

[Пред.](#)[Начало](#)[След.](#)[Наверх](#)[Сведения о документе](#)

## Стартовые сценарии

---

### Стартовые сценарии

---

Стартовые сценарии осуществляют запуск и останов *системных служб* – программ, предоставляющих различные “услуги” (сервисы) пользователям системы. В качестве примера системных служб можно привести `syslog` – службу инициализации системного журнала или `network` – службу инициализации сети.

Для управления списком служб, которые должны запускаться при старте системы (а точнее – при её загрузке на том или ином уровне выполнения), используется команда **chkconfig** вида **chkconfig имя\_службы команда**. Например, командой **chkconfig sendmail on** запускается почтовый сервер, осуществляющий отправление и доставку почты по протоколу SMTP. Более подробно работа команды **chkconfig** рассмотрена ниже.

Для запуска, перезапуска или останова служб в процессе работы применяется утилита **service**. Её общий вид следующий: **service имя\_службы{start|stop|reload|restart|status}**. Например, если понадобилось перезапустить службу системного журнала, это можно сделать следующим образом:

```
# service syslogd restart
Stopping system logger service:          [ DONE ]
Starting system logger service:          [ DONE ]
```

Обратите внимание, что для управления службами необходимы права суперпользователя.

Управление системными службами при помощи утилит **chkconfig** и **service** – наиболее простой в повседневной работе вариант. Далее речь пойдет о более сложном и универсальном способе – работе со стартовыми сценариями.

Стартовые сценарии ALT Linux устроены примерно так же, как в RedHat Linux, по простой “.d”-схеме. Суть её в следующем. Любое атомарное действие по изменению способностей системы – запуск и останов определённой службы, автоматическое изменение или сохранение настроек и т. п. – оформляется в виде командного сценария в каталоге `/etc/rc.d/init.d`. К такому сценарию предъявляются определённые требования: как минимум, он должен распознавать в качестве первого параметра слова *start* и *stop*. Будучи запущен с параметром *start*, этот сценарий выполняет действия по запуску службы или активации настроек, а с параметром *stop* он останавливает службу или записывает активные настройки в файл.

Когда система загружается, процесс `init` выполняет сначала сценарий `/etc/rc.d/rc.sysinit`, а затем – `/etc/rc.d/rcномер`, где *номер* соответствует *уровню выполнения* (профилю использования системы, обычно он равен 3, “многопользовательский с сетью” или 5, “многопользовательский с сетью и графической средой”). Всё это сводится к выполнению в *лексикографическом* (грубо говоря, алфавитном) порядке *всех* сценариев из каталога `/etc/rc.d/rcномер.d/`, начинающихся на букву *K* (Kill) с параметром *stop* и *всех* сценариев, начинающихся на *S* (Start) с параметром *start*. Принято после буквы *K* или *S*

вставлять двузначное число, а затем – имя службы, а сами файлы в `rcномер.d/` делать символьными ссылками на сценарии в `/etc/rc.d/init.d/` (точнее, на файлы в `../init.d/`, так достигается совместимость с системами, в которых `init.d` и `rc*.d` лежат не в `/etc/rc.d`, а в `/etc` или вообще где угодно). Останов системы рассматривается как переход на уровень выполнения 0, а перезагрузка – на уровень выполнения 6, так что имена файлов в каталогах `rc0.d` и `rc6.d` начинаются сплошь на К, кроме `S00killall`, `S01reboot` или `S01halt`. Таким образом достигается строгий порядок выполнения действий без создания единого стартового сценария (как это делалось, например, в ранних версиях BSD).

В ALT Linux эти рекомендации носят *обязательный* характер, потому что размещение сценариев в каталогах `rc*.d` происходит не вручную, а автоматически. Для этого в начало каждого сценария вносятся комментарии особого вида: `# description:`, после которого следует краткое описание, и `chkconfig:`, после которого следует три числа. Первое – набор цифр (без пробелов), соответствующих уровням выполнения, на которых сценарий *будет* использоваться (т. е. создаваться ссылка, начинающаяся на S). Второе – двузначное число, которое будет добавляться в имя ссылки после S, а третье – двузначное число, которое будет добавляться в имя ссылки после K.

```
#!/bin/sh
#
# crond                Start/Stop the cron clock daemon.
#
# chkconfig: 2345 40 60
# description: cron is a standard UNIX program that runs user-specified \
#              programs at periodic scheduled times. vixie cron adds a \
#              number of features to the basic UNIX cron, including better \
#              security and more powerful configuration options.
# processname: crond
# config: /etc/crontab
# pidfile: /var/run/crond.pid
```

Команда **chkconfig имя\_службы on** автоматически создаёт ссылки типа S для всех уровней выполнения, указанных в первом числе поля `chkconfig:`, а для остальных уровней выполнения – ссылки типа K. Команда **chkconfig имя\_службы off** заводит во всех каталогах `rc*.d` ссылки типа K. Стоит заметить, что другие поля похожего формата остались в стартовых сценариях от других, более сложных систем автоматической загрузки и в ALT Linux не используются. В приведённом примере сценарий `netfs` (монтирование сетевых файловых систем) запускается на уровнях 3, 4, и 5, причём при запуске служба имеет номер 25, а при останове – 75.

```
$ grep chkconfig: /etc/rc.d/init.d/netfs
# chkconfig: 345 25 75
$ chkconfig --list netfs
netfs          0:выкл  1:выкл  2:выкл  3:вкл   4:вкл   5:вкл   6:выкл
$ ls -l /etc/rc.d/rc*.d/*netfs*
lrwxrwxrwx 1 root root 15 Авг 15 19:06 /etc/rc.d/rc0.d/K75netfs -> ../init.d/netfs
lrwxrwxrwx 1 root root 15 Авг 15 19:06 /etc/rc.d/rc1.d/K75netfs -> ../init.d/netfs
lrwxrwxrwx 1 root root 15 Авг 15 19:06 /etc/rc.d/rc2.d/K75netfs -> ../init.d/netfs
lrwxrwxrwx 1 root root 15 Авг 15 19:06 /etc/rc.d/rc3.d/S25netfs -> ../init.d/netfs
lrwxrwxrwx 1 root root 15 Авг 15 19:06 /etc/rc.d/rc4.d/S25netfs -> ../init.d/netfs
lrwxrwxrwx 1 root root 15 Авг 15 19:06 /etc/rc.d/rc5.d/S25netfs -> ../init.d/netfs
lrwxrwxrwx 1 root root 15 Авг 15 19:06 /etc/rc.d/rc6.d/K75netfs -> ../init.d/netfs
```

Кроме того, правила оформления стартовых сценариев ALT Linux рекомендуют, чтобы они распознавали ещё один параметр – *status*, позволяющий определить состояние службы (по крайней мере, узнать, запущена ли она или нет). Корректный стартовый сценарий всё равно должен уметь это определять, иначе попытка запуска K-сценария при *остановленной* службе будет выдавать сообщение об ошибке, даже если спецификой службы это предусмотрено. Например, супердемон `xinetd` при запуске проверяет, зарегистрирована ли в нём хотя бы

одна сетевая служба, а не найдя таковых, завершает работу. Для отражения того, что запуск службы прошёл успешно, в ALT Linux заведён специальный каталог, `/var/lock/subsys`, в котором стартовые сценарии создают т. н. lock-файлы. Если lock-файл службы есть в этом каталоге, значит, она запущена, и K-сценарий действительно должен её останавливать. Поэтому рекомендуется запускать и останавливать системные службы не напрямую, запуском демонов, и даже не путём запуска сценария из `init.d`, а с помощью утилиты **service**. Впрочем, вызов **service имя\_службы команда** эквивалентен `/etc/rc.d/init.d/имя_службы команда`, просто командная строка выходит короче.

При написании стартового сценария рекомендуется пользоваться файлом `/etc/init.d/functions`, в котором определены основные функции для работы с `/var/lock/subsys` и унифицированным выводом сообщений на экран. Обратите внимание, что по соображениям совместимости обычно используется имя каталога `/etc/init.d`, а не `/etc/rc.d/init.d`, при этом первое имя – всего лишь символьная ссылка на второе.

---

[Пред.](#)[Начало](#)[След.](#)[Наверх](#)[Сведения о документе](#)