



Базовое Администрирование Linux

Занятие 6



Дмитрий
Молчанов

Управление системой: софт и сервисы



1. Методики управления софтом

1. Самостоятельная сборка
2. Пакеты, управление пакетами.

2. Сервисы

1. Системы управления сервисами
2. Запуск сервисов
3. Остановка сервисов

3. Логи

Пакеты vs make/make install



	PROs	CONs
Пакеты	<ul style="list-style-type: none">• Контроль над установленными файлами• Легкая проверка целостности• Pre/post-inst/remove• обработка зависимостей	<ul style="list-style-type: none">• Версия софта заморожена в релизе
Самосбор	<ul style="list-style-type: none">• Быстро установить самую актуальную версию	<ul style="list-style-type: none">• Сложность тиражирования установки• Часто отсутствие init-скриптов для нужной системы

Если все же самосбор, то



- старайтесь изолировать софт от системы, например собирайте в `/usr/local/$software` или `/opt/$software`
- не храните ненужного в системе
- используйте те библиотеки, которая система предоставляет

- Где хранить исходники?
 - ~/src
 - /usr/src/[soft/]
- Типичная последовательность:
 - configure
 - make
 - make install

- Типичная последовательность:
 - configure
 - ./configure -help
 - --prefix – куда собирать

- Типичная последовательность:
 - `configure`
 - `./configure --help`
 - `--prefix` – куда собирать
 - `make`
 - доставить зависимости и снова `make`
 - `make install` – не всегда необходимо

- Репозитории пакетов
- Скриптлеты
 - pre/post-install/remove/uninstall
- Свойства пакетов
 - имя
 - Версия
 - Состояние (установлен, сконфигурирован)
 - Зависимости

Управление пакетами



- dpkg
 - apt
 - apt-cache – поиск, инфо о пакете
 - apt-get – управление пакетами
 - aptitude

Основные задачи



- install,update
 - dpkg -i
 - apt-get install
- upgrade
 - apt-get upgrade
 - apt-get distupgrade
- remove
 - dpkg -r
 - apt-get remove

Основные задачи



- list/info
 - dpkg -l
 - dpkg-query
 - apt-cache showpkg
- search
 - apt-cache search

Сервис – процесс выполняющийся в фоне, отвязанный от терминала, имеющий `ppid = 1`.

Программа должна либо сама уметь демонизироваться, либо это можно сделать с помощью утилиты `nohup`.

Сервис – процесс выполняющийся в фоне, отвязанный от терминала, имеющий `ppid = 1`

Системы инициализации системы:

- SysV init
 - Исторически «первая», самая распространенная
- upstart
 - Применялась в ubuntu, fedora, но в итоге используют systemd
- systemd

Запуск системы **sysv-init**



- Уровни выполнения
 - 0 – shutdown
 - 1,S – single mode
 - 2,3 – multiuser
 - 4,5 – multiuser + X
 - 6 – reboot
- Наборы действий
 - /etc/rcN.d
 - /etc/init.d
 - /etc/rc.local

Уровни выполнения



Вход в уровень выполнения N:

- выполнить все `/etc/rcN.d/S*`

Выход из уровня выполнения N:

- выполнить все `/etc/rcN.d/K*`

S – Start

K – Kill

Каждый скрипт является симлинком в «одноименный» `/etc/init.d` - скрипт

/etc/init.d - скрипты инициализации



Каждый скрипт, который является скриптом инициализации сервиса и должен обрабатывать следующие действия:

- start – запуск сервиса
- stop – остановка сервиса
- status – текущее состояние сервиса
- restart - перезапуск

Но так же может обрабатывать разные дополнительные действия, например reload, configtest и т.п.

telinit – смена текущего runlevel'a

service \$service (start|stop|restart|status) –
управление сервисом

update-rc.d – добавление/удаление
сервисов в загрузку в определенных
runlevel'ax

- rsyslog
- /var/log
 - syslog
 - auth
 - mail

Полезные ссылки



- https://ru.wikipedia.org/wiki/Уровень_выполнения