

[КАК СТАТЬ АВТОРОМ](#)[Технотекст](#)[Обучение ИБ: что с ним, на ваш взгляд, не так?](#)**68.85**

Рейтинг

**Слёрм**

Учебный центр для тех, кто работает в IT

**edeshina**

13 янв 2023 в 14:50

## Хозяйке на заметку: разбираемся с конфигурационными файлами Linux

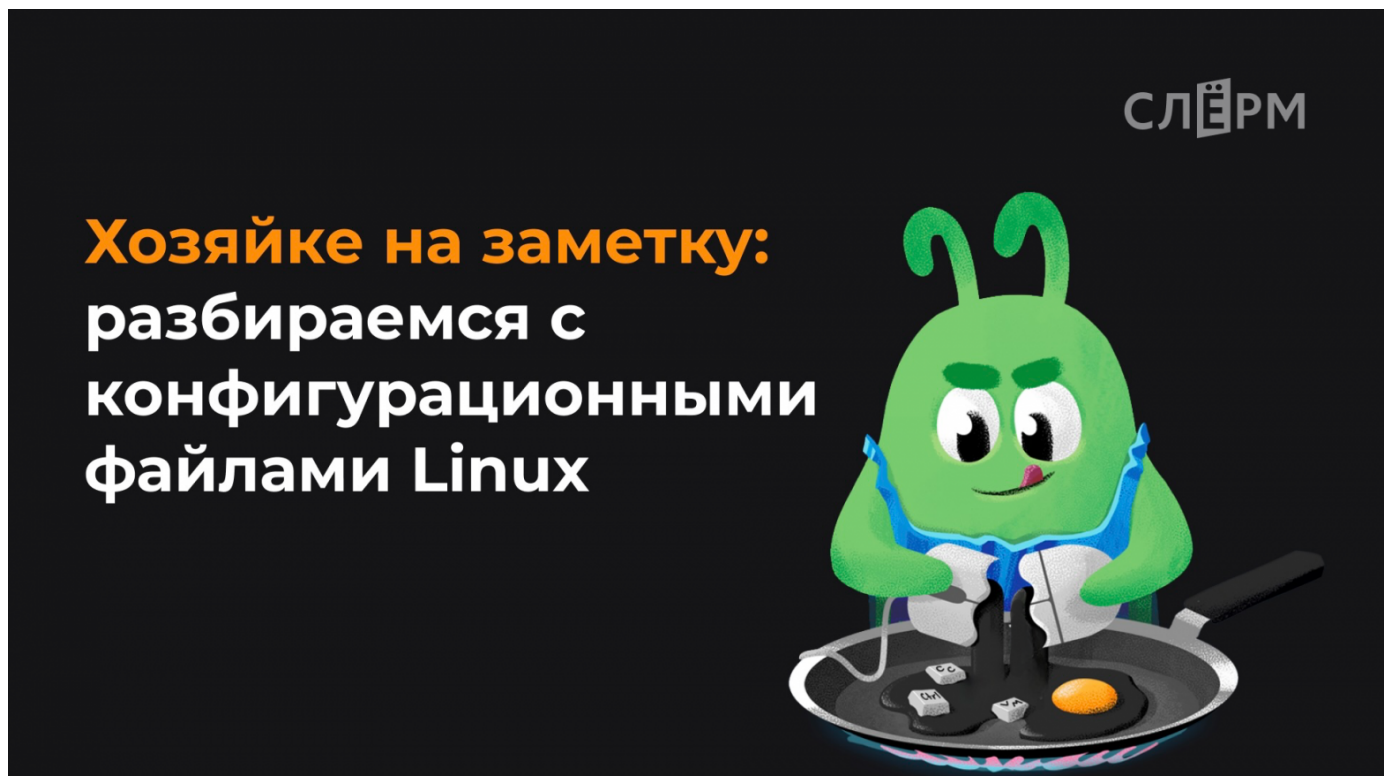
🕒 11 мин 👁 23K

Блог компании Слёрм, Настройка Linux\*, Системное администрирование\*, Разработка под Linux\*, DevOps

[Обзор](#)[Перевод](#)

Автор оригинала: Subodh Soni

Понимая, как работают конфигурационные файлы, вы можете существенно повысить уровень владения Linux. В статье рассказываем, как конфигурационные файлы управляют пользовательскими разрешениями, системными приложениями, демонами и другими административными задачами в многопользовательской среде.



## Введение в конфигурационные файлы

Само ядро можно назвать «программой». Зачем ядру конфигурационные файлы? Ему нужно знать список пользователей и групп в системе, управлять правами доступа к файлам (определять, может ли файл открываться конкретным пользователем в соответствии с разрешениями UNIX\_USERS).

Такие файлы считываются не программами, а функцией, предоставляемой системной библиотекой и используемой ядром. Предположим, программа, которой требуется зашифрованный пароль пользователя, не должна открывать файл `/etc/passwd`. Вместо этого следует вызвать функцию системной библиотеки `getpw()`. Этот тип функции также известен как **системный вызов**. Через системную библиотеку ядро открывает файл `/etc/passwd` и затем ищет пароль запрошенного пользователя.

Большинство конфигурационных файлов Linux находятся в каталоге `/etc`, если не указано иное. В зависимости от сферы применения и служб, на которые они влияют, конфигурационные файлы делят на несколько категорий, о которых и пойдёт речь дальше.

### Доступ к файлам

<code>/etc/host.conf</code>	Сообщает серверу сетевого домена, как искать имена хостов. Сначала идёт <code>/etc/hosts</code> , затем имя сервера. Файл можно изменить через <code>netconf</code> .
<code>/etc/hosts</code>	Содержит список известных хостов в локальной сети. Может использоваться, если IP-адрес системы не генерируется динамически.
<code>/etc/hosts.allow</code>	Справочная страница, аналогичная <code>hosts_access</code> .
<code>/etc/hosts.deny</code>	Справочная страница, аналогичная <code>hosts_access</code> .

### Загрузка и вход/выход из системы

<code>/etc/issue &amp; /etc/issue.net</code>	Такие файлы считываются <code>mingetty</code> (и аналогичными программами) для отображения приветственной строки пользователю, подключающемуся с терминала ( <code>issue</code> ) или через сеанс <code>telnet(issue.net)</code> . Они включают в себя несколько строк с указанием номера выпуска Red Hat, названия и идентификатора ядра. Используются <code>rc.local</code> .
<code>/etc/redhat-release</code>	Включает одну строку с указанием номера и названия выпуска Red Hat. Используется <code>rc.local</code> .

<b>/etc/rc.d/rc</b>	Предназначен для перевода системы с одного уровня выполнения на другой. В процессе начальной загрузки переводит систему из однопользовательского режима на уровень, задаваемый по умолчанию.
<b>/etc/rc.d/rc.local</b>	Помогает конфигурировать запуск системы – запускать дополнительные программы и подключать библиотеки к работающим процессам. Также позволяет уничтожать работающие процессы с помощью утилиты <code>slay</code> и перезапускать их с другими параметрами, однако такой подход считается громоздким.
<b>/etc/rc.d/rc.sysinit</b>	Используется для локальной инициализации системы.
<b>/etc/rc.d/rc/rc X.d</b>	Скрипты запускаются из <code>rc</code> ( <code>X</code> означает любое число от 1 до 5). Эти каталоги являются специфичными каталогами «уровня выполнения». Когда система запускается, она определяет уровень выполнения, который необходимо инициировать, а затем вызывает все сценарии запуска, присутствующие в каталоге для этого уровня выполнения. Например, система запускается, и после сообщений о запуске появляется сообщение «вход на уровень выполнения 3» – это означает, что будут вызваны все сценарии инициализации в каталоге <code>/etc/rc.d/rc3.d/</code> .

## Файловая система

Ядро предоставляет интерфейс для отображения структур данных, которые могут быть полезны при определении системных параметров – прерываний, статистики памяти и др. Этот интерфейс представляет собой отдельную файловую систему, известную как `/proc filesystem`. Многие системные утилиты используют значения, присутствующие в этой файловой системе или отображающие системную статистику. Например, в файле `/proc/modules` перечислены загруженные в данный момент модули. Информация считывается командой `lsmod`, которая затем отображает её в удобочитаемом формате.

<b>/etc/mtab</b>	Содержимое файла не обновляется средствами ядра ОС, а поддерживается в актуальном состоянии средствами утилиты <code>mount</code> . Когда файловые системы монтируются и размонтируются, изменения немедленно отражаются в этом файле.
<b>/etc/fstab</b>	Содержит список файловых систем, которые в настоящее время «монтируются» компьютером. Файл важен, поскольку во время загрузки компьютер запускает команду <code>mount -a</code> , которая монтирует каждую файловую систему, отмеченную цифрой «1» в предпоследнем столбце <code>fstab</code> .

<b>/etc/mtools.conf</b>	Считается конфигурационным файлом для всех операций (mkdir, copy, format и др.) в файловой системе типа DOS.
-------------------------	--

## Системное администрирование

<b>/etc/group</b>	Определяет группы в системе Linux. Один пользователь может присутствовать более чем в одной группе, если он выполняет несколько задач. Например, если «user» является администратором, а также членом группы проекта «project 1», то его запись в файле группы будет выглядеть так: user: * : group-id : project1
<b>/etc/nologin</b>	Основная задача файла – показать сообщение пользователям, которые пытаются войти в систему во время завершения работы. Как только сообщение отображается, процедура входа в систему завершается, что останавливает вход пользователя в систему.  Пример: если файл /etc/nologin существует, то login(1) разрешит доступ только пользователю root. Другим пользователям будет показано содержимое этого файла, но во входе в систему им будет отказано.
<b>/etc/passwd</b>	Содержит информацию об учётной записи пользователя, включая пароли (если они не «затенены»).
<b>/etc/rpmrc</b>	Конфигурация команды rpm. Все параметры командной строки rpm устанавливаются вместе в этом файле, поэтому они применяются глобально, когда любая команда rpm запускается в системе.
<b>/etc/securetty</b>	Содержит имена устройств tty-линий (по одной на строку, без ведущего /dev/), на которых root разрешён вход.
<b>/etc/usertty /etc/shadow</b>	Содержит информацию о зашифрованном пароле для учётных записей пользователей и, при необходимости, информацию об истечении срока действия пароля. Включены следующие поля:  – имя для входа в систему  – зашифрованный пароль

	<ul style="list-style-type: none"> <li>– количество дней с 1 января 1970 года, когда пароль был изменен в последний раз</li> <li>– количество дней до того, как пароль может быть изменен</li> <li>– количество дней, после которых пароль должен быть изменен</li> <li>– количество дней до истечения срока действия пароля о том, что пользователь предупрежден</li> <li>– количество дней после истечения срока действия пароля.</li> </ul>
<b>/etc/shells</b>	Содержит список возможных «оболочек», доступных системе.
<b>/etc/motd</b>	Сообщение дня – используется, если администратор хочет передать какое-то сообщение всем пользователям сервера Linux.

## Сеть

<b>/etc/gated.conf</b>	Представляет собой конфигурационные данные для работы демона gated, предназначенного для маршрутизации протоколов RIP, BGP, EGP, HELLO и OSPF.
<b>/etc/gated.version</b>	Содержит номер версии демона gated.
<b>/etc/gateway</b>	Опционально используется демоном routed.
<b>/etc/networks</b>	Хранит доменные имена и IP-адреса сетей, с которыми у вашей системы есть соединение. Сети имеют сокращенные IP-адреса. В зависимости от типа сети в IP-адресах может использоваться одно, два или три числа. IP-адреса записываются в файле /etc/networks вместе с соответствующими им доменными именами сетей.
<b>/etc/protocols</b>	Хранит список доступных на данный момент протоколов. Никогда не должен меняться.
<b>/etc/resolv.conf</b>	Определяет, как система использует DNS для разрешения имен хостов и IP-адресов.
<b>/etc/rpc</b>	Содержит инструкции/правила для RPC, которые можно использовать при вызовах NFS, удаленном монтировании

	файловой системы и др.
<b>/etc/exports</b>	<p>В этом файле перечислены все каталоги, экспортируемые с сервера. Одна строка задает один каталог.</p> <p>В файле /etc/exports один каталог может быть указан дважды: один раз для NFS версии 2 или NFS версии 3 и один раз – для NFS версии 4. Сервер автоматически экспортирует указанные каталоги каждый раз при запуске сервера NFS. Затем клиенты могут смонтировать экспортированные каталоги.</p>
<b>/etc/services</b>	Содержит информацию о многочисленных службах, которые клиентские приложения могут использовать на компьютере. Преобразует имена сетевых служб в номер порта/протокол.
<b>/etc/inetd.conf</b>	Конфигурационный файл для суперсервера inetd. Изменять этот файл следует только при необходимости добавления или удаления описаний серверов.
<b>/etc/sendmail.cf</b>	Файл конфигурации почтовой программы sendmail. Содержит большое количество опций, представленных в неудобном для восприятия виде, поэтому анализировать содержимое данного файла и редактировать его довольно сложно.
<b>/etc/sysconfig/network</b>	Глобальный файл конфигурации. Позволяет установить, хотите ли вы использовать сетевое соединение (NETWORKING=yes or no).
<b>/etc/sysconfig/network-scripts/if*</b>	Скрипты настройки сети Red Hat.

## Системные команды

Системные команды предназначены для управления системой и обеспечения того, чтобы все работало должным образом. Все программы вроде как `login` (выполняющей аутентификацию пользователя на консоли) или `bash` (обеспечивающей взаимодействие между пользователем и компьютером) являются системными командами, поэтому связанные с ними файлы особенно важны.

В таблице перечисляем файлы, представляющий наибольший интерес для пользователей и администраторов.

<b>/etc/lilo.conf</b>	Содержит параметры командной строки системы по умолчанию, а также различные образы для загрузки. Вы можете просмотреть
-----------------------	--

	весь список, нажав Tab в командной строке LILO.
<b>/etc/logrotate.conf</b>	Поддерживает файлы журнала, присутствующие в каталоге /var/log.
<b>/etc/identd.conf</b>	Identd – сервер, который реализует предлагаемый TCP/IP протокол идентификации пользователя IDENT. identd работает путем поиска конкретных соединений TCP/IP и возврата имени пользователя процесса, владеющего соединением. При необходимости он может возвращать другую информацию вместо имени пользователя.
<b>/etc/ld.so.conf</b>	Конфигурация для динамического компоновщика
<b>/etc/inittab</b>	Хронологически первый конфигурационный файл в UNIX. Первая программа, запускаемая после включения компьютера UNIX, – это init, которая знает, что делать благодаря inittab. Файл /etc/inittab управляет порождением процессов программой init.
<b>/etc/termcap</b>	База данных, содержащая все возможные типы терминалов и их возможности.

## Демоны

**Демон** – программа, которая всегда работает в фоновом режиме. Обычно её задачи связаны с сетевой областью: они ожидают подключений, чтобы через них можно было предоставлять услуги. Для Linux доступно множество демонов, начиная от веб-серверов и заканчивая ftp-серверами.

<b>/etc/syslogd.conf</b>	Содержит информацию, используемую демоном системного журнала syslogd(1M) для пересылки системного сообщения соответствующим файлам журнала и/или пользователям
<b>/etc/httpd.conf</b>	Файл конфигурации для веб-сервера Apache. Обычно находится не в /etc, а в /usr/local/httpd/conf/ или /etc/httpd/conf/. Чтобы определить точное нахождение, вам нужно проверить конкретную установку Apache.
<b>/etc/conf.modules or /etc/modules.conf</b>	Файл конфигурации ядра. Заботится о загрузке дополнительных модулей ядра «на лету», когда это необходимо.

В Linux (и UNIX в целом) существует бесчисленное множество «пользовательских» программ. Наиболее распространенным файлом конфигурации пользовательской программы считается `/etc/lynx.cfg`. Это файл конфигурации для `lynx` – хорошо известного текстового браузера. С помощью этого файла вы можете определить прокси-сервер, используемый набор символов и др.

Пример кода ниже показывает часть файла `lynx.cfg`, который может быть отредактирован для изменения настроек прокси-сервера системы Linux. Эти настройки применяются ко всем пользователям, запускающим `lynx` в соответствующих оболочках, если только пользователь не переопределит конфигурационный файл по умолчанию, указав `--cfg = "mylynx.cfg"`.

```
.h1 proxy
.h2 HTTP_PROXY
.h2 HTTPS_PROXY
.h2 FTP_PROXY
.h2 GOPHER_PROXY
.h2 NEWS_PROXY
.h2 NNTP_PROXY
# Lynx version 2.2 and beyond supports the use of proxy servers that can act as
# firewall gateways and caching servers. They are preferable to the older
# gateway servers. Each protocol used by Lynx can be mapped separately using
# PROTOCOL_proxy environment variables (see Lynx Users Guide). If you have
# not set them externally, you can set them at run time via this configuration file.
# They will not override external settings. The no_proxy variable can be used
# to inhibit proxying to selected regions of the Web (see below). Note that on
# VMS these proxy variables are set as process logicals rather than symbols, to
# preserve lowercasing, and will outlive the Lynx image.
#
.ex 15
http_proxy:http://proxy3.in.ibm.com:80/
ftp_proxy:http://proxy3.in.ibm.com:80/
#http_proxy:http://penguin.in.ibm.com:8080
#ftp_proxy:http://penguin.in.ibm.com:8080/

.h2 NO_PROXY
# The no_proxy variable can be a comma-separated list of strings defining
# no-proxy zones in the DNS domain name space. If a tail substring of the
# domain-path for a host matches one of these strings, transactions with that
# node will not be proxied.
.ex
no_proxy:demiurge.in.ibm.com, demiurge
```

## Изменение конфигурационных файлов



При изменении конфигурационных файлов убедитесь, что программа, использующая эту конфигурацию, перезапущена, если она не контролируется системным администратором или ядром. Обычный пользователь, как правило, не имеет привилегий для запуска или остановки системных программ и/или демонов.

Ядро

Изменение конфигурационных файлов в ядре сразу влияет на систему. Например, изменение файла `passwd` для добавления пользователя немедленно активирует этого пользователя. Также есть некоторые настраиваемые параметры ядра в каталоге `/proc/sys` в любой системе Linux. Доступ к записи ко всем этим файлам предоставляется только суперпользователю; другие пользователи имеют доступ только для чтения

Также в каталоге `/proc/sys` в любой системе Linux есть некоторые настраиваемые параметры ядра. Редактирование этих файлов доступно только суперпользователю; другие пользователи имеют доступ только для чтения. Файлы в этом каталоге классифицируются таким же образом, как и исходный код ядра Linux. Каждый файл в этом каталоге представляет собой структуру данных ядра, которую можно динамически модифицировать для изменения производительности системы.

**Важно:** прежде чем менять какое-либо значение в любом из файлов, убедитесь, что вы знаете о нём всё, чтобы избежать непоправимого ущерба для системы.

Файлы в каталоге `/proc/sys/kernel/`

Название файла	Описание
<code>threads-max</code>	максимальное количество задач, которые может выполнить ядро
<code>ctrl-alt-del</code>	If 1, то нажатие этой последовательности клавиш полностью перезагружает систему
<code>sysrq</code>	If 1, то Alt-SysRq активен
<code>osrelease</code>	отображает версию операционной системы
<code>ostype</code>	отображает тип операционной системы
<code>hostname</code>	имя хоста системы
<code>domainname</code>	сетевой домен, частью которого является система
<code>modprobe</code>	указывает, должен ли <code>modprobe</code> автоматически запускаться и загружать необходимые модули

## Демоны и системные программы

Некоторые демоны во время работы внимательно следят за конфигурационными файлами и автоматически перезагружают их при изменении. Однако большинство демонов не выполняют автоматическую перезагрузку. Нам нужно как-то «сообщить» им, что конфигурационный файл изменился и его следует перезагрузить. Это можно сделать путем перезапуска служб с помощью команды `service`.

Например, если мы изменили конфигурацию сети, нам нужно указать:

```
service network restart.
```

**Примечание:** сервисы чаще всего представляют собой скрипты, находящиеся в каталоге `/etc/rc.d/init.d/*`, и запускаются программой инициализации при загрузке системы. Чтобы перезапустить сервис, вы также можете сделать следующее:

```
/etc/rc.d/init.d/<script-for-the-service> start | stop | status
```

`start`, `stop` и `status` – значения, которые эти скрипты принимают в качестве входных данных для выполнения действия.

## Пользовательские программы

Пользовательская или системная программа считывает свой конфигурационный файл при каждом запуске. Однако некоторые системные программы запускаются при включении компьютера, и их поведение зависит от того, что они читают в конфигурационных файлах в `/etc/`. Таким образом, при первом запуске пользовательской программы конфигурация по умолчанию считывается из файлов, находящихся в каталоге `/etc/`. Позже пользователь может настроить программы с помощью `rc` и `.` (dot) files.

## Пользовательские конфигурационные файлы: `.` (dot) files and `rc` files

«Обычный» пользователь не может зайти в `/etc` и изменить конфигурационные файлы. С точки зрения файловой системы они принадлежат пользователю `root`. Вот почему большинство пользовательских программ определяют два конфигурационных файла:

- на «системном» уровне, расположенный в `/etc/`;
- «личный» для пользователя, который можно найти в домашнем каталоге.

Разберём на примере утилиты `wget`. В `/etc/` есть файл `/etc/wgetrc`. В домашнем каталоге есть файл с именем `.wgetrc`, который описывает настроенную конфигурацию (которая будет загружена только тогда, когда пользователь выполнит команду `wget`). Другие пользователи также могут иметь файл `.wgetrc` в своем домашнем каталоге (`/home/other`). Этот файл будет прочитан, только тогда, когда пользователь запустит команду `wget`. Иными словами, файл `/etc/wgetrc` предоставляет значения «по умолчанию» для `wget`, а файл `/home/xxx/.wgetrc` перечисляет «настройки» для определенного пользователя.

Важно понимать, что это «общее правило», и оно не обязательно верно для всех случаев. Например, у программы `pine` нет файлов в `/etc/`, только пользовательская конфигурация в домашнем каталоге пользователя в файле с именем `.pinerc`. Другие программы могут иметь только файл конфигурации по умолчанию в `/etc/` и могут не позволять пользователям «настраивать» их.

## Коротко о главном

В этой статье мы разобрали конфигурационные файлы, которые управляют пользовательскими разрешениями, системными приложениями, демонами и др. А также рассмотрели особенности процесса изменения конфигурационных файлов.

▸ [Администрирование Linux Mega](#)

**Теги:** конфигурационные файлы, linux, linux kernel, linux для всех, linux-сервер, ядро linux, ядро линукс

**Хабы:** Блог компании Слёрм, Настройка Linux, Системное администрирование, Разработка под Linux, DevOps

## Редакторский дайджест

Присылаем лучшие статьи раз в месяц

Электронная почта



Слёрм

Учебный центр для тех, кто работает в IT

Сайт



**30** **-2.4**  
Карма Рейтинг

**Катерина @edeshina**

Редактор в MobileUp

## Комментарии 15



**baldr**

13 янв 2023 в 15:21

Какая-то некрофилия.. Оригинальная статья 2001 года.

`/etc/rc.d` , серьезно? SystemD уже давно шествует по планете, независимо от того нравится это кому-то или нет.

Lilo? Еще остались те кто помнит что это такое?

Наиболее распространенным файлом конфигурации пользовательской программы считается `/etc/lynx.cfg`.

Мда.



Ответить



**16tomatotonn**

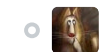
13 янв 2023 в 18:41

В качестве вводной статьи-экскурсии, тем не менее, более чем сойдёт, для изучающих тему.

Тем более что полагаю, у статьи нет цели описать "все-все существующие модные или современные конфигулируемые сервисы-демоны". Изучающему человеку это понадобится позже и он сам разберётся, пользуясь базой из статьи. Так что монопенисуально.



Ответить



**baldr**

13 янв 2023 в 19:24

Что, например, из статьи вы считаете полезным?

Вся эта статья - практически, список файлов. Половина (или больше) из которых относится к `init.d`, который сейчас уже новичок точно не встретит, а те файлы оттуда, которые еще существуют - очень не рекомендуется трогать, поскольку `systemd` ими сама рулит исключительно для совместимости.

Остальные файлы - тоже либо не существуют в современных дистрибутивах, либо описаны очень невнятно.

Вот, правда - ничерта для "изучающего человека" здесь найти не удастся, а только запутает еще больше.



Ответить



**16tomatotonnns**

13 янв 2023 в 22:00

Само описание файлов и их приблизительное местоположение.  
Ты не будешь искать то, о существовании чего не подозреваешь.  
И, соответственно, появляется худо-бедное понимание происходящего.

В противном случае, например, древнюю книжку K&R можно просто выкинуть в ведро, потому что "все давно пишут ОС на JS" : )



Ответить

**djalín**

14 янв 2023 в 23:35

Lilo? Еще остались те кто помнит что это такое?

Моя помнить - загрузчик в мандраке 2005 года.

Я вот тоже думал - о круто, а начал читать... Ну может что то с редхата может быть полезно, и то только потому что я его в глаза не видел (а вдруг там SystemD не завезли еще).

Больше смущает что курс рекламируют, надеюсь там немного посвежей материал ...



Ответить

**DungeonLords**

15 янв 2023 в 15:22

SystemD требует использовать glibc. Но многие системы (каждый второй роутер?) используют musl, то есть никакого SystemD. См. также BusyBox init.



Ответить

**Johan\_Palych**

13 янв 2023 в 15:25

| /etc/rc.d/rc.sysinit Используется для локальной инициализации системы.

Из криокамеры повалил азотный туман, обдав ноги волной холода ...

От отчаяния набрал в терминале:

```
sudo systemctl daemon-reload
```

Закрыв глаза и нажал на Enter



Ответить

**DungeonLords**

16 янв 2023 в 11:53

тем временем новость "Выпуск дистрибутива MX Linux 21.3"  
В дистрибутиве используется система инициализации sysVinit.



Ответить

**Johan\_Palych**

16 янв 2023 в 14:15



Glory, glory, hallelujah!

Init Freedom is about restoring a sane approach to PID1 that respects portability, diversity and freedom of choice.

<https://www.devuan.org/os/init-freedom>

The GNU/Linux distributions mentioned here support init options that are **portable, compatible, small, fast, and secure**.

- antiX: `sysvinit` and `runit`
- Artix: `openrc`, `runit` and `s6`
- Devuan: `sysvinit` (default), `openrc`, `runit`. Please test `sinit` and `66-devuan`
- Dragora: `sysvinit` + `perp`
- Gentoo: `openrc` (see [Gentoo without systemd](#))
- Guix: `shepherd`
- Hyperbola: `openrc`
- KNOPPIX: `knoppix-autoconfig`
- MX Linux: `sysvinit` (default), `systemd`
- Obarun: `s6 supervision suite`
- Parabola: `openrc`, `sysvinit`, `s6`, `systemd` (default)
- PCLinuxOS: `sysvinit`
- Slackware: `sysvinit`
- Stali: `sinit`
- Void Linux: `runit`



Ответить

**Rastler**

13 янв 2023 в 17:41

Мало того, что устарело, так ещё многое зависит от дистрибутива. RedHat-style это одно и Debian-style другое.



Ответить

**ermouth**

14 янв 2023 в 03:55

Самое главное то забыли: etc – это давно не et cetera, а edit to config )



Ответить



borisovEvg  
14 янв 2023 в 17:37

Последнее время на Хабре прямо мода какая-то от обучающих центров, статьи про Linux, сомнительного, а иногда даже вредного качества.



Ответить



RedPandaHere  
16 янв 2023 в 14:32

Вместо этого следует вызвать функцию системной библиотеки `getpw()`. Этот тип функции также известен как системный вызов.

Системный вызов это когда из юзерспейса идет обращение к ядру, а `getpw` это библиотечная функция, работающая в юзерспейсе, которая при своей работе делает несколько системных вызовов.



Ответить



nnstepan  
17 янв 2023 в 17:45

Подходит только к редхат подобным дистрибутивам не первой свежести, 6 ветки например



Ответить



6eJik  
20 янв 2023 в 15:05

Вот вроде бы серьёзная компания, с очень неплохими курсами, а статья прям со свалки забытых вещей.

Статья в первую очередь нацелена на человека, пытающегося разобраться с устройством системы, а в итоге, только путает!

База это конечно вещь нужная, но актуальность базовых принципов тоже нужно соблюдать. Мне кажется, ни один Ваш курс не использует системы с `init.d`.

В целом неплохо, но я предлагаю автору актуализировать статью на текущие реалии.



Ответить



Только полноправные пользователи могут оставлять комментарии. Войдите, пожалуйста.

## Публикации

ЛУЧШИЕ ЗА СУТКИ

ПОХОЖИЕ

**alizar**

4 часа назад

## Мигель де Икаса и его мечта – Linux на десктопах

**Простой**

6 мин



2.5K

Обзор

**+34**

7



5

**Tirarex**

23 часа назад

## Fan Control – лучший софт для управления вентиляторами в компьютере

**Простой**

8 мин



11K

Тutorial

**+34**

112



63

3 часа назад

## Лучшие IT-работодатели России 2023: ежегодный рейтинг Хабр Карьеры



7 мин



4.1K

**+32**

4



3

**jm\_sub**

23 часа назад

## Итоги ежегодного опроса РНР-сообщества: результаты за 2023



5 мин



4.9K

Аналитика

**+31**

22



5

**RationalAnswer**

8 часов назад

## Илона Маска лишают честно заработанных \$50 млрд, а российские банки будут ловить уехавших через геолокацию



10 мин



15K

Дайджест



 +28 17 45**Корсеньиу**

5 часов назад

## Универсальная плата для E-Ink экранов. Ч1. Разработка системы питания

 Средний 13 мин 1.3K

Кейс

 +26 22 9**DRoman0v**

19 часов назад

## Dell, IBM и Google заставили сотрудников вернуться в офисы: что в итоге?

 3 мин 20K +19 23 82**mr-pickles**

3 часа назад

## Оффлайновое использование Git

 Простой 6 мин 1.8K

Перевод

 +15 25 5**CuriousGeorgiy**

3 часа назад

## Тестирование менеджера транзакций

 Средний 6 мин 477

Кейс

 +14 10 0**PanDubs**

19 часов назад

## Двуязычный словарь от ChatGPT за 7 (нет) простых шагов

Средний

29 мин

3.2К

Кейс

+12

28

11

Показать еще

ИНФОРМАЦИЯ

Сайт	slurm.io
Дата регистрации	15 ноября 2012
Дата основания	4 июня 2018
Численность	51–100 человек
Местоположение	Россия
Представитель	Антон Скобин

ССЫЛКИ

Подкаст «В поисках бирюзовых компаний»  
www.youtube.com  
IT-курсы Слёрма  
to.slurm.io

ВИДЖЕТ



ВИДЖЕТ

# Мониторинг В Grafana



СТАРТ 1 ПОТОКА  
19 ФЕВРАЛЯ



Обучиться новому скиллу

## БЛОГ НА ХАБРЕ

25 янв в 16:20

Как мониторить Jenkins с помощью Prometheus, Node exporter и Grafana

👁 2.9K 💬 11

16 янв в 16:20

<https://habr.com/ru/companies/slurm/articles/710398/>

Изучаем Ansible Inventory: основы и примеры использования

 6.9K

 1

11 янв в 16:04

Концепция Twelve-Factor App и Kubernetes

 3K

 1

28 дек 2023 в 16:22

Что происходит, когда вы создаёте Pod в Kubernetes?

 11K

 2

21 дек 2023 в 16:26

Работаете с микросервисами? Идём в сторону Go

 5.4K

 5

Ваш аккаунт

Войти  
Регистрация

Разделы

Статьи  
Новости  
Хабы  
Компании  
Авторы  
Песочница

Информация

Устройство сайта  
Для авторов  
Для компаний  
Документы  
Соглашение  
Конфиденциальность

Услуги

Корпоративный блог  
Медийная реклама  
Нативные проекты  
Образовательные  
программы  
Стартапам



Настройка языка

Техническая поддержка