейсе. Однако в Red Hat Enterprise Linux 8.0 некоторые функции могут быть доступны только из **вирт-менеджер** или командная строка.

([JIRA:RHELPLAN-10304](#))

ГЛАВА 7. ИЗВЕСТНЫЕ ВОПРОСЫ (МАШИННЫЙ ПЕРЕВОД)

Эта часть описывает известные проблемы в Red Hat Enterprise Linux 8.

7.1. УСТАНОВЩИК И СОЗДАНИЕ ОБРАЗА (МАШИННЫЙ ПЕРЕВОД)

auth а также authconfig Команды Kickstart требуют хранилища AppStream

authselect-compat пакет требуется **auth** а также **authconfig** Команды кикстарта во время установки. Без этого пакета установка завершится неудачно, если **auth** или же **authconfig** используются. Однако по замыслу **authselect-compat** Пакет доступен только в хранилище AppStream.

Чтобы обойти эту проблему, убедитесь, что репозитории BaseOS и AppStream доступны для установщика или используйте **authselect** Команда Kickstart во время установки.

(BZ#1640697)

Копирование содержимого Binary DVD.iso файл в раздел пропускает .treeinfo а также .discinfo файлы

При копировании содержимого файла образа RHEL 8.0 Binary DVD.iso в раздел для локальной установки, «*» в **cp <path>/* <mounted partition>/dir** команда не может скопировать **.treeinfo** а также **.discinfo** файлы, которые необходимы для успешной установки. В результате репозитории BaseOS и AppStream не загружаются, и в журнале появляется сообщение об отладке. **anaconda.log** файл является единственной записью проблемы.

Чтобы обойти проблему, скопируйте отсутствующие **.treeinfo** а также **.discinfo** файлы в раздел.

(BZ#1692746)

7.2. ЯДРО (МАШИННЫЙ ПЕРЕВОД)

i40iw модуль не загружается автоматически при загрузке

Из-за того, что многие сетевые карты i40e не поддерживают iWarp и i40iw модуль не полностью поддерживает приостановку / возобновление, этот модуль не загружается автоматически по умолчанию, чтобы обеспечить правильную работу приостановки / возобновления. Чтобы обойти эту проблему, вручную отредактируйте **/lib/udev/rules.d/90-rdma-hw-modules.rules** файл для включения автоматической загрузки i40iw

Также обратите внимание, что если на том же компьютере установлено другое устройство RDMA с устройством i40e, устройство RDMA, отличное от i40e, запускает RDMA сервис, который загружает все включенные модули стека RDMA, включая i40iw модуль.

(BZ#1623712)

Система иногда перестает отвечать на запросы, когда подключено много устройств

Когда Red Hat Enterprise Linux 8 настраивает большое количество устройств, на системной консоли появляется большое количество сообщений консоли. Это происходит, например, когда существует большое количество номеров логических единиц (LUN) с несколькими путями к каждому LUN. Поток консольных сообщений, в дополнение к другой работе, которую выполняет ядро, может привести к тому, что сторожевой таймер ядра вызовет панику ядра, поскольку ядро кажется зависшим.

Поскольку сканирование происходит в начале цикла загрузки, система перестает отвечать на запросы, когда подключено много устройств. Это обычно происходит во время загрузки.

Если **kdump** включен на вашем компьютере во время события сканирования устройства после загрузки, жесткая блокировка приводит к захвату **vmcore** образ.

Чтобы обойти эту проблему, увеличьте таймер блокировки сторожевого таймера. Для этого добавьте **watchdog_thresh=N** опция в командной строке ядра. замещать **N** с количеством секунд:

- Если у вас менее тысячи устройств, используйте **30**
- Если у вас более тысячи устройств, используйте **60**

Для хранения число устройств - это количество путей ко всем LUN: как правило, число **/dev/sd*** устройства.

После применения обходного пути система больше не перестает отвечать на запросы при настройке большого количества устройств.

(BZ#1598448)

KSM иногда игнорирует запросы памяти NUMA

Когда функция общей памяти ядра (KSM) включена с параметром «merge_across_nodes = 1», KSM игнорирует политики памяти, установленные функцией `mbind()`, и может объединять страницы из некоторых областей памяти в узлы неравномерного доступа к памяти (NUMA) которые не соответствуют политике.

Чтобы обойти эту проблему, отключите KSM или установите для параметра `merge_across_nodes` значение «0», если используется привязка памяти NUMA с QEMU. В результате политики памяти NUMA, настроенные для виртуальной машины KVM, будут работать должным образом.

(BZ#1153521)

7.3. УПРАВЛЕНИЕ ПРОГРАММНЫМ ОБЕСПЕЧЕНИЕМ (МАШИННЫЙ ПЕРЕВОД)

Бег yum list под некорневым пользователем вызывает YUM авария

При запуске **yum list** команда под пользователем без полномочий `root` после **libdnf** пакет обновлен, YUM может закончиться неожиданно. Если вы нажали эту ошибку, запустите **yum list** под `root`, чтобы решить проблему. В результате последующие попытки запуска **yum list** под пользователем без полномочий `root` больше не вызывает YUM авария.

(BZ#1642458)

yum(8) Страница man неправильно упоминает yum module profile команда

yum(8) на странице руководства неверно указано, что **YUM** инструмент управления пакетами включает в себя **yum module profile** Команда для предоставления информации о профилях модуля. Тем не менее, эта команда больше не доступна и при использовании, **YUM** отображает сообщение об ошибке о неверной команде. Для получения подробной информации о профилях модулей, используйте новый **yum module info --profile** команда вместо

(BZ#1622580)

yum-plugin-aliases недоступен в данный момент

yum-plugin-aliases пакет, который обеспечивает **alias** Команда для добавления пользовательских псевдонимов yum, в настоящее время недоступна. Следовательно, в настоящее время невозможно использовать псевдонимы.

(BZ#1647760)

yum-plugin-changelog недоступен в данный момент

yum-plugin-changelog Пакет, который позволяет просматривать журналы изменений пакета до и после обновления пакета, в настоящее время недоступен.

(BZ#1581191)

7.4. ИНФРАСТРУКТУРНЫЕ УСЛУГИ (МАШИННЫЙ ПЕРЕВОД)

Настроенный не устанавливает параметры командной строки загрузки ядра

Настроенный инструмент не поддерживает спецификацию загрузчика (BLS), которая включена по умолчанию. Как следствие, **Настроенный** не устанавливает определенные параметры командной строки загрузки ядра, что вызывает некоторые проблемы, такие как снижение производительности или отсутствие изоляции ядер ЦП. Чтобы обойти эту проблему, отключите BLS и перезапустите **Настроенный**

1. Установите *неряшливый* пакет.
2. Удалите следующую строку из **/etc/default/grub** файл:


```
GRUB_ENABLE_BLSCFG=true
```
3. Повторно сгенерируйте **grub2.cfg** запустите файл для систем без EFI:

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

или для систем EFI:

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

4. Запустить снова **Настроенный** запустив:

```
перезапуск systemctl настроен
```

В следствии, **Настроенный** устанавливает параметры загрузки ядра, как ожидалось.

(BZ#1576435)

7.5. ОБОЛОЧКИ И ИНСТРУМЕНТЫ КОМАНДНОЙ СТРОКИ (МАШИННЫЙ ПЕРЕВОД)

Python связывание net-snmp пакет недоступен

Net-SNMP Набор инструментов не обеспечивает привязку для **Python 3**, по умолчанию **Python** реализация в RHEL 8. Как следствие, **python-net-snmp**, **python2-net-snmp**, или же **python3-net-snmp** пакеты недоступны в RHEL 8.

(BZ#1584510)

7.6. ДИНАМИЧЕСКИЕ ЯЗЫКИ ПРОГРАММИРОВАНИЯ, ВЕБ-СЕРВЕРЫ И СЕРВЕРЫ БАЗ ДАННЫХ (МАШИННЫЙ ПЕРЕВОД)

Серверы баз данных не устанавливаются параллельно

mariadb а также **mysql** модули не могут быть установлены параллельно в RHEL 8.0 из-за конфликтующих пакетов RPM.

По своей конструкции невозможно установить более одной версии (потока) одного и того же модуля параллельно. Например, вам нужно выбрать только один из доступных потоков из **postgresql** модуль, либо **10** (по умолчанию) или **9.6** Параллельная установка компонентов возможна в Red Hat Software Collections для RHEL 6 и RHEL 7. В RHEL 8 различные версии серверов баз данных могут использоваться в контейнерах.

(BZ#1566048)

Проблемы в mod_cgid протоколирование

Если **mod_cgid** Модуль Apache httpd используется в многопоточном многопоточном модуле (MPM), который используется по умолчанию в RHEL 8, возникают следующие проблемы с журналированием:

- **stderr** вывод сценария CGI не имеет префикс стандартной информации о времени.
- **stderr** вывод сценария CGI неправильно перенаправлен в файл журнала, относящийся к **VirtualHost** , если настроен.

(BZ#1633224)

IO::Socket::SSL Модуль Perl не поддерживает TLS 1.3

В RHEL 8 были реализованы новые функции протокола TLS 1.3, такие как возобновление сеанса или аутентификация после рукопожатия. **OpenSSL** библиотека, но не в **Net::SSLeay** Perl, и, следовательно, недоступны в **IO::Socket::SSL** Perl модуль. Следовательно, аутентификация сертификата клиента может завершиться неудачно, а восстановление сеансов может быть медленнее, чем при использовании протокола TLS 1.2.

Чтобы обойти эту проблему, отключите использование TLS 1.3, установив **SSL_version** вариант к **!TLSv1_3** значение при создании **IO::Socket::SSL** объект.

(BZ#1632600)

7.7. УПРАВЛЕНИЕ ИДЕНТИФИКАЦИЕЙ (МАШИННЫЙ ПЕРЕВОД)

Кэш учетных данных KCM не подходит для большого количества учетных данных в одном кэше учетных данных.

Если кэш учетных данных содержит слишком много учетных данных, выполняются операции Kerberos, такие как Kinit, сбой из-за жесткого ограничения на буфер, используемый для передачи данных между SSSD-KCM компонент и базовая база данных.

Чтобы обойти эту проблему, добавьте `ccache_storage = memory` вариант в KCM раздел `/etc/sss/sss.conf` файл. Это инструктирует KCM респондент хранит только кэши учетных данных в памяти, а не постоянно. Если вы сделаете это, перезапустите систему или SSSD-KCM очищает кэши учетных данных. Обратите внимание, что KCM может обрабатывать кэши размером до 64 кБ.

(BZ#1448094)

Конфликтующие значения тайм-аута не позволяют SSSD подключаться к серверам

Некоторые значения времени ожидания по умолчанию, относящиеся к операциям аварийного переключения, используемым демоном System Security Services (SSSD), конфликтуют. Следовательно, значение тайм-аута, зарезервированное для SSSD для связи с одним сервером, не позволяет SSSD пытаться использовать другие серверы до истечения времени ожидания операции соединения. Чтобы обойти проблему, установите значение `ldap_opt_timeout` параметр тайм-аута выше, чем значение `dns_resolver_timeout` параметр и установите значение `dns_resolver_timeout` параметр выше, чем значение `dns_resolver_op_timeout` параметр.

(BZ#1382750)

Использование смарт-карты для входа в веб-интерфейс IdM не работает

Когда пользователь пытается войти в веб-интерфейс Identity Management (IdM) с использованием сертификата, хранящегося на его смарт-карте, код интерфейса D-Bus для System Security Services Daemon использует некорректный обратный вызов для поиска пользователя. Следовательно, поиск вылетает. Чтобы обойти проблему, используйте другие методы аутентификации.

(BZ#1642508)

IdM сервер не работает в FIPS

Из-за неполной реализации коннектора SSL для Tomcat сервер IdM с установленным сервером сертификатов не работает на машинах с включенным режимом FIPS.

(BZ#1673296)

nuxwdog Сбой службы в средах HSM и требует установки keyutils пакет в среде без HSM

`nuxwdog` сторожевой сервис был интегрирован в систему сертификатов. Как следствие, `nuxwdog` больше не предоставляется как отдельный пакет. Чтобы использовать службу сторожевого таймера, установите `pki-server` пакет.

Обратите внимание, что `nuxwdog` сервис имеет следующие известные проблемы:

- **nuxwdog** служба не работает, если вы используете аппаратный модуль хранения (HSM). Для этой проблемы нет обходного пути.
- В среде без HSM Red Hat Enterprise Linux 8.0 не устанавливает автоматически **keyutils** Пакет как зависимость. Чтобы установить пакет вручную, используйте **dnf install keyutils** команда.

(BZ#1652269)

7.8. КОМПИЛЯТОРЫ И ИНСТРУМЕНТЫ РАЗРАБОТКИ (МАШИННЫЙ ПЕРЕВОД)

Синтетические функции, генерируемые GCC, путают SystemTap

Оптимизация GCC может генерировать синтетические функции для частично встроенных копий других функций. Такие инструменты, как SystemTap и GDB, не могут отличить эти синтетические функции от реальных функций. Как следствие, SystemTap может размещать зонды как в точках входа в синтетическую, так и в реальную функцию, и, таким образом, регистрировать несколько попаданий зондов для одного вызова реальной функции.

Чтобы обойти эту проблему, сценарии SystemTap должны быть адаптированы к таким мерам, как обнаружение рекурсии и подавление проб, связанных с встроенными частичными функциями. Например, скрипт

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

Можно попытаться избежать описанной проблемы следующим образом:

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

Обратите внимание, что в этом примере сценария не учитываются все возможные сценарии, такие как пропущенные kprobes или kretprobes или подлинная предполагаемая рекурсия.

(BZ#1169184)

Itrace инструмент не сообщает о вызовах функций

Из-за улучшений бинарного упрочнения, применяемых ко всем компонентам RHEL, **Itrace** Инструмент больше не может обнаруживать вызовы функций в двоичных файлах, поступающих из компонентов RHEL. Как следствие, **Itrace** вывод пуст, потому что он не сообщает о каких-либо обнаруженных вызовах при использовании таких двоичных файлов. В настоящее время нет обходного пути.

Как примечание, **ltrace** может правильно сообщать о вызовах в пользовательских двоичных файлах, созданных без соответствующих флагов защиты.

(BZ#1618748, BZ#1655368)

7.9. ФАЙЛОВЫЕ СИСТЕМЫ И ХРАНИЛИЩЕ (МАШИННЫЙ ПЕРЕВОД)

Невозможно обнаружить цель iSCSI с помощью `iscsiuio` пакет

Red Hat Enterprise Linux 8 не разрешает одновременный доступ к областям регистрации PCI. Как следствие, **could not set host net params (err 29)** ошибка была установлена, и соединение с порталом обнаружения не удалось. Чтобы обойти эту проблему, установите параметр ядра **iomem=relaxed** в командной строке ядра для разгрузки iSCSI. Это конкретно включает в себя любую разгрузку с использованием **bnx2i** Водитель. В результате соединение с порталом обнаружения теперь успешно и **iscsiuio** Пакет теперь работает правильно.

(BZ#1626629)

Параметр монтирования XFS DAX несовместим с общими экстендами данных копирования при записи

Файловая система XFS, отформатированная с помощью функции общих экстендов копирования при записи, несовместима с **-o dax** опция монтирования. Как следствие, монтирование такой файловой системы с **-o dax** выходит из строя.

Чтобы обойти проблему, отформатируйте файловую систему с помощью **reflink=0** Опция метаданных для отключения общих экстендов данных копирования при записи:

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

В результате монтируется файловая система с **-o dax** успешно

(BZ#1620330)

7.10. СЕТЕЙ (МАШИННЫЙ ПЕРЕВОД)

`nftables` не поддерживает многомерный набор типов IP

`nftables` инфраструктура фильтрации пакетов не поддерживает типы наборов с конкатенациями и интервалами. Следовательно, вы не можете использовать многомерные типы IP-наборов, такие как **hash:net,port**, с `nftables`

Чтобы обойти эту проблему, используйте `iptables` рамки с `ipset` инструмент, если вам требуется многомерный набор типов IP.

(BZ#1593711)

7.11. БЕЗОПАСНОСТЬ (МАШИННЫЙ ПЕРЕВОД)

OpenSCAP `rpmverifypackage` не работает правильно

`chdir` а также `chroot` системные вызовы вызываются дважды `rpmverifypackage` зонд. Следовательно, ошибка возникает, когда зонд используется во время OpenSCAP

сканировать с использованием пользовательского содержимого Open Vulnerability and Assessment Language (OVAL).

Чтобы обойти эту проблему, не используйте `rpmverifypackage_test` OVAL проверить в вашем контенте или использовать только контент из `scap-security-guide` пакет где `rpmverifypackage_test` не используется

(BZ#1646197)

libssh не соответствует общесистемной политике шифрования

`libssh` библиотека не поддерживает общесистемные параметры криптографической политики. Как следствие, набор поддерживаемых алгоритмов не изменяется, когда администратор изменяет уровень политик шифрования, используя `update-crypto-policies` команда.

Чтобы обойти эту проблему, набор рекламируемых алгоритмов должен быть установлен индивидуально каждым приложением, которое использует `libssh` В результате, когда для системы задан уровень политики LEGACY или FUTURE, приложения, использующие `libssh` вести себя непоследовательно по сравнению с `OpenSSH`

(BZ#1646563)

SCAP Workbench не в состоянии генерировать основанные на результатах исправления из настроенных профилей

Следующая ошибка возникает при попытке создать роли исправления на основе результатов из настроенного профиля с помощью `SCAP Workbench` инструмент:

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

Чтобы обойти эту проблему, используйте `oscap` команда с `--tailoring-file` вариант.

(BZ#1640715)

OpenSCAP rpmverifyfile не работает

`OpenSCAP` сканер неправильно меняет текущий рабочий каталог в автономном режиме, а `fchdir` функция не вызывается с правильными аргументами в `OpenSCAP rpmverifyfile` зонд. Следовательно, сканирование произвольных файловых систем с использованием `oscap-chroot` команда не выполняется, если `rpmverifyfile_test` используется в контенте SCAP. В следствии, `oscap-chroot` прерывается в описанном сценарии.

(BZ#1636431)

Утилита для проверки безопасности и соответствия контейнеров недоступна

В Red Hat Enterprise Linux 7 `oscap-docker` Утилита может использоваться для сканирования контейнеров Docker на основе технологий Atomic. В Red Hat Enterprise Linux 8, связанной с Docker и Atomic `OpenSCAP` Команды недоступны. В следствии, `oscap-docker` или эквивалентная утилита для проверки контейнеров на предмет безопасности и соответствия в настоящее время недоступна в RHEL 8.

(BZ#1642373)

ГЛАВА 8. ЗАМЕТНЫЕ ИЗМЕНЕНИЯ В КОНТЕЙНЕРАХ (МАШИННЫЙ ПЕРЕВОД)

Набор изображений контейнеров доступен для Red Hat Enterprise Linux (RHEL) 8.0. Заметные изменения включают в себя:

- Docker не входит в RHEL 8.0. Для работы с контейнерами используйте **podman**, **buildah**, **skopeo**, а также **RunC** инструменты.

Для получения информации об этих инструментах и об использовании контейнеров в RHEL 8 см. [Building, running, and managing containers](#)

- **podman** инструмент был выпущен как полностью поддерживаемая функция. **podman** Инструмент управляет модулями, изображениями контейнеров и контейнерами на одном узле. Он построен на **libpod** библиотека, которая позволяет управлять контейнерами и группами контейнеров, называемыми модулями.

Чтобы научиться использовать **podman**, увидеть [Building, running, and managing containers](#)

- В RHEL 8 GA недавно появились универсальные базовые изображения (UBI) Red Hat. UBI заменяют некоторые ранее предоставленные Red Hat изображения, такие как стандартные и минимальные базовые образы RHEL. В отличие от старых изображений Red Hat, UBI свободно распространяются. Это означает, что они могут быть использованы в любой среде и доступны где угодно. Вы можете использовать их, даже если вы не являетесь клиентом Red Hat.

Для документации UBI, см. [Building, running, and managing containers](#)

- В RHEL 8 GA доступны дополнительные образы контейнеров, которые предоставляют компоненты AppStream, для которых образы контейнеров распространяются вместе с **Коллекции программного обеспечения Red Hat** в RHEL 7. Все эти изображения RHEL 8 основаны на **ubi8** базовое изображение.
- Контейнерные образы ARM для 64-битной архитектуры ARM полностью поддерживаются в RHEL 8.
- **rhel-tools** Контейнер был удален в RHEL 8. **sos** а также **redhat-support-tool** инструменты предоставляются в **support-tools** контейнер. Системные администраторы также могут использовать этот образ в качестве основы для построения образа контейнера системных инструментов.
- Поддержка контейнеров без корней доступна как предварительный просмотр технологии в RHEL 8. Контейнеры без рута - это контейнеры, которые создаются и управляются обычными пользователями системы без прав администратора.

ПРИЛОЖЕНИЕ А. СПИСОК БИЛЕТОВ ПО КОМПОНЕНТАМ

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

Составная часть	Билеты
gdm	BZ#1589678
glibc	BZ#1512004, BZ#1376834, BZ#1512010, BZ#1304448, BZ#1512009, BZ#1512006, BZ#1514839, BZ#1533608
go-toolset	BZ#1512570
httpd	BZ#1633224, BZ#1632754
iproute	BZ#1640991
iptables	BZ#1644030, BZ#1564596
iscsi-initiator-utils	BZ#1626629, BZ#1582099
kernel-rt	BZ#1592977
kernel	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
kexec-tools	BZ#1520209
kmod-kvdo	BZ#1534087, BZ#1639512
libdnf	BZ#1642458
libreswan	BZ#1657854
libssh	BZ#1485241
ltrace	BZ#1618748, BZ#1584322
lvm2	BZ#1643553, BZ#1643543, BZ#1643545, BZ#1643547, BZ#1643549, BZ#1643562, BZ#1643576
mariadb	BZ#1637034
net-snmp	BZ#1584510
nfs-utils	BZ#1592011, BZ#1639432
nftables	BZ#1593711

Составная часть	Билеты
nginx	BZ#1545526
nodejs-10-module	BZ#1622118
nss	BZ#1645153
nuxwdog	BZ#1652269
openldap	BZ#1570056
opensc	BZ#1595626
openscap	BZ#1646197, BZ#1636431, BZ#1642373, BZ#1618484, BZ#1614273, BZ#1618464
openssh	BZ#1622511, BZ#1228088
pacemaker	BZ#1543494
pcs	BZ#1578891, BZ#1591308, BZ#1615420, BZ#1158816, BZ#1542288, BZ#1549535, BZ#1620190, BZ#1566430, BZ#1595829, BZ#1436217, BZ#1578955, BZ#1596050, BZ#1554310, BZ#1638852, BZ#1640477, BZ#1619620
perl-IO-Socket-SSL	BZ#1632600
perl	BZ#1511131
pki-core	BZ#1565073, BZ#1623444, BZ#1566360, BZ#1394069, BZ#1669257 БЗ № 1656856, BZ#1673296
pykickstart	BZ#1637872, BZ#1612061
qemu-kvm	BZ#1508139, BZ#1559240, BZ#1497911, BZ#1578855, BZ#1651994, BZ#1621817, BZ#1508137, BZ#1519004
redhat-release	BZ#1636338
rsyslog	BZ#1613880, BZ#1542497, BZ#1614179, BZ#1619645
scap-security-guide	BZ#1618528, BZ#1618518
scap-workbench	BZ#1640715
selinux-policy	BZ#1592244, BZ#1594111, BZ#1549772, BZ#1626446

Составная часть	Билеты
setup	BZ#1591969
sos	BZ#1559836
sssd	BZ#1448094, BZ#1382750, BZ#1642508, BZ#1620123
subversion	BZ#1571415
swig-3.0-module	BZ#1660051
tomcatjss	BZ#1424966, BZ#1636564
tuned	BZ#1576435, BZ#1565598
valgrind	BZ#1500481, BZ#1538009
virt-manager	BZ#1599777, BZ#1643609
wpa_supplicant	BZ#1582538
Другой	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, BZ#1686057 , BZ#1582530, BZ#1581496, BZ#1650618, BZ#1650675, BZ#1650701 , 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