Play with Bash. SystemD.

SKILLFACTORY

Содержание

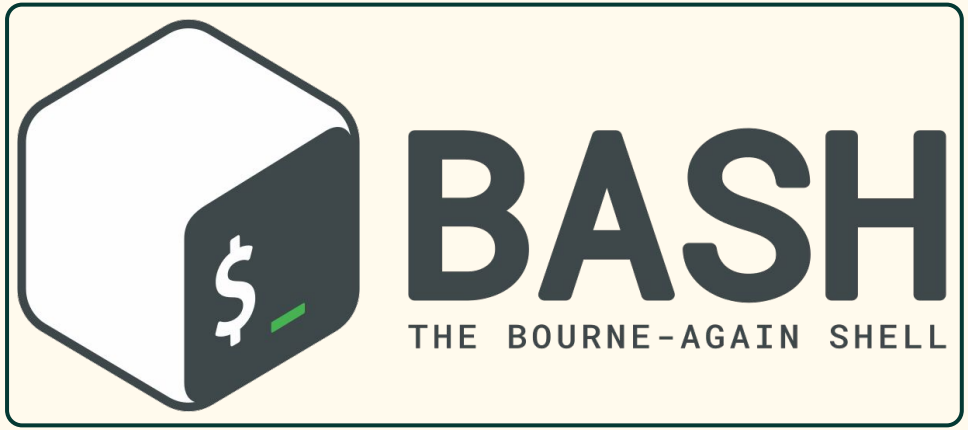
- Bash — сценарий оболочки
- SystemD — система инициализации Linux

Оболочка bash

Burn Shell (sh) – старая и распространенная оболочка Unix систем.

Burn Again Shell (bash) – расширенная версия Burn Shell. Оболочка по умолчанию в системах Linux.

.bashrc – конфигурационный сценарий оболочки Bash, который запускается при входе пользователя в систему.



Сценарий оболочки

Исполняемый файл с набором команд, которые необходимо выполнить в определенном порядке и/или условиями.

#!/bin/bash – шебанг – указывает путь к интерпретатору оболочки

#!/usr/bin/env bash – универсальный вариант

Для запуска сценария оболочки необходимы права «+x»

Переменные окружения

Именованный объект, содержащий текстовую информацию, которую могут использовать программы в рамках сессии

PATH – содержит пути исполняемых файлов

PWD – полный путь текущей директории

HOME – полный путь домашнего каталога

SHELL – имя командной оболочки текущего сеанса

USER – имя пользователя

Символ ***$*** используется для разыменования переменной

echo $PATH

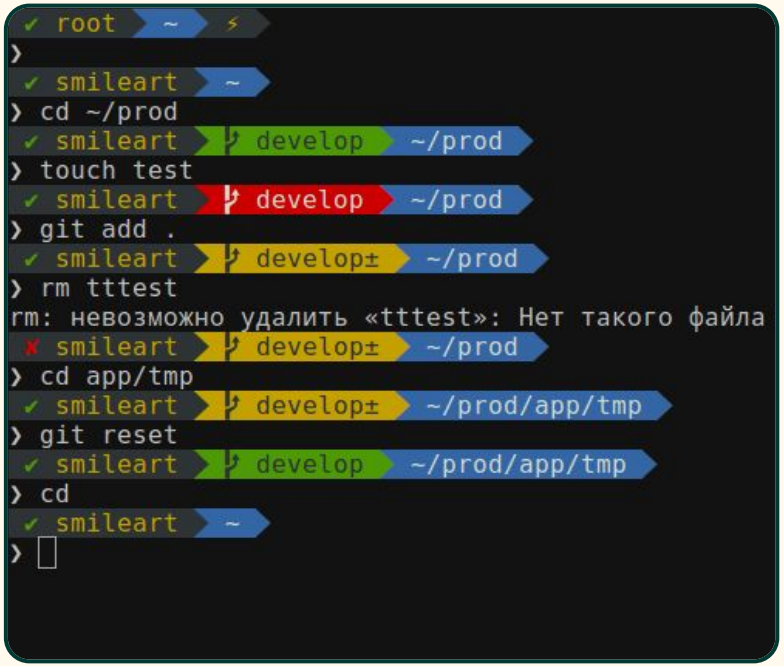
Язык **Bash** чувствителен к регистру. Переменные могут содержать цифры, но не могут начинаться с нее.

Приглашение

PS1 – содержит шаблон приглашения

Задается в файле *.bashrc* в формате:

PS1="\u@\H :"



Параметры и ключи

$0 – имя сценария оболочки

$1 – первый параметр

$2 – второй параметр

…

$9 – девятый параметр

$@ – содержит все параметры по отдельности

$* – содержит все параметры в виде одной строки

$? – статус ошибки

Ключи:

- ➔ ***-a** : вывести все объекты*
- ➔ ***-c**: произвести подсчет*
- ➔ ***-d**: указать директорию*
- ➔ ***-f**: указать файл для считывания данных*
- ➔ ***-h**: вывести справку по команде*
- ➔ ***-r**: рекурсивная обработка*
- ➔ ***-y**: ответить «да» на все вопросы*

alias, export, type

Alias

Выводит и задает псевдонимы

alias ll=ls -la

Export

Выводит и задает переменные среды

export PATH=$PATH:/mnt

Type

Вывод информации о типе команды

type cat

read, echo

read — команда чтения строки из стандартного ввода.

- ➔ **-a**: массив
- ➔ **-d**: разделитель
- ➔ **-p**: строка приветствия перед вводом
- ➔ **-t**: время ожидания
- ➔ **-u**: чтение из файлового дескриптора
- ➔ **-s**: не выводит ввод в терминал

echo — команда записи в стандартный вывод.

- ➔ **-n**: не добавлять символ новой строки
- ➔ **-e**: интерпретировать специальные символы
  - ◆ *n* – новая строка
  - ◆ *a* – сигнал
  - ◆ *0nnn* – символ ASCII
  - ◆ *r* – возврат каретки
  - ◆ *t* – символ табуляции

Условный оператор

if *условие;*

then *команда*

else *команда*

fi

условие

[-d abcd] :является ли файл abcd каталогом?
[-r abcd] :есть ли доступ на чтение к файлу abcd?
[-s abcd] :файл abcd имеет ненулевой размер (он не пуст)?
[“$STRING” = STRING1] :равны ли строки STRING и STRING1?
[“$DIGIT” -eq “$DIGIT1”] :равны ли числа DIGIT и DIGIT1?

Условный оператор

if *условие*

then *команда*

elif *условие*

then *команда*

else *команда*

fi

условие

&& – логическое И

|| – логическое ИЛИ

!= – не равно

== – равно

! – отрицание

case, select

```
number=one
case $number in
    one) echo 1;;
    two) echo 2;;
    *) echo something wrong ;;
esac
```

```
select number in 1 2 3
do
  case $number in
    1) echo One;;
    2) echo Two;;
    3) echo Three;;
    *) break ;;
  esac
done
```

For

for *переменная* **in** *список значений*

do *команды*

done

for *number* **in** *1 two "line № 3"*

do *echo This is $number*

done

for *line* **in** *$(cat /var/users)*

do *echo In this line: $line*

done

While

while *условие*

do *команда*

done

i=1

while *[-f file]*

do *echo $i*

((i++))

done

Функции

```
func()    {
    i=1
    ...
return $value
        }
func 2 5
```

- ➔ Функция задается перед вызовом.
- ➔ **$1** хранит первый аргумент функции, **$2** – второй и так далее.

Содержание

- Bash — сценарий оболочки
- SystemD — система инициализации Linux

Система инициализации

Система инициализации (init) – набор сценариев, которые выполняются при старте системы

Досистемный загрузчик – BIOS, UEFI – проверка оборудования

Загрузчик первого уровня – чтение загрузочного сектора

Загрузчик второго уровня – GRUB, GRUB2 – выбор ОС

Инициализация ядра – процесс запуска ядра

Процесс init – запуск операционной системы

Systemd

Systemd – системный демон ОС Linux.

Цель: распараллеливание и отложенный запуск

Порядок запуска и поведение службы описывается с помощью специальных сценариев – **юнитов**.

Возможности:

- ➔ Параллельный запуск независимых служб
- ➔ Собственный журнал
- ➔ Поддержка отложенного запуска
- ➔ Сохранение состояния сервисов для восстановления
- ➔ Управление сетью
- ➔ Управление DNS
- ➔ Планирование заданий

Юниты

service – обычная служба
target – обеспечение точек синхронизации
a**utomount** – точка автоматического монтирования
device – файл устройства
mount - точка монтирования
path – отслеживание пути файла или папки
slice – группа системных служб system
snapshot – сохраненное состояние запущенных служб
socket – сокет для взаимодействия между процессами

[Unit] #ssh_service
Description=OpenSSH server daemon
Documentation=man:sshd(8)
man:sshd_config(5)
After=network.target sshd-keygen.service
Wants=sshd-keygen.service

[Service]
EnvironmentFile=/etc/sysconfig/sshd
ExecStart=/usr/sbin/sshd -D $OPTIONS
ExecReload=/bin/kill -HUP $MAINPID
KillMode=process
Restart=on-failure
RestartSec=42s

[Install]
WantedBy=multi-user.target

[Unit]

[Unit] – первый раздел юнита:

- ➔ **Description** – описание
- ➔ **Documentation** – местоположение документации
- ➔ **Requires** – список зависимостей (юниты)
- ➔ **Wants** – менее строгая, чем Requires
- ➔ **BindsTo** – приводит к завершению, если завершится одна из зависимостей
- ➔ **Before** – указанные юниты не будут запущены, пока текущий не будет отмечен как запущенный
- ➔ **After** – указанные юниты будут запущены до запуска текущего юнита
- ➔ **Conflict** – список конфликтных юнитов

[Install]

[Install] – третья секция для определения уровня, на котором запустится настраиваемый сервис:

- ➔ **WantedBy** – определяет, когда сервис будет запущен

Уровни:

- ➔ **poweroff.target** – отключение системы
- ➔ **rescue.target** – режим восстановления, однопользовательский
- ➔ **multi-user.target** – сетевой режим без графической оболочки
- ➔ **graphical.target** – сетевой режим с графической оболочкой
- ➔ **reboot.target** – перезагрузка
- ➔ **emergency.target** – аварийная командная строка

[Service]

Раздел **[Service] –** секция описывающая конфигурацию службы:

- ➔ **Type** – как запустится демон:
  - Simple – незамедлительный запуск
  - Forking – после запуска демон ответвляется, родительский процесс завершается
  - one-shot – одноразовое выполнение
  - notify – запустится незамедлительно и сообщит о своей готовности systemd
- ➔ **PIDfile** – ссылка на основной процесс
- ➔ **WorkingDirectory** – рабочая директория
- ➔ **User** – пользователь, от чьего имени нужно запускать
- ➔ **Group** – группа
- ➔ **Environment** – переменная окружения
- ➔ **ExecStart** – команда старта сервиса
- ➔ **ExecReload** – команда перезапуска сервиса
- ➔ **ExecStop** – команда остановки сервиса
- ➔ **TimeoutSec** – задержка перед ExecStart и ExecStop
- ➔ **Restart** – настройки перезапуска

[Path]

Раздел **[Path]** отслеживает изменения файлов и каталогов:

- ➔ **PathExist** – проверка наличия пути
- ➔ **PathChanged** – файл изменен (при закрытом файле)
- ➔ **PathModified** – файл изменен, но активируется при записи файлов
- ➔ **DirectoryNotEmpty** – каталог не пустой
- ➔ **Unit** – запуск зависимого юнита
- ➔ **MakeDirectory** – создание структуры каталогов перед просмотром
- ➔ **DirectoryMode** – определяет режим прав на создаваемые каталоги

Приоритет расположения юнитов:

1. /etc/systemd/system – наивысший приоритет.
2. /run/system/system – динамически создаваемые юниты.
3. /lib/systemd/system – системные юниты, устанавливаемые вместе с приложениями.

Команды

- **systemctl** – менеджер служб
  - list-units – список служб
  - start – запустить
  - stop – остановить
  - reload – обновить конфигурацию
  - restart – перезапустить службу
  - status – статус службы
  - enable – активация автозагрузки
  - disable – выключение автозагрузки
- **systemd-resolved –** служба DNS
- **systemd-networkd –** сетевая служба
- **systemd-analyze –** мониторинг статистики системной службы
- **service –** управление службами
- **systemd-detec-virt –** обнаружение выполнения в виртуализированный среде
- **journalctl –** журнал сервисов systemd

Ссылки

- Bash-скрипты
- Кастомизация приветственной строки
- Встроенные команды Bash
- Программирование на bash
- Написание скриптов
- Bash сценарии
- Функции
- Юниты Systemd
- Systemd
- Написание юнитов