

# Утилита rsync: синхронизация директорий и backup-ы

15.12.2019

Теги: CLI • Linux • Директория • Команда • Конфигурация • Настройка • Сервер • Файл

Утилита rsync – это программное обеспечение с открытым исходным кодом, которое можно использовать для синхронизации файлов и директорий с локального компьютера на удаленный и наоборот. Примечательная особенность rsync – возможность передавать файлы с шифрованием, используя ssh в качестве транспорта.

Синтаксис команды

\$ rsync **опции** **источник** **приемник**

Копировать

## Примеры использования

Перед тем, как копировать файлы, всегда стоит сделать пробный прогон (ключ `-n`) в режиме с показом подробностей (`-v`):

\$ rsync **-avn** /path/to/src/ **user@example.com:/path/to/dst**

Копировать

В этом режиме rsync покажет список файлов, которые он копирует. После этого можно запускать настоящее копирование:

\$ rsync **-av** /path/to/src/ **user@example.com:/path/to/dst**

Копировать

В этой команде ключ `-a` подразумевает рекурсивное копирование всех файлов и каталогов включая их атрибуты: владелец, группа, дата создания и изменения. Ключ `-v` даст подробный отчет о работе по мере выполнения и по окончании.

### Примечание

Обратите внимание на слэш в конце первого аргумента – без него файлы источника `/path/to/src/файлы` будут размещены в `/path/to/dst/src/файлы`. А при наличии слэша в конце первого аргумента – файлы будут в `/path/to/dst/файлы`, что скорее всего и нужно.

По умолчанию rsync попытается использовать транспорт ssh. При использовании другого порта для ssh, его нужно указать:

## Категории блога

- Web-разработка
  - HTML и CSS
  - JavaScript и TypeScript
  - PHP и MySQL
  - CMS Битрикс
  - CMS WoprdPress
  - Yii2 и Laravel
  - Python и Flask
  - Web-аналитика
  - Разное
- ОС Linux
  - Команды
  - Сценарии
  - Разное
- 1С:Предприятие
  - Программирование
  - Язык запросов
  - Разное
- Локальная сеть
  - Разное

## Облако тегов

- 1С:Предприятие (31)
- Yii2 (69)
- API (29)
- БазаДанных (95)
- Bash (43)
- Битрикс (66)
- CLI (126)
- Блог (29)
- CMS (139)
- Верстка (43)
- CSS (50)
- ИнтернетМагаз... (84)
- Frontend (75)
- КаталогТоваров (87)
- HTML (66)
- Класс (30)
- JavaScript (150)
- Клиент (28)
- Laravel (72)
- Ключ (28)
- Linux (172)
- Команда (88)
- MySQL (76)
- Компонент (60)
- PHP (125)
- Конфигурация (69)
- React.js (66)
- Корзина (32)
- SSH (27)
- ЛокальнаяСеть (33)
- Ubuntu (70)
- Модуль (34)
- Web-разработка (509)
- Навигация (31)
- Настройка (145)
- WordPress (73)

```
$ rsync -av -e "ssh -p 2222" /path/to/src/ user@example.com:/path/to/dest/
```

Точно также можно скопировать файлы из удаленного сервера на локальную машину:

```
$ rsync -av user@example.com:/remote/machine/path/ /local/machine/path/
```

Обратите внимание, что источник всегда является первым аргументом, а приемник – вторым.

#### Примечание

Для автоматического копирования, например по [cron](#), необходимо настроить авторизацию с использованием ключей.

## Основные опции

Опция [-a](#) или [--archive](#) – рекурсивно копирует файлы, включая разыменованные символьные ссылки, с сохранением связанных с файлами сведений об их владельцах, групповых и индивидуальных правах и времени последнего изменения. Использование этого ключа равносильно указанию ключей [--group](#), [--links](#), [--owner](#), [-perms](#), [--recursive](#) и [--times](#).

Опция [-b](#) или [--backup](#) – переименовывает файлы, которые в противном случае должны быть удалены или перезаписаны, добавляя к их существующим именам знак тильды (~). Эта опция обычно используется вместе с [--backup-dir](#) – вместо переименования [rsync](#) перемещает эти файлы в указанный каталог.

Опция [--backup-dir=dir](#) – используется вместе с [--backup](#) и перемещает в директорию [dir](#) те файлы, которые в противном случае должны быть удалены или перезаписаны. После перемещения старой версии файла в каталог [dir](#) новая версия из источника копируется в приемник.

#### Примечание

Если [dir](#) указан с использованием относительного пути, то это путь относительно приемника.

Опция [--copy-unsafe-links](#) – для каждого файла, являющегося символьной ссылкой на файл за пределами той иерархии, которой принадлежит источник, производится копирование файла, на который указывает ссылка, а не символьной ссылки на него.

Опция [--delete](#) заставляет [rsync](#) удалять файлы в приемнике, которые отсутствуют в источнике.

Опция [-n](#) или [--dry-run](#) – запуск утилиты без записи на диск. С опцией [--verbose](#) позволяет посмотреть, что будет сделано, если запустить [rsync](#) без [--dry-run](#). Особенно полезно запускать перед использованием [--delete](#).

Опция [-g](#) или [--group](#) – сохранение групповых связей копируемых файлов.

[ПанельУправле...](#) (29) [Установка](#) (69)[Плагин](#) (33)[Файл](#) (51)[Пользователь](#) (26)[Форма](#) (58)[Практика](#) (101)[Фреймворк](#) (192)[Сервер](#) (77)[Функция](#) (36)[Событие](#) (28)[ШаблонСайта](#) (68)[Теория](#) (106)[Все теги](#)

### Категории статей

[Web-разработка](#)[1С:Предприятие](#)[ОС Linux](#)[Регулярные выражения](#)[Разное](#)

Опция `-l` или `--links` — для каждого файла, являющегося символьной ссылкой, копирование символьной ссылки, а не файла, на который указывает ссылка, даже если файл, на который указывает ссылка, не присутствует в источнике.

Опция `--link-dest=dir` — если файл существует в источнике, но его нет в приемнике, rsync ищет в каталоге `dir` файл с точно таким же именем. Если точная копия файла будет найдена, rsync создает из файла в `dir` жесткую ссылку на него в приемнике. Если точная копия не будет найдена, rsync копирует файл в приемник.

Опция `-o` или `--owner` — сохранение данных о владельце копируемых файлов (только для пользователей с root-правами).

Опция `-p` или `--perms` — сохранение данных о правах доступа копируемых файлов.

Опция `-r` или `--recursive` — рекурсивный спуск по каталогу-источнику и копирование всех файлов в иерархии.

Опция `-t` или `--times` — сохранение данных о времени последнего изменения копируемых файлов.

Опция `-u` или `--update` — пропускать файл, если в приемнике оказался файл новее, чем в источнике.

Опция `-v` или `--verbose` — вывод информации о том, что в данный момент делает rsync.

Опция `-z` или `--compress` — сжатие файлов в процессе копирования.

## Зеркальное копирование

Утилиту rsync можно использовать для создания зеркальной копии каталога. На удаленной системе должен быть установлен ssh-сервер (к нему можно подключиться с использованием утилиты ssh). Если нужно запускать копирование по расписанию, следует настроить авторизацию с использованием ключей.



Опция `--update` заставляет rsync не перезаписывать самые последние версии файлов на удаленной системе более старыми версиями таких же файлов, взятых из локальной системы. Опция `-delete` заставляет rsync удалять на удаленном сервере файлы, отсутствующие на локальной системе.

## Резервное копирование

После проведения исходного резервного копирования rsync может эффективно осуществлять дополнительное резервное копирование, экономя место на диске и время копирования. При дополнительном (инкрементном) копировании сохраняются только те файлы, которые претерпели изменения со времени последнего резервного копирования.

С помощью опции `--link-dest=dir` можно превращать каждое дополняющее резервное копирование в полноценное, путем создания жестких ссылок между дополняющими резервными копиями

и неизменившимися файлами, имеющимися в исходной полноценной резервной копии.

Если `dir` задается как относительный путь, то он задается относительно файла-приемника.

```
$ rsync --archive --link-dest=../backup /path/to/src/ user@exa
```

При запуске этой команды `rsync` проходит сверху вниз по содержимому каталога `/path/to/src`, проверяя каждый найденный файл. Для каждого файла в каталоге `/path/to/src` утилита ищет в каталоге `/path/to/dst` точную копию этого файла.

- Если она находит точную копию в каталоге `/path/to/dst` — переходит к следующему файлу
- Если она не находит точную копию в каталоге `/path/to/dst` — ищет точную копию в каталоге `backup`
  - Если она находит точную копию файла в каталоге `backup`, утилита создает жесткую ссылку из файла в каталоге `backup` на файл в каталоге `/path/to/dst`
  - Если она не находит точную копию файла в каталоге `backup`, утилита копирует файл из каталога `/path/to/src` в каталог `/path/to/dst`

Небольшой `bash`-скрипт, демонстрирующий создание полноценной и дополняющей резервной копии:

```
#!/bin/bash
rm -rf backup-two-days-ago
mv backup-one-day-ago backup-two-days-ago
mv backup-latest-state backup-one-day-ago
rsync --archive --link-dest=../backup-one-day-ago /some/path/d
```

Можно немного доработать этот скрипт, чтобы хранить не три, а тридцать резервных копий:

```
#!/bin/bash

SOURCE='/home/evgeniy/data'
BACKUP='/home/evgeniy/backup'
COUNT=30 # кол-во бэкапов

rm -rf ${BACKUP}/backup-${COUNT}
for (( i = COUNT; i > 1; i-- )); do
    mv ${BACKUP}/backup-$(i - 1) ${BACKUP}/backup-${i}
done

rsync --archive --link-dest=../backup-2 ${SOURCE}/ ${BACKUP}/b
```

#### Похожие записи

- Настройка SFTP-сервера в Ubuntu 18.04 LTS
- Systemd. Служба логирования `journald`
- Монтирование NFS на сервере Ubuntu 18.04 LTS
- Создание SSH-туннеля. Часть 3 из 4
- Локализация Ubuntu Server 18.04 LTS
- Создание SSH-туннеля. Часть 2 из 4
- Создание SSH-туннеля. Часть 1 из 4

04.04.2024, 12:13

Утилита rsync: синхронизация директорий и backup-ы. Категория: ОС Linux • Разное

Поиск: [CLI](#) • [Linux](#) • [Директория](#) • [Команда](#) • [Файл](#) • [Backup](#) • [rsync](#) •  
[Сервер](#) • [Резервное копирование](#) • [Настройка](#) • [Конфигурация](#)

Узелки на память: [Web-разработка](#), [1С:Предприятие](#), [ОС Linux](#)