

Тонкости работы с Urpmi

urpmi вне всяких сомнений является одним из наиболее важных инструментов ROSA. С помощью **urpmi** вы с лёгкостью сможете управлять пакетами. Если вы овладеете с **urpmi**, вы никогда не будете испытывать «ад зависимостей», на который жалуются многие неопытные пользователи. Например, простая команда **urpmi sylpheed** установит почтовый клиент Sylpheed^[1], а вместе с ним и все необходимые библиотеки.

Где получить дополнительную информацию об urpmi

Urpmi — важный инструмент для всех пользователей ROSA, и он требует какое-то время для изучения. Эта страница даёт обзор наиболее часто используемых параметров. Следующие ресурсы позволяют получить дополнительную информацию об urpmi:

- на **man**-страницах можно узнать обо всех параметрах urpmi; страницы руководств — самые актуальные источники информации;
- на странице обновление ROSA с помощью urpmi обсуждается обновление ROSA с одной основной версии до другой, используя urpmi.



Примечание

Страницы руководств (man-страницы) можно открывать во многих веб-браузерах, просто введёте *man:urpmi* в командной строке браузера.

Помимо небольшого введения на этой странице даётся обзор информации, отсутствующей в двух, приведённых выше, источниках. Предполагается, что вы имеете представление как пользоваться страницами руководств, и что вы ознакомились с выше указанной страницей. Если вы не сделали этого раньше, то рекомендуется сделать это именно сейчас. После этого вы можете вернуться к чтению этой страницы.



Предупреждение

Данная страница может оказаться полезной, так как имеется ряд неочевидных проблем, с которыми вы можете столкнуться.

Использование urpmi

Краткий список базовых функций

Команда	Что делает
urpmi --auto-update	объединяет функциональность urpmi.update -a и urpmi --auto-select
urpmq -i xxx.rpm	показывает информацию о пакете xxx.rpm, находящемся в текущем каталоге
urpmq -il xxx.rpm	показывает информацию о пакете xxx.rpm, находящемся в текущем каталоге и список файлов, которые он устанавливает
urpmq --changelog xxx.rpm	выводит журнал изменений пакета, находящегося в текущем каталоге
urpmq --whatrequires xxx.rpm	показывает, какие пакеты требуются пакету xxx.rpm, находящемуся в текущем каталоге
urpmf /путь/к/файлу	какой пакет установил «файл» в каталог «/путь/к»
urpmi --fuzzy --test xxx	показывает все rpm, совпадающие со строкой «xxx»

<code>rpm -qf /path/to/file</code>	то же, что и <code>urpmf</code> , но ищет среди установленных пакетов
<code>urpmi.update updates</code>	обновляет локальные данные о доступных пакетах хранилища «updates»
<code>urpmc</code>	обновляет локальные данные о доступных пакетах, принадлежащих ко всем источникам
Команда	Что делает
<code>urpme xxx</code>	удаляет пакет «xxx» (и все пакеты, зависящие от этого пакета)
<code>urpmi --keep xxx.rpm</code>	устанавливает пакет «xxx.rpm», содержащийся в текущем каталоге, со всеми зависимостями этого пакета, но если что-нибудь в результате установки должно быть удалено, пакет установлен не будет.
<code>urpmi --update --auto-select</code>	устанавливает доступные обновления с включённых источников
<code>urpmi --test --keep --auto --auto-select</code>	обновить все пакеты с включённых источников, но ничего не устанавливать и не удалять, просто сообщить, что это будет работать
<code>urpmi --keep --auto --auto-select</code>	обновить все пакеты из включённых источников, но ничего не удалять, просто сообщить если что-то не работает
<code>urpmi --auto-select</code>	автоматически выбрать пакеты, которые можно обновить

urpmi.addmedia

Всё начинается с добавления репозитория (или «хранилища», «зеркала»). Репозиторий можно добавить с помощью программы **urpmi.addmedia**.

Список источников можно получить на странице Easy Urpmi ^[2] или urpmi.edumandriva.ru ^[3]. Формат команды для добавления репозитория имеет следующий вид:

```
urpmi.addmedia имя_источника ftp://ftp.сайт.ком/путь/к/ROSA/RPMS
```

Но удобнее использовать программу с графическим интерфейсом (команда **edit-urpm-sources.pl**). Эту же программу можно запустить из центра управления ROSA (раздел «Управление программами», пункт «Настройки источников настройки/обновления ПО»).

Структура зеркал ftp для ROSA включает в себя следующее:

- **main** — содержит наиболее важные программы, поддерживаемые ROSA;
- **contrib** — содержит дополнительные программы, добавленные участниками сообщества ROSA, но которые не обязательно должны получать обновления по безопасности;
- **non-free** — содержит некоторые программы, которые не являются свободными.

Каждое хранилище подразделяется ещё на 4:

- **release** — здесь лежат пакеты в том состоянии, в котором они находились на момент выхода релиза;
- **updates** — здесь лежат пакеты, обновлённые со дня выхода релиза по причинам безопасности или исправления ошибок. Это хранилище должно быть обязательно включено, даже если скорость вашего Интернет-соединения низкая. При добавлении хранилища с обновлениями добавьте переключатель **--update**, чтобы **urpmi** мог отличить это хранилище от обычных хранилищ;
- **backports** — здесь лежат некоторые пакеты с новыми версиями программ, бэкпортированными из cooker;
- **testing** — используется для тестирования свежих обновлений, что позволяет участникам, сообщаящим об ошибках, проверить корректность обновлений.

Чтобы устанавливать пакеты с зеркала, **urpmi** необходимо предоставить один из файлов, хранящих в сжатом виде либо *минимально необходимый*, либо *полный* набор данных о пакетах.

Для более полного понимания организации структуры зеркала рекомендуется ознакомиться со страницами политики структуры зеркала и политики хранилища программ.

Восстановление потерянных источников (cdfs)

Если в списке хранилищ отсутствуют некоторые CD или DVD, то их можно добавить в список с помощью команды:

```
urpmi.addmedia --distrib cdrom removable://mnt/cdrom
```

. В момент запуска команды на исполнения в приводе компакт-дисков должен находиться носитель, который нужно добавить в список.



Примечание

Приведённый формат команды соответствует системам, версия которых ниже 2007.1. Все отсоединяемые устройства теперь монтируются в каталоге */media*.

Копирование CD в домашний каталог и использование копий в urpmi

Если вам не нравится жонглировать дисками при обновлении системы и у вас есть свободное место, создайте каталог (например: */home/uid/CDS/*), скопируйте в этот каталог рекурсивно каталоги *base/* и *ROSA/* (т. е. вместе со всем содержимым их вложенных подкаталогов) с первого установочного CD. Затем, скопируйте все каталоги, начиная с *RPMS2* вплоть до *RPMS8*, со всех CD/DVD в каталог *ROSA/*. После извлечения или отключения установочных CD выполните следующую команду от имени суперпользователя root:

```
urpmi.addmedia --distrib HD file://home/uid/CDS
```

Так как в ROSA, начиная с версии **2010.1**, структура каталогов на CD была изменена, список каталогов, которые нужно рекурсивно скопировать, выглядит теперь так:

```
CD1:/media/main
CD2:/media/main2
CD3:/Applications
    /Drivers
    /media/commercial
    /media/dkms
    /media/drivers
    /main3
CD4:/kde3.3
    /media/main4
CD5:/media/main5
CD6:/Applications
    /media/main6
```

По окончании копирования выполните следующие команды:

```
cd /home/uid/CDS
genhdlist
```

Затем, воспользуйтесь менеджером источников. К нему можно получить доступ из центра управления ROSA (*центр управления ROSA --> Управление программами --> Настройка источников установки/обновления ПО*). Выберите созданный вами каталог и отключите все записи, имеющие отношение к CD.

Другой метод заключается в копировании образов ISO в различные точки монтирования. Подробнее смотри копирование CD на жёсткий диск и последующее их монтирование.

Копирование rpms (включая установочные CD) в отдельный каталог. Использование копии в urpmi

Создайте каталог для хранения всех rpms, например `~/RPMS`

Чтобы скопировать сразу «кучу» rpms (с установочных CD, например):

```
find /INSTALL_CDS/ -name *.rpm -print -exec cp {} ~/RPMS \;
```

Если вы скопировали установочные CD на жёсткий диск, или у вас есть их образы ISO на диске (о том, как монтировать образы, смотри копирование CD на жёсткий диск), это можно сделать за один шаг.

В этом примере CD монтируются как `/INSTALL_CDS/CD1`, `/INSTALL_CDS/CD2` и т. д.

```
cd ~/RPMS
```

```
genhdlist
```

Это приведёт к созданию файлов `hdlist.cz` и `synthesis.hdlist.cz`, которые основываются на том, что они найдут в каталоге, в котором находятся.

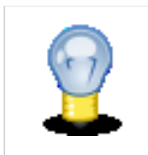
Найдите публичные ключи для rpms и скопируйте их в `~/RPMS/pubkey`. Они должны находиться в `INSTALL_CDS/CD1/media/media_info`:

```
mkdir ~/RPMS/pubkey
```

```
cp /INSTALL_CDS/CD1/media/media_info/pubkey* ~/RPMS/pubkey/
```

Затем, от имени суперпользователя добавьте источник:

```
urpmi.addmedia local_rpms file:///home/uid/RPMS/
```



Примечание

Обратите внимание на вкладку «Обсуждение» соответствующей английской статьи.

Добавление источника для дистрибутива

urpmi может добавить источники (т. е. main, updates, contrib ...) с выбранного зеркала с помощью одной-единственной команды. Наберите в командной строке следующую команду от имени суперпользователя:

```
urpmi.addmedia --distrib ftp://'MIRROR_SITE'/pub/ROSA/official/'VERSION'/'ARCH'
```

где:

- 'MIRROR_SITE' действующий URL сервера ftp;
- 'VERSION' версия вашей системы ROSA, для которой вы хотите добавить источник
- 'ARCH' одна из двух архитектур: i586 или x86_64, или возможно другая, которая существует не для всех версий ROSA (например sparc, ppc и т. д.).

urpmi.update

Команда **urpmi.update** обновляет список пакетов репозитория. Этот список меняется всякий раз, когда на сервере меняются пакеты. Таким образом, список необходимо обновлять каждый раз, когда вы захотите установить новый пакет с репозитория, в котором возможны изменения среди пакетов, например ROSA Cooker. Если вы пользуетесь репозиторием стабильного релиза, то такой репозиторий обычно не изменяется, поэтому нет необходимости обновлять список пакетов перед установкой какого-либо пакета. Это работает примерно следующим образом:

```
urpmi.update название_источника
```

или

```
urpmi.update -a
```

Аргумент -a означает "все источники".

urpmi.removemedia

Если Вы хотите избавиться от репозитория, используйте следующую команду:

```
urpmi.removemedia nameofmedia
```

urpmf

С помощью **urpmf** можно узнать, к какому пакету принадлежит тот или иной файл, а также посмотреть описание пакета и т. п.

Допустим, что вы хотите скомпилировать программу, написанную на языке программирования C, а компилятор жалуется на отсутствие файла *jpeglib.h*. Достаточно набрать следующую команду:

```
urpmf jpeglib.h
```

Например, вывод команды выглядит следующим образом:

```
libjpeg62-devel:/usr/include/jpeglib.h  
mozilla-devel:/usr/include/mozilla-1.4a/jpeg/jpeglib.h
```

Это означает, что заголовочный файл *jpeglib.h* является частью пакета **libjpeg62-devel**. Теперь, чтобы его установить, нужно набрать следующую команду:

```
urpmi libjpeg62-devel
```

Другой пример. Допустим, что вы хотите установить почтовый клиент, но вы не знаете ни одного почтового клиента для Linux. Вы можете выполнить поиск по описанию пакетов, используя выражения «mail» и «client». Например, команда **urpmf --summary mail -a client** возвратила следующее:

```
evolution:Integrated GNOME mail client, calendar and address book.  
squirrelmail:Squirrelmail is a webmail client for PHP4.  
sylpheed-claws:Enhanced version of the Sylpheed e-mail client  
comsat:A mail checker client and comsat mail checking server.  
cscmail:CSCMail is a GTK email client written in Perl  
sylpheed:A GTK+ based, lightweight, and fast e-mail client  
tradeclient:Email Client with PIM features for X
```

Параметр **--summary** используется для поиска по описанию. **-a** означает логическое «И».

urpmi

Это — главная команда программы управления пакетами, которая используется для установки пакетов вместе со всеми зависимостями:

```
urpmi имя_пакета
```

Например, если вы считаете, что **Sylpheed** — достойный для установки почтовый клиент, его можно установить с помощью команды: **urpmi sylpheed**.

Замечание: команда **urpmi имя_пакета** не обязательно установит последнюю версию пакета. **Urpmi** сравнивает номер версии установленного пакета с номером версии пакета из включённого источника. Исключение составляет тот факт, что иногда обновление пакета доступно под немного иным названием, обе версии пакета, старая и новая, остаются доступными во включённом источнике. Это делается по причинам обратной совместимости.

Here is a notable example **urpmi autoconf** will not result in the latest version being installed, for the highest version of that package name is autoconf2.5-2.60.

Вместо этого следует использовать команду **urpmi autoconf2.5**, которая приведёт к установке последней версии 2.5*.



Примечание

Заметьте, что в вашей системе могут быть установлены 2 версии: ROSA автоматически выберет наиболее подходящую. Для получения более подробной информации по данному примеру смотрите `/usr/share/doc/autoconf-2.13/IMPORTANT.README.MDK`.

Ниже даётся несколько замечаний об использовании команды **urpmq —fuzzy**, разъясняющих, как избежать в будущем подобных проблем с версиями.

Ещё один пример использования **urpmi** — **обновление системы с помощью последних обновлений по безопасности и баг-фиксам**.

```
urpmi.update updates && urpmi --update --auto-select
```

В этом примере предполагается, что у **urpmi** есть источник под названием «updates», который был объявлен при добавлении как «источник обновлений». Поскольку содержимое источника обновлений меняется часто, необходимо периодически проверять наличие новых обновлений с помощью **urpmi.update**. Символы **&&** означают «если первая команда успешно выполнена, выполнить вторую команду». Во второй команде ключ **--update** означает «искать только в источниках, заявленных как обновляемые», а ключ **--auto-select** означает «выбрать доступные обновления для пакетов, установленных в системе». **Urpmi** выведет список пакетов, доступных для обновления, и спросит разрешения о продолжении своей работы.

Если добавить ключ **--auto**, все обновляемые пакеты будут обновлены, **включая** все требуемые зависимости. Если вы предпочитаете использовать графический интерфейс, воспользуйтесь программой **rpmrake** из центра управления ROSA: [Управление программами](#) → [Обновление системы](#).

urpmi --parallel

В параллельном режиме работы **urpmi** обновления загружаются на одну из машин, а она в свою очередь раздаёт их остальным компьютерам, входящим в локальную сеть (смотри [4]).

Команда **urpmi --parallel** установит обновления на всех определённых вами компьютерах. Подробнее смотри man-страницу **urpmi** (**man urpmi**), а также [5].

Параллельный режим **urpmi** работает следующим образом: вы запускаете на выполнение команду **urpmi**, которая выполняется параллельно на нескольких машинах. Машина, на которой вы запустили команду, тестирует результат на каждой машине из группы по очереди, загружает все необходимые пакеты для всех машин группы, предоставляет соответствующие пакеты каждой машине, затем, вызывает **urpmi** на машине для выполнения актуальной установки. Это отличная идея для быстрой установки программ на всех ваших компьютерах, или даже для поддержания программного обеспечения на всех компьютерах в актуальном состоянии с помощью нескольких команд, кроме того, это позволяет экономно использовать полосу пропускания канала, так как каждый необходимый пакет загружается только один раз. Единственный недостаток в настоящий момент — вы не можете включить сервер в группу, что является недостатком для малых домашних локальных сетей.

Итак, как же этим пользоваться. На самом деле, всё просто. Сначала проверьте, что вы можете достучаться с сервера до каждой клиентской машины через **ssh** от имени суперпользователя (вы можете ввести идентификационную фразу или пароль, или можно использовать **ssh-add** для настройки ключей). Итак, надо установить **urpmi-parallel-ssh** на серверной машине, и исправить файл `/etc/urpmi/parallel.cfg`, он должен выглядеть примерно таким образом:

```
local:ssh:toy:htpc
```

первый параметр — имя группы (оставлено на ваше усмотрение), второй параметр оставьте «ssh». Оставшиеся параметры — имя хостов машин в группе. Их может быть сколько угодно (сервер не должен входить в их список: **urpmi** просто откажется работать, обнаружив свои lock-файлы).

Далее, вы можете использовать этот файл. На сервере наберите команду:

```
urpmi --parallel local somepackage
```

До тех пор, пока **urpmi** на сервере будет иметь доступ ко всем пакетам, необходимым всем клиентским машинам из своих источников **urpmi**, всё должно работать гладко. Простейший способ гарантировать это — иметь на всех машинах (на сервере и клиентах) одни и те же источники **urpmi**.

Кроме того, можно постоянно поддерживать группу машин в актуальном состоянии. Например, чтобы содержать все машины обновлёнными, в домашней сети, которая включает машины zen (сервер), toy и htpc, запустите следующую последовательность команд на zen:

```
fanout "localhost toy htpc" "urpmi.update -a"
```

программа **fanout** запускает одну команду на нескольких машинах одновременно, используя **ssh**.

```
urpmi -auto-select -keep -noclean -v
```

noclean оставляет пакеты на машине zen после установки; это значит, что когда команда **parallel** будет запущена в следующий раз, она не станет скачивать эти пакеты заново.

```
urpmi --parallel local -auto-select -keep -v
```

urpme

urpme — команда, предназначенная для удаления программного обеспечения с компьютера. Также как и **urpmi**, **urpme** обрабатывает зависимости и извещает пользователя, что установленное программное обеспечение зависит от того программного обеспечения, которое он собирается удалить. В таком случае, у пользователя есть возможность отменить процедуру удаления или удалить пакет вместе с другими зависящими от него пакетами. Использование:

urpme название_пакета

Например, в системе установлено 2 почтовых клиента: Sylpheed и Evolution. В этом случае, если один из клиентов не используется, его можно удалить. К тому же, таким образом можно освободить немного места на жёстком диске. Чтобы удалить пакет Evolution, воспользуйтесь следующей командой:

urpme evolution

urpmq

urpmq позволяет Вам запрашивать базу данных **rpm**, благодаря чему Вы можете выяснить информацию о пакетах, установленных Вами, а так же о других вещах в базе данных, например о том, какие у Вас имеются источники установки.

```
[root@isis root]# urpmq --list-media
contrib
cooker
plf
```

Предостережение: Используя **urpmq** для поиска пакетов, удостоверьтесь, что понимаете разницу между использованием его с опцией *--fuzzy* и без нее. Вы можете не найти некоторые пакеты, которые Вы искали если не будете осторожными. Если **urpmq** находит пакет, название которого точно соответствует Вашему запросу, по умолчанию она выводит только название этого пакета, и Вы не увидите других пакетов, которые содержат то же название.

Например:

```
# Представьте, что Ваш лучший друг рассказал Вам о vegastrike, классной 3D-игре.
[root@localhost augustin]# urpmq vegastrike
vegastrike
# Круто! Есть пакет ROSA с точно таким именем!
# Но Вы не видите других пакетов, которые содержат то же мя!

# Запросим еще раз: обратите внимание на пропущенную 'e' в конце названия пакета
[root@localhost augustin]# urpmq vegastrik
No package named vegastrik
The following packages contain vegastrik: vegastrike, vegastrike-data
You should use "-a" to use all of them
# В этот раз результат запроса сильно отличается
# Мы увидели уже не один, а два пакета.

# Сравните следующий запрос с предыдущими двумя
[root@localhost augustin]# urpmq --fuzzy vegastrike
vegastrike
```



```
vegastrike-data
    // Обратите внимание на разницу, которую обеспечивает --fuzzy
```

urpmq --fuzzy так же очень полезна, когда нужно узнать какие другие доступные версии пакета имеются: иногда новая мажорная версия не предоставляет полной обратной совместимости. По этой причине, новая версия может распространяться под немного другим именем пакета. Сейчас, Вы уверены, что не пропустите ничего.

Например:

```
[root@localhost augustin]# urpmi mplayer
    #Это не установит последнюю версию проигрывателя
[root@localhost augustin]# urpmq --fuzzy mplayer
The following packages contain mplayer:
kmplayer
mplayer
mplayer-fonts
mplayer-gui
mplayer-skins
mplayer1.0
mplayer1.0-gui
mplayerplugin
transcode
xmms-mplayer
    # Видите! Новый релиз 1.0 распространяется под другим названием
    # Вы предупреждены!
```

gurpmi

gurpmi — графический фронт-энд для программы **urpmi**. Может использоваться из командной строки для установки и поиска пакетов. Используется программой **rpmdrake** для отображения сообщений о необходимости выполнения некоторых действий со стороны пользователя.

rpmdrake

Для тех же задач ROSA предлагает набор графических инструментов. Подробнее см. Rpm Drake.

smart

Пакеты **smart**, **smart-update** и **smart-gui**, первоначально использовавшиеся фирмой Conectiva, представляют собой инструменты, основанные на языке программирования Python. Пакеты доступны в [contrib](#). Эти инструменты отличаются высокой скоростью работы, достаточной надёжностью, а также имеют некоторые преимущества перед оригинальными инструментами ROSA. Официально не поддерживаются. Домашняя страница: Smart Home ^[6]. На данный момент доступны во многих других дистрибутивах.

Вопросы безопасности

Проблемы безопасности

Чтобы программа **urpmi** могла устанавливать пакеты, она должна быть запущена с привилегиями суперпользователя. Хитрый злоумышленник может обмануть пользователя, подсунув ему пакет для установки. Такие пакеты на первый взгляд выглядят как обычные пакеты, которыми регулярно пользуются множество пользователей, но в данном случае опасность заключается в том, что этот пакет может содержать программу-троян или какой-либо другой вредоносный код. Как только пакет будет установлен в системе, приложение может выпустить в систему какой-нибудь вирус, червя или даже программу-шпиона... До настоящего времени нам неизвестно о попытках использования таких эксплоитов, но в ближайшие несколько лет, когда GNU/Linux получит более широкое распространение, возможно, кто-то захочет попытаться это претворить в жизнь. К счастью, таких людей ждёт неудача, потому что внимание, уделяемое безопасности, всегда было неотъемлемой частью Linux-сообщества. Большинство пакетов имеют подпись, которая подтверждает подлинность пакета. Подробнее см. GnuPG.

Злоумышленник может попытаться внедрить вредоносную программу в пакет. Перед установкой **urpmi** проверит целостность пакета, используя хэш-код MD5, и включённый в пакет ключ gpg. Подробнее см. соответствующий раздел.

Иногда при работе с **urpmi** могут быть получены следующие строки:

```
rpmdb: /var/lib/rpm/Pubkeys: unexpected file type or format
rpmdb: /var/lib/rpm/Pubkeys: unexpected file type or format
rpmdb: /var/lib/rpm/Pubkeys: unexpected file type or format
rpmdb: /var/lib/rpm/Pubkeys: unexpected file type or format
rpmdb: /var/lib/rpm/Pubkeys: unexpected file type or format

rpmdb: /var/lib/rpm/Pubkeys: unexpected file type or format
rpmdb: /var/lib/rpm/Pubkeys: unexpected file type or format
rpmdb: /var/lib/rpm/Pubkeys: unexpected file type or format
```

Непонятно, в чём же здесь проблема: может быть файл был повреждён во время обновления, а может, процесс **urpmi** был принудительно завершён нетерпеливым пользователем.

Решение проблемы: удалить файл `/var/lib/rpm/Pubkeys`, повторно импортировать открытые ключи. См. GnuPG.

Хэширование MD5

Хэш MD5 представляет собой код, включённый в rpm-пакет. Этот код позволяет **urpmi** проверить хэш пакета. Если по каким-либо причинам файл был повреждён, **rpm** и **urpmi** откажутся его устанавливать.

Пример:

```
[root@localhost augustin]# urpmi kdesdk
rpmdb: /var/lib/rpm/Pubkeys: unexpected file type or format
error: cannot open Pubkeys index using db3 - Invalid
argument (22)
The following packages have bad signatures:
/var/cache/urpmi/rpms/kdesdk-3.1.3-9mdk.i586.rpm: Invalid
signature ((SHA1) DSA sha1 MD5 GPG GPG#70771ff3 NOT OK)
Do you want to continue installation ? (y/N) y
installing /var/cache/urpmi/rpms/kdesdk-3.1.3-9mdk.i586.rpm
```

```
error: /var/cache/urpmi/rpms/kdesdk-3.1.3-9mdk.i586.rpm: MD5
digest: BAD Expected(97f2ba5a91888cd3af40f89be6b65868) !=
(393221db35071aa90eaa73816a9a5ba8)
unable to install package
/var/cache/urpmi/rpms/kdesdk-3.1.3-9mdk.i586.rpm
```

В данном случае было получено два уведомления *Invalid signature ((SHA1) DSA sha1 MD5 GPG GPG#70771ff3 NOT OK)* и *MD5 digest: BAD Expected(97f2ba5a91888cd3af40f89be6b65868) != (393221db35071aa90eaa73816a9a5ba8)*. Файл *kdesdk-3.1.3-9mdk.i586.rpm* повреждён и установлен не будет.

Решение: удалить файл из */var/cache/urpmi/rpms/* и скачать его заново. Попытайтесь ещё раз установить его с помощью **urpmi** или **rpm --import <filename>**

Если это не помогло, значит используемое зеркало содержит повреждённый файл. Попробуйте скачать файл вручную с двух-трёх других зеркал и сохранить его в каталоге */var/cache/urpmi/rpms/*. Всегда удаляйте вручную предыдущий скачанный файл перед последующей попыткой использования другого зеркала. Затем, установите его снова с помощью **urpmi**.

Если после нескольких попыток будет возникать одна и та же проблема, возможно, что повреждённый файл попал на каждый ftp-сервер. Обратитесь в форум или список рассылки, возможно, что другие пользователи также столкнулись с этой проблемой. В таком случае, скорее всего новый файл появится в ближайшее время.

Решение проблем

База данных RPM заблокирована

От имени суперпользователя можно ввести:

```
killall urpmi urpmi.update urpme rpm urpmi.addmedia
rm -f /var/lib/urpmi/.LOCK /var/lib/rpm/.RPMLOCK
```

Если это не разблокирует базу, можно перезагрузить систему.

С другой стороны, такое использование программы **killall** повышает риск повреждения базы данных RPM, так как вы не позволяете программам, работающим с базой, завершить свою работу. Поэтому будьте осторожны, используя данный приём.

Пересборка базы данных

Может статься, что база rpm-пакетов поломалась. В таком случае, вы можете получить сообщение, что не установленный пакет уже установлен или наоборот. Чтобы исправить, воспользуйтесь командами:

```
rm -rf /var/lib/rpm/__db*
rpm --rebuilddb
```

* Неплохой идеей может быть периодической резервное копирование директории */var/lib/rpm*. В этом случае вы всегда сможете вернуть предыдущую рабочую версию такими командами:

```
rpm -ivh --justdb --noscripts --notriggers
```



Примечание

В современном rpm5, опция **rebuilddb** устарела. Для восстановления базы, необходимо использовать соответствующую утилиту **db_recover** из **berkeleydb** внутри директории */var/lib/rpm*. Например, если в системе используется **libdb-5.1**, то команда будет выглядеть следующим образом: **cd /var/lib/rpm && db51_recover -ev**

Ошибка "medium contrib uses an invalid list"

```
rm /var/lib/urpmi/list.contrib
```

Эта команда устранил ошибку, не влияя на установку пакетов.

RPM package verification

Очень полезным при проверке сломавшейся системы может оказаться проверка установленных пакетов на соответствие базе данных пакетов.

```
rpm -Va
```

Эта команда сообщит вам, какие пакеты изменились с момента установки. Она покажет все несоответствия установленных в системе файлов тому, что должно быть, согласно базе urpmi. При непредвиденном отключении питания, зависании системы и других форс-мажорных обстоятельствах (и человеческом факторе) некоторые файлы могут повредиться или исчезнуть. Узнав, какие пакеты повреждены, вы можете переустановить их и исправить работу системы. Таким образом можно восстановить работоспособность даже незагружающейся системы, войдя в неё при помощи какого-нибудь Live-CD. Для переустановки пакетов используйте команду

```
urpmi --replacepkgs пакет_1 пакет_2 ...
```

Для проверки одного пакета используйте команду **rpm -V имя_пакета** (достаточно только имени пакета).

Для проверки контрольной суммы пакета (md5sum), хэша и цифровой подписи gpg:

```
rpm -K foo.123.rpm
```

или

```
rpm -K foo*
```

Обновление до последней версии Росы с помощью **urpmi**

Urpmi также может быть использован для обновления вышей машины до более новой версии Росы. Для этого необходимо добавить репозитории с новыми пакетами и запустить **urpmi --auto-update**.

Другие возможности **urpmi**

Установка пакета из сети и из локального файла

urpmi может быть использован для установки локальных файлов **rpm** с разрешением всех зависимостей. Например, если вы собрали новую версию пакета **foo-1.0-1bar.rpm**, вы можете его установить в систему с помощью **urpmi foo-1.0-1bar.rpm**. Более того, в качестве аргумента можно задать ссылку на http или ftp сервер (и даже ssh, при условии, что и на сервере, и на клиенте установлен rsync).

Получение списка зависимостей перед установкой

С помощью **urpmq** вы можете получить детальную информацию о зависимостях пакетов rpm.

- **urpmq -d** выведет список всех пакетов, которые необходимы для установки заданного rpm.
- используйте опцию **-m** для вывода только тех пакетов, которые еще не установлены в системе.
- **--sources** выведет ссылки, с которых будут скачаны необходимые пакеты.

Таким образом, команда **urpmq -d -m --sources** выведет вам список ссылок на пакеты, которые необходимо доустановить в систему. Такой список полезен, если у вас плохое соединение с Интернет и вы хотите скачать пакеты где-то в другом месте.

Получение информации о пакетах

С помощью **urpmq -i** вы можете получить детальную информацию о пакете из репозитория. Попробуйте **urpmq -i bash**.

Установка ПО не в формате RPM

По возможности, для установки ПО вы должны использовать RPM-пакеты из репозитория ROSA. Мы уверены, что наши репозитории содержат все необходимое для большинства пользователей. Тем не менее, иногда возникает необходимость поставить программу, отсутствующую в репозитории.

Старайтесь не устанавливать ПО в обход базы данных RPM. Также не устанавливайте пакеты при помощи **rpm --nodeps**; мы не можем гарантировать, что установленное таким образом ПО будет корректно функционировать и не сломает систему.

Компилирование исходных кодов

Вместо уже знакомой тройки команд:

```
./configure
make
make install
```

используйте **checkinstall**:

```
./configure
make
checkinstall
```

После этого вы получите пакет, который можно установить с помощью **urpmi**, а потом удалить с помощью этой же программы. Исходный код некоторых приложений уже содержит **spec**-файл, и вы можете попробовать его использовать для сборки RPM-пакета:

```
rpmbuild -tb ballname.tar.gz
```

Установка двоичных пакетов

Такие программы должны находиться в каталоге [*/usr/local*](#). Этот каталог предназначен для хранения неофициальных пакетов.

Проприетарные драйвера

Все проприетарные драйвера находятся в хранилище [*non-free*](#).

Пересборка RPMS

Если у вас есть rpm-пакет, собранный для системы, отличающей от вашей, его можно пересобрать. Подробнее см. сборка пакетов.

Копирование CD/DVD на жёсткий диск и их последующее монтирование

Если вам не нравится постоянно менять носители в приводе, вы можете сохранить образы носителей на жёстком диске, настроив **urpmi** на использование образов носителей вместо физических носителей.

Сначала, создайте каталог для образов ISO носителей:

```
mkdir ~/iso
```

Скопируйте все носители в соответствующие каталоги:

```
dd if=/dev/cdrom of=~/iso/cd1
dd if=/dev/cdrom of=~/iso/cd2
dd if=/dev/cdrom of=~/iso/cd3
...
```

От имени суперпользователя создайте точки монтирования для каждого носителя:

```
mkdir /mnt/cd1
mkdir /mnt/cd2
mkdir /mnt/cd3
...
```

Смонтируйте носители с помощью команд:

```
mount -t iso9660 -o loop /home/uid/iso/cd1 /mnt/cd1
mount -t iso9660 -o loop /home/uid/iso/cd2 /mnt/cd2
mount -t iso9660 -o loop /home/uid/iso/cd3 /mnt/cd3
...
```

Внесите изменения в файл [*/etc/urpmi/urpmi.cfg*](#).

Пример исходного файла:

```
Installation\ CD\ 1\ (cdrom1) removable://mnt/cdrom/media/main {
    hdlist: hdlist.Installation CD 1 (cdrom1).cz
    key-ids: 70771ff3
    removable: /dev/hdc
    with_hdlist: ../../media/media_info/hdlist1.cz
}
```

Изменённый файл:

```
Installation\ CD\ 1\ (cdrom1) file://mnt/cd1/media/main {  
    hdlist: hdlist.Installation CD 1 (cdrom1).cz  
    key-ids: 70771fff3  
    with_hdlist: ../../media/media_info/hdlist1.cz  
}
```

Другими словами, измените *removable://mnt/cdrom* на *file://mnt/cd1* для каждого носителя cd1, cd2, cd3 и т. д. Также удалите строки, содержащие запись *removable*:

```
removable: /dev/hdc
```

Примечания

- [1] <http://sylpheed.sraoss.jp/en/>
- [2] <http://easyurpmi.zarb.org/>
- [3] <http://urpmi.mandriva.ru>
- [4] <http://archives.mandrivalinux.com/expert/2006-03/msg00001.php>
- [5] <http://www.happyassassin.net/2005/05/04/a-quick-guide-to-urpmi-parallel>
- [6] <http://labix.org/smart>

Источники и основные авторы

Тонкости работы с Urmі Источник: <http://wiki.rosalab.ru/ru/index.php?oldid=7994> Редакторы: D uragan, PastorDi, Михаил Александрович

Источники, лицензии и редакторы изображений

Image:Idea.png Источник: <http://wiki.rosalab.ru/ru/index.php?title=Файл:Idea.png> Лицензия: неизвестно Редакторы: PastorDi, StasFomin

Image:Dialog-warning.png Источник: <http://wiki.rosalab.ru/ru/index.php?title=Файл:Dialog-warning.png> Лицензия: неизвестно Редакторы: PastorDi, StasFomin

Лицензия

Creative Commons Attribution-ShareAlike 3.0 License (CC-BY-SA)
<http://creativecommons.org/licenses/by-sa/3.0/>
