

Реферат

«Система инициализации SystemD»

Ярослав Антонович Меркулов

НПИбд-03-24

Введение

Современные операционные системы требуют эффективных и надежных механизмов управления процессами и службами, особенно в момент загрузки системы. Система инициализации играет ключевую роль в обеспечении правильного запуска, остановки и мониторинга различных компонентов операционной системы.

Одной из наиболее популярных и широко используемых систем инициализации в современных дистрибутивах Linux является SystemD. Она пришла на смену традиционным системам, таким как SysVinit, предлагая более гибкий, быстрый и модульный подход к управлению службами. Сегодня мы рассмотрим историю, основные принципы работы SystemD, преимущества и влияние на процесс загрузки операционной системы.

История создания

Давайте поговорим об истории создания и популяризации systemd в линуксе. 15 лет назад, в далеком 2010 году systemd был разработан Ленартом Поттерингом.



Рисунок 1 – Ленарт Поттеринг

Эта система инициализации устроила революцию в мире линукса.

В издании UNIX от 1979 года задач у `init` — процесса с PID 1 — две, это выполнить при старте `/etc/rc` и создать процессы `getty` и `login` (DrArgentum, 2024). Позже, по мере развития линукса, к задачам стали прибавляться все новые и новые. И в конце концов осталась одна основная задача инициализатора — привести систему в полную готовность к работе, запустить базовые процессы и пользовательское пространство.

Но время неумолимо двигалось вперед. Теперь стали появляться сервисы, с множествами юнитами и скриптами, и при помощи старой SysV `init` стало неудобно управлять системой. Сервисы в SysV запускались последовательно, так что, если захочется контролировать время запуска одного

сервиса — придется контролировать все сервисы. А также это было просто медленно. Зависимости между сервисами в SysV были сложными. Также SysV просто запускал и отключал сервисы, он не имел ни контроля ресурсов, не журналирования, не управления сетью. Также SysV имел множество разрозненных утилит, в отличие от systemd.

И под конец — SysV имел проблемы с надежностью и безопасностью. Пример ненадежности — это небезопасное выключение системы. SysVinit полагается исключительно на загрузочные скрипты, чтобы отмонтировать файловые системы и остановить процессы. Это крайне небезопасно и может вылиться в потерю данных в худших случаях.

Кроме SysV был еще UpStart, разработанный Canonical. Его главная концепция — event-driven подход. Запуск системы — это событие, новое устройство — событие, запуск демона — событие. Процессы могли подписываться на события и запускаться только при явной необходимости. Upstart совместим с sysvinit-скриптами, благодаря этому он был быстро интегрирован во многие дистрибутивы Linux.

Тем не менее, после того как большинство дистрибутивов Linux по тем или иным причинам перешли на systemd, разработка сошла на нет: последний релиз был в 2014-м году, и сегодня Upstart используется только в Chrome OS.

Подведем итог. После нескольких месяцев борьбы, дистрибутивы стали мало-помалу принимать systemd — Fedora, OpenSuSE. В августе 2012 года Arch Linux переключился также на systemd.

Что такое Systemd?

Systemd — это система инициализации и менеджер служб для операционных систем на базе Linux. Он предназначен для управления процессами и службами, упрощая запуск, остановку и мониторинг служб, а также их конфигурацию. Systemd стал стандартной системой инициализации для многих современных дистрибутивов Linux, таких как Fedora, CentOS, Ubuntu и Debian (Fedorov, 2024).

Основные характеристики Systemd

- **Параллельный запуск:** Systemd способен одновременно запускать несколько служб, что существенно сокращает время загрузки системы по сравнению с классическими системами инициализации.
- **Управление службами:** Он предоставляет команды для управления службами (например, `start`, `stop`, `restart`, `status`) и позволяет отслеживать их состояние.
- **Системные юниты:** Systemd использует концепцию юнитов, которые представляют собой отдельные объекты (например, службы, таймеры, точки монтирования и др.). Каждый юнит имеет свой конфигурационный файл, который определяет, как он должен вести себя.
- **Логирование:** Systemd включает встроенную систему журналирования — `journald`, которая собирает и хранит логи системы и служб, что упрощает диагностику и мониторинг.
- **Зависимости:** Он управляет зависимостями между юнитами, позволяя определять, какие службы должны быть запущены или остановлены в определенном порядке.

- **Таймеры:** вместо традиционных cron-заданий, Systemd предоставляет возможность использовать таймеры для планирования выполнения задач.

Преимущества использования Systemd

- **Ускорение загрузки:** Параллельный запуск служб сокращает время загрузки системы.
- **Упрощенное управление:** Централизованное управление службами через единый интерфейс.
- **Улучшенное логирование:** Интегрированная система логирования позволяет более удобно управлять журналами.
- **Гибкость:** Возможность конфигурирования и расширения благодаря поддержке различных типов юнитов.

Утилиты Systemd

SystemD включает в себя множество утилит, которые помогают управлять службами, процессами и конфигурацией системы. Вот основные утилиты SystemD и их функции:

1. **systemctl**: Это основная утилита для управления службами и состоянием системы. С её помощью можно запускать, останавливать, перезапускать службы, а также проверять их статус, включать или отключать автозапуск при загрузке.

Примеры команд:

1. `systemctl start <имя_службы>` — запустить службу.
 2. `systemctl stop <имя_службы>` — остановить службу.
 3. `systemctl status <имя_службы>` — показать статус службы.
 4. `systemctl enable <имя_службы>` — включить автозапуск службы при загрузке.
2. **journalctl**: Утилита для просмотра логов, собранных системой журналирования `journald`. Она позволяет фильтровать и искать записи по различным критериям, таким как время, уровень важности и идентификатор службы.

Пример команды:

1. `journalctl -u <имя_службы>` — показать логи для конкретной службы.
3. **systemd-analyze**: Утилита для анализа времени загрузки системы и диагностики проблем с производительностью. Она предоставляет информацию о времени, затраченном на загрузку различных компонентов.

Пример команды:

1. `systemd-analyze blame` — показать время загрузки всех служб в порядке убывания.
4. **loginctl**: Утилита для управления сессиями пользователей и состоянием входа в систему. С её помощью можно просматривать активные сессии, завершать их или блокировать пользователей.
5. **hostnamectl**: Утилита для управления настройками имени хоста системы. Позволяет изменять имя хоста и просматривать информацию о системе.
6. **systemd-tmpfiles**: Утилита для управления временными файлами и каталогами в системе. Она может создавать, удалять или очищать временные файлы на основе конфигурационных файлов.
7. **localectl**: Утилита для настройки локализации системы (языка и раскладки клавиатуры). Позволяет изменять параметры локали и раскладки клавиатуры.
8. **timedatectl**: Утилита для управления настройками времени и даты на системе. Позволяет устанавливать часовой пояс, синхронизировать время с сетевыми серверами времени (NTP) и изменять формат отображения времени.

Эти утилиты делают SystemD мощным инструментом для администраторов систем, позволяя эффективно управлять службами и процессами в Linux-системах.

Вывод

Система инициализации SystemD представляет собой значительный шаг вперед в управлении службами и процессами в операционных системах Linux. Благодаря своей модульной архитектуре, поддержке параллельного запуска и динамическому управлению зависимостями, SystemD обеспечивает более быструю загрузку системы и упрощает администрирование. Утилиты, входящие в состав SystemD, такие как `systemctl`, `journalctl` и `loginctl`, предоставляют администраторам мощные инструменты для управления состоянием системы, мониторинга логов и настройки пользовательских сессий.

Несмотря на критику со стороны некоторых пользователей за сложность и монолитность, SystemD продолжает развиваться и внедрять новые функции, что делает её важным компонентом современных дистрибутивов Linux (Opointer.net, 2010). В результате, SystemD не только улучшает производительность систем, но и способствует более удобному и эффективному управлению ими. В будущем можно ожидать дальнейшего развития этой системы инициализации, что позволит ей оставаться актуальной в быстро меняющемся мире технологий.

Список литературы

1. *0pointer.net*. (30 Апрель 2010 г.). Получено из Rethinking PID 1:
<https://0pointer.de/blog/projects/systemd.html>
2. DrArgentum. (1 Июль 2024 г.). *systemD с 0 до 1: библия сисадмина*. Получено из
habr.com: <https://habr.com/ru/companies/timeweb/articles/824146/>
3. Fedorov, D. (1 Ноябрь 2024 г.). *Что такое Systemd простыми словами*.
Получено из serverspace.ru:
https://serverspace.ru/support/glossary/systemd/?utm_source=google.com&utm_medium=organic&utm_campaign=google.com&utm_referrer=google.com