

Следующий: [Обзор](#) , Предыдущий: [\(каталог\)](#) , Вверх: [\(каталог\)](#) [ [Содержание](#) ][ [Индекс](#) ]

# Wget 1.24.5

Этот файл документирует утилиту GNU Wget для загрузки сетевых данных.

Авторские права © 1996–2011, 2015, 2018–2024 Free Software Foundation, Inc.

Разрешается копировать, распространять и/или изменять этот документ в соответствии с условиями лицензии GNU Free Documentation License версии 1.3 или любой более поздней версии, опубликованной Free Software Foundation; без неизменяемых разделов, без текстов на передней обложке и без текстов на задней обложке. Копия лицензии включена в раздел под названием «Лицензия на бесплатную документацию GNU».

## Оглавление

### 1 Обзор

### 2 Вызов

[2.1 Формат URL-адреса](#)

[2.2 Синтаксис опций](#)

[2.3 Основные параметры запуска](#)

[2.4 Параметры журнала и входного файла](#)

[2.5 Варианты загрузки](#)

[2.6 Параметры каталога](#)

[2.7 Параметры HTTP](#)

[2.8 Параметры HTTPS \(SSL/TLS\)](#)

[2.9 Параметры FTP](#)

[2.10 Параметры FTPS](#)

[2.11 Параметры рекурсивного поиска](#)

[2.12 Рекурсивные параметры принятия/отклонения](#)

[2.13 Статус выхода](#)

### 3 Рекурсивная загрузка

### 4 следующих ссылки

[4.1 Объединение хостов](#)

[4.2 Типы файлов](#)

[4.3 Ограничения на основе каталогов](#)

4.4 Относительные ссылки

4.5 Переход по FTP-ссылкам

## 5. Отметка времени

5.1 Использование отметок времени

5.2 Внутреннее устройство HTTP-метки времени

5.3 Внутренние устройства отметки времени FTP

## 6. Файл запуска

6.1 Расположение Wgetrc

6.2 Синтаксис Wgetrc

6.3 Команды Wgetrc

6.4 Пример Wgetrc

## 7 примеров

7.1 Простое использование

7.2 Расширенное использование

7.3 Очень продвинутое использование

## 8 Разных

8.1 Прокси

8.2 Распространение

8.3 Веб-сайт

8.4 Списки рассылки

Основной список

Устаревшие списки

8.5 Интернет-релейный чат

8.6 Сообщения об ошибках

8.7 Портативность

8.8 Сигналы

## 9 приложений

9.1 Исключение роботов

9.2 Вопросы безопасности

9.3 Участники

## Приложение А Копирование данного руководства

А.1 Лицензия на бесплатную документацию GNU

## Указатель концепций

Следующая: [Вызов](#) , Предыдущая: [Wget 1.24.5](#) , Вверх: [Wget 1.24.5](#) [ [Содержание](#) ][ [Индекс](#) ]

# 1 Обзор

GNU Wget – бесплатная утилита для неинтерактивной загрузки файлов из Интернета. Он поддерживает протоколы HTTP , HTTPS и FTP , а также получение через HTTP- прокси.

Эта глава представляет собой частичный обзор возможностей Wget.

- Wget неинтерактивен, то есть может работать в фоновом режиме, пока пользователь не вошел в систему. Это позволяет вам начать извлечение и отключиться от системы, позволяя Wget завершить работу. Напротив, большинство веб-браузеров требуют постоянного присутствия пользователя, что может стать большой помехой при передаче большого количества данных.
- Wget может переходить по ссылкам на страницах HTML , XHTML и CSS , чтобы создавать локальные версии удаленных веб-сайтов, полностью воссоздавая структуру каталогов исходного сайта. Иногда это называют «рекурсивной загрузкой». При этом Wget соблюдает Стандарт исключения роботов (/robots.txt). Wget можно поручить преобразовать ссылки в загруженных файлах в локальные файлы для просмотра в автономном режиме.
- Сопоставление имен файлов по подстановочным знакам и рекурсивное зеркалирование каталогов доступны при получении через FTP . Wget может читать информацию о временных метках, предоставленную как HTTP , так и FTP- серверами, и сохранять ее локально. Таким образом, Wget может видеть, изменился ли удаленный файл с момента последнего получения, и автоматически получить новую версию, если она изменилась. Это делает Wget подходящим для зеркалирования FTP- сайтов, а также домашних страниц.
- Wget был разработан для обеспечения устойчивости при медленных или нестабильных сетевых соединениях; если загрузка не удалась из-за проблем с сетью, она будет повторять попытки до тех пор, пока не будет получен весь файл. Если сервер поддерживает повторную загрузку, он даст ему указание продолжить загрузку с того места, где она остановилась.
- Wget поддерживает прокси-серверы, которые могут снизить нагрузку на сеть, ускорить поиск и обеспечить доступ через брандмауэры. Wget по умолчанию использует пассивную загрузку по FTP , активный FTP является опцией.
- Wget поддерживает IP версии 6, следующее поколение IP. IPv6 автоматически определяется во время компиляции и может быть отключен как во время сборки, так и во время выполнения. Двоичные файлы, созданные с поддержкой IPv6, хорошо работают как в средах только с IPv4, так и в средах с двойным семейством.

- Встроенные функции предлагают механизмы для настройки ссылок, по которым вы хотите перейти (см. [Следующие ссылки](#) ).
- Ход отдельных загрузок отслеживается с помощью индикатора прогресса. Интерактивные загрузки отслеживаются с помощью шкалы в виде «термометра», тогда как неинтерактивные загрузки отслеживаются точками, каждая точка представляет собой фиксированный объем полученных данных (по умолчанию 1 КБ). Любой датчик можно настроить в соответствии с вашими предпочтениями.
- Большинство функций полностью настраиваются либо с помощью параметров командной строки, либо с помощью файла инициализации `..wgetrc` (см. [Файл запуска](#) ). Wget позволяет вам определять *глобальные* файлы запуска (`/usr/local/etc/wgetrc` по умолчанию) для настроек сайта. Вы также можете указать расположение файла запуска с помощью опции `-config`. Чтобы отключить чтение файлов конфигурации, используйте `-no-config`. Если указаны оба параметра `-config` и `-no-config`, параметр `-no-config` игнорируется.
- Наконец, GNU Wget – свободное программное обеспечение. Это означает, что каждый может использовать его, распространять и/или изменять в соответствии с условиями Стандартной общественной лицензии GNU, опубликованной Фондом свободного программного обеспечения (см. файл КОПИРОВАНИЕ, который поставляется вместе с GNU Wget, подробнее).

---

Следующая: [Рекурсивная загрузка](#) , Предыдущая: [Обзор](#) , Вверх: [Wget 1.24.5](#) [ [Содержание](#) ] [ [Индекс](#) ]

## 2 Вызов

По умолчанию Wget вызывается очень просто. Основной синтаксис:

```
wget [ опция ]... [ URL ]...
```

Wget просто загрузит все URL-адреса , указанные в командной строке. *URL* – это *унифицированный указатель ресурса* , как определено ниже.

Однако вы можете изменить некоторые параметры Wget по умолчанию. Сделать это можно двумя способами: навсегда, добавив соответствующую команду в `wgetrc` (см. [«Файл запуска»](#)) или указав его в командной строке.

- [Формат URL-адреса](#)
- [Синтаксис опций](#)
- [Основные параметры запуска](#)

- [Параметры журнала и входного файла](#)
- [Варианты загрузки](#)
- [Параметры каталога](#)
- [Параметры HTTP](#)
- [Параметры HTTPS \(SSL/TLS\)](#)
- [Параметры FTP](#)
- [Опции FTPS](#)
- [Параметры рекурсивного поиска](#)
- [Рекурсивные параметры принятия/отклонения](#)
- [Статус выхода](#)

---

Далее: [Синтаксис параметров](#) , Предыдущий: [Вызов](#) , Вверх: [Вызов](#) [ [Содержание](#) ] [ [Индекс](#) ]

## 2.1 Формат URL

*URL* – это аббревиатура от Uniform Resource Locator. Унифицированный указатель ресурса – это компактное строковое представление ресурса, доступного через Интернет. Wget распознает синтаксис URL согласно RFC1738 . Это наиболее широко используемая форма (квадратные скобки обозначают необязательные части):

```
http://хост[:порт]/каталог/файл
ftp://хост[:порт]/каталог/файл
```

Вы также можете закодировать свое имя пользователя и пароль в URL-адресе :

```
ftp://пользователь:пароль@хост/путь
http://пользователь:пароль@хост/путь
```

Либо *пользователь* , либо *пароль* , либо и то, и другое можно не указывать. Если вы не укажете имя пользователя или пароль HTTP , аутентификация не будет отправлена. Если вы не укажете имя пользователя FTP , 'анонимный' будет использован. Если вы не укажете пароль FTP , ваш адрес электронной почты будет указан в качестве пароля по умолчанию.<sup>1</sup>

**Важное примечание :** если вы укажете в командной строке URL- адрес , содержащий пароль , имя пользователя и пароль будут видны всем пользователям системы в виде ps. В многопользовательских системах это представляет собой большой риск для безопасности. Чтобы обойти эту проблему, используйте `wget -i` - и передайте URL -адреса на стандартный ввод `Wget`, каждый в отдельной строке, заканчивающейся *C-d*.

Вы можете закодировать небезопасные символы в URL-адресе как '%ху', являющийся шестнадцатеричным представлением значения ASCIIху символа . Некоторые распространенные небезопасные символы включают в себя: '%' (цитируется как '%25'), ':' (цитируется как '%3A'), и '@' (цитируется как '%40'). Полный список небезопасных символов см. в RFC1738 .

Wget также поддерживает эту турефункцию для URL- адресов FTP . По умолчанию документы FTP извлекаются в двоичном режиме (введите 'я'), что означает, что они загружаются без изменений. Еще один полезный режим – «а' ( *ASCII* ), который преобразует разделители строк между различными операционными системами и, таким образом, полезен для текстовых файлов. Вот пример:

```
ftp://хост/каталог/файл;тип=а
```

Также поддерживаются два альтернативных варианта спецификации URL-адресов по историческим (истерическим?) причинам и их широкому использованию.

Синтаксис только для FTP (поддерживается NcFTP):

```
хост:/каталог/файл
```

Синтаксис только для HTTP (представленный Netscape):

```
хост[:порт]/каталог/файл
```

Эти две альтернативные формы устарели и могут прекратить поддержку в будущем.

Если вы не понимаете разницу между этими обозначениями или не знаете, какой из них использовать, просто используйте обычный формат, который вы используете в своем любимом браузере, например Lynxили Netscape.

Следующая: [Основные параметры запуска](#) , Предыдущая: [Формат URL](#) , Вверх: [Вызов](#) [ [Содержание](#) ][ [Индекс](#) ]

## 2.2 Синтаксис опций

Поскольку Wget использует GNU getopt для обработки аргументов командной строки, каждая опция имеет не только короткую, но и длинную форму. Длинные варианты удобнее запоминать, но их ввод требует времени. Вы можете свободно смешивать различные стили параметров или указывать параметры после аргументов командной строки. Таким образом, вы можете написать:

```
wget -r --tries=10 http://fly.srk.fer.hr/ -o журнал
```

Пробел между опцией, принимающей аргумент, и аргументом можно опустить. Вместо '-o журнал' ты можешь написать '-олог'.

Вы можете объединить несколько опций, не требующих аргументов, например:

```
wget -drc URL-адрес
```

Это полностью эквивалентно:

```
wget -d -r -c URL
```

Поскольку параметры могут быть указаны после аргументов, вы можете завершить их с помощью '--'. Итак, следующие люди попытаются загрузить URL '-Икс', сообщая о невыполнении бревно:

```
wget -o журнал -- -x
```

Все параметры, которые принимают списки, разделенные запятыми, соблюдают соглашение, согласно которому указание пустого списка очищает его значение. Это может быть полезно для очистки wgetrc-настройки. Например, если ваш wgetrc устанавливается exclude\_directories на/cgi-bin, в следующем примере он сначала сбрасывается, а затем устанавливается для исключения/~никто и/~кто-то. Вы также можете очистить списки в wgetrc (см. [Синтаксис Wgetrc](#) ).

```
wget -X "" -X /~никто,/~кто-то
```

Большинство опций, которые не принимают аргументы, являются *логическими* опциями, названными так потому, что их состояние можно зафиксировать с помощью переменной типа «да» или «нет» («логическая»). Например, '--follow-ftp' сообщает Wget перейти по FTP-ссылкам из файлов HTML и, с другой стороны, '--no-glob' сообщает ему, что не следует выполнять подстановку файлов по URL-адресам FTP. Логический параметр может быть *утвердительным* или *отрицательным* (начиная с '--нет'). Все такие варианты имеют несколько общих свойств.

Если не указано иное, предполагается, что поведение по умолчанию противоположно тому, что обеспечивает опция. Например, документально подтвержденное существование '--follow-ftp' предполагает, что по умолчанию *не* следует переходить по FTP-ссылкам со страниц HTML.

Утвердительные варианты можно отменить, добавив в начале '--нет-' к названию опции; отрицательные варианты можно отменить, опустив '--нет-' префикс. Это может показаться излишним: если по умолчанию для утвердительного варианта установлено не делать что-либо, то зачем предоставлять возможность явного отключения этого действия? Но файл запуска может фактически изменить значение по умолчанию. Например, используя follow\_ftp = on в .wgetrc заставляет Wget *переходить по* FTP-ссылкам по умолчанию и использовать '--no-follow-ftp' – единственный способ восстановить заводские настройки по умолчанию из командной строки.

---

Далее: [Параметры журнала и входного файла](#) , Предыдущий: [Синтаксис параметров](#) , Вверх: [Вызов](#) [ [Содержание](#) ][ [Индекс](#) ]

## 2.3 Основные параметры запуска

'-В'

'--версия'

Отобразите версию Wget.

'-час'

'--помощь'

Распечатайте справочное сообщение с описанием всех параметров командной строки Wget.

'-б'

'--фон'



Перейти в фоновый режим сразу после запуска. Если выходной файл не указан через '-O', вывод перенаправляется на wget-журнал.

'-е команда'

'--выполнить команду'

Выполните команду, как если бы она была частью wgetrc (см. [Файл запуска](#)). Вызванная таким образом команда будет выполнена *после* команд в wgetrc, таким образом, имея приоритет над ними. Если вам нужно указать более одной команды wgetrc, используйте несколько экземпляров '-Это'.

---

Следующая: [Параметры загрузки](#), Предыдущая: [Основные параметры запуска](#), Вверх: [Вызов](#)  
[ [Содержание](#) ] [ [Индекс](#) ]

## 2.4 Параметры журнала и входного файла

'-о файл журнала'

'--output-file= файл журнала'

Записывать все сообщения в *файл журнала*. Сообщения обычно передаются в стандартную ошибку.

'-файл журнала'

'--append-output= файл журнала'

Добавить в *файл журнала*. Это то же самое, что '-O', только он добавляется в *файл журнала*, а не перезаписывает старый файл журнала. Если *файл журнала* не существует, создается новый файл.

'-д'

'--отлаживать'

Включите вывод отладки, то есть различную информацию, важную для разработчиков Wget, если он не работает должным образом. Возможно, ваш системный администратор решил скомпилировать Wget без поддержки отладки, и в этом случае '-д' не будет работать. Обратите внимание, что компиляция с поддержкой отладки всегда безопасна – Wget, скомпилированный с поддержкой отладки, *не* будет печатать какую-либо отладочную информацию, если она не запрошена с помощью '-д'. См. [раздел «Сообщения об ошибках»](#) для получения дополнительной информации о том, как использовать '-д' для отправки отчетов об ошибках.

'-q'

'--тихий'

Отключите вывод Wget.

'-в'

'--подробный'

Включите подробный вывод со всеми доступными данными. Вывод по умолчанию является подробным.

'-nv'

'--не многословный'

Отключите многословие, но не будьте полностью тихим (используйте '-q' для этого), что означает, что сообщения об ошибках и основная информация по-прежнему печатаются.

'--report-speed= тип'

Выходная полоса пропускания как *тип* . Единственное допустимое значение – 'биты'.

'-я файл'

'--input-file= файл'

Чтение URL -адресов из локального или внешнего *файла* . Если '-' указан как *файл* , URL -адреса считываются из стандартного ввода. (Использовать './-' для чтения из файла с буквальным именем '-'.)

Если используется эта функция, в командной строке не требуется указывать URL -адреса. Если URL -адреса есть как в командной строке, так и во входном файле, они будут получены в первую очередь. Если '--force-html' не указан, то *файл* должен состоять из серии URL-адресов, по одному в строке.

Однако, если вы укажете '--force-html', документ будет считаться 'HTML'. В этом случае у вас могут возникнуть проблемы с относительными ссылками, которые можно решить либо добавив в документы, либо указав '<base href="url">--base= URL' в командной строке.

Если *файл* является внешним, документ будет автоматически рассматриваться как 'HTML' если Content-Type совпадает 'текст/html'. Более того, местоположение *файла* будет неявно использоваться в качестве базовой href, если ничего не указано.

'--input-metalink= файл'

Загружает файлы, содержащиеся в локальном *файле* Metalink . Поддерживаются версии Metalink 3 и 4.

'--keep-badhash'

Сохраняет скачанные файлы Металинка с плохим хешем. Он добавляет .badhash к имени файлов Metalink, у которых не совпадает контрольная сумма, за исключением случаев, когда существующие файлы не перезаписываются.

**'--metalink-over-http'**

Выдает запрос HTTP HEAD вместо GET и извлекает метаданные Metalink из заголовков ответов. Затем он переключается на загрузку Metalink. Если действительные метаданные Metalink не найдены, происходит возврат к обычной загрузке HTTP. Включает 'Тип контента: приложение/metalink4+xml' загрузка/обработка файлов.

**'--metalink-index= номер'**

Установите Металинк 'приложение/metalink4+xml' порядковый номер метаурла. От 1 до общего количества доступных «application/metalink4+xml». Укажите 0 или 'инфа', чтобы выбрать первый хороший. Метаурлы, например, из '--metalink-over-http', возможно, был отсортирован по значению приоритетного ключа; имейте это в виду, чтобы выбрать правильный НОМЕР.

**'--Предпочтительное Местоположение'**

Установите предпочтительное местоположение для ресурсов Metalink. Это имеет эффект, если доступно несколько ресурсов с одинаковым приоритетом.

**'--xattr'**

Включите использование расширенных атрибутов файловой системы, чтобы сохранить исходный URL-адрес и значение HTTP-заголовка Referer, если оно используется.

Имейте в виду, что URL-адрес может содержать личную информацию, например токены доступа или учетные данные.

**'-Ф'****'--force-html'**

Когда ввод считывается из файла, принудительно обрабатывайте его как файл HTML . Это позволяет вам получать относительные ссылки из существующих файлов HTML на локальном диске, добавляя их в HTML или используя '<base href="url">--база'параметр командной строки.

**'-В URL-адрес'****'--base= URL- адрес'**

Разрешает относительные ссылки, используя URL-адрес в качестве точки ссылки при чтении ссылок из файла HTML, указанного через '-я'/'--input-file'опция (вместе с '--force-html', или когда входной файл был удален удаленно с сервера, описывающего его как HTML ). Это эквивалентно наличию тега BASEво входном файле HTML с URL-адресом в качестве значения атрибута href.

Например, если вы укажете 'http://foo/bar/a.html' для URL-адреса , а Wget читает '../baz/b.html' из входного файла, оно будет разрешено как 'http://foo/baz/b.html'.

**'--config= ФАЙЛ'**

Укажите местоположение файла запуска, который вы хотите использовать вместо файлов по умолчанию. Используйте `--no-config`, чтобы отключить чтение файлов конфигурации. Если указаны оба параметра `--config` и `--no-config`, параметр `--no-config` игнорируется.

### `--rejected-log= файл журнала`

Регистрирует все отклонения URL-адресов в *файле журнала* в виде значений, разделенных запятыми. Значения включают причину отклонения, URL-адрес и родительский URL-адрес, в котором он был найден.

---

Далее: [Параметры каталога](#) , Предыдущее: [Параметры ведения журнала и входного файла](#) ,  
Вверх: [Вызов](#) [ [Содержание](#) ][ [Индекс](#) ]

## 2.5 Параметры загрузки

### `--bind-address= АДРЕС`

При создании клиентских TCP/IP-подключений привязывайтесь к *ADDRESS* на локальном компьютере. *ADDRESS* может быть указан как имя хоста или IP-адрес. Эта опция может быть полезна, если ваша машина привязана к нескольким IP-адресам.

### `--bind-dns-address= АДРЕС`

[только для `libcares`] Этот адрес переопределяет маршрут для DNS-запросов. Если вам когда-нибудь понадобится обойти стандартные настройки из `/etc/resolv.conf`, эта опция вместе с `--dns-серверы` твой друг. *ADDRESS* должен быть указан как адрес IPv4 или IPv6. Чтобы эта опция была доступна, Wget должен быть собран с использованием `libcares`.

### `--dns-servers= АДРЕСА`

[только для `libcares`] Указанные адреса переопределяют стандартные адреса серверов имен, например, настроенные в `/etc/resolv.conf`. *АДРЕСЫ* могут быть указаны как адреса IPv4 или IPv6, разделенные запятыми. Чтобы эта опция была доступна, Wget должен быть собран с использованием `libcares`.

### `-t номер`

### `--tries= число`

Установите количество попыток *равным* . Укажите 0 или 'инфа' для бесконечных повторов. По умолчанию повтор попытки составляет 20 раз, за исключением фатальных ошибок, таких как «соединение отклонено» или «не найдено» (404), которые не повторяются.

### `-O файл`

### `--output-document= файл`

Документы не будут записаны в соответствующие файлы, но все будут объединены и записаны в *файл*. Если '-' используется как *файл*, документы будут распечатаны в стандартный вывод, отключив преобразование ссылок. (Использовать './-' для печати в файл с буквальным именем '-'.)

Использование '-O' не означает просто «использовать имя *файла* вместо того, что указано в URL-адресе»; скорее, это аналогично перенаправлению оболочки: 'wget -O файл http://foo' предназначен для работы как 'wget -O - http://foo > файл'; файл будет немедленно обрезан, и *весь* загруженный контент будет записан туда.

По этой причине, '-H' (для проверки временных меток) не поддерживается в сочетании с '-O': поскольку *файл* всегда создается заново, он всегда будет иметь новую временную метку. При использовании этой комбинации будет выдано предупреждение.

Аналогично, используя '-r' или '-p' с '-O' может работать не так, как вы ожидаете: Wget не будет просто загружать первый файл в *файл*, а затем загружать остальные под их обычные имена: *весь* загруженный контент будет помещен в *файл*. Это было отключено в версии 1.11, но было восстановлено (с предупреждением) в 1.11.2, поскольку в некоторых случаях такое поведение действительно может иметь какое-то применение.

Сочетание с '-NC' принимается только в том случае, если данный выходной файл не существует.

Обратите внимание, что комбинация с '-к' разрешено только при загрузке одного документа, поскольку в этом случае он просто преобразует все относительные URI во внешние; '-к' не имеет смысла для нескольких URI, когда все они загружаются в один файл; '-к' можно использовать только в том случае, если выходные данные представляют собой обычный файл.

## '-NC'

### '--не затирать'

Если файл загружается более одного раза в один и тот же каталог, поведение Wget зависит от нескольких параметров, включая '-NC'. В некоторых случаях локальный файл будет *уничтожен* или перезаписан при повторной загрузке. В остальных случаях он сохранится.

При запуске Wget без '-H', '-NC', '-r', или '-p', загрузка того же файла в тот же каталог приведет к сохранению исходной копии файла, а второй копии будет присвоено имя '*файл*.1'. Если этот файл будет загружен еще раз, третья копия будет называться '*файл*.2', и так далее. (Это также поведение с '-nd', даже если '-r' или '-p' действуют.) Когда '-NC' указан, это поведение подавляется, и Wget откажется загружать новые копии '*файл*'. Таким образом, «no-clobber» в этом режиме на самом деле является неправильным употреблением – предотвращается не

затирание (поскольку числовые суффиксы уже предотвращают затирание), а скорее предотвращается сохранение нескольких версий.

При запуске Wget с помощью '-р' или '-п', но без '-Н', '-nd', или '-NC', повторная загрузка файла приведет к тому, что новая копия просто перезапишет старую. Добавление '-NC' предотвратит такое поведение, вместо этого сохраняя исходную версию и игнорируя все новые копии на сервере.

При запуске Wget с помощью '-Н', с или без '-р' или '-п', решение о том, загружать или нет более новую копию файла, зависит от локальной и удаленной отметки времени и размера файла (см. « [Отметка времени](#) »). '-NC' нельзя указывать одновременно с '-Н'.

Сочетание с '-O'/'--output-document' принимается только в том случае, если данный выходной файл не существует.

Обратите внимание, что когда '-NC' указан, файлы с суффиксами '.html' или '.htm' будут загружены с локального диска и проанализированы, как если бы они были получены из Интернета.

### '--backups= *резервные копии*'

Прежде чем (пере)записывать файл, создайте резервную копию существующего файла, добавив '.1' суффикс ('\_1' на VMS) к имени файла. Такие файлы резервных копий ротируются в '.2', '.3', и так далее, вплоть до *резервных копий* (и за ее пределами).

### '--no-netrc'

Не пытайтесь получить учетные данные от.netrcфайл. По умолчанию .netrcВ файле выполняется поиск учетных данных в случае, если они не были переданы в командной строке и требуется аутентификация.

### '-с'

### '--продолжать'

Продолжайте получать частично загруженный файл. Это полезно, если вы хотите завершить загрузку, начатую предыдущим экземпляром Wget или другой программой. Например:

```
wget -c ftp://sunsite.doc.ic.ac.uk/lS-lR.Z
```

Если существует файл с именемлS-lR.Zв текущем каталоге Wget предположит, что это первая часть удаленного файла, и попросит сервер продолжить извлечение со смещением, равным длине локального файла.

Обратите внимание, что вам не нужно указывать эту опцию, если вы просто хотите, чтобы текущий вызов Wget повторил загрузку файла, если соединение будет потеряно в середине процесса. Это поведение по умолчанию. '-с' влияет только на возобновление загрузок, начатых *до* этого вызова Wget и чьи локальные файлы все еще находятся.

Без '-с', предыдущий пример просто загрузит удаленный файл vls-lR.Z.1, оставляя усеченное vls-lR.Z файл один.

Если вы используете '-с' для непустого файла и сервер не поддерживает продолжение загрузки, Wget перезапустит загрузку с нуля и полностью перезапишет существующий файл.

Начиная с Wget 1.7, если вы используете '-с' для файла такого же размера, что и файл на сервере, Wget откажется загружать файл и распечатает поясняющее сообщение. То же самое происходит, когда файл на сервере меньше, чем локально (вероятно, потому, что он был изменен на сервере с момента вашей последней попытки загрузки) – поскольку «продолжение» не имеет смысла, загрузка не происходит.

С другой стороны медали, используя '-с', любой файл, размер которого на сервере больше, чем локальный, будет считаться неполной загрузкой, и только  $(\text{length}(\text{remote}) - \text{length}(\text{local}))$  байты будут загружены и прикреплены к концу локального файла. Такое поведение может быть желательным в определенных случаях – например, вы можете использовать 'wget -с', чтобы загрузить только новую часть, добавленную к файлу сбора данных или файлу журнала.

Однако если файл на сервере больше по размеру, потому что он был *изменен*, а не просто *добавлен*, вы получите искаженный файл. Wget не имеет возможности проверить, действительно ли локальный файл является допустимым префиксом удаленного файла. Вы должны быть особенно осторожны с этим при использовании '-с' в сочетании с '-р', поскольку каждый файл будет рассматриваться как кандидат на «неполную загрузку».

Еще один случай, когда вы получите искаженный файл, если попытаетесь использовать '-с' – это если у вас неумелый HTTP-прокси, который вставляет в локальный файл строку «передача прервана». В будущем для этого случая может быть добавлена опция «отката».

Обратите внимание, что '-с' работает только с FTP-серверами и HTTP-серверами, поддерживающими Range-заголовки.

### '--start-pos= СМЕЩЕНИЕ'

Начните загрузку с нулевой позиции *OFFSET*. Смещение может быть выражено в байтах, килобайтах с суффиксом «к» или мегабайтах с суффиксом «м» и т. д.

'--start-pos' имеет более высокий приоритет над '--продолжать'. Когда '--start-pos' и '--продолжать' оба указаны, wget выдаст предупреждение, а затем продолжит

работу, как если бы '--продолжать' отсутствовал.

Требуется поддержка сервера для продолжения загрузки, в противном случае '--start-pos' не могу помочь. Видеть '-с' для подробностей.

### '--progress= тип'

Выберите тип индикатора прогресса, который вы хотите использовать. Юридическими индикаторами являются «точка» и «черта».

По умолчанию используется индикатор «бар». Он рисует графический индикатор выполнения ASCII (также известный как дисплей «термометра»), указывающий статус извлечения. Если вывод не является телетайпом, по умолчанию будет использоваться точечная полоса.

Использовать '--progress=точка', чтобы переключиться на «точечный» дисплей. Он отслеживает извлечение, печатая точки на экране, каждая точка представляет собой фиксированный объем загруженных данных.

Тип прогресса также может принимать один или несколько параметров. Параметры различаются в зависимости от выбранного *типа*. Параметры *типа* передаются путем добавления их к типу, разделенному двоеточием (:), например: '--progress= тип : параметр1 : параметр2'.

При использовании точечного поиска вы можете установить стиль, указав тип как 'точка: стиль'. Разные стили придают разное значение одной точке. Поскольку defaultкаждая точка соответствует 1 КБ, в кластере содержится десять точек, а в строке – 50 точек. Стиль *binary* имеет более «компьютерную» ориентацию – 8 тыс. точек, кластеры по 16 точек и 48 точек на строку (что составляет 384 тыс. строк). Этот мегастиль подходит для загрузки больших файлов – каждая точка соответствует извлеченным 64 КБ, в кластере восемь точек и 48 точек в каждой строке (таким образом, каждая строка содержит 3 М). Если мегаэтого недостаточно, вы можете использовать стиль *giga* : каждая точка представляет собой 1 миллион извлеченных данных, в кластере восемь точек и 32 точки в каждой строке (таким образом, каждая строка содержит 32 миллиона).

С '--progress=бар', на данный момент есть два возможных параметра: *Force* и *noscroll*.

Если выходные данные не являются TTY, индикатор выполнения всегда возвращается к «точке», даже если '--progress=бар' был передан Wget во время вызова. Это поведение можно переопределить и принудительно вывести вывод «бар», используя параметр «force» как '--progress=бар:сила'.

По умолчанию 'бар' стиль прогресс-бара прокручивает имя файла слева направо для загружаемого файла, если имя файла превышает максимальную длину, отведенную для его отображения. В некоторых случаях, например, с '--progress=бар:сила', возможно, вам не захочется, чтобы имя файла прокручивалось в индикаторе



выполнения. Передав параметр `noscroll`, Wget можно заставить отображать как можно большую часть имени файла без его прокрутки.

Обратите внимание, что вы можете установить стиль по умолчанию, используя `progress` команду в `wgetrc`. Этот параметр можно переопределить из командной строки. Например, чтобы принудительно вывести панель без прокрутки, используйте `'--progress=bar:force:noscroll'`.

### **'--show-прогресс'**

Заставьте wget отображать индикатор выполнения с любой степенью детализации.

По умолчанию wget отображает индикатор выполнения только в подробном режиме. Однако можно захотеть, чтобы wget отображал индикатор выполнения на экране в сочетании с любыми другими режимами детализации, такими как `'--не многословный'` или `'--тихий'`. Это свойство часто требуется при вызове wget для загрузки нескольких маленьких/больших файлов. В таком случае можно просто вызвать wget с этим параметром, чтобы получить более чистый вывод на экране.

Эта опция также заставит печатать индикатор выполнения `hastderg` при использовании вместе с `'--выходной файл'` вариант.

### **'-Н'**

### **'--метка времени'**

Включите отметки времени. Подробности см . в разделе [Временные метки](#) .

### **'--no-if-modified-since'**

Не отправлять заголовок If-Modified-Since в `'-Н'` режим. Вместо этого отправьте предварительный запрос HEAD. Это имеет эффект только в `'-Н'` режим.

### **'--no-use-server-timestamps'**

Не устанавливайте временную метку локального файла по временной метке на сервере.

По умолчанию, когда файл загружается, его временные метки устанавливаются в соответствии с метками удаленного файла. Это позволяет использовать `'--метка времени'` при последующих вызовах wget. Однако иногда полезно использовать временную метку локального файла на момент его фактической загрузки; для этой цели `'--no-use-server-timestamps'` опция была предоставлена.

### **'-С'**

### **'--server-response'**

Распечатайте заголовки, отправленные HTTP- серверами, и ответы, отправленные FTP -серверами.

### **'--паук'**

При вызове с этой опцией Wget будет вести себя как веб- *паук* , что означает, что он не будет загружать страницы, а просто проверит их наличие. Например, вы можете использовать Wget для проверки своих закладок:

```
wget --spider --force-html -i bookmarks.html
```

Эта функция требует гораздо большей работы, чтобы Wget приблизился к функциональности настоящих веб-пауков.

### '-Т секунд'

#### '--timeout= *секунды*'

Установите тайм-аут сети на *секунды* . Это эквивалентно указанию '--dns-таймаут', '--connect-таймаут', и '--время чтения', Все одновременно.

При взаимодействии с сетью Wget может проверить тайм-аут и прервать операцию, если она занимает слишком много времени. Это предотвращает такие аномалии, как зависание операций чтения и бесконечные соединения. Единственный тайм-аут, включенный по умолчанию, — это 900-секундный тайм-аут чтения. Установка тайм-аута на 0 полностью отключает его. Если вы не знаете, что делаете, лучше не менять настройки таймаута по умолчанию.

Все параметры, связанные с тайм-аутом, принимают десятичные значения, а также значения долей секунды. Например, '0,1' секунды — это законный (хотя и неразумный) выбор тайм-аута. Тайм-ауты в доли секунды полезны для проверки времени ответа сервера или для тестирования задержки в сети.

#### '--dns-timeout= *секунды*'

Установите тайм-аут поиска DNS в *секундах* . Поиск DNS, который не завершится в течение указанного времени, завершится неудачей. По умолчанию при поиске DNS нет тайм-аута, кроме того, который реализуется системными библиотеками.

#### '--connect-timeout= *секунды*'

Установите тайм-аут подключения в *секундах* . TCP-соединения, для установления которых требуется больше времени, будут прерваны. По умолчанию тайм-аут соединения отсутствует, кроме того, который реализован системными библиотеками.

#### '--read-timeout= *секунды*'

Установите тайм-аут чтения (и записи) в *секунды* . «Время» этого тайм-аута относится к *времени простоя* : если в какой-либо момент загрузки данные не получены в течение более указанного количества секунд, чтение не удастся и загрузка возобновляется. Эта опция не влияет напрямую на продолжительность всей загрузки.

Конечно, удаленный сервер может решить разорвать соединение раньше, чем этого требует этот вариант. Тайм-аут чтения по умолчанию составляет 900 секунд.

### '--limit-rate= *сумма*'

Ограничьте скорость загрузки *количеством* байт в секунду. Сумма может быть выражена в байтах, килобайтах с помощью 'к' или мегабайты с 'м' суффикс. Например, '--limit-rate=20 тыс.' ограничит скорость извлечения до 20 КБ/с. Это полезно, когда по какой-либо причине вы не хотите, чтобы Wget использовал всю доступную пропускную способность.

Эта опция позволяет использовать десятичные числа, обычно в сочетании с суффиксами степени; например, '--limit-rate=2,5 тыс.' является юридической ценностью.

Обратите внимание, что Wget реализует ограничение, отключая соответствующее количество времени после чтения сети, которое заняло меньше времени, чем указано в скорости. В конечном итоге эта стратегия приводит к замедлению передачи TCP примерно до указанной скорости. Однако для достижения этого баланса может потребоваться некоторое время, поэтому не удивляйтесь, если ограничение скорости не работает с очень маленькими файлами.

### '-w *секунд*'

### '--wait= *секунд*'

Подождите указанное количество секунд между извлечениями. Рекомендуется использовать эту опцию, поскольку она снижает нагрузку на сервер за счет уменьшения частоты запросов. Вместо секунд время можно указать в минутах с помощью суффикса m, в часах с помощью h суффикса или в днях с использованием dсуффикса.

Указание большого значения для этой опции полезно, если сеть или целевой хост не работают, так что Wget может ждать достаточно долго, чтобы разумно ожидать, что сетевая ошибка будет исправлена перед повторной попыткой. На интервал ожидания, указанный этой функцией, влияет --random-wait, см.

### '--waitretry= *секунды*'

Если вы не хотите, чтобы Wget ждал между *каждым* получением, а только между повторными попытками неудачных загрузок, вы можете использовать эту опцию. Wget будет использовать *линейную отсрочку*, ожидая 1 секунду после первого сбоя в данном файле, а затем ожидая 2 секунды после второго сбоя в этом файле, вплоть до указанного вами максимального количества *секунд*.

По умолчанию Wget принимает значение 10 секунд.

### '--random-wait'

Некоторые веб-сайты могут выполнять анализ журналов для идентификации поисковых программ, таких как Wget, путем поиска статистически значимых сходств во времени между запросами. Эта опция приводит к тому, что время между запросами варьируется от 0,5 до 1,5 \* секунд *ожидания*, где *ожидание* было указано с помощью '--ждать', чтобы скрыть присутствие Wget от такого анализа.

В статье 2001 года в публикации, посвященной разработке на популярной потребительской платформе, был предоставлен код для выполнения такого анализа на лету. Его автор предложил блокировать на уровне адресов класса C, чтобы гарантировать блокировку программ автоматического поиска, несмотря на изменение адресов, предоставленных DHCP.

'--random-wait' Опция ' была вдохновлена этой опрометчивой рекомендацией заблокировать доступ к веб-сайту многим посторонним пользователям из-за действий одного из них.

### '--no-прокси'

\*\_проху Не используйте прокси, даже если определена соответствующая переменная среды.

См. [«Прокси»](#) для получения дополнительной информации об использовании прокси с Wget.

### '-Q квота'

#### '--quota= квота'

Укажите квоту загрузки для автоматического получения. Значение может быть указано в байтах (по умолчанию), килобайтах (с помощью 'к' суффикс) или мегабайты (с 'м' суффикс).

Обратите внимание, что квота никогда не повлияет на загрузку отдельного файла. Итак, если вы укажете 'wget -Q10k https://example.com/ls-lR.gz', Все ls-lR.gz будет загружен. То же самое происходит, даже если в командной строке указано несколько URL-адресов. Квота проверяется только в конце каждого загруженного файла, поэтому это никогда не приведет к частичной загрузке файла. Таким образом, вы можете смело набирать 'wget -Q2m -i сайты' – загрузка будет прервана после полной загрузки файла, исчерпавшего квоту.

Установка квоты на 0 или на 'инфа' ограничивает квоту загрузки.

### '--no-dns-кэш'

Отключите кэширование DNS-запросов. Обычно Wget запоминает IP-адреса, которые он искал в DNS, поэтому ему не нужно повторно обращаться к DNS-серверу для одного и того же (обычно небольшого) набора хостов, с которых он получает данные. Этот кеш существует только в памяти; новый запуск Wget снова свяжется с DNS.

Однако сообщалось, что в некоторых ситуациях нежелательно кэшировать имена хостов даже на время работы кратковременного приложения, такого как Wget. С

помощью этой опции Wget выполняет новый поиск DNS (точнее, новый вызов `gethostbyname` или `getaddrinfo`) каждый раз, когда устанавливает новое соединение. Обратите внимание, что этот параметр *не* повлияет на кэширование, которое может выполняться разрешающей библиотекой или внешним слоем кэширования, например NSCD.

Если вы не совсем понимаете, что делает эта опция, возможно, она вам не понадобится.

### '--restrict-file-names= *режимы*'

Измените символы, найденные в удаленных URL-адресах, которые должны быть экранированы во время создания локальных имен файлов. Символы, использование которых *ограничено* этой опцией, экранируются, т.е. заменяются на '%XX', где 'XX' – шестнадцатеричное число, соответствующее ограниченному символу. Эту опцию также можно использовать для принудительного перевода всех букв в нижний или верхний регистр.

По умолчанию Wget экранирует недопустимые или безопасные символы в именах файлов в вашей операционной системе, а также управляющие символы, которые обычно не печатаются. Эта опция полезна для изменения этих значений по умолчанию, возможно, потому, что вы загружаете в неродной раздел, или потому, что вы хотите отключить экранирование управляющих символов, или вы хотите дополнительно ограничить символы только теми, которые находятся в диапазоне значений ASCII .

Режимы представляют собой набор текстовых значений, разделенных запятыми . Приемлемые значения: 'юникс', 'окна', 'нет контроля', 'ascii', 'строчные буквы', и 'верхний регистр'. Ценности 'юникс' и 'окна' являются взаимоисключающими (одно переопределяет другое), как и 'строчные буквы' и 'верхний регистр'. Последние являются особыми случаями, поскольку они не меняют набор экранируемых символов, а скорее заставляют пути к локальным файлам преобразовываться либо в нижний, либо в верхний регистр.

Если указано «unix», Wget экранирует символ '/' и управляющие символы в диапазонах 0–31 и 128–159. Это значение по умолчанию в Unix-подобных операционных системах.

Когда указано «windows», Wget экранирует символы '\', '|', '/', ':', '?', '"', '\*', '<', '>', а также управляющие символы в диапазонах 0–31 и 128–159. В дополнение к этому, Wget в режиме Windows использует '+' вместо ':' для разделения хоста и порта в именах локальных файлов и использует '@' вместо '?', чтобы отделить запрашиваемую часть имени файла от остальной части. Поэтому URL-адрес, который будет сохранен как 'www.xemacs.org:4300/search.pl?input=blah' в режиме Unix будет сохранен как 'www.xemacs.org+4300/search.pl@input=blah' в режиме Windows. Этот режим установлен по умолчанию в Windows.

Если вы укажете 'нет контроля', то экранирование управляющих символов также отключается. Эта опция может иметь смысл при загрузке URL-адресов, имена которых

содержат символы UTF-8, в системе, которая может сохранять и отображать имена файлов в UTF-8 (некоторые возможные значения байтов, используемые в последовательностях байтов UTF-8, попадают в диапазон значений, обозначенных от Wget как «элементы управления»).

'asciiРежим' используется для указания того, что любые байты, значения которых находятся за пределами диапазона символов ASCII (то есть больше 127), должны быть экранированы. Это может быть полезно при сохранении имен файлов, кодировка которых не соответствует той, которая используется локально.

'-4'

'--inet4-только'

'-6'

'--inet6-only'

Принудительное подключение к адресам IPv4 или IPv6. С '--inet4-только' или '-4', Wget будет подключаться только к хостам IPv4, игнорируя записи AAAA в DNS и отказываясь подключаться к адресам IPv6, указанным в URL-адресах. И наоборот, с '--inet6-only' или '-6', Wget будет подключаться только к хостам IPv6 и игнорировать записи A и адреса IPv4.

Ни один из вариантов обычно не требуется. По умолчанию Wget с поддержкой IPv6 будет использовать семейство адресов, указанное в DNS-записи хоста. Если DNS отвечает адресами как IPv4, так и IPv6, Wget будет проверять их последовательно, пока не найдет тот, к которому можно подключиться. (Также см. --prefer-familyвариант, описанный ниже.)

Эти параметры можно использовать для преднамеренного принудительного использования семейств адресов IPv4 или IPv6 в системах двойного семейства, обычно для облегчения отладки или устранения неисправности конфигурации сети. Только один из '--inet6-only' и '--inet4-только' можно указать одновременно. Ни один из вариантов недоступен в Wget, скомпилированном без поддержки IPv6.

'--prefer-family=none/IPv4/IPv6'

Если вам предоставлен выбор из нескольких адресов, сначала подключитесь к адресам с указанным семейством адресов. Порядок адресов, возвращаемый DNS, по умолчанию используется без изменений.

Это позволяет избежать ложных ошибок и попыток подключения при доступе к хостам, которые разрешают адреса IPv6 и IPv4 из сетей IPv4. Например, 'www.kame.net' решает '2001:200:0:8002:203:47ff:fea5:3085' и '203.178.141.194'. Если предпочтительным семейством является IPv4, сначала используется адрес IPv4; если предпочтительное семейство – IPv6сначала используется адрес IPv6; если указано значение none, порядок адресов, возвращаемый DNS, используется без изменений.

В отличие '-4' и '-6', эта опция не запрещает доступ к какому-либо семейству адресов, а только меняет *порядок* доступа к адресам. Также обратите внимание, что изменение порядка, выполняемое этой опцией, *стабильно* – оно не влияет на порядок адресов одного и того же семейства. То есть относительный порядок всех адресов IPv4 и всех адресов IPv6 остается неизменным во всех случаях.

### '--retry-connrefused'

Считайте «соединение отклонено» временной ошибкой и повторите попытку. Обычно Wget отказывается от URL-адреса, когда не может подключиться к сайту, поскольку отсутствие подключения воспринимается как признак того, что сервер вообще не работает и повторные попытки не помогут. Этот вариант предназначен для зеркалирования ненадежных сайтов, серверы которых имеют тенденцию исчезать на короткие периоды времени.

### '--user= *пользователь*'

### '--password= *пароль*'

Укажите имя пользователя *и* пароль *для* получения файлов по FTP и HTTP . Эти параметры можно переопределить с помощью '--ftp-пользователь' и '--ftp-пароль' параметры для FTP -соединений и '--http-пользователь' и '--http-пароль' параметры для HTTP- соединений.

### '--ask-пароль'

Запрашивать пароль для каждого установленного соединения. Невозможно указать, когда '--пароль' используется, потому что они взаимоисключающие.

### '--use-askpass= *команда*'

Запросить пользователя и пароль с помощью указанной команды. Если команда не указана, используется команда из переменной среды WGET\_ASKPASS. Если WGET\_ASKPASS не установлен, используется команда из переменной среды SSH\_ASKPASS.

Вы можете установить команду по умолчанию для use-askpass в wgetrc. Этот параметр можно переопределить из командной строки.

### '--no-ири'

Отключите поддержку интернационализованного URI (IRI). Использовать '--является' включить это. Поддержка IRI активирована по умолчанию.

Вы можете установить состояние поддержки IRI по умолчанию, используя иги команду в wgetrc. Этот параметр можно переопределить из командной строки.

### '--local-encoding= *кодировка*'

Заставьте Wget использовать *кодировку* в качестве системной кодировки по умолчанию. Это влияет на то, как Wget преобразует URL-адреса, указанные в

качестве аргументов, из локали в UTF-8 для поддержки IRI.

Wget использует функцию `nL_langinfo()`, а затем `CHARSET` переменную среды, чтобы получить локаль. Если это не удастся, используется ASCII .

Вы можете установить локальную кодировку по умолчанию, используя `local_encoding` команду в `wgetrc`. Этот параметр можно переопределить из командной строки.

#### '--remote-encoding= *кодировка*'

Заставьте Wget использовать *кодировку* в качестве кодировки удаленного сервера по умолчанию. Это влияет на то, как Wget преобразует URI, найденные в файлах, из удаленной кодировки в UTF-8 во время рекурсивной выборки. Эта опция полезна только для поддержки IRI и интерпретации символов, отличных от ASCII .

Для HTTP удаленное кодирование можно найти в Content-Type заголовке HTTP и Content-Type `http-equivметатеge HTML`.

Вы можете установить кодировку по умолчанию, используя `remoteencoding` команду в `wgetrc`. Этот параметр можно переопределить из командной строки.

#### '--отсоединить'

Заставьте Wget отсоединить файл вместо удаления существующего файла. Эта опция полезна для загрузки в каталог с жесткими ссылками.

---

Далее: [Параметры HTTP](#) , Предыдущий: [Параметры загрузки](#) , Вверх: [Вызов](#) [ [Содержание](#) ][ [Индекс](#) ]

## 2.6 Параметры каталога

#### '-nd'

#### '--no-каталоги'

Не создавайте иерархию каталогов при рекурсивном извлечении. Если эта опция включена, все файлы будут сохраняться в текущем каталоге без затирания (если имя отображается более одного раза, имена файлов получают расширения `'.n'`).

#### '-Икс'

#### '--force-каталоги'

Противоположно `'-nd'`— создать иерархию каталогов, даже если иначе она не была бы создана. Например: `'wget -x http://fly.srk.fer.hr/robots.txt'` сохранит загруженный файл в `fly.srk.fer.hr/robots.txt`.

#### '-nH'

#### '--no-host-каталоги'



Отключите создание каталогов с префиксом хоста. По умолчанию вызов Wget с помощью '-p http://fly.srk.feg.hr/' создаст структуру каталогов, начинающуюся cfly.srk.feg.hr/. Эта опция отключает такое поведение.

### '--протокол-каталоги'

Используйте имя протокола как компонент каталога имен локальных файлов. Например, с помощью этой опции: 'wget -r http:// *хост*' сохраню в 'http/ *хост* /...' а не просто '*хозяин* /...'.

### '--cut-dirs= *число*'

Игнорировать компоненты справочника *номеров* . Это полезно для получения детального контроля над каталогом, в котором будет сохраняться рекурсивный поиск.

Возьмем, к примеру, каталог по адресу 'ftp://ftp.xemacs.org/pub/xemacs/'. Если вы получите его с помощью '-p', он будет сохранен локально под ftp.xemacs.org/pub/xemacs/. В то время '-nH' опция может удалить ftp.xemacs.org/ часть, ты все еще застрял с паб/хемас. Это где '--cut-dirs' пригодится; это заставляет Wget не «видеть» *количество* компонентов удаленного каталога. Вот несколько примеров того, как '--cut-dirs' опция работает.

```
Нет опций -> ftp.xemacs.org/pub/xemacs/  
-nH -> паб/хемас/  
-nH --cut-dirs=1 -> хемас/  
-nH --cut-dirs=2 -> .  
  
--cut-dirs=1 -> ftp.xemacs.org/xemacs/  
...
```

Если вы просто хотите избавиться от структуры каталогов, этот вариант аналогичен комбинации '-nd' и '-P'. Однако, в отличие от '-nd', '--cut-dirs' не проигрывает с подкаталогами - например, с '-nH --cut-dirs=1', абета/подкаталог будет помещен в хемас/бета, как и следовало ожидать.

### '-P *префикс*'

### '--directory-prefix= *префикс*'

Установите префикс каталога на *prefix* . *Префикс каталога* – это каталог, в котором будут сохранены все остальные файлы и подкаталоги, т. е. верхняя часть дерева поиска. По умолчанию '.' (текущий каталог).

Далее: [Параметры HTTPS \(SSL/TLS\)](#) , Предыдущий: [Параметры каталога](#) , Вверх: [Вызов](#) [ [Содержание](#) ][ [Индекс](#) ]

## 2.7 Параметры HTTP

### '--default-page= *имя*'

Используйте *имя* в качестве имени файла по умолчанию, когда оно неизвестно (например, для URL-адресов, оканчивающихся косой чертой), вместо `index.html`.

### '-И'

### '--adjust-расширение'

Если файл типа 'приложение/xhtml+xml' или 'текст/html' загружается, и URL-адрес не заканчивается регулярным выражением '\.[Xч][Тт][Мм][Лл]?', эта опция приведет к появлению суффикса '.html', который будет добавлен к локальному имени файла. Это полезно, например, когда вы зеркалируете удаленный сайт, использующий '.asp', но вы хотите, чтобы зеркальные страницы были доступны для просмотра на стандартном сервере Apache. Еще одно хорошее применение – при загрузке материалов, созданных с помощью CGI. URL-адрес типа 'http://site.com/article.cgi?25' будет сохранен как `статья.cgi?25.html`.

Обратите внимание, что имена файлов, измененные таким образом, будут загружаться повторно каждый раз, когда вы повторно зеркалируете сайт, поскольку Wget не может определить, что локальный *X.html* файл соответствует удаленному URL '*Икс*' (поскольку он еще не знает, что URL-адрес выдает выходные данные типа 'текст/html' или 'приложение/xhtml+xml').

Начиная с версии 1.12, Wget также гарантирует, что все загруженные файлы типа 'текст/css' заканчиваются суффиксом '.css', и опция была переименована в '--html-расширение', чтобы лучше отразить его новое поведение. Старое имя опции по-прежнему приемлемо, но теперь его следует считать устаревшим.

Начиная с версии 1.19.2, Wget также гарантирует, что все загруженные файлы с пометкой Content-Encoding 'br', 'компресс', 'сдувать' или 'gzip' заканчиваются суффиксом '.br', '.C', '.zlib' и '.gz' соответственно.

В какой-то момент в будущем эта опция вполне может быть расширена за счет включения суффиксов для других типов контента, включая типы контента, которые не анализируются Wget.

### '--http-user= *пользователь*'

### '--http-password= *пароль*'

Укажите имя пользователя и пароль – *пароль* на HTTP- сервере. В зависимости от типа запроса Wget закодирует их, используя либо basic(небезопасную), digestлибо схему аутентификации Windows NTLM.

Другой способ указать имя пользователя и пароль – в самом URL-адресе (см. [Формат URL-адреса](#)). Любой метод раскрывает ваш пароль любому, кто потрудится запустить ps. Чтобы пароли не были видны, используйте '--use-askpass' или сохранить их в wgetrc или .netrc и обязательно защитите эти файлы от других пользователей с помощью chmod. Если пароли действительно важны, не оставляйте их в этих файлах – отредактируйте файлы и удалите их после того, как Wget начнет загрузку.

### '--no-http-keep-alive'

Отключите функцию «поддержания активности» для HTTP-загрузок. Обычно Wget просит сервер поддерживать соединение открытым, чтобы при загрузке более одного документа с одного и того же сервера они передавались по одному и тому же TCP-соединению. Это экономит время и в то же время снижает нагрузку на сервер.

Эта опция полезна, когда у вас по каким-то причинам не работают постоянные (keep-alive) соединения, например из-за ошибки сервера или из-за неспособности серверных скриптов справиться с соединениями.

### '--no-cache'

Отключите кэш на стороне сервера. В этом случае Wget отправит удаленному серверу соответствующие директивы ('Управление кэшем: без кэша' и 'Прагма: без кэша'), чтобы получить файл из удаленной службы, а не возвращать кэшированную версию. Это особенно полезно для получения и удаления устаревших документов на прокси-серверах.

Кэширование разрешено по умолчанию.

### '--net-persistent'

Отключите использование файлов cookie. Файлы cookie – это механизм поддержания состояния на стороне сервера. Сервер отправляет клиенту файл cookie, используя заголовок Set-Cookie, и клиент отвечает тем же файлом cookie при дальнейших запросах. Поскольку файлы cookie позволяют владельцам серверов отслеживать посетителей, а сайтам обмениваться этой информацией, некоторые считают их нарушением конфиденциальности. По умолчанию используются файлы cookie; однако *сохранение* файлов cookie по умолчанию не включено.

### '--load-cookies файл'

Загружайте файлы cookie из *файла* до первого получения HTTP. *Файл* представляет собой текстовый файл в формате, первоначально использовавшемся Netscape. файлы cookie.txt файл.

Обычно вы будете использовать эту опцию при зеркалировании сайтов, требующих входа в систему для доступа к части или всему их контенту. Процесс входа в систему обычно выполняется веб-сервером, выдающим HTTP- файл cookie после получения и проверки ваших учетных данных. Затем файл cookie повторно

отправляется браузером при доступе к этой части сайта и таким образом подтверждает вашу личность.

Зеркальное отображение такого сайта требует, чтобы Wget отправлял те же файлы cookie, которые ваш браузер отправляет при взаимодействии с сайтом. Это достигается за счет '--load-cookies' – просто укажите Wget на местоположение файлы cookie.txt файл, и он отправит те же файлы cookie, которые ваш браузер отправил бы в той же ситуации. Разные браузеры хранят текстовые файлы cookie в разных местах:

#### Нетскейп 4.x.

Файлы cookie находятся в ~/.netscape/cookies.txt.

#### Мозилла и Нетскейп 6.x.

Файл cookie Mozilla также называется файлы cookie.txt, расположенный где-то под ~/.mozilla, в каталоге вашего профиля. Полный путь обычно выглядит примерно так: ~/.mozilla/default/*некоторая-странный-строка* /cookies.txt.

#### Интернет Эксплорер.

Вы можете создать файл cookie, который Wget сможет использовать, используя меню «Файл», «Импорт и экспорт», «Экспорт файлов cookie». Это было протестировано с Internet Explorer 5; не гарантируется работа с более ранними версиями.

#### Другие браузеры.

Если вы используете другой браузер для создания файлов cookie, '--load-cookies' будет работать только в том случае, если вы сможете найти или создать файл cookie в формате Netscape, который ожидает Wget.

Если вы не можете использовать '--load-cookies', возможно, еще есть альтернатива. Если ваш браузер поддерживает «менеджер файлов cookie», вы можете использовать его для просмотра файлов cookie, используемых при доступе к зеркалируемому вами сайту. Запишите имя и значение файла cookie и вручную поручите Wget отправить эти файлы cookie в обход «официальной» поддержки файлов cookie:

```
wget --no-cookies --header "Куки: имя = значение "
```

#### '--save-cookies *файл*'

Перед выходом сохраните файлы cookie в *файл* . Это не позволит сохранить файлы cookie, срок действия которых истек или срок действия которых не истек (так называемые «сессионные файлы cookie»), но также см.--keep-session-cookies'.

### '--keep-session-cookies'

Если указано, вызывает '--save-cookies', чтобы также сохранить файлы cookie сеанса. Сеансовые файлы cookie обычно не сохраняются, поскольку они предназначены для хранения в памяти и забывания при выходе из браузера. Их сохранение полезно на сайтах, которые требуют входа в систему или посещения домашней страницы, прежде чем вы сможете получить доступ к некоторым страницам. При использовании этой опции несколько запусков Wget считаются одним сеансом браузера для сайта.

Поскольку формат файлов cookie обычно не содержит файлов cookie сеанса, Wget помечает их временной меткой истечения срока действия, равной 0.--load-cookies' распознает их как файлы cookie сеанса, но это может сбить с толку другие браузеры. Также обратите внимание, что загруженные таким образом файлы cookie будут рассматриваться как другие файлы cookie сеанса, а это означает, что если вы хотите:--save-cookies' чтобы сохранить их снова, вы должны использовать '--keep-session-cookies' снова.

### '--ignore-длина'

К сожалению, некоторые HTTP-серверы (точнее, CGIContent-Length -программы) рассылают фиктивные заголовки, из-за чего Wget сходит с ума, поскольку считает, что не весь документ был получен. Вы можете заметить этот синдром, если Wget снова и снова пытается получить один и тот же документ, каждый раз заявляя, что (в противном случае нормальное) соединение закрылось на одном и том же байте.

При использовании этой опции Wget будет игнорировать Content-Lengthзаголовок, как если бы его никогда не существовало.

### '--header= строка заголовка'

Отправляйте *строку заголовка* вместе с остальными заголовками в каждом HTTP-запросе. Предоставленный заголовок отправляется «как есть», что означает, что он должен содержать имя и значение, разделенные двоеточием, и не должен содержать символы новой строки.

Вы можете определить более одного дополнительного заголовка, указав '--header' больше чем единожды.

```
wget --header='Accept-Charset: iso-8859-2' \  
      --header='Accept-Language: hr' \  
      http://fly.srk.fer.hr/
```

Указание пустой строки в качестве значения заголовка очистит все предыдущие заголовки, определенные пользователем.

Начиная с Wget 1.10, эту опцию можно использовать для переопределения заголовков, которые в противном случае генерируются автоматически. В этом примере Wget инструктируется подключиться к локальному хосту, но указать 'фу.бар' в Hostзаголовке:

```
wget --header="Хост: foo.bar" http://localhost/
```

В версиях Wget до 1.10 такое использование '--header' вызвал отправку повторяющихся заголовков.

### '--compression= *тип*'

Выберите тип сжатия, который будет использоваться. Юридические ценности *авто*, 'gzip' и 'никто'.

Если 'авто' или 'gzip' указаны, Wget просит сервер сжать файл, используя формат сжатия gzip. Если сервер сжимает файл и отвечает Content-Encoding соответствующим образом установленным полем заголовка, файл будет распакован автоматически.

Если 'никто', wget не будет запрашивать у сервера сжатие файла и не будет распаковывать ответы сервера. Это значение по умолчанию.

Поддержка сжатия в настоящее время является экспериментальной. Если он включен, сообщите о любых ошибках по адресу [bug-wget@gnu.org](mailto:bug-wget@gnu.org).

### '--max-redirect= *число*'

Указывает максимальное количество перенаправлений, которым должен следовать ресурс. По умолчанию установлено значение 20, что обычно намного больше, чем необходимо. Однако в тех случаях, когда вы хотите разрешить больше (или меньше), этот вариант можно использовать.

### '--proxy-user= *пользователь*'

### '--proxy-password= *пароль*'

Укажите логин пользователя и пароль- *пароль* для аутентификации на прокси-сервере. Wget закодирует их, используя basicсхему аутентификации.

Соображения безопасности аналогичны тем, которые приведены для '--http-пароль' относится и сюда.

### '--refer= *URL*'

Включите заголовок «Referer: *url* » в HTTP-запрос. Полезно для получения документов с обработкой на стороне сервера, которая предполагает, что они всегда извлекаются интерактивными веб-браузерами и отображаются правильно только тогда, когда Referer установлен на одну из страниц, которые на них указывают.

**'--save-headers'**

Сохраните заголовки, отправленные HTTP- сервером, в файл, предшествующий фактическому содержимому, с пустой строкой в качестве разделителя.

**'-U строка агента'****'--user-agent= строка-агента'**

Идентифицировать как *строку агента* для HTTP -сервера.

Протокол HTTP позволяет клиентам идентифицировать себя с помощью User-Agent поля заголовка. Это позволяет различать программное обеспечение WWW , обычно в статистических целях или для отслеживания нарушений протокола. Wget обычно идентифицируется как 'Wget/ версия', где *version* – текущий номер версии Wget.

Однако известно, что некоторые сайты навязывают политику адаптации результатов в соответствии с User-Agent предоставленной информацией. Хотя в теории это не такая уж и плохая идея, серверы злоупотребляют ею, отказывая в предоставлении информации другим клиентам, кроме (исторически) Netscape или, чаще, Microsoft Internet Explorer. Эта опция позволяет вам изменить User-Agent строку, выданную Wget. Использование этой опции не рекомендуется, если вы действительно не знаете, что делаете.

Указание пустого пользовательского агента с помощью '--user-agent=""' инструктирует Wget не отправлять User-Agent заголовков в HTTP- запросах.

**'--post-data= строка'****'--post-file= файл'**

Используйте POST в качестве метода для всех HTTP-запросов и отправляйте указанные данные в теле запроса. '--post-данные' отправляет *строку* как данные, тогда как '--post-file' отправляет содержимое *файла* . В остальном они работают точно так же. В частности, они *оба* ожидают содержимого в форме `key1=value1&key2=value2` с процентным кодированием специальных символов; единственная разница состоит в том, что один ожидает его содержимого как параметра командной строки, а другой принимает его содержимое из файла. В частности, '--post-file' *не* предназначен для передачи файлов в виде вложений: они должны отображаться как `key=value` данные (с соответствующим процентным кодированием), как и все остальное. Wget в настоящее время не поддерживает multipart/form-data передачу данных POST; только application/x-www-form-urlencoded. Только один из '--post-данные' и '--post-file', следует указать.

Обратите внимание, что wget не требует, чтобы содержимое имело форму `key1=value1&key2=value2`, и не проверяет его. Wget просто передаст любые предоставленные ему данные. Однако большинство серверов ожидают, что данные POST будут в указанном выше формате при обработке HTML-форм.

При отправке POST-запроса с использованием '--post-file', Wget рассматривает файл как двоичный файл и отправляет каждый символ в запросе POST, не удаляя конечные символы новой строки или перевода строки. Любые другие управляющие символы в тексте также будут отправлены в запросе POST без изменений.

Имейте в виду, что Wget необходимо заранее знать размер данных POST.

Следовательно, аргументом --post-file должен быть обычный файл; указание FIFO или что-то вроде /dev/стандартный ввод не сработает. Не совсем понятно, как обойти это ограничение, присущее HTTP/1.0. Хотя HTTP/1.1 вводит *фрагментированную* передачу, которая не требует предварительного знания длины запроса, клиент не может использовать фрагментированную передачу, если не знает, что обращается к серверу HTTP/1.1. И он не может знать этого, пока не получит ответ, который, в свою очередь, требует завершения запроса – проблема курицы и яйца.

Примечание. Начиная с версии 1.15, если Wget перенаправляется после завершения запроса POST, его поведение будет зависеть от кода ответа, возвращаемого сервером. В случае 301 Moved Permanently, 302 Moved Temporarily или 307 Temporary Redirect, Wget, в соответствии с RFC2616, продолжит отправлять запросы POST. Если сервер хочет, чтобы клиент изменил метод запроса при перенаправлении, он должен отправить код ответа 303 See Other.

В этом примере показано, как войти на сервер с помощью POST, а затем приступить к загрузке нужных страниц, предположительно доступных только авторизованным пользователям:

```
# Войдите на сервер. Это можно сделать только один раз.
wget --save-cookies cookies.txt \
      --post-data 'user=foo&password=bar' \
      http://example.com/auth.php

# Теперь возьмите страницу или страницы, которые нас интересуют.
wget --load-cookies cookies.txt \
      -p http://example.com/interesting/article.php
```

Если сервер использует файлы cookie сеанса для отслеживания аутентификации пользователя, описанное выше не будет работать, потому что '--save-cookies' не сохранит их (как и браузеры), а файл cookie.txt будет пуст. В этом случае используйте '--keep-session-cookies' вместе с '--save-cookies', чтобы принудительно сохранить файлы cookie сеанса.

### '--method= HTTP-метод'

Для целей сценариев RESTful Wget позволяет отправлять другие методы HTTP без необходимости явно устанавливать их с помощью '--header=Строка заголовка'. Wget



будет использовать любую строку, переданную ему после '--метод' в качестве метода HTTP для сервера.

**'--body-data= *Строка данных*'**

**'--body-file= *Файл данных*'**

Должен быть установлен, когда дополнительные данные необходимо отправить на сервер вместе с методом, указанным с помощью '--метод'. '--body-данные' отправляет *строку* как данные, тогда как '--body-файл' отправляет содержимое *файла*. В остальном они работают точно так же.

В настоящее время, '--body-файл' *не* предназначен для передачи файлов целиком. Wget в настоящее время не поддерживает multipart/form-data передачу данных; только application/x-www-form-urlencoded. В будущем это может быть изменено, чтобы wget отправлял '--body-файл' как полный файл вместо отправки его содержимого на сервер. Имейте в виду, что Wget необходимо заранее знать содержимое данных BODY и, следовательно, аргумент '--body-файл' должен быть обычным файлом. Видеть '--post-file' для более подробного объяснения. Только один из '--body-данные' и '--body-файл', следует указать.

Если Wget перенаправляется после завершения запроса, Wget приостановит текущий метод и отправит запрос GET до завершения перенаправления. Это справедливо для всех кодов ответа перенаправления, за исключением 307 Temporary Redirect, который используется для явного указания того, что метод запроса *не* должен меняться. Другое исключение – когда для метода установлено значение POST, и в этом случае правила перенаправления, указанные в разделе «--post-данные' следуют.

**'--content-disposition'**

Если этот параметр включен, Content-Disposition включается экспериментальная (не полнофункциональная) поддержка заголовков. В настоящее время это может привести к дополнительным обращениям к серверу для HEAD-запроса и, как известно, страдает от нескольких ошибок, поэтому в настоящее время он не включен по умолчанию.

Эта опция полезна для некоторых программ CGI, загружающих файлы, которые используют Content-Disposition заголовки для описания того, каким должно быть имя загружаемого файла.

В сочетании с '--metalink-over-http' и '--trust-server-names', 'Тип контента: приложение/metalink4+xml' файл получает имя с использованием Content-Disposition поля имени файла, если оно доступно.

**'--content-при ошибке'**

Если этот параметр включен, wget не будет пропускать контент, когда сервер отвечает кодом состояния http, указывающим на ошибку.

**'--trust-server-names'**

Если этот параметр установлен, при перенаправлении имя локального файла будет основано на URL-адресе перенаправления. По умолчанию имя локального файла основано на исходном URL-адресе. При рекурсивном извлечении это может быть полезно, поскольку на многих веб-сайтах перенаправленные URL-адреса соответствуют базовой файловой структуре, а URL-адреса ссылок – нет.

### '--auth-no-challenge'

Если указана эта опция, Wget будет отправлять базовую информацию аутентификации HTTP (имя пользователя и пароль в виде открытого текста) для всех запросов, точно так же, как Wget 1.10.2 и более ранние версии делали это по умолчанию.

Использование этой опции не рекомендуется и предназначено только для поддержки некоторых малоизвестных серверов, которые никогда не отправляют запросы HTTP-аутентификации, но принимают незапрошенную информацию для аутентификации, скажем, в дополнение к аутентификации на основе форм.

### '--retry-on-host-error'

Считайте ошибки хоста, такие как «Временный сбой разрешения имен», нефатальными временными ошибками.

### '--retry-on-http-error= код[,код,...]'

Рассматривайте данные коды ответов HTTP как нефатальные временные ошибки. В качестве аргумента укажите список трехзначных кодов ответов HTTP, разделенных запятыми. Полезно для решения особых обстоятельств, когда требуются повторные попытки, но сервер отвечает кодом ошибки, который обычно не повторяется Wget. Такими ошибками могут быть 503 (Служба недоступна) и 429 (Слишком много запросов). Повторные попытки, включенные этой опцией, выполняются с учетом обычных ограничений времени повторов и количества повторов Wget.

Использование этой опции предназначено только для поддержки особых случаев использования и, как правило, не рекомендуется, поскольку может привести к вынужденным повторным попыткам даже в тех случаях, когда сервер действительно пытается уменьшить свою нагрузку. Пожалуйста, используйте разумно и только в том случае, если вы знаете, что делаете.

---

Следующая: [Параметры FTP](#) , Предыдущая: [Параметры HTTP](#) , Вверх: [Вызов](#) [ [Содержание](#) ]  
[ [Индекс](#) ]

## 2.8 Параметры HTTPS (SSL/TLS)

Для поддержки зашифрованных загрузок HTTP (HTTPS) Wget должен быть скомпилирован с внешней библиотекой SSL. Текущее значение по умолчанию – GnuTLS. Кроме того, Wget также поддерживает HSTS (HTTP Strict Transport Security). Если Wget скомпилирован без поддержки SSL, ни одна из этих опций не будет доступна.

### '--secure-protocol= *протокол*'

Выберите безопасный протокол, который будет использоваться. Юридические ценности авто', 'SSLv2', 'SSLv3', 'TLSv1', 'TLSv1\_1', 'TLSv1\_2', 'TLSv1\_3' и 'ПФС'. Если 'авто', библиотеке SSL предоставляется возможность автоматического выбора соответствующего протокола, что достигается путем отправки приветствия TLSv1. Это значение по умолчанию.

Указание 'SSLv2', 'SSLv3', 'TLSv1', 'TLSv1\_1', 'TLSv1\_2' или 'TLSv1\_3' заставляет использовать соответствующий протокол. Это полезно при общении со старыми и ошибочными реализациями SSL-сервера, из-за которых базовой библиотеке SSL сложно выбрать правильную версию протокола. К счастью, такие серверы встречаются довольно редко.

Указание 'ПФС' требует использования так называемых наборов шифров Perfect Forward Security. Короче говоря, PFS повышает безопасность, создавая одноразовый ключ для каждого SSL-соединения. Это оказывает немного большее влияние на процессор на клиенте и сервере. Мы используем заведомо безопасные шифры (например, без MD4) и протокол TLS. Этот режим также явно исключает методы обмена ключами, не относящиеся к PFS, такие как RSA.

### '--только https'

В рекурсивном режиме отслеживаются только ссылки HTTPS.

### '--шифры'

Установите строку списка шифров. Обычно эта строка устанавливает наборы шифров и другие параметры SSL/TLS, которые пользователь желает использовать, в установленном порядке предпочтений (GnuTLS называет это «строкой приоритета»). Эта строка будет дословно передана в механизм SSL/TLS (OpenSSL или GnuTLS), и, следовательно, ее формат и синтаксис зависят от этого. Wget не будет каким-либо образом обрабатывать или манипулировать ими. Для получения дополнительной информации обратитесь к документации OpenSSL или GnuTLS.

### '--no-check-сертификат'

Не проверяйте сертификат сервера по доступным центрам сертификации. Также не требуйте, чтобы имя хоста URL-адреса совпадало с общим именем, представленным в сертификате.

Начиная с Wget 1.10, по умолчанию выполняется проверка сертификата сервера на соответствие признанным центрам сертификации, прерывание SSL-квитирования и прерывание загрузки в случае неудачной проверки. Хотя это обеспечивает более безопасную загрузку, это нарушает совместимость с некоторыми сайтами, которые работали с предыдущими версиями Wget, особенно с теми, которые используют самозаверяющие, просроченные или иным образом недействительные сертификаты. Эта опция включает «небезопасный» режим работы, который превращает ошибки проверки сертификата в предупреждения и позволяет вам продолжить.

Если вы столкнулись с ошибками «проверки сертификата» или сообщениями о том, что «общее имя не соответствует запрошенному имени хоста», вы можете использовать эту опцию, чтобы обойти проверку и продолжить загрузку. *Используйте эту опцию только в том случае, если вы уверены в подлинности сайта или если вас действительно не волнует действительность его сертификата.* Не проверять сертификаты при передаче конфиденциальных или важных данных почти всегда плохая идея. Для самозаверяющих/внутренних сертификатов вам следует загрузить сертификат и проверить его, а не принудительно использовать этот небезопасный режим. Если вы действительно уверены, что не хотите проверять сертификаты, вы можете указать `--check-certificate=quiet`, чтобы сообщить wget не выводить никаких предупреждений о недействительных сертификатах, хотя в большинстве случаев это неправильный поступок.

### `--certificate= файл`

Используйте сертификат клиента, хранящийся в *файле*. Это необходимо для серверов, настроенных на требование сертификатов от подключающихся к ним клиентов. Обычно сертификат не требуется, и этот переключатель является необязательным.

### `--certificate-type= тип`

Укажите тип сертификата клиента. Юридические ценности ПЕМ' (предполагается по умолчанию) и 'ТО', также известен как 'АСН1'.

### `--private-key= файл`

Прочтите закрытый ключ из *файла*. Это позволяет вам предоставить закрытый ключ в файле отдельно от сертификата.

### `--private-key-type= тип`

Укажите тип закрытого ключа. Принятые значения: 'ПЕМ' (по умолчанию) и 'ТО'.

### `--ca-certificate= файл`

Используйте *файл* в качестве файла с пакетом центров сертификации («СА») для проверки узлов. Сертификаты должны быть в формате PEM.

Без этой опции Wget ищет сертификаты СА в указанных системой местах, выбранных во время установки OpenSSL.

### `--ca-directory= каталог`

Указывает каталог, содержащий сертификаты СА в формате PEM. Каждый файл содержит один сертификат ЦС, а имя файла основано на хеш-значении, полученном из сертификата. Это достигается путем обработки каталога сертификатов с помощью `s_grehash` утилиты, входящей в состав OpenSSL. С использованием '`--ca-каталог`' более эффективен, чем '`--as-сертификаты`', когда установлено много сертификатов, поскольку это позволяет Wget получать сертификаты по требованию.

Без этой опции Wget ищет сертификаты CA в указанных системой местах, выбранных во время установки OpenSSL.

### '--crl-file= *файл*'

Указывает файл CRL в *файле* . Это необходимо для сертификатов, которые были отозваны центрами сертификации.

### '--pinnedpubkey=файл/хеши'

Сообщает wget использовать указанный файл открытого ключа (или хеши) для проверки узла. Это может быть путь к файлу, который содержит один открытый ключ в формате PEM или DER, или любое количество хэшей sha256 в кодировке Base64, которым предшествует «sha256//» и которые разделены знаком «;».

При согласовании соединения TLS или SSL сервер отправляет сертификат, указывающий его личность. Из этого сертификата извлекается открытый ключ, и если он не совсем соответствует открытому ключу(ам), предоставленному для этой опции, wget прервет соединение перед отправкой или получением каких-либо данных.

### '--random-file= *файл*'

[Только для OpenSSL и LibreSSL] Используйте *файл* в качестве источника случайных данных для заполнения генератора псевдослучайных чисел в системах без/dev/urandom.

В таких системах для инициализации библиотеки SSL требуется внешний источник случайных данных. Случайность может быть обеспечена с помощью EGD (см.--egd-*файл* ниже) или прочитать из внешнего источника, указанного пользователем. Если этот параметр не указан, Wget ищет случайные данные в \$RANDFILEили, если этот параметр не установлен, в\$HOME/.rnd.

Если вы получаете сообщение «Не удалось заполнить OpenSSL PRNG; отключение SSL». ошибка, вам следует предоставить случайные данные, используя некоторые из методов, описанных выше.

### '--egd-file= *файл*'

[Только OpenSSL] Использовать *файл* в качестве сокета EGD. EGD означает *Entropy Gathering Daemon* , программу пользовательского пространства, которая собирает данные из различных непредсказуемых системных источников и делает их доступными для других программ, которым они могут понадобиться. Программному обеспечению шифрования, такому как библиотека SSL, необходимы источники неповторяющейся случайности для создания генератора случайных чисел, используемого для создания криптостойких ключей.

OpenSSL позволяет пользователю указать свой собственный источник энтропии с помощью RAND\_FILEпеременной среды. Если эта переменная не установлена или если указанный файл не обеспечивает достаточную случайность, OpenSSL будет считывать случайные данные из сокета EGD, указанного с помощью этой опции.

Если этот параметр не указан (и не используется эквивалентная команда запуска), соединение с EGD никогда не осуществляется. EGD не требуется в современных системах Unix, поддерживающих/dev/urandom.

### '--no-hsts'

Wget по умолчанию поддерживает HSTS (HTTP Strict Transport Security, RFC 6797). Использовать '--no-hsts', чтобы Wget действовал как UA, не соответствующий HSTS. Как следствие, Wget будет игнорировать все Strict-Transport-Security заголовки и не будет применять существующую политику HSTS.

### '--hsts-file= *файл*'

По умолчанию Wget хранит свою базу данных HSTS в ~/.wget-hsts. Вы можете использовать '--hsts-файл', чтобы переопределить это. Wget будет использовать предоставленный файл в качестве базы данных HSTS. Такой файл должен соответствовать правильному формату базы данных HSTS, используемому Wget. Если Wget не может проанализировать предоставленный файл, поведение не указано.

База данных HSTS Wget представляет собой обычный текстовый файл. Каждая строка содержит запись HSTS (т. е. сайт, который выдал заголовок Strict-Transport-Security, следовательно, указал конкретную политику HSTS, которая должна применяться). Строки, начинающиеся с тире (#), игнорируются Wget. Обратите внимание, что, несмотря на удобство и удобочитаемость, ручной взлом базы данных HSTS, как правило, не является хорошей идеей.

Строка ввода HSTS состоит из нескольких полей, разделенных одним или несколькими пробелами:

```
<hostname> SP [<port>] SP <include subdomains> SP <created> SP <max-age>
```

Поля имени *хоста* и *порта* указывают имя хоста и порт, к которым применяется данная политика HSTS. Поле порта может быть нулевым, и в большинстве случаев так оно и есть. Это означает, что номер порта не будет учитываться при принятии решения о том, следует ли применять такую политику HSTS к данному запросу (будет оцениваться только имя хоста). Если *порт* отличается от нуля, будут оцениваться и имя целевого хоста, и порт, и политика HSTS будет применяться только в том случае, если они оба совпадают. Эта функция была включена только в целях тестирования/разработки. Набор тестов Wget (втеств/) создает базы данных HSTS с явными портами с целью обеспечения правильного поведения Wget. Применение политик HSTS к портам, отличным от портов по умолчанию, не рекомендуется RFC 6797 (см. Приложение В «Различия между политикой HSTS и политикой одного и того же происхождения»). Таким образом, эту функцию не следует использовать в производственных средах, а значение *порта* обычно равно нулю. Последние три поля делают то, что от них ожидается. Поле *include\_subdomains* может иметь значение 1 или 0 и указывает, должны ли поддомены целевого домена также быть частью данной политики HSTS. Поля «*созданный*» и «*максимальный возраст*» содержат значения временной метки, когда такая запись была создана (впервые замечена Wget), а

также определенное HSTS значение «max-age», которое указывает, как долго эта политика HSTS должна оставаться активной, измеряется в прошедших секундах. поскольку временная метка хранится в *созданном файле* . По истечении этого времени эта политика HSTS перестанет быть действительной и в конечном итоге будет удалена из базы данных.

Если вы предоставляете свою собственную базу данных HSTS через '--hsts-файл', имейте в виду, что Wget может изменить предоставленный файл, если произойдет какое-либо изменение между политиками HSTS, запрошенными удаленными серверами, и политиками в файле. Когда Wget завершает работу, он эффективно обновляет базу данных HSTS, перезаписывая файл базы данных новыми записями.

Если предоставленный файл не существует, Wget создаст его. Этот файл будет содержать новые записи HSTS. Если записи HSTS не были созданы ( Strict-Transport-Security ни один из серверов не отправил заголовки), то ни один файл не будет создан, даже пустой. Такое поведение применимо к файлу базы данных по умолчанию (~/.wget-hsts) также: он не будет создан, пока какой-либо сервер не применит политику HSTS.

Принимаются меры, чтобы не переопределить возможные изменения, внесенные другими процессами Wget одновременно в базу данных HSTS. Прежде чем записать обновленные записи HSTS в файл, Wget перечитывает его и объединит изменения.

Использование собственной базы данных HSTS и/или изменение существующей не рекомендуется. Дополнительную информацию о потенциальных угрозах безопасности, возникающих в результате такой практики, см. в разделе 14 «Вопросы безопасности» RFC 6797, особенно в разделе 14.9 «Творческое манипулирование хранилищем политик HSTS».

#### '--warc-file= *файл*'

Используйте *файл* в качестве целевого файла WARC.

#### '--warc-header= *строка*'

Используйте *строку* в качестве записи warcinfo.

#### '--warc-max-size= *размер*'

Установите максимальный размер файлов WARC *размером* .

#### '--warc-cdx'

Запись индексных файлов CDX.

#### '--warc-dedup= *файл*'

Не храните записи, перечисленные в этом файле CDX.

#### '--no-warc-сжатие'

Не сжимайте файлы WARC с помощью GZIP.

**'--no-warc-дайджесты'**

Не рассчитывайте дайджесты SHA1.

**'--no-warc-keep-log'**

Не храните файл журнала в записи WARC.

**'--warc-tempdir= *каталог*'**

Укажите расположение временных файлов, созданных модулем записи WARC.

---

Следующая: [Параметры рекурсивного извлечения](#) , Предыдущая: [Параметры HTTPS \(SSL/TLS\)](#)  
, Вверх: [Вызов](#) [ [Содержание](#) ][ [Индекс](#) ]

## 2.9 Параметры FTP

**'--ftp-user= *пользователь*'****'--ftp-password= *пароль*'**

Укажите имя пользователя и пароль- *пароль* на FTP- сервере. Без этого или соответствующего параметра запуска пароль по умолчанию равен «-wget@», обычно используется для анонимного FTP.

Другой способ указать имя пользователя и пароль – в самом URL-адресе (см. [Формат URL-адреса](#) ). Любой метод раскрывает ваш пароль любому, кто потрудится запустить ps. Чтобы пароли не были видны, храните их в.wgetrcили.netrcи обязательно защитите эти файлы от других пользователей с помощью chmod. Если пароли действительно важны, не оставляйте их в этих файлах – отредактируйте файлы и удалите их после того, как Wget начнет загрузку.

**'--no-remove-листинг'**

Не удаляйте временные.листингфайлы, созданные при извлечении по FTP . Обычно эти файлы содержат необработанные списки каталогов, полученные с FTP -серверов. Не удалять их может быть полезно в целях отладки или когда вы хотите иметь возможность легко проверять содержимое каталогов удаленного сервера (например, чтобы убедиться, что зеркало, которое вы используете, завершено).

Обратите внимание, что даже несмотря на то, что Wget записывает в известное имя этого файла, это не является дырой в безопасности в сценарии, когда пользователь делает .листингсимволическая ссылка на/etc/парольили что-то в этом роде и попросить gootзапустить Wget в его или ее каталоге. В зависимости от используемых опций Wget откажется писать в.листинг, что приведет к сбою операции подстановки/рекурсии/отметки времени, или символическая ссылка будет удалена и заменена фактической .листингфайл, иначе список будет записан в .листинг. *число*файл.



Несмотря на то, что эта ситуация не является проблемой, root'ник иногда не следует запускать Wget в каталоге недоверенного пользователя. Пользователь может сделать что-то простое, например, связать `index.htmlk/etc/пароль` и попросить root'запустить Wget с помощью `'-H'` или `'-p'` поэтому файл будет перезаписан.

### '--no-glob'

Отключите подстановку FTP . Подстановка означает использование специальных символов, подобных оболочке ( *шаблонов* ), например `'*'`, `'?'`, `'['` и `']'`, чтобы получить одновременно несколько файлов из одного каталога, например:

```
wget ftp://gnjilux.srk.fer.hr/*.msg
```

По умолчанию подстановка включена, если URL-адрес содержит подстановочный символ. Эту опцию можно использовать для постоянного включения или отключения подстановки.

Возможно, вам придется заключить URL-адрес в кавычки , чтобы защитить его от расширения вашей оболочкой. Подстановка заставляет Wget искать список каталогов, который зависит от системы. Вот почему в настоящее время он работает только с FTP- серверами Unix (и теми, которые эмулируют lsвывод Unix).

### '--no-passive-ftp'

Отключите использование *пассивного* режима передачи по FTP. Пассивный FTP требует, чтобы клиент подключался к серверу для установления соединения для передачи данных, а не наоборот.

Если машина подключена к Интернету напрямую, как пассивный, так и активный FTP должны работать одинаково хорошо. За большинством конфигураций брандмауэра и NAT пассивный FTP имеет больше шансов на работу. Однако в некоторых редких конфигурациях брандмауэра активный FTP работает, а пассивный FTP – нет. Если вы подозреваете, что это так, используйте эту опцию или установите ее `passive_ftp=off` в своем файле инициализации.

### '--preserve-разрешения'

Сохраняйте разрешения для удаленных файлов вместо разрешений, установленных `umask`.

### '--retr-symlinks'

По умолчанию при рекурсивном получении каталогов FTP и обнаружении символической ссылки символическая ссылка проходится и извлекаются файлы, на которые указывает. В настоящее время Wget не проходит по символическим ссылкам на

каталоги для их рекурсивной загрузки, хотя эта функция может быть добавлена в будущем.

Когда `'--retr-symlinks=нет'` указан, связанный файл не загружается. Вместо этого в локальной файловой системе создается соответствующая символическая ссылка. Файл, на который указывает указанный файл, не будет получен, если только этот рекурсивный поиск не встретит его отдельно и все равно не загрузит. Этот параметр представляет угрозу безопасности, поскольку вредоносный FTP-сервер может заставить Wget записывать файлы за пределами предполагаемых каталогов через специально созданный файл `.LISTING`.

Обратите внимание, что при получении файла (а не каталога), поскольку он был указан в командной строке, а не потому, что к нему была произведена рекурсия, этот параметр не имеет никакого эффекта. В этом случае всегда пропускаются символические ссылки.

## 2.10 Параметры FTPS

### `'--ftps-implicit'`

Этот параметр указывает Wget использовать FTPS неявно. Неявный FTPS заключается в инициализации SSL/TLS с самого начала управляющего соединения. Этот параметр не отправляет AUTH TLSкоманду: предполагается, что сервер использует FTPS и напрямую запускает соединение SSL/TLS. Если попытка успешна, сессия продолжается как обычный FTPS ( PBSZи PROTотправляются и т.д.). Неявный FTPS больше не является обязательным требованием для реализаций FTPS, поэтому многие серверы могут его не поддерживать. Если `'--ftps-implicit'` и не указан явный номер порта, будет использоваться порт по умолчанию для неявного FTPS, 990, вместо порта по умолчанию для «нормального» (явного) FTPS, который такой же, как и для FTP, 21.

### `'--no-ftps-resume-ssl'`

Не возобновляйте сеанс SSL/TLS в канале данных. При запуске соединения для передачи данных Wget пытается возобновить сеанс SSL/TLS, ранее запущенный в управляющем соединении. Возобновление сеанса SSL/TLS позволяет избежать выполнения совершенно нового рукопожатия за счет повторного использования параметров SSL/TLS предыдущего сеанса. Обычно FTPS-серверы этого хотят, поэтому Wget делает это по умолчанию. Однако в редких случаях может потребоваться запустить совершенно новый сеанс SSL/TLS при каждом подключении к данным. Это то, что `'--no-ftps-resume-ssl'` для.

### `'--ftps-clear-data-connection'`

Все подключения к данным будут в виде обычного текста. Только управляющее соединение будет находиться под SSL/TLS. Wget отправит PROT Сдля этого команду, которая должна быть одобрена сервером.

### '--ftp-fallback-to-ftp'

Вернитесь к FTP, если FTPS не поддерживается целевым сервером. По соображениям безопасности этот параметр не используется по умолчанию. Поведение по умолчанию – выход с ошибкой. Если сервер не отвечает успешно на исходную AUTH TLSкоманду или, в случае неявного FTPS, если первоначальная попытка подключения SSL/TLS отклонена, считается, что такой сервер не поддерживает FTPS.

Следующая: [Параметры рекурсивного принятия/отклонения](#) , Предыдущая: [Параметры FTP](#) ,  
Вверх: [Вызов](#) [ [Содержание](#) ][ [Индекс](#) ]

## 2.11 Параметры рекурсивного поиска

### '-r'

### '--рекурсивный'

Включите рекурсивное получение. Дополнительные сведения см . [в разделе Рекурсивная загрузка](#) . Максимальная глубина по умолчанию равна 5.

### '-л *глубина*'

### '--level= *глубина*'

Установите максимальное количество подкаталогов, в которые Wget будет рекурсивно *возвращаться* . Чтобы предотвратить случайную загрузку очень больших веб-сайтов при использовании рекурсии, по умолчанию глубина ограничена 5, т. е. она будет проходить не более 5 каталогов в глубину, начиная с предоставленного URL-адреса. Набор '-л 0' или '-л инф' для бесконечной глубины рекурсии.

```
wget -r -l 0 http:// сайт /1.html
```

В идеале можно было бы ожидать, что это загрузится только1.html. но, к сожалению, это не так, потому что '-л 0' эквивалентно '-л инф'- то есть бесконечная рекурсия. Чтобы загрузить одну HTML- страницу (или несколько из них), укажите их все в командной строке и оставьте '-р' и '-л'. Чтобы загрузить необходимые элементы для просмотра одной HTML- страницы, см.реквизиты страницы'.

### '--delete-после'

Эта опция указывает Wget удалить каждый загружаемый файл *после* этого. Это полезно для предварительной загрузки популярных страниц через прокси, например:

```
wget -r -nd --delete-after http://whatever.com/~popular/page/
```

'-p' вариант - рекурсивно получить, и '-nd', чтобы не создавать каталоги.

Обратите внимание, что '--delete-после' удаляет файлы на локальном компьютере. Он не выдает 'ОТ НЕГО', например, для удаленных FTP-сайтов. Также обратите внимание, что когда '--delete-после' указано, '--convert-links' игнорируется, поэтому '.orig' файлы просто не создаются.

'-к'

### '--convert-links'

После завершения загрузки преобразуйте ссылки в документе, чтобы сделать их пригодными для локального просмотра. Это влияет не только на видимые гиперссылки, но и на любую часть документа, которая ссылается на внешний контент, например встроенные изображения, ссылки на таблицы стилей, гиперссылки на контент, отличный от HTML, и т. д.

Каждая ссылка будет изменена одним из двух способов:

- Ссылки на файлы, загруженные Wget, будут изменены и будут ссылаться на файл, на который они указывают, как на относительную ссылку.

Пример: если загруженный файл/foo/doc.htmlссылки на /bar/img.gif, тоже скачал, то ссылка вдокумент.html будет изменен, чтобы указать на '../bar/img.gif'. Этот вид преобразования надежно работает для произвольных комбинаций каталогов.

- Ссылки на файлы, которые не были загружены Wget, будут изменены и будут включать имя хоста и абсолютный путь к местоположению, на которое они указывают.

Пример: если загруженный файл/foo/doc.htmlссылки на /bar/img.gif(или чтобы../bar/img.gif), то ссылка в документ.htmlбудет изменен, чтобы указать на `http:// имя хоста /bar/img.gif`.

Благодаря этому локальный просмотр работает надежно: если связанный файл был загружен, ссылка будет ссылаться на его локальное имя; если он не был загружен, ссылка будет ссылаться на его полный адрес в Интернете, а не на неработающую ссылку. Тот факт, что прежние ссылки преобразуются в относительные, гарантирует, что вы сможете переместить загруженную иерархию в другой каталог.

Обратите внимание, что только в конце загрузки Wget сможет узнать, какие ссылки были загружены. По этой причине работа, проделанная '-к' будет выполнено в конце всех загрузок.

### '--convert-file-only'

Эта опция преобразует только часть URL-адресов, посвященную имени файла, оставляя остальные URL-адреса нетронутыми. Эту часть имени файла иногда называют «базовым именем», хотя мы избегаем здесь этого термина, чтобы не вызывать путаницы.

Особенно хорошо это работает в сочетании с '--adjust-расширение', хотя эта связь не применяется. Полезно заполнять кэши Интернета файлами, загруженными с разных хостов.

Пример: если какая-то ссылка указывает на `//foo.com/bar.cgi?xyzc` '--adjust-расширение' утверждается, и его местное назначение должно быть `./foo.com/bar.cgi?xyz.css`, то ссылка будет преобразована в `//foo.com/bar.cgi?xyz.css`. Обратите внимание, что была изменена только часть имени файла. Остальная часть URL-адреса осталась нетронутой, включая сетевой путь ( `//` ), который в противном случае был бы обработан Wget и преобразован в эффективную схему (т. е. `http://`).

'-к'

'--backup-преобразовано'

При преобразовании файла создайте резервную копию исходной версии с помощью файла '.orig' суффикс. Влияет на поведение '-Н' (см. [Внутреннее устройство отметки времени HTTP](#) ).

'-м'

'--зеркало'

Включите параметры, подходящие для зеркалирования. Эта опция включает рекурсию и отметки времени, устанавливает бесконечную глубину рекурсии и сохраняет списки каталогов FTP . В настоящее время это эквивалентно `'-г -N -l inf --no-remove-listing'`.

'-п'

'--page-реквизиты'

Эта опция заставляет Wget загружать все файлы, необходимые для правильного отображения данной HTML- страницы. Сюда входят такие вещи, как встроенные изображения, звуки и таблицы стилей, на которые имеются ссылки.

Обычно при загрузке одной HTML- страницы не загружаются все необходимые документы, необходимые для ее правильного отображения. С использованием '-р' вместе с '-л' может помочь, но поскольку Wget обычно не различает внешние и встроенные документы, обычно остаются «листовые документы», в которых отсутствуют реквизиты.

Например, скажем документ1.html содержит <IMG>тег, ссылающийся на1.гифи <A>тег, указывающий на внешний документ2.html. Скажи это2.htmlпохож, но его изображение2.гифи это связано с3.html. Скажем, это продолжается до какого-то сколь угодно большого числа.

Если выполнить команду:

```
wget -r -l 2 http :  
//сайт/1.html
```

затем 1.html, 1.гиф, 2.html, 2.гиф, и 3.html будет загружен. Как вы видите, 3.html без необходимых реквизитов 3.гиф потому что Wget просто подсчитывает количество прыжков (до 2) от 1.html чтобы определить, где остановить рекурсию. Однако с помощью этой команды:

```
wget -r -l 2 -p http :  
//сайт/1.html
```

все вышеуказанные файлы и 3.html это необходимое 3.гиф будет загружен. Сходным образом,

```
wget -r -l 1 -p http:// сайт /1.html
```

вызовет 1.html, 1.гиф, 2.html, и 2.гиф для скачивания. Можно подумать, что:

```
wget -r -l 0 -p http :  
//сайт/1.html
```

скачал бы просто 1.html и 1.гиф, но, к сожалению, это не так, потому что '-л 0' эквивалентно '-л инф' - то есть бесконечная рекурсия. Чтобы загрузить одну HTML-страницу (или несколько из них, все указано в командной строке или в '-я' URL-адрес входного файла) и его (или их) реквизиты просто оставьте '-р' и '-л':

```
wget -p http :  
//сайт/1.html
```

Обратите внимание, что Wget будет вести себя так, как будто '-р' был указан, но будет загружена только эта единственная страница и ее реквизиты. Ссылки с этой страницы на внешние документы не будут использоваться. На самом деле, чтобы загрузить одну страницу и все ее реквизиты (даже если они существуют на

отдельных веб-сайтах) и убедиться, что лот отображается правильно локально, этот автор любит использовать несколько параметров в дополнение к '-п':

```
wget -E -H -k -K -p http:// сайт / документ
```

Чтобы завершить эту тему, стоит знать, что идея ссылки на внешний документ в Wget – это любой URL-адрес, указанный в <A>теге, <AREA>теге или <LINK>теге, отличным от <LINK REL="stylesheet">.

### '--strict-комментарии'

Включите строгий анализ HTML- комментариев. По умолчанию комментарии прекращаются при первом появлении '-->'.

Согласно спецификациям, комментарии HTML выражаются в виде *объявлений* SGML . Декларация – это специальная разметка, которая начинается с '<!' и заканчивается на '>', такой как '<!DOCTYPE ...>', который может содержать комментарии между парой '--' разделители. Комментарии HTML – это «пустые объявления», объявления SGML без какого-либо текста без комментариев. Поэтому, '<!--фу-->' является действительным комментарием, как и '<!--один-- --два-->', но '<!--1--2-->' не является.

С другой стороны, большинство авторов HTML не воспринимают комментарии как нечто иное, как текст, разделенный знаком '<!--' и '-->', что не совсем то же самое. Например, что-то вроде '<!------->' работает как действительный комментарий, если количество тире кратно четырем (!). В противном случае комментарий технически длится до следующего '--', который может находиться на другом конце документа. Из-за этого многие популярные браузеры полностью игнорируют спецификацию и реализуют то, что ожидают пользователи: комментарии, разделенные знаком '<!--' и '-->'.

До версии 1.9 Wget строго интерпретировал комментарии, что приводило к отсутствию ссылок на многих веб-страницах, которые нормально отображались в браузерах, но, к сожалению, содержали несоответствующие требованиям комментарии. Начиная с версии 1.9, Wget пополнил ряды клиентов, реализующих «наивные» комментарии, завершающие каждый комментарий при первом появлении '-->'.

Если по какой-либо причине вам нужен строгий анализ комментариев, используйте эту опцию, чтобы включить ее.

---

Следующая: [Статус выхода](#) , Предыдущая: [Параметры рекурсивного извлечения](#) , Вверх: [Вызов](#) [ [Содержание](#) ] [ [Индекс](#) ]

## 2.12 Опции рекурсивного принятия/отклонения

'-A *акклист* --принять *акклист*'

'-R *рейлист* --отклонить *рейлист*'

Укажите разделенные запятыми списки суффиксов имен файлов или шаблонов, которые следует принять или отклонить (см. [Типы файлов](#) ). Обратите внимание: если какой-либо из подстановочных символов '\*', '?', '[' или ']', появляется в элементе *acclist* или *rejlist* , он будет рассматриваться как шаблон, а не как суффикс. В этом случае вам придется заключить шаблон в кавычки, чтобы оболочка не могла его расширить, как в примере: -A "\*.mp3" или '-A '\*.mp3'.

'--accept-regex *URLregex*'

'--reject-regex *URLregex*'

Укажите регулярное выражение, чтобы принять или отклонить полный URL-адрес.

'--regex-type тип *регулярного выражения*'

Укажите тип регулярного выражения. Возможные типы: 'позикс' или 'рсге'. Обратите внимание: чтобы иметь возможность использовать 'рсге', wget должен быть скомпилирован с поддержкой libpcre.

'-D *список доменов*'

'--domains= *список-доменов*'

Установите домены для отслеживания. *список\_доменов* – это список доменов, разделенных запятыми. Обратите внимание, что он *не* включается '-ЧАС'.

'--exclude-domains *список-доменов*'

Укажите домены, за которыми *не* следует следить (см. [Связывающие хосты](#) ).

'--follow-ftp'

Переходите по FTP -ссылкам из HTML- документов. Без этой опции Wget будет игнорировать все FTP- ссылки.

'--follow-tags= *список*'

Wget имеет внутреннюю таблицу пар HTML- тег/атрибут, которую он учитывает при поиске связанных документов во время рекурсивного поиска. Однако если пользователь хочет, чтобы учитывалась только часть этих тегов, ему или ей следует указать такие теги в *списке*, разделенном запятыми , с помощью этой опции.

'--ignore-tags= *список*'

Это противоположность '--follow-теги' вариант. Чтобы пропустить определенные HTML- теги при рекурсивном поиске документов для загрузки, укажите их в *списке* , разделенном запятыми .



Раньше этот вариант был лучшим выбором для загрузки одной страницы и ее реквизитов с помощью командной строки, например:

```
wget --ignore-tags=a,area -H -k -K -r http:// сайт / документ
```

Однако автор этого варианта наткнулся на страницу с тегами вроде <LINK REL="home" HREF="/">и пришел к выводу, что указания тегов, которые следует игнорировать, недостаточно. Нельзя просто сказать Wget игнорировать <LINK>, потому что тогда таблицы стилей не будут загружены. Теперь лучшим вариантом для загрузки одной страницы и ее реквизитов является выделенный '--page-реквизиты' вариант.

### '--ignore-case'

Игнорировать регистр при сопоставлении файлов и каталогов. Это влияет на поведение опций -R, -A, -I и -X, а также на подстановку значений при загрузке с FTP-сайтов. Например, с помощью этой опции: '-A "\*.txt"'будет соответствовать 'файл1.txt', но и 'файл2.TXT', 'файл3.TxT', и так далее. Кавычки в примере предназначены для предотвращения расширения шаблона оболочкой.

### '-ЧАС'

### '--span-hosts'

Включите охват хостов при рекурсивном извлечении (см. [«Объединение хостов»](#)).

### '-Л'

### '--родственник'

Переходите только по относительным ссылкам. Полезно для получения определенной домашней страницы без каких-либо отвлекающих факторов, даже с тех же хостов (см. [Относительные ссылки](#) ).

### '-Я перечисляю'

### '--include-directory= список'

Укажите разделенный запятыми список каталогов, которым вы хотите следовать при загрузке (см. [Ограничения на основе каталогов](#) ). Элементы *списка* могут содержать подстановочные знаки.

### '-X список'

### '--exclude-directories= список'

Укажите список каталогов, разделенных запятыми, которые вы хотите исключить из загрузки (см. [Ограничения на основе каталогов](#) ). Элементы *списка* могут содержать подстановочные знаки.

'-например'

'--без родителя'

Никогда не переходите в родительский каталог при рекурсивном извлечении. Это полезная опция, поскольку она гарантирует, что будут загружены только файлы, *находящиеся ниже* определенной иерархии. Дополнительные сведения см. в [разделе «Ограничения на основе каталогов»](#) .

---

Предыдущий: [Параметры рекурсивного принятия/отклонения](#) , Вверх: [Вызов](#) [ [Содержание](#) ] [ [Индекс](#) ]

## 2.13 Статус выхода

Wget может вернуть один из нескольких кодов ошибок, если возникнут проблемы.

- 0  
Никаких проблем не возникло.
- 1  
Общий код ошибки.
- 2  
Ошибка анализа – например, при анализе параметров командной строки строка `'.wgetrc'` или `'.netrc'`...
- 3  
Ошибка ввода-вывода файла.
- 4  
Ошибка сети.
- 5  
Ошибка проверки SSL.
- 6  
Ошибка аутентификации имени пользователя и пароля.
- 7  
Ошибки протокола.
- 8  
Сервер выдал ответ об ошибке.

За исключением 0 и 1, коды выхода с меньшими номерами имеют приоритет над кодами с более высокими номерами при возникновении нескольких типов ошибок.

В версиях Wget до 1.12 статус выхода Wget имел тенденцию быть бесполезным и непоследовательным. Рекурсивные загрузки практически всегда возвращали 0 (успех), независимо от каких-либо возникших проблем, а нерекурсивные загрузки возвращали только статус, соответствующий последней попытке загрузки.

---

Далее: [По ссылкам](#) , Предыдущее: [Вызов](#) , Вверх: [Wget 1.24.5](#) [ [Содержание](#) ][ [Индекс](#) ]

## 3 Рекурсивная загрузка

GNU Wget способен перемещаться по частям Интернета (или по одному HTTP- или FTP-серверу), переходя по ссылкам и структуре каталогов. Мы называем это *рекурсивным поиском* или *рекурсией* .

С помощью URL-адресов HTTP Wget извлекает и анализирует HTML или CSS из заданного URL-адреса , получая файлы, на которые ссылается документ, с помощью разметки, например `<a href=URL>` или значений CSS URI , указанных с помощью `'hrefsrc URL()'` функциональная запись. Если только что загруженный файл также имеет тип `text/html`, `application/xhtml+xml` или `text/css`, он будет проанализирован и прослежен дальше.

Рекурсивный поиск содержимого HTTP и HTML / CSS осуществляется *в ширину* . Это означает, что Wget сначала загружает запрошенный документ, затем документы, связанные с этим документом, затем документы, связанные ими, и так далее. Другими словами, Wget сначала загружает документы на глубине 1, затем на глубине 2 и так далее до указанной максимальной глубины.

Максимальная *глубина* , на которую может опускаться извлечение, указывается с помощью `-l` вариант. Максимальная глубина по умолчанию составляет пять слоев.

При рекурсивном получении URL-адреса FTP Wget извлечет все данные из заданного дерева каталогов (включая подкаталоги до указанной глубины) на удаленном сервере, локально создавая его зеркальное изображение. Получение FTP также ограничено этим параметром. В отличие от рекурсии HTTP , рекурсия FTP выполняется в глубину. `depth`

По умолчанию Wget создаст локальное дерево каталогов, соответствующее тому, которое находится на удаленном сервере.

Рекурсивное извлечение может найти множество применений, наиболее важным из которых является зеркалирование. Это также полезно для WWW-презентаций и любых других возможностей, когда медленные сетевые соединения необходимо обойти, сохраняя файлы локально.

Следует предупредить, что рекурсивные загрузки могут перегрузить удаленные серверы. Из-за этого многие администраторы не одобряют их и могут запретить доступ к вашему сайту, если обнаружат очень быструю загрузку больших объемов контента. При загрузке с интернет-серверов рассмотрите возможность использования '-В' опция, позволяющая ввести задержку между доступами к серверу. Загрузка займет немного больше времени, но администратора сервера ваше хамство не смутит.

Конечно, рекурсивная загрузка может вызвать проблемы на вашем компьютере. Если оставить его работать без контроля, он может легко заполнить диск. При загрузке из локальной сети это также может занимать полосу пропускания системы, а также потреблять память и процессор.

Постарайтесь указать критерии, соответствующие типу загрузки, которую вы пытаетесь достичь. Если вы хотите загрузить только одну страницу, используйте '--page-реquisity' без какой-либо дополнительной рекурсии. Если вы хотите загружать файлы в один каталог, используйте '-наприме', чтобы избежать загрузки файлов из других каталогов. Если вы хотите загрузить все файлы из одного каталога, используйте '-л 1', чтобы убедиться, что глубина рекурсии никогда не превышает единицу. Дополнительные сведения об этом см. в [разделе «Следующие ссылки»](#).

Рекурсивный поиск следует использовать с осторожностью. Не говорите, что вас не предупреждали.

---

Следующая: [Метка времени](#) , Предыдущая: [Рекурсивная загрузка](#) , Вверх: [Wget 1.24.5](#) [ [Содержание](#) ][ [Индекс](#) ]

## 4 перехода по ссылкам

При рекурсивном извлечении нежелательно получать массу ненужных данных. Большую часть времени пользователи точно знают, что они хотят загрузить, и хотят, чтобы Wget переходил только по определенным ссылкам.

Например, если вы хотите скачать музыкальный архив с сайта 'fly.srk.feg.hr', вы не захотите загружать все домашние страницы, на которые ссылается неясная часть архива.

Wget обладает несколькими механизмами, которые позволяют вам точно настроить, по каким ссылкам он будет следовать.

- [Охватывающие хосты](#)
- [Типы файлов](#)
- [Ограничения на основе каталогов](#)
- [Относительные ссылки](#)
- [Переход по FTP-ссылкам](#)

Следующая: [Типы файлов](#) , Предыдущая: [Переход по ссылкам](#) , Вверх: [Переход по ссылкам](#)  
[ [Содержание](#) ][ [Индекс](#) ]

## 4.1 Объединение хостов

Рекурсивный поиск Wget обычно отказывается посещать хосты, отличные от того, который вы указали в командной строке. Это разумное значение по умолчанию; без него каждый запрос может превратить ваш Wget в уменьшенную версию Google.

Однако посещение разных хостов или *объединение хостов* иногда бывает полезным вариантом. Возможно, изображения подаются с другого сервера. Возможно, вы зеркалируете сайт, состоящий из страниц, связанных между тремя серверами. Возможно, у сервера есть два эквивалентных имени, и страницы HTML взаимозаменяемы.

### Переход к любому хосту---ЧАС'

'-ЧАСОпция ' включает охват хостов, что позволяет рекурсивному запуску Wget посещать любой хост, на который указывает ссылка. Если не будут применены достаточные критерии ограничения рекурсии, эти внешние хосты обычно будут связываться с еще большим количеством хостов, и так далее, пока Wget не перехватит гораздо больше данных, чем вы предполагали.

### Ограничить охват определенными доменами---Д'

'-ДОпция ' позволяет вам указать домены, которые будут отслеживаться, тем самым ограничивая рекурсию только хостами, принадлежащими этим доменам. Очевидно, это имеет смысл только в сочетании с '-ЧАС'. Типичным примером может быть загрузка содержимого 'www.example.com', но разрешая загрузку с 'images.example.com', и т. д.:

```
wget -rH -Dexample.com http://www.example.com/
```

Вы можете указать несколько адресов, разделив их запятой, например: '-Домен1.com,домен2.com'.

### Продолжайте загрузку с определенных доменов---exclude-domains'

Если есть домены, которые вы хотите исключить специально, вы можете сделать это с помощью '--exclude-domains', который принимает аргументы того же типа, что и '-Д', но *исключит* все перечисленные домены. Например, если вы хотите загрузить все хосты из 'foo.edu' домен, за исключением 'sunsite.foo.edu', вы можете сделать это следующим образом:

```
wget -rH -Dfoo.edu --exclude-domains sunsite.foo.edu \  
http://www.foo.edu/
```

Следующая: [Ограничения на основе каталогов](#) , Предыдущая: [Охватывающие узлы](#) , Вверх: [Переход по ссылкам](#) [ [Содержание](#) ][ [Индекс](#) ]

## 4.2 Типы файлов

При загрузке материалов из Интернета вам часто захочется ограничить получение только определенными типами файлов. Например, если вы заинтересованы в загрузке файлов GIF , вы не будете в восторге от множества документов PostScript, и наоборот.

Wget предлагает два варианта решения этой проблемы. В каждом описании опции указаны короткое имя, длинное имя и эквивалентная команда в `wgetrc`.

'-Акклист'

'--принять *список*'

'принять = *внести в список*'

'--ассепт-regex *URLregex*'

'принять-регулярное выражение = *URLregex*'

Аргумент в пользу '--приниматьПараметр ' представляет собой список файловых суффиксов или шаблонов, которые Wget загрузит во время рекурсивного извлечения. Суффикс — это конечная часть файла и состоит из «обычных» букв, например: «гифка' или '.jpg'. Соответствующий шаблон содержит подстановочные знаки, подобные оболочке, например: 'книги\*' или 'железный\*196[0-9]\*'.

Итак, указав 'wget - GIF, JPG' заставит Wget загружать только файлы, заканчивающиеся на 'гифка' или 'jpg', то есть GIF и JPEG . С другой стороны, 'wget -A "zelazny\*196[0-9]\*"' будет загружать только файлы, начинающиеся с 'железный' и содержащий цифры от 1960 до 1969 где угодно внутри. Посмотрите руководство вашей оболочки, чтобы узнать, как работает сопоставление с образцом.

Конечно, любое количество суффиксов и шаблонов можно объединить в список, разделенный запятыми, и передать в качестве аргумента '-A'.

Аргумент в пользу '--ассепт-regexПараметр ' — это регулярное выражение, которое сопоставляется с полным URL-адресом.

'-R *список треков*'

'--reject *rejlist*'

'отклонить = *повторный список*'

## '--reject-regex *URLregex*'

### 'отклонить-регулярное выражение = *регулярное выражение*'

'--отклонять' опция работает так же, как '--принимать', только логика у него обратная; Wget загрузит все файлы, *кроме* тех, которые соответствуют суффиксам (или шаблонам) в списке.

Итак, если вы хотите загрузить всю страницу, за исключением громоздких файлов MPEG и .AU, вы можете использовать 'wget -R миль на галлон, MPEG, AU'.

Аналогично, чтобы скачать все файлы, кроме тех, которые начинаются с 'Бьорк', использовать 'wget -R "Бьорк\*"'. Кавычки предназначены для предотвращения расширения оболочки.

Аргумент в пользу '--assert-regexПараметр' – это регулярное выражение, которое сопоставляется с полным URL-адресом.

'-A' и '-P' параметры можно комбинировать для достижения еще более точной настройки файлов для извлечения. Например: 'wget -A "\*zelazny\*" -R .ps' загрузит все файлы, имеющие 'железный' как часть их имени, но *не* файлов PostScript.

Обратите внимание, что эти два параметра не влияют на загрузку файлов HTML (определяется параметром '.htm' или '.html' префикс имени файла). Такое поведение может быть нежелательным для всех пользователей и может быть изменено в будущих версиях Wget.

Также обратите внимание, что строки запроса (строки в конце URL-адреса, начинающиеся с вопросительного знака ('?')) не включаются в имя файла для правил принятия/отклонения, хотя на самом деле они влияют на имя, выбранное для локального файла. Ожидается, что будущая версия Wget предоставит возможность сопоставления строк запроса.

Наконец, стоит отметить, что списки принятия/отклонения сопоставляются *дважды* с загруженными файлами: один раз с частью имени файла URL-адреса, чтобы определить, следует ли загружать файл в первую очередь; затем, после того как он был принят и успешно загружен, имя локального файла также проверяется по спискам принятия/отклонения, чтобы определить, следует ли его удалить. Обоснование заключалось в том, что, поскольку «.htm' и '.html' файлы всегда загружаются независимо от правил принятия/отклонения, их следует удалить *после* загрузки и проверить на наличие ссылок, если они соответствуют спискам принятия/отклонения. Однако это может привести к неожиданным результатам, поскольку локальные имена файлов могут отличаться от исходных имен файлов URL-адресов следующими способами, каждый из которых может изменить соответствие правилу принятия/отклонения:

- Если локальный файл уже существует и '--по-каталоги' был указан, к исходному имени будет добавлен числовой суффикс.

- Если '--adjust-расширение' было указано, локальное имя файла может иметь '.html', добавлено к нему. Если Wget вызывается с помощью '-E -A.php', имя файла, например 'index.php' будет принято, но при загрузке будет называться 'index.php.html', который больше не соответствует, поэтому файл будет удален.
- Строки запроса не участвуют в сопоставлении URL-адресов, но включаются в локальные имена файлов и, таким образом, *способствуют* сопоставлению имен файлов.

Такое поведение также считается нежелательным и может измениться в будущей версии Wget.

---

Следующая: [Относительные ссылки](#) , Предыдущая: [Типы файлов](#) , Вверх: [Переход по ссылкам](#)  
[ [Содержание](#) ] [ [Указатель](#) ]

## 4.3 Ограничения на основе каталогов

Независимо от других средств перехода по ссылкам, часто бывает полезно установить ограничение на получение файлов в зависимости от каталогов, в которых эти файлы расположены. Для этого может быть много причин – домашние страницы могут быть организованы в разумной структуре каталогов. ; или некоторые каталоги могут содержать бесполезную информацию, например/cgi-bin или /устройствокаталоги.

Wget предлагает три различных варианта удовлетворения этого требования. В каждом описании опции указаны короткое имя, длинное имя и эквивалентная команда в.wgetrc.

**'-Я перечисляю'**

**'--включить список'**

**'include\_directories = список'**

'-ЯОпция ' принимает разделенный запятыми список каталогов, включенных в поиск. Любые другие каталоги будут просто игнорироваться. Каталоги являются абсолютными путями.

Итак, если вы хотите скачать с 'http://хост/люди/бозо/' переходя только по ссылкам на коллег бозо в/люди каталог и поддельные скрипты в/cgi-bin, вы можете указать:

```
wget -I /people,/cgi-bin http://host/people/bozo/
```

**'-X список'**

**'--исключить список'**

**'исключать\_каталоги = список'**



'-ИКС' опция является полной противоположностью '-Я' – это список каталогов, *исключенных* из загрузки. Например, если вы не хотите, чтобы Wget загружал файлы `c/cgi-bin` каталог, укажите '-X /cgi-bin' в командной строке.

То же, что и с '-А'/'-Р', эти два параметра можно объединить, чтобы получить более точную настройку загрузки подкаталогов. Например, если вы хотите загрузить все файлы из/пабиерархия, за исключением /паб/бесполезный, указать '-I/паб -X/паб/бесполезный'.

'-например'

'--без родителя'

'no-parent = включено'

Самый простой и часто очень полезный способ ограничения каталогов – запретить получение ссылок, которые относятся к иерархии *выше* начального каталога, т.е. запрет восхождения на родительский каталог/каталоги.

'--без родителя' опция (коротко '-например') полезно в этом случае. Его использование гарантирует, что вы никогда не покинете существующую иерархию. Предположим, вы запускаете Wget с:

```
wget -r --no-parent http://somehost/~luzer/my-archive/
```

Вы можете быть уверены, что ни одна из ссылок на /~домашняя страница-его-девушек/или/~luzer/все-мои-трег/будет следовать. Будет скачан только тот архив, который вас интересует. По сути, '--без родителя' похож на '-I/~luzer/мой-архив', только он обрабатывает перенаправления более разумным образом.

**Обратите внимание**, что для HTTP (и HTTPS) косая черта в конце очень важна для '--без родителя'. В HTTP нет понятия «каталог» – Wget полагается на то, что вы указываете, что является каталогом, а что нет. В 'http://foo/бар/', Wget рассмотрит 'бар' быть каталогом, находясь в 'http://foo/бар' (без косой черты), 'бар' будет считаться именем файла (поэтому '--без родителя' было бы бессмысленно, поскольку его родительским элементом является '/').

---

Далее: [Переход по FTP-ссылкам](#), Предыдущий: [Ограничения на основе каталога](#), Вверх: [Переход по ссылкам](#) [ [Содержание](#) ][ [Индекс](#) ]

## 4.4 Относительные ссылки

Когда '-Л' включен, всегда выполняются только относительные ссылки. Относительные ссылки здесь определяются как те, которые не относятся к корню веб-сервера. Например, эти ссылки являются относительными:

```
<a href="foo.gif">  
<a href="foo/bar.gif">  
<a href="../foo/bar.gif">
```

Эти ссылки не являются относительными:

```
<a href="/foo.gif">  
<a href="/foo/bar.gif">  
<a href="http://www.example.com/foo/bar.gif">
```

Использование этой опции гарантирует, что рекурсивное извлечение не будет охватывать хосты, даже без '-ЧАС'. В простых случаях это также позволяет загрузкам «просто работать» без необходимости конвертировать ссылки.

Эта опция, вероятно, не очень полезна и может быть удалена в будущем выпуске.

---

Предыдущий: [Относительные ссылки](#) , Вверх: [Переход по ссылкам](#) [ [Содержание](#) ][ [Указатель](#) ]

## 4.5 Переход по FTP-ссылкам

Правила FTP несколько специфичны, как и положено. FTP -ссылки в HTML- документы часто включаются в справочных целях, и часто их неудобно загружать по умолчанию.

Чтобы переходить по FTP -ссылкам из HTML- документов, необходимо указать '--follow-ftp' вариант. После этого FTP -ссылки будут охватывать хосты независимо от '-ЧАС' параметр. Это логично, поскольку FTP -ссылки редко указывают на тот же хост, где находится HTTP -сервер. По тем же причинам '-Л'Параметры не влияют на такие загрузки. С другой стороны, принятие домена ('-Д') и правила суффиксов ('-А' и '-Р') применяются нормально.

Также обратите внимание, что последующие ссылки на FTP- каталоги не будут в дальнейшем получаться рекурсивно.

---

Следующая: [Файл запуска](#) , Предыдущая: [По ссылкам](#) , Вверх: [Wget 1.24.5](#) [ [Содержание](#) ][ [Индекс](#) ]

## 5 Временные метки

Одним из наиболее важных аспектов зеркалирования информации из Интернета является обновление ваших архивов.

Скачивать весь архив снова и снова только для того, чтобы заменить несколько изменившихся файлов – это дорого, как с точки зрения траты трафика и денег, так и времени на обновление. Вот почему все инструменты зеркалирования предлагают возможность постепенного обновления.

Такой механизм обновления означает, что удаленный сервер сканируется в поисках *новых* файлов. Вместо старых будут загружены только новые файлы.

Файл считается новым, если соблюдено одно из этих двух условий:

1. Файл с таким именем еще не существует локально.
2. Файл с таким именем существует, но удаленный файл был изменен позже, чем локальный файл.

Для реализации этого программе необходимо знать время последней модификации как локальных, так и удаленных файлов. Мы называем эту информацию *меткой времени* файла.

Отметка времени в GNU Wget включается с помощью '--метка времени' ('-H') или через `timestamping` = опдирективу в `.wgetrc`. При использовании этой опции для каждого файла, который он собирается загрузить, Wget будет проверять, существует ли локальный файл с таким же именем. Если да, и удаленный файл не новее, Wget не будет его загружать.

Если локальный файл не существует или размеры файлов не совпадают, Wget загрузит удаленный файл независимо от того, что указано в временных метках.

- [Использование отметок времени](#)
- [Внутренние метки времени HTTP](#)
- [Внутренние метки времени FTP](#)

---

Далее: [Внутреннее устройство отметки времени HTTP](#) , Предыдущее: [Отметка времени](#) ,  
Вверх: [Отметка времени](#) [ [Содержание](#) ][ [Индекс](#) ]

## 5.1 Использование отметок времени

Использовать временные метки просто. Допустим, вы хотите загрузить файл, чтобы в нем сохранялась дата изменения.

```
wget -S http://www.gnu.ai.mit.edu/
```

Простой пример `ls -l` показывает, что временная метка локального файла соответствует состоянию Last-Modified заголовка, возвращенному сервером. Как видите, информация о временных метках сохраняется локально, даже без '-N' (по крайней мере, для HTTP).

Через несколько дней вы захотите, чтобы Wget проверил, изменился ли удаленный файл, и загрузил его, если да.

```
wget -N http://www.gnu.ai.mit.edu/
```

Wget запросит у сервера дату последнего изменения. Если локальный файл имеет ту же временную метку, что и сервер, или более новую, удаленный файл не будет загружен повторно. Однако, если удаленный файл более новый, Wget продолжит его получение.

То же самое касается и FTP. Например:

```
wget "ftp://ftp.ifi.uio.no/pub/emacs/gnus/*"
```

(Кавычки вокруг этого URL-адреса предназначены для предотвращения попыток оболочки интерпретировать '\*'.)

После загрузки список локальных каталогов покажет, что временные метки совпадают с метками на удаленном сервере. Повторный ввод команды с помощью '-N' заставит Wget повторно загрузить *только* те файлы, которые были изменены с момента последней загрузки.

Если вы хотите зеркалировать архив GNU каждую неделю, вам следует использовать следующую команду: еженедельно:

```
wget --timestamping -r ftp://ftp.gnu.org/pub/gnu/
```

Обратите внимание, что отметка времени будет работать только для файлов, для которых сервер предоставляет метку времени. Для HTTP это зависит от получения Last-Modified заголовка. Для FTP это зависит от получения списка каталогов с датами в формате, который Wget может проанализировать (см. [Внутренние метки времени FTP](#)).

---

Следующая: [Внутреннее устройство отметки времени FTP](#) , Предыдущая: [Использование отметки времени](#) , Вверх: [Добавление отметки времени](#) [ [Содержание](#) ][ [Индекс](#) ]

## 5.2 Внутреннее устройство HTTP-меток времени

Метка времени в HTTP реализуется путем проверки Last-Modified заголовка. Если вы хотите получить файл `foo.html` через HTTP Wget проверит, `foo.html` существует локально. Если это не так, `foo.html` будут извлечены безоговорочно.

Если файл существует локально, Wget сначала проверит его локальную метку времени (аналогично тому, как `ls -l`он проверяется), а затем отправит HEAD запрос на удаленный сервер, запрашивая информацию об удаленном файле.

Заголовок Last-Modified проверяется, чтобы определить, какой файл был изменен совсем недавно (что делает его «более новым»). Если удаленный файл более новый, он будет загружен; если он старше, Wget сдастся.

Когда `'--backup-преобразовано' ('-K')` указывается вместе с `'-H'`, файл сервера `'Икс'` сравнивается с локальным файлом `'X.ориг'`, если он существует, вместо сравнения с локальным файлом `'Икс'`, который всегда будет отличаться, если он был преобразован `'--convert-links' ('-к')`.

Возможно, отметка времени HTTP If-Modified-Since должна быть реализована с использованием запроса.

---

Предыдущий: [Внутренние устройства отметки времени HTTP](#) , Вверх: [Отметки времени](#) [ [Содержание](#) ] [ [Индекс](#) ]

## 5.3 Внутренние устройства отметки времени FTP

Теоретически отметка времени FTP работает почти так же, как HTTP , только FTP не имеет заголовков – отметки времени необходимо получать из списков каталогов.

Если загрузка по FTP является рекурсивной или использует подстановку, Wget будет использовать команду FTP LIST , чтобы получить список файлов для каталога, содержащего нужные файлы. Он попытается проанализировать листинг, рассматривая его как `ls -l`выходные данные Unix, извлекая временные метки. Остальное точно так же, как и для HTTP . Обратите внимание, что при получении отдельных файлов с FTP- сервера без использования подстановки или рекурсии файлы списков не будут загружены (и, следовательно, файлы не будут иметь отметку времени), если только `'-H'` - уточняется.

Предположение о том, что каждый список каталогов представляет собой список в стиле Unix, может показаться чрезвычайно ограничивающим, но на практике это не так, поскольку многие не-Unix FTP -серверы используют формат списка Unixoid, потому что большинство (все?) клиентов его понимают. Имейте в виду, что RFC959 не определяет стандартного способа получения списка файлов, не говоря уже о временных метках. Мы можем только надеяться, что будущий стандарт определит это.

Еще одно нестандартное решение включает в себя использование MDTMкоманды, поддерживаемой некоторыми FTP- серверами (в том числе популярным `wu-ftp`), которая возвращает точное время указанного файла. Wget может поддерживать эту команду в будущем.

---

Следующая: [Примеры](#) , Предыдущая: [Временная метка](#) , Вверх: [Wget 1.24.5](#) [ [Содержание](#) ][ [Индекс](#) ]

## 6 Файл запуска

Как только вы узнаете, как изменить настройки Wget по умолчанию с помощью аргументов командной строки, вы, возможно, захотите сделать некоторые из этих настроек постоянными. Вы можете сделать это удобным способом, создав файл запуска `Wget:.wgetrc`.

Кроме `.wgetrc`— «основной» файл инициализации, удобно иметь специальное средство для хранения паролей. Таким образом, Wget читает и интерпретирует содержимое `$HOME/.netrc`, если найдет. Ты можешь найти `.netrc` в руководствах вашей системы.

Wget читает `.wgetrc` при запуске, распознавая ограниченный набор команд.

- [Расположение Wgetrc](#)
- [Синтаксис Wgetrc](#)
- [Команды Wgetrc](#)
- [Пример Wgetrc](#)

---

Далее: [Синтаксис Wgetrc](#) , Предыдущий: [Файл запуска](#) , Вверх: [Файл запуска](#) [ [Содержание](#) ][ [Индекс](#) ]

### 6.1 Расположение Wgetrc

При инициализации Wget будет искать *глобальный* файл запуска. `/usr/local/etc/wgetrc` по умолчанию (или какой-либо префикс, отличный от `/usr/` локальный, если там не был установлен Wget) и читать команды оттуда, если он существует.

Затем он будет искать файл пользователя. Если переменная среды `WGETRC` установлена, Wget попытается загрузить этот файл. В противном случае дальнейшие попытки предприниматься не будут.

Если `WGETRC` не установлено, Wget попытается загрузить `$HOME/.wgetrc`.

Тот факт, что пользовательские настройки загружаются после общесистемных, означает, что в случае коллизии пользовательский `wgetrc` *переопределяет* общесистемный `wgetrc`

(в/usr/local/etc/wgetrcпо умолчанию). Фашистские админы, прочь!

Следующая: [Команды Wgetrc](#) , Предыдущая: [Расположение Wgetrc](#) , Вверх: [Файл запуска](#) [ [Содержание](#) ][ [Индекс](#) ]

## 6.2 Синтаксис Wgetrc

Синтаксис команды wgetrc прост:

переменная = значение

Переменная также *будет* называться *командой* . Допустимые *значения* различны для разных команд.

Команды нечувствительны к регистру, подчеркиванию и минусам. Таким образом 'Дир\_\_Префикс', 'Префикс НАПРАВЛЕНИЯ' и 'префикс директории' одинаковы. Пустые строки, строки, начинающиеся с '#' и строки, содержащие только пробелы, отбрасываются.

Команды, ожидающие список, разделенный запятыми, очистят список при пустой команде. Итак, если вы хотите сбросить список отклонений, указанный в глобальныхwgetrc, вы можете сделать это с помощью:

ОТКЛОНИТЬ =

Далее: [Пример Wgetrc](#) , Предыдущее: [Синтаксис Wgetrc](#) , Вверх: [Файл запуска](#) [ [Содержание](#) ][ [Индекс](#) ]

## 6.3 Команды Wgetrc

Полный набор команд приведен ниже. Юридические значения указаны после '"="'. Простые логические значения можно установить или отключить с помощью 'на' и 'выключенный' или '1' и '0'.

Некоторые команды принимают псевдопроизвольные значения. Значения *адреса* могут быть именами хостов или IP-адресами, состоящими из четырех точек. *n* может быть любым положительным целым числом или 'инфа' для бесконечности, где это возможно. *строковые* значения могут быть любой непустой строкой.

Большинство этих команд имеют прямые эквиваленты в командной строке. Кроме того, любую команду wgetrc можно указать в командной строке с помощью '--выполнять'переключатель (см. [Основные параметры запуска](#) .)

**принять/отклонить = строка**

Такой же как '-A'/'-P' (см. [Типы файлов](#) ).

**add\_hostdir = вкл/выкл**

Включить/отключить имена файлов с префиксом хоста. '-nH'отключает его.

**Ask\_password = вкл/выкл**

Запрашивать пароль для каждого установленного соединения. Невозможно указать, когда '--пароль' используется, потому что они взаимоисключающие. Эквивалент '--ask-пароль'.

**auth\_no\_challenge = вкл/выкл**

Если указана эта опция, Wget будет отправлять базовую информацию аутентификации HTTP (имя пользователя и пароль в виде открытого текста) для всех запросов. Видеть '--auth-no-challenge'.

**фон = вкл/выкл**

Включить/отключить переход в фоновый режим – то же, что и '-b(что позволяет это сделать).

**backup\_converted = вкл/выкл**

Включить/выключить сохранение предварительно конвертированных файлов с суффиксом '.orig'-такой же как '-K(что позволяет это сделать).

**резервные копии = количество**

Используйте до *количества* резервных копий для файла. Резервные копии чередуются путем добавления инкрементного счетчика, который начинается с '1'. По умолчанию '0'.

**база = строка**

Учитывайте относительные URL- адреса во входных файлах (указанные через 'вход'команда или '--input-file'/'-я'опция вместе с 'Force\_html' или '--force-html') как относительно *строки* – то же самое, что '--base= строка'.

**привязка\_адрес = адрес**

Привязка к *адресу* , например '--bind-address= адрес'.

**ca\_certificate = файл**



Установите для файла пакета центра сертификации *file* . Такой же как '--ca-certificate= *файл*'.

### **ca\_directory = *каталог***

Установите каталог, используемый для центров сертификации. Такой же как '--ca-directory= *каталог*'.

### **кэш = *вкл/выкл***

Если установлено значение «выключено», запретить кэширование сервера. Смотрите '--no-кэш' вариант.

### **сертификат = *файл***

Установите имя файла сертификата клиента *file* . Такой же как '--certificate= *файл*'.

### **тип\_сертификата = *строка***

Укажите тип сертификата клиента, допустимые значения: 'ПЕМ' (по умолчанию) и 'ТО(он же ASN1)'. Такой же как '--certificate-type= *строка*'.

### **check\_certificate = *вкл/выкл***

Если для этого параметра установлено значение «выкл.», сертификат сервера не проверяется на соответствие указанным центрам сертификации клиента. По умолчанию установлено «включено». Такой же как '--check-сертификат'.

### **Connect\_timeout = *п***

Установите тайм-аут подключения – то же, что и '--connect-таймаут'.

### **content\_disposition = *вкл/выкл***

Включить распознавание (нестандартного) 'Содержание-Расположение' HTTP-заголовков – если установлено значение 'на', такой же как '--content-disposition'.

### **Trust\_server\_names = *вкл/выкл***

Если этот параметр включен, имя локального файла создается на основе URL-адресов перенаправления, а не исходных URL-адресов.

### **продолжить = *вкл/выкл***

Если установлено значение «включено», принудительное продолжение ранее существовавших частично извлеченных файлов. Видеть '-с', прежде чем установить его.

### **Convert\_links = *вкл/выкл***

Преобразуйте неотносительные ссылки локально. Такой же как '-к'.

**куки = вкл/выкл**

Если установлено значение «Выкл.», запретить использование файлов cookie. Смотрите '--печенье' вариант.

**Cut\_dirs = л**

Игнорируйте *компоненты* удаленного каталога. Эквивалент '--cut-dirs= н'.

**отладка = вкл/выкл**

Режим отладки, такой же, как '-д'.

**страница\_по умолчанию = строка**

Имя страницы по умолчанию – то же, что и '--default-page= строка'.

**delete\_after = вкл/выкл**

Удалить после загрузки – то же самое, что '--delete-после'.

**dir\_prefix = строка**

Верхняя часть дерева каталогов – то же, что '-Р строка'.

**директория = вкл/выкл**

Включение и выключение dirstruct – то же самое, что и '-Икс' или '-nd', соответственно.

**dns\_cache = вкл/выкл**

Включите/выключите кэширование DNS. Поскольку кэширование DNS включено по умолчанию, эта опция обычно используется для его отключения и эквивалентна '--no-dns-кэш'.

**dns\_timeout = л**

Установите тайм-аут DNS – такой же, как '--dns-таймаут'.

**домены = строка**

Такой же как '-Д(см. [раздел «Связывающие хосты»](#)).

**точка\_байт = л**

Укажите количество байтов, «содержащихся» в точке, как оно будет видно при извлечении (по умолчанию 1024). Вы можете постфиксировать значение с помощью 'к' или 'м', представляющий килобайты и мегабайты соответственно. С помощью настроек точек вы можете настроить извлечение точек в соответствии со своими потребностями или использовать предопределенные *стили* (см. [«Параметры загрузки»](#)).

**dot\_spacing = л**

Укажите количество точек в одном кластере (по умолчанию 10).

**dots\_in\_line = *n***

Укажите количество точек, которые будут печататься в каждой строке на протяжении всего поиска (50 по умолчанию).

**egd\_file = *файл***

Используйте *строку* в качестве имени файла сокета EGD. Такой же как '`--egd-file=файл`'.

**исключить\_каталоги = *строка***

Укажите список каталогов, разделенных запятыми, которые вы хотите исключить из загрузки – аналогично '`-X строка`' (см. [Ограничения на основе каталогов](#)).

**исключать\_домены = *строка***

Такой же как '`--exclude-domains= строка`' (см. [раздел «Связывающие хосты»](#)).

**Follow\_ftp = *вкл/выкл***

Переходите по FTP -ссылкам из HTML- документов – то же самое, что '`--follow-ftp`'.

**Follow\_tags = *строка***

При рекурсивном извлечении следуйте только определенным тегам HTML , например:--  
`follow-tags= строка`'.

**Force\_html = *вкл/выкл***

Если установлено значение «on», принудительно рассматривать имя входного файла как документ HTML – то же самое, что и «-Ф».

**ftp\_password = *строка***

Установите для пароля FTP значение *string* . Без этой настройки пароль по умолчанию равен '`-wget@`', что является полезным значением по умолчанию для анонимного доступа по FTP .

Эта команда называлась `passwd`до Wget 1.10.

**ftp\_proxy = *строка***

Используйте *строку* в качестве FTP- прокси вместо той, которая указана в среде.

**ftp\_user = *строка***

Установите для пользователя FTP значение *string* .

Эта команда называлась `login`до Wget 1.10.

**glob = *вкл/выкл***

Включить/выключить подстановку – то же, что и '--glob' и '--no-glob'.

**заголовок = строка**

Определите заголовок для HTTP-загрузок, например, используя '--header= строка'.

**сжатие = строка**

Выберите тип сжатия, который будет использоваться. Юридические ценности авто' (по умолчанию), 'gzip', и 'никто'. Такой же как '--compression= строка'.

**корректировка\_расширения = вкл/выкл**

Добавить '.html' расширение 'текст/html' или 'приложение/xhtml+xml' файлы, в которых его нет, '.css' расширение 'текст/css' файлы, в которых его нет, и '.br', '.C', '.zlib' или '.gz' в сжатые файлы типа '-И'. Ранее назывался 'html\_extension' (все еще приемлемо, но устарело).

**http\_keep\_alive = вкл/выкл**

Включите или отключите функцию поддержания активности (по умолчанию включено). Выключение эквивалентно '--no-http-keep-alive'.

**http\_пароль = строка**

Установите пароль HTTP , эквивалентный '--http-пароль= строка'.

**http\_проху = строка**

Используйте строку в качестве HTTP- прокси вместо той, которая указана в среде.

**http\_user = строка**

Установите для пользователя HTTP строку , эквивалентную '--http-user= строка'.

**https\_only = вкл/выкл**

В рекурсивном режиме отслеживаются только ссылки HTTPS (по умолчанию отключено).

**https\_проху = строка**

Используйте строку в качестве прокси-сервера HTTPS вместо той, которая указана в среде.

**ignore\_case = вкл/выкл**

Если этот параметр включен, файлы и каталоги сопоставляются без учета регистра; такой же как '--ignore-case'.

**ignore\_length = вкл/выкл**

Если включено, Content-Length заголовок игнорируется; такой же как '--ignore-длина'.

**ignore\_tags = строка**

Игнорируйте определенные теги HTML при рекурсивном извлечении, например '--ignore-tags= строка'.

**include\_directories = строка**

Укажите разделенный запятыми список каталогов, которым вы хотите следовать при загрузке — аналогично '-Я *натягиваю*'.

**ири = вкл/выкл**

Если этот параметр включен, включите поддержку интернационализованного URI (IRI); такой же как '--является'.

**inet4\_only = вкл/выкл**

Принудительное подключение к адресам IPv4, по умолчанию отключено. Вы можете поместить это в глобальный файл инициализации, чтобы отключить попытки Wget разрешить и подключиться к хостам IPv6. Доступно только в том случае, если Wget был скомпилирован с поддержкой IPv6. Такой же как '--inet4-только' или '-4'.

**inet6\_only = вкл/выкл**

Принудительное подключение к адресам IPv6, по умолчанию отключено. Доступно только в том случае, если Wget был скомпилирован с поддержкой IPv6. Такой же как '--inet6-only' или '-6'.

**ввод = файл**

Прочитайте URL -адреса из *строки* , например '-я *файл*'.

**Keep\_session\_cookies = вкл/выкл**

Если указано, вызывает 'save\_cookies = включено', чтобы также сохранить файлы cookie сеанса. Видеть '--keep-session-cookies'.

**limit\_rate = ставка**

*Ограничьте скорость загрузки до уровня* , не превышающего байтов в секунду. Такой же как '--limit-rate= ставка'.

**load\_cookies = файл**

Загрузить файлы cookie из *файла* . Видеть '--load-cookies *файл*'.

**local\_encoding = кодировка**

Заставьте Wget использовать *кодировку* в качестве системной кодировки по умолчанию. Видеть '--local-кодировка'.

**файл журнала = файл**

Установите файл журнала в *файл* , так же, как '-о *файл*'.

**max\_redirect = *число***

Указывает максимальное количество перенаправлений, которым должен следовать ресурс. Видеть '--max-redirect= *число*'.

**зеркало = вкл/выкл**

Включите/выключите зеркальное отображение. Такой же как '-м'.

**netrc = вкл/выкл**

Включите или выключите чтение netrc.

**no\_clobber = вкл/выкл**

Такой же как '-NC'.

**no\_parent = вкл/выкл**

Запретить получение данных за пределами иерархии каталогов, например '--без родителя' (см. [Ограничения на основе каталогов](#) ).

**no\_proxy = *строка***

Используйте *строку* в качестве списка доменов, разделенных запятыми, чтобы избежать загрузки прокси, вместо того, который указан в среде.

**выходной\_документ = *файл***

Установите имя выходного файла – такое же, как '-O *файл*'.

**page\_реквизиты = вкл/выкл**

Загрузите все вспомогательные документы, необходимые для правильного отображения одной HTML- страницы – так же, как '-п'.

**пассивный\_ftp = вкл/выкл**

Измените настройку пассивного FTP , эквивалентную '--passive-ftp' вариант.

**пароль = *строка***

Укажите *строку* пароля для получения файлов по FTP и HTTP . Эту команду можно переопределить с помощью 'ftp\_password' и 'http\_пароль' команда для FTP и HTTP соответственно.

**post\_data = *строка***

Используйте POST в качестве метода для всех HTTP-запросов и отправляйте *строку* в теле запроса. Такой же как '--post-data= *строка*'.

**post\_file = *файл***

Используйте POST в качестве метода для всех HTTP-запросов и отправляйте содержимое *файла* в теле запроса. Такой же как '--post-file= *файл*'.

**предпочтительный\_семейство = нет/IPv4/IPv6**

Если вам предоставлен выбор из нескольких адресов, сначала подключитесь к адресам с указанным семейством адресов. Порядок адресов, возвращаемый DNS, по умолчанию используется без изменений. Такой же как '--prefer-семья', в котором подробно описано, почему это полезно.

**частный\_ключ = файл**

Установите файл закрытого ключа в *file* . Такой же как '--private-key= файл'.

**Private\_key\_type = строка**

Укажите тип закрытого ключа, допустимые значения: 'ПЕМ' (по умолчанию) и 'ТО(он же ASN1)'. Такой же как '--private-type= строка'.

**прогресс = строка**

Установите тип индикатора прогресса. Юридические типы 'точка' и 'бар'. Эквивалент '--progress= строка'.

**каталог\_протокола = вкл/выкл**

Если установлено, используйте имя протокола в качестве компонента каталога имен локальных файлов. Такой же как '--протокол-каталоги'.

**ргоху\_password = строка**

Установите пароль аутентификации прокси-сервера в *строку* , например '--ргоху-password= строка'.

**ргоху\_user = строка**

Установите имя пользователя для аутентификации прокси-сервера в *строку* , например '--ргоху-user= строка'.

**тихо = вкл/выкл**

Тихий режим – то же, что и '-q'.

**квота = квота**

Укажите квоту загрузки, которую полезно проставить в глобальном wgetrc. Если указана квота загрузки, Wget прекратит получение после того, как сумма загрузки превысит квоту. Квота может быть указана в байтах (по умолчанию), килобайтах 'к' добавлено) или мегабайты ('м', добавлено). Таким образом 'квота = 5 млн.' установит квоту в 5 мегабайт. Обратите внимание, что файл запуска пользователя имеет приоритет над системными настройками.

**случайный\_файл = файл**

Используйте *файл* как источник случайности в системах, где отсутствует /dev/случайный.

**случайное\_ожидание = вкл/выкл**

Включите или отключите случайное время ожидания между запросами. Такой же как `'--random-wait'`.

**read\_timeout = *n***

Установите таймаут чтения (и записи) – такой же, как `'--read-timeout= n'`.

**повторный уровень = *n***

Уровень рекурсии (глубина) – то же, что `'-ln'`.

**рекурсивный = вкл/выкл**

Рекурсивное включение/выключение – то же, что `'-r'`.

**ссылка = *строка***

Установить HTTP 'Ссылаться:' заголовок, как `'--referer= строка'`. (Обратите внимание, что именно люди, написавшие спецификацию HTTP, неправильно написали слово «реферер».)

**относительный\_only = вкл/выкл**

Переходите только по относительным ссылкам – то же самое, что `'-L'` (см. [Относительные ссылки](#)).

**Remote\_encoding = *кодировка***

Заставьте Wget использовать *кодировку* в качестве кодировки удаленного сервера по умолчанию. Видеть `'--remote-кодирование'`.

**удалить\_листинг = вкл/выкл**

Если этот параметр включен, удалите списки FTP, загруженные Wget. Отключение его аналогично `'--no-remove-листинг'`.

**strict\_file\_names = unix/windows**

Ограничьте имена файлов, генерируемые Wget, из URL-адресов. Видеть `'--restrict-file-names'` для более подробного описания.

**retr\_symlinks = вкл/выкл**

Если этот параметр включен, извлекать символические ссылки, как если бы они были обычными файлами; такой же как `'--retr-symlinks'`.

**retry\_connrefused = вкл/выкл**

Если этот параметр включен, считайте «отказ в соединении» временной ошибкой – так же, как `'--retry-connrefused'`.

**роботы = вкл/выкл**



Укажите, соблюдается ли соглашение `robots` в Wget, по умолчанию «включено». Этот переключатель управляет как `robots.txt` 'не следует' аспект спецификации. Дополнительные сведения об этом см. в [разделе Исключение роботов](#) . Прежде чем выключать эту функцию, убедитесь, что вы знаете, что делаете.

**save\_cookies = файл**

Сохранять файлы cookie в *файл* . Такой же как `--save-cookies файл` .

**save\_headers = вкл/выкл**

Такой же как `--save-headers` .

**Secure\_protocol = строка**

Выберите безопасный протокол, который будет использоваться. Юридические ценности авто' (по умолчанию), 'SSLv2', 'SSLv3', и 'TLSv1'. Такой же как `--secure-protocol= строка` .

**server\_response = вкл/выкл**

Выберите, печатать ли HTTP и FTP -серверов – то же самое, что и `-C` .

**show\_all\_dns\_entries = вкл/выкл**

Когда DNS-имя разрешено, покажите все IP-адреса, а не только первые три.

**span\_hosts = вкл/выкл**

Такой же как `-ЧАС` .

**паук = вкл/выкл**

Такой же как `--паук` .

**strict\_comments = вкл/выкл**

Такой же как `--strict-комментарии` .

**таймаут = л**

Установите для всех применимых значений тайм-аута значение *л* . , так же, как `-Т л` .

**отметка времени = вкл./выкл.**

Включить/выключить отметки времени. Такой же как `-Н` (см. [Метка времени](#) ) .

**use\_server\_timestamps = вкл/выкл**

Если установлено значение 'выключенный', Wget не будет устанавливать временную метку локального файла по временной метке на сервере (так же, как `--no-use-server-timestamps` ) .

**пытается = *n***

Установите количество повторов для каждого URL – так же, как '*-т n*'.

**use\_проху = вкл/выкл**

Если установлено значение «выключено», не используйте прокси-сервер, даже если установлены переменные среды, связанные с прокси-сервером. В этом случае это то же самое, что использовать '*--по-прокси*'.

**пользователь = строка**

Укажите *строку* имени пользователя для получения файлов по FTP и HTTP . Эту команду можно переопределить с помощью '*ftp\_user*' и '*http\_user*' команда для FTP и HTTP соответственно.

**user\_agent = строка**

Идентификатор пользовательского агента, отправленный на HTTP-сервер, аналогичен '*--user-agent= строка*'.

**подробный = вкл/выкл**

Включить/выключить многословие – то же самое, что '*-в*'/'*-nv*'.

**подожди = *n***

Подождите *n* секунд между запросами – то же самое, что и '*-w n*'.

**wait\_retry = *n***

Подождите до *n* секунд между попытками только неудачных извлечений – то же самое, что '*--waitretry= n*'. Обратите внимание, что это включено по умолчанию в глобальном *wgetrc*.

---

Предыдущий: [Команды Wgetrc](#) , Вверх: [Файл запуска](#) [ [Содержание](#) ][ [Индекс](#) ]

## 6.4 Пример Wgetrc

Это образец файла инициализации, включенный в дистрибутив. Он разделен на два раздела: один для глобального использования (подходит для глобального файла запуска) и один для локального использования (подходит для *\$HOME/.wgetrc*). Будьте осторожны с тем, что вы меняете.

Обратите внимание, что почти все строки закомментированы. Чтобы команда имела какой-либо эффект, вы должны удалить '#' символ в начале строки.

```
###
```

```
### Пример файла инициализации Wget .wgetrc
```

###

## Вы можете использовать этот файл, чтобы изменить поведение wget по умолчанию  
## избегайте необходимости вводить множество параметров командной строки. Этот  
## не содержит полного списка команд – посмотрите руководство

## чтобы узнать, что можно поместить в этот файл. Вы можете найти это здесь:

## \$ info wget.info 'Файл запуска'

## Или онлайн здесь:

## <https://www.gnu.org/software/wget/manual/wget.html#Startup-File>

##

## Файл инициализации Wget может находиться в /usr/local/etc/wgetrc

## (глобально, для всех пользователей) или \$HOME/.wgetrc (для одного пользоват

##

## Чтобы использовать настройки из этого файла, вам придется их раскомментировать  
##, а также изменять их, в большинстве случаев, как значения на

## закомментированные строки являются значениями по умолчанию (например, «выкл

##

## Команда не учитывает регистр, подчеркивание и минус.

## Например, ftp\_проху, ftp-проху и ftpпроху – это одно и то же.

##

## Глобальные настройки (полезно для настройки в /usr/local/etc/wgetrc).

## Хорошо подумайте, прежде чем их менять, так как они могут уменьшить значени

## функциональность и заставить ее вести себя вопреки документации:

##

# Вы можете установить квоту получения для новичков, указав значение

# Необязательно, за которым следует «K» (килобайты) или «M» (мегабайты).

# квота по умолчанию не ограничена.

#altitude = ниже

# Вы можете уменьшить (или увеличить) количество повторов по умолчанию, когда

# загрузка файла (по умолчанию 20).

# попыток = 20

# Уменьшение максимальной глубины рекурсивного поиска удобно

# не позволяйте новичкам заходить слишком «глубоко», когда они невольно начина

# рекурсивный поиск. Значение по умолчанию – 5.

#reclevel = 5

# По умолчанию Wget использует «пассивную FTP» передачу, при которой клиент

# иницирует соединение для передачи данных с сервером, а не с другим

```
# наоборот. Это требуется в системах за NAT, где клиент
# компьютер не может быть легко доступен из Интернета. Однако некоторые
# программное обеспечение межсетевых экранов явно поддерживает активный FTP и
# проблемы с поддержкой пассивной передачи. Если вы находитесь в таком
# среде используйте «passive_ftp = off», чтобы вернуться к активному FTP.
#passive_ftp = выключено

# Приведенная ниже команда «wait» заставляет Wget ждать между каждым соединени
# Если вместо этого вы хотите, чтобы Wget ждал только между попытками неудачно
# загрузок, установите для waitretry максимальное количество секунд ожидания (
# будет использовать «линейную задержку», ожидая 1 секунду после первого сбоя
# на файле, через 2 секунды после второго сбоя и т.д. до этого максимума).
#waitretry = 10

##
## Локальные настройки (пользователь может установить их в своем $HOME/.wgetrc
## *крайне* нежелательно выносить эти настройки в глобальный файл, т.к.
## они потенциально опасны для «обычных» пользователей.
##
## Даже при настройке собственного ~/.wgetrc вы должны знать, что вам нужно.
## делают, прежде чем сделать это.
##

# Установите это значение, чтобы использовать метку времени по умолчанию:
#timestamping = выключено

# Хорошая идея заставить Wget отправлять ваш адрес электронной почты в поле «0
# заголовок с вашим запросом (чтобы администраторы сервера могли связаться
# вам в случае ошибок). Wget *не* отправляет 'From:' по умолчанию.
#header = От: Ваше имя <имя_пользователя@сайт.домен>

# Вы можете настроить другие заголовки, например Accept-Language. Асепт-Язык
# *не* отправляется по умолчанию.
#header = Асепт-Language: en

# Вы можете установить прокси-серверы по умолчанию, которые Wget будет использ
# Они переопределяют значение в среде.
#https_proxy = http://проху.yoyodyne.com:18023/
#http_proxy = http://проху.yoyodyne.com:18023/
#ftp_proxy = http://проху.yoyodyne.com:18023/

# Если вы вообще не хотите использовать прокси, отключите этот параметр.
```

```
#use_proxy = включено

# Вы можете настроить внешний вид поиска. Допустимые параметры – по умолчанию,
# бинарный, мега и микро.
#dot_style = по умолчанию

# Если отключить этот параметр, Wget не сможет загружать /robots.txt. Будь уве
# знать *точно*, что такое /robots.txt и как он используется, прежде чем вноси
# по умолчанию!
#роботы = включено

# Может быть полезно заставить Wget ждать между соединениями. Установите это н
# количество секунд, в течение которых Wget будет ждать.
#подождите = 0

# Вы можете принудительно создать структуру каталогов, даже если один
# получено, если включить этот параметр.
#dirstruct = выкл.

# Вы можете включить рекурсивное извлечение по умолчанию (не делайте этого, ес
# вы не уверены, что знаете, что это значит), включив этот параметр.
#рекурсивный = выключен

# Чтобы всегда создавать резервную копию файла X как X.orig перед преобразован
# to -k / --convert-links / Convert_links = если указано),
# установите эту переменную в значение:
#backup_converted = выключено

# Чтобы Wget по умолчанию переходил по FTP-ссылкам из HTML-файлов, установите
# для включения:
#follow_ftp = выключено

# Чтобы сначала попробовать адреса ipv6:
#prefer-family = IPv6

# Установить состояние поддержки IRI по умолчанию
#iri = выключено

# Принудительно использовать системную кодировку по умолчанию
#localencoding = UTF-8

# Принудительно использовать кодировку удаленного сервера по умолчанию
#remoteencoding = UTF-8
```

```
# Включите, чтобы запретить переход по ссылкам, отличным от HTTPS, в рекурсивн
#httponly = выключено

# Настройте безопасность HTTPS (авто, SSLv2, SSLv3, TLSv1, PFS)
#secureprotocol = авто
```

Следующая: [Разное](#) , Предыдущая: [Файл запуска](#) , Вверх: [Wget 1.24.5](#) [ [Содержание](#) ][ [Индекс](#) ]

## 7 примеров

Примеры разделены на три раздела в зависимости от их сложности.

- [Простое использование](#)
- [Расширенное использование](#)
- [Очень продвинутое использование](#)

Следующая: [Расширенное использование](#) , Предыдущая: [Примеры](#) , Вверх: [Примеры](#) [ [Содержание](#) ][ [Указатель](#) ]

### 7.1 Простое использование

- Предположим, вы хотите загрузить URL-адрес . Просто введите:

```
wget http://fly.srk.fer.hr/
```

- Но что будет, если соединение медленное, а файл длинный? Соединение, вероятно, прервется, прежде чем весь файл будет получен более одного раза. В этом случае Wget будет пытаться получить файл до тех пор, пока не получит его целиком или не превысит количество повторов по умолчанию (20). Чтобы гарантировать благополучную доставку всего файла, легко изменить количество попыток на 45:

```
wget --tries=45 http://fly.srk.fer.hr/jpg/flyweb.jpg
```

- Теперь давайте оставим Wget работать в фоновом режиме и запишем его прогресс в файл журнала.бревно. Утомительно печатать '--пытается', поэтому мы будем

использовать '-т'.

```
wget -t 45 -o журнал http://fly.srk.fer.hr/jpg/flyweb.jpg &
```

Амперсанд в конце строки гарантирует, что Wget работает в фоновом режиме. Чтобы ограничить количество повторов, используйте '-т инф'.

- Использование FTP так же просто. Wget позаботится о логине и пароле.

```
wget ftp://gnjilux.srk.fer.hr/welcome.msg
```

- Если вы укажете каталог, Wget получит список каталогов, проанализирует его и преобразует в HTML . Пытаться:

```
wget ftp://ftp.gnu.org/pub/gnu/  
ссылки index.html
```

---

Следующая: [Очень продвинутое использование](#) , Предыдущая: [Простое использование](#) ,  
Вверх: [Примеры](#) [ [Содержание](#) ][ [Указатель](#) ]

## 7.2 Расширенное использование

- У вас есть файл, содержащий URL-адреса, которые вы хотите загрузить?  
Использовать '-я' выключатель:

```
wget -i файл
```

Если вы укажете '-' в качестве имени файла, URL -адреса будут считаны из стандартного ввода.

- Создайте пятиуровневое зеркальное отображение веб-сайта GNU с той же структурой каталогов, что и оригинал, только с одной попыткой для каждого документа, сохраняя журнал действий в упал:

```
wget -r https://www.gnu.org/ -o gnu-log
```

- То же, что и выше, но преобразуйте ссылки в загруженных файлах, чтобы они указывали на локальные файлы, чтобы вы могли просматривать документы в автономном режиме:

```
wget --convert-links -r https://www.gnu.org/ -o gnulog
```

- Получите только одну HTML- страницу, но убедитесь, что все элементы, необходимые для отображения страницы, такие как встроенные изображения и внешние таблицы стилей, также загружены. Также убедитесь, что загруженная страница ссылается на загруженные ссылки.

```
wget -p --convert-links http://www.example.com/dir/page.html
```

HTML - страница будет сохранена в `www.example.com/dir/page.html`, а изображения, таблицы стилей и т. д. где-то под `www.example.com/`, в зависимости от того, где они находились на удаленном сервере.

- То же, что и выше, но без `www.example.com/каталог`. На самом деле, я в любом случае не хочу иметь все эти случайные каталоги сервера – просто сохраните *все* эти файлы вскачать/ подкаталог текущего каталога.

```
wget -p --convert-links -nH -nd -Pdownload \  
http://www.example.com/dir/page.html
```

- Получите `index.html` файла '`www.lycos.com`', показывая исходные заголовки сервера:

```
wget -S http://www.lycos.com/
```

- Сохраните заголовки сервера вместе с файлом, возможно, для постобработки.

```
wget --save-headers http://www.lycos.com/  
еще index.html
```

- Получить первые два уровня '`wuarchive.wustl.edu`', сохраняя их в `/tmp`.



```
wget -r -l2 -P/tmp ftp://wuarchive.wustl.edu/
```

- Вы хотите загрузить все GIF- файлы из каталога на HTTP- сервере. Ты пытался 'wget http://www.example.com/dir/\*.gif', но это не сработало, поскольку HTTP -поиск не поддерживает подстановку. В этом случае используйте:

```
wget -r -l1 --no-parent -A.gif http://www.example.com/dir/
```

Более многословно, но эффект тот же. '-r -l1' означает рекурсивное извлечение (см. [Рекурсивная загрузка](#) ) с максимальной глубиной 1. '--без родителя' означает, что ссылки на родительский каталог игнорируются (см. [«Ограничения на основе каталогов»](#) ) и '-A.gif' означает загрузку только файлов GIF . '-A "\*.gif"' тоже сработало бы.

- Предположим, вы находились в процессе загрузки, когда Wget был прерван. Теперь вы не хотите уничтожать уже имеющиеся файлы. Это было бы:

```
wget -nc -r https://www.gnu.org/
```

- Если вы хотите закодировать свое имя пользователя и пароль в HTTP или FTP , используйте соответствующий синтаксис URL-адреса (см. [Формат URL-адреса](#) ).

```
wget ftp://hnksic:mypassword@unix.example.com/.emacs
```

Однако обратите внимание, что такое использование не рекомендуется в многопользовательских системах, поскольку оно раскрывает ваш пароль любому, кто просматривает выходные данные ps.

- Вы хотите, чтобы выходные документы отправлялись в стандартный вывод, а не в файлы?

```
wget -O - http://jagor.srce.hr/ http://www.srce.hr/
```

Вы также можете объединить эти два варианта и создать конвейеры для получения документов из удаленных горячих списков:

```
wget -O - http://cool.list.com/ | wget --force-html -i -
```

Предыдущий: [Расширенное использование](#) , Вверх: [Примеры](#) [ [Содержание](#) ][ [Указатель](#) ]

## 7.3 Очень продвинутое использование

- Если вы хотите, чтобы Wget сохранял зеркало страницы (или подкаталогов FTP ), используйте '--зеркало'('-м'), что является сокращением от '-г -л инф -N'. Вы можете поместить Wget в файл crontab и попросить его перепроверять сайт каждое воскресенье:

```
кронтаб  
0 0 * * 0 wget --mirror https://www.gnu.org/ -o /home/me/weeklog
```

- В дополнение к вышесказанному вы хотите, чтобы ссылки были преобразованы для локального просмотра. Но, прочитав это руководство, вы знаете, что преобразование ссылок не очень хорошо сочетается с метками времени, поэтому вам также нужно, чтобы Wget выполнил резервное копирование исходных файлов HTML перед преобразованием. Вызов Wget будет выглядеть так:

```
wget --mirror --convert-links --backup-converted \  
https://www.gnu.org/ -o /home/me/weeklog
```

- Но вы также заметили, что локальный просмотр работает не очень хорошо, если HTML- файлы сохраняются с расширениями, отличными от '.html', возможно, потому, что они служили index.cgi. Итак, вы хотите, чтобы Wget переименовал все файлы, обслуживаемые с типом контента 'текст/html' или 'приложение/xhtml+xml' *кимя* .html.

```
wget --mirror --convert-links --backup-converted \  
--adjust-extension -o /home/me/weeklog \  
https://www.gnu.org/
```

Или, с меньшим количеством ввода:

```
wget -m -k -K -E https://www.gnu.org/ -o /home/me/weeklog
```

Следующая: [Приложения](#) , Предыдущая: [Примеры](#) , Вверх: [Wget 1.24.5](#) [ [Содержание](#) ][ [Индекс](#) ]

## 8 Разное

В этой главе собрано все то, что больше нигде не поместилось.

- [Прокси](#)
- [Распределение](#)
- [Веб-сайт](#)
- [Списки рассылки](#)
- [Интернет-релейный чат](#)
- [Сообщить об ошибках](#)
- [Портативность](#)
- [Сигналы](#)

Следующая: [Распространение](#) , Предыдущая: [Разное](#) , Вверх: [Разное](#) [ [Содержание](#) ][ [Индекс](#) ]

### 8.1 Прокси

*Прокси* – это специальные HTTP- серверы, предназначенные для передачи данных с удаленных серверов локальным клиентам. Одним из типичных применений прокси является снижение нагрузки на сеть для пользователей с медленным соединением. Это достигается путем направления всех HTTP- и FTP -запросов через прокси-сервер, который кэширует передаваемые данные. Когда кэшированный ресурс будет запрошен снова, прокси вернет данные из кеша. Другое использование прокси-серверов – компании, которые отделяют (по соображениям безопасности) свои внутренние сети от остального Интернета. Чтобы получить информацию из Интернета, их пользователи подключаются и получают удаленные данные с помощью авторизованного прокси-сервера.

Wget поддерживает прокси-серверы для получения данных как по HTTP , так и по FTP . Стандартный способ указать местоположение прокси-сервера, который распознает Wget, – использовать следующие переменные среды:

**http\_proxy**

## https\_proxu

Если установлено, переменные `http_proxu` и `https_proxu` должны содержать URL-адреса прокси-серверов для HTTP- и HTTPS-соединений соответственно.

## ftp\_proxu

Эта переменная должна содержать URL-адрес прокси-сервера для FTP-соединений. Довольно часто `http_proxu` и `ftp_proxu` устанавливаются на один и тот же URL-адрес.

## no\_proxu

Эта переменная должна содержать разделенный запятыми список расширений домена, для которых прокси-сервер *не* должен использоваться. Например, если значение `no_proxu` равно `'.mit.edu'`, прокси-сервер не будет использоваться для получения документов из MIT.

В дополнение к переменным среды расположение и настройки прокси-сервера могут быть указаны внутри самого Wget.

**'--no-прокси'**

**'прокси = вкл/выкл'**

Эту опцию и соответствующую команду можно использовать для подавления использования прокси-сервера, даже если установлены соответствующие переменные среды.

**'http\_proxu = URL -адрес'**

**'https\_proxu = URL -адрес'**

**'ftp\_proxu = URL -адрес'**

**'no\_proxu = строка'**

Эти переменные файла запуска позволяют переопределить настройки прокси, заданные средой.

Некоторые прокси-серверы требуют авторизации, чтобы вы могли их использовать. Авторизация состоит из *имени пользователя* и *пароля*, которые должен отправить Wget. Как и в случае с авторизацией HTTP, существует несколько схем аутентификации. Для авторизации через прокси Basic на данный момент реализована только схема аутентификации.

Вы можете указать свое имя пользователя и пароль либо через URL-адрес прокси-сервера, либо через параметры командной строки. Предполагая, что прокси-сервер находится по адресу `'проxu.companу.com'` на порту 8001 URL-адрес прокси-сервера, содержащий данные авторизации, может выглядеть следующим образом:

```
http://hniksic:mypassword@proxy.company.com:8001/
```

Альтернативно вы можете использовать 'прокси-пользователь' и 'прокси-пароль' опции и эквивалентные `wgetrc` настройки `proxy_user` и `proxy_password` установить имя пользователя и пароль прокси.

---

Следующая: [Веб-сайт](#) , Предыдущая: [Прокси](#) , Вверх: [Разное](#) [ [Содержание](#) ][ [Индекс](#) ]

## 8.2 Распределение

Как и все утилиты GNU, последнюю версию Wget можно найти на главном архивном сайте GNU [ftp.gnu.org](http://ftp.gnu.org) и на его зеркалах. Например, Wget 1.24.5 можно найти по адресу <https://ftp.gnu.org/pub/gnu/wget/wget-1.24.5.tar.gz>.

---

Следующая: [Списки рассылки](#) , Предыдущая: [Рассылка](#) , Вверх: [Разное](#) [ [Содержание](#) ][ [Индекс](#) ]

## 8.3 Веб-сайт

Официальный веб-сайт GNU Wget находится по адресу <https://www.gnu.org/software/wget/> . Однако наиболее полезная информация находится на сайте «The Wget Wgiki», <http://wget.addictivecode.org/> .

---

Следующая: [Интернет-чат](#) , Предыдущая: [Веб-сайт](#) , Вверх: [Разное](#) [ [Содержание](#) ][ [Индекс](#) ]

## 8.4 Списки рассылки

- [Основной список](#)
- [Устаревшие списки](#)

### Основной список

Основной список рассылки для обсуждения, отчетов об ошибках или вопросов о GNU Wget находится по адресу [bug-wget@gnu.org](mailto:bug-wget@gnu.org) . Чтобы подписаться, отправьте электронное письмо по адресу [bug-wget-join@gnu.org](mailto:bug-wget-join@gnu.org) или посетите <https://lists.gnu.org/mailman/listinfo/bug-wget> .

Вам не нужно подписываться, чтобы отправить сообщение в список; однако обратите внимание, что отписавшиеся сообщения модерируются, и может пройти некоторое время,

прежде чем они попадут в список – **обычно около дня** . Если вы хотите, чтобы ваше сообщение появилось немедленно, подпишитесь на список перед публикацией. Архивы списка можно найти по адресу <https://lists.gnu.org/archive/html/bug-wget/> .

Шлюз NNTP/Usenetish также доступен через [Gmane](#) . Вы можете просмотреть архивы Gmane по адресу <http://news.gmane.org/gmane.comp.web.wget.general> . Обратите внимание, что в архивы Gmane удобно включать сообщения как из текущего списка, так и из предыдущего. Сообщения также появляются в архивах Gmane раньше, чем на <https://lists.gnu.org> .

## Устаревшие списки

Раньше список рассылки [wget@sunsite.dk](mailto:wget@sunsite.dk) использовался в качестве основного списка обсуждения, а другой список, [wget-patches@sunsite.dk](mailto:wget-patches@sunsite.dk) , использовался для отправки и обсуждения исправлений для GNU Wget.

Сообщения с адреса [wget@sunsite.dk](mailto:wget@sunsite.dk) архивируются по адресу

<https://www.mail-archive.com/wget%40sunsite.dk/> и по адресу

<http://news.gmane.org/gmane.comp.web.wget.general> (который также продолжает архивировать текущий список, [bug-wget@gnu.org](mailto:bug-wget@gnu.org) ).

Сообщения с адреса [wget-patches@sunsite.dk](mailto:wget-patches@sunsite.dk) архивируются по адресу

<http://news.gmane.org/gmane.comp.web.wget.patches> .

---

Следующая: [Сообщения об ошибках](#) , Предыдущая: [Списки рассылки](#) , Вверх: [Разное](#) [ [Содержание](#) ][ [Индекс](#) ]

## 8.5 Интернет-чат

В дополнение к спискам рассылки у нас также есть канал поддержки, настроенный через IRC по адресу [#wget](irc://irc.freenode.org). Приходите проверить это!

---

Следующая: [Портативность](#) , Предыдущая: [Интернет-чат](#) , Вверх: [Разное](#) [ [Содержание](#) ][ [Индекс](#) ]

## 8.6 Сообщения об ошибках

Вы можете отправлять отчеты об ошибках через систему отслеживания ошибок GNU Wget (см. <https://savannah.gnu.org/bugs/?func=additem&group=wget> ) или в наш список рассылки [bug-wget@gnu.org](mailto:bug-wget@gnu.org) .

Посетите <https://lists.gnu.org/mailman/listinfo/bug-wget> , чтобы получить дополнительную информацию (как подписаться, список архивов и т. д.).

Прежде чем отправлять отчет об ошибке, попробуйте следовать нескольким простым рекомендациям.

1. Пожалуйста, постарайтесь убедиться, что поведение, которое вы видите, действительно является ошибкой. Если Wget выходит из строя, это ошибка. Если Wget не ведет себя так, как описано в документации, это ошибка. Если что-то работает странно, но вы не уверены в том, как оно должно работать, это вполне может быть ошибка, но вы можете перепроверить документацию и списки рассылки (см. [Списки рассылки](#) ).
2. Постарайтесь повторить ошибку в как можно более простых обстоятельствах. Например, если Wget выходит из строя во время загрузки `'wget -r10 -kKE -t5 --no-проху http://example.com -o /tmp/log'`, вам следует попытаться увидеть, повторяется ли сбой и произойдет ли он, с помощью более простого набора параметров. Вы можете даже попытаться начать загрузку со страницы, на которой произошел сбой, чтобы проверить, не вызвала ли эта страница каким-то образом сбой.

Кроме того, хотя мне, вероятно, будет интересно узнать содержание вашего `.wgetгс`файл, просто выгружать его в отладочное сообщение, вероятно, плохая идея. Вместо этого вам следует сначала попытаться увидеть, повторяется ли ошибка с `.wgetгс`двинулся с пути. Только если окажется, что `.wgetгс`настройки влияют на ошибку, пришлите мне соответствующие части файла.

3. Пожалуйста, начните Wget с `'-д'` и отправьте нам полученный результат (или его соответствующие части). Если Wget был скомпилирован без поддержки отладки, скомпилируйте его заново – это *очень важно*. легче отслеживать ошибки.

Примечание. Обязательно удалите всю потенциально конфиденциальную информацию из журнала отладки, прежде чем отправлять ее на адрес ошибки. Он `-д`не будет изо всех сил стараться собирать конфиденциальную информацию, но журнал *будет* содержать довольно полную расшифровку общения Wget с сервером, которая может включать пароли и фрагменты загруженных данных. Поскольку адрес ошибки находится в публичном архиве, вы можете предположить, что все отчеты об ошибках доступны общественности.

4. Если произошел сбой Wget, попробуйте запустить его в отладчике, например, `gdb 'which wget'` согеи введите команду, `where`чтобы получить обратную трассировку. Это может не сработать, если системный администратор отключил файлы ядра, но попробовать можно.

---

Следующая: [Сигналы](#) , Предыдущая: [Отчеты об ошибках](#) , Вверх: [Разное](#) [ [Содержание](#) ] [ [Индекс](#) ]

## 8.7 Переносимость

Как и все программы GNU, Wget работает в системе GNU. Однако, поскольку для сборки и настройки он использует GNU Autotools и в основном избегает использования «специальных» функций какой-либо конкретной Unix, он должен компилироваться (и работать) на всех распространенных разновидностях Unix.

Различные версии Wget были скомпилированы и протестированы под многими типами Unix-систем, включая GNU/Linux, Solaris, SunOS 4.x, Mac OS X, OSF (также известный как Digital Unix или Tru64), Ultrix, \*BSD, IRIX, AIX и другие. Некоторые из этих систем больше не широко используются и могут не поддерживать последние версии Wget. Если Wget не скомпилируется в вашей системе, мы хотели бы знать об этом.

Благодаря любезным участникам эта версия Wget компилируется и работает на 32-битных платформах Microsoft Windows. Он был успешно скомпилирован с использованием компиляторов MS Visual C++ 6.0, Watcom, Borland C и GCC. Естественно, в нем урезаны некоторые функции, доступные в Unix, но он должен служить заменой для людей, использующих Windows. Обратите внимание, что части Wget, специфичные для Windows, не гарантированно будут поддерживаться в будущем, хотя на практике это происходит уже много лет. Обо всех вопросах и проблемах при использовании Windows следует сообщать в список рассылки Wget по адресу [wget@sunsite.dk](mailto:wget@sunsite.dk), где с ними могут ознакомиться волонтеры, поддерживающие функции, связанные с Windows.

Поддержку сборки на MS-DOS через DJGPP предоставил Жисл Ванем; порт для VMS поддерживается Стивеном Шведой и доступен по адресу <https://antinode.info/dec/sw/wget.html>.

---

Предыдущий: [Портативность](#) , Вверх: [Разное](#) [ [Содержание](#) ][ [Индекс](#) ]

## 8.8 Сигналы

Поскольку целью Wget является фоновая работа, он улавливает сигнал зависания (SIGHUP) и игнорирует его. Если вывод был на стандартном выводе, он будет перенаправлен в файл с именем wget-журнал. В противном случае SIGHUP игнорируется. Это удобно, если вы хотите перенаправить вывод Wget после его запуска.

```
$ wget http://www.gnu.org/dist/gnus.tar.gz &
```

```
...
```

```
$ убить -HUP %
```

```
Получен сигнал SIGHUP, который перенаправляет вывод в wget-log.
```

Кроме этого, Wget не будет каким-либо образом вмешиваться в сигналы. C-с, kill -TERM и kill -KILL должен убить его одинаково.



Далее: [Копирование этого руководства](#) , Предыдущее: [Разное](#) , Вверх: [Wget 1.24.5](#) [ [Содержание](#) ][ [Индекс](#) ]

## 9 Приложения

В этой главе содержатся некоторые ссылки, которые я считаю полезными.

- [Исключение роботов](#)
- [Вопросы безопасности](#)
- [Авторы](#)

---

Следующая: [Вопросы безопасности](#) , Предыдущая: [Приложения](#) , Вверх: [Приложения](#) [ [Содержание](#) ][ [Указатель](#) ]

### 9.1 Исключение роботов

Чрезвычайно легко заставить Wget бесцельно бродить по веб-сайту, поглощая все доступные данные. 'wget -г сайт', и все готово. Большой? Не для администратора сервера.

Пока Wget извлекает только статические страницы и делает это с разумной скоростью (см.--ждатьвариант), особых проблем нет. Проблема в том, что Wget не может отличить самую маленькую статическую страницу от самой требовательной CGI. На моем знакомом сайте есть раздел, обрабатываемый CGI Perl-скриптом, который на лету преобразует Info-файлы в HTML . Сценарий медленный, но работает достаточно хорошо для пользователей, которые время от времени просматривают информационный файл. Однако, когда чья-то рекурсивная загрузка Wget наткнется на индексную страницу, которая ссылается на все файлы Info через сценарий, система ставится на колени, не предоставляя пользователю ничего полезного (эту задачу преобразования файлов Info можно выполнить локально и получить доступ к в Информационную документацию для всего установленного в системе программного обеспечения GNU можно получить с помощью этой infoкоманды).

Чтобы избежать подобных происшествий, а также сохранить конфиденциальность документов, которые необходимо защитить от «хороших» роботов, была изобретена концепция *исключения роботов* . Идея состоит в том, что администраторы серверов и авторы документов могут указать, какие части сайта они хотят защитить от роботов, а к каким разрешить доступ.

Самый популярный механизм и *де-факто* стандартом, поддерживаемым всеми основными роботами, является «Стандарт исключения роботов» (RES), написанный Мартейном Костером и др. в 1994 году. Он определяет формат текстового файла, содержащего директивы, которые указывают роботам, какие URL-пути следует избегать. Чтобы роботы могли найти

спецификации, их необходимо поместить в `/robots.txt` в корне сервера, который роботы должны загрузить и проанализировать.

Хотя Wget не является веб-роботом в строгом смысле этого слова, он может загружать большие части сайта без вмешательства пользователя для загрузки отдельной страницы. По этой причине Wget учитывает RES при рекурсивной загрузке. Например, когда вы выдаете:

```
wget -r http://www.example.com/
```

Сначала индекс `'www.example.com'` будет загружен. Если Wget обнаружит, что хочет загрузить дополнительные документы с этого сервера, он запросит `'http://www.example.com/robots.txt'` и, если он найден, используйте его для дальнейших загрузок. `robots.txt` загружается только один раз на каждый сервер.

До версии 1.8 Wget поддерживал первую версию стандарта, написанную Мартейном Костером в 1994 году и доступную по адресу <http://www.robotstxt.org/orig.html>. Начиная с версии 1.8, Wget поддерживает дополнительные директивы, указанные в интернет-проекте `<draft-koster-robots-00.txt>` под названием «Метод управления веб-роботами». Проект, который, насколько мне известно, никогда не входил в RFC, доступен по адресу <http://www.robotstxt.org/norobots-rfc.txt>.

В настоящее руководство больше не включен текст Стандарта исключения роботов.

Второй, менее известный механизм позволяет автору отдельного документа указать, хочет ли он, чтобы по ссылкам из файла следовал робот. Это достигается с помощью METAтега, например:

```
<meta name="robots" content="nofollow">
```

Это объясняется более подробно на <http://www.robotstxt.org/meta.html>. Wget поддерживает этот метод исключения роботов в дополнение к обычному `/robots.txt` исключение.

Если вы знаете, что делаете, и действительно хотите отключить исключение роботов, установите `robots` для переменной значение `'выключенный'` в вашей `.wgetrc`. Вы можете добиться того же эффекта из командной строки, используя команду-е переключатель, например: `'wget -e robots=отключить URL ...'`.

---

Следующая: [Участники](#) , Предыдущая: [Исключение роботов](#) , Вверх: [Приложения](#) [ [Содержание](#) ][ [Указатель](#) ]

## 9.2 Вопросы безопасности

При использовании Wget вы должны знать, что он отправляет незашифрованные пароли по сети, что может представлять проблему безопасности. Вот основные проблемы и некоторые решения.

1. Пароли в командной строке видны с помощью `ps`. Лучший способ обойти это – использовать `wget -i` -и передавать URL -адреса на стандартный ввод Wget, каждый в отдельной строке, заканчивающейся `C-d` . Другой обходной путь – использовать `netrc` сохранить пароли; однако хранение незашифрованных паролей также считается угрозой безопасности.
2. При использовании небезопасной *базовой* схемы аутентификации незашифрованные пароли передаются через сетевые маршрутизаторы и шлюзы.
3. Пароли FTP также никоим образом не шифруются . На данный момент хорошего решения для этой проблемы не существует.
4. Хотя «обычный» вывод Wget пытается скрыть пароли, журналы отладки показывают их во всех формах. Этой проблемы можно избежать, если вы будете осторожны при отправке журналов отладки (да, даже если вы отправляете их мне).

---

Предыдущий: [Вопросы безопасности](#) , Вверх: [Приложения](#) [ [Содержание](#) ][ [Указатель](#) ]

## 9.3 Участники

GNU Wget был написан Хрвое Никшичем [hniksic@xemacs.org](mailto:hniksic@xemacs.org) ,

Однако разработка Wget никогда бы не зашла так далеко, если бы не помощь многих людей в виде отчетов об ошибках, предложений по функциям, исправлений или писем со словами «Спасибо!».

Особая благодарность выражается следующим людям (без особого порядка):

- Дэн Харклесс – предоставил много кода и документации чрезвычайно высокого качества, а также `--page-requisites` сопутствующих опций. Некоторое время он был главным сопровождающим и выпустил Wget 1.6.
- Ян Эбботт – внес исправления ошибок, исправления, связанные с Windows, и предоставил прототип реализации рекурсивной загрузки в ширину. Совместное сопровождение Wget в течение цикла выпуска 1.8.
- Команда [dotsrc.org](http://dotsrc.org), в частности Карстен Тигесен, пожертвовала системные ресурсы, такие как список рассылки, веб-пространство, FTP- пространство и репозитории контроля версий, а также потратила много времени на то, чтобы они

действительно заработали. Кристиан Райнигер оказал неоценимую помощь при настройке Subversion.

- Хейко Герольд – на протяжении многих лет предоставлял высококачественные сборки Windows, а также предоставлял отчеты об ошибках и сборках.
- Шон МакХорс – отчеты об ошибках и исправления.
- Каве Р. Гази – оперативная обработка ansi2kpr. Множество исправлений переносимости.
- Гордон Мацигкейт–.netrcподдерживать.
- Златко Чалушич, Томислав Вуец и Дражен Качар – представлены предложения и «философские» дискуссии.
- Дарко Будор – первоначальный порт на Windows.
- Антонио Розелла – помощь и предложения, а также первоначальный итальянский перевод.
- Томислав Петрович, Марио Микочевич – множество сообщений об ошибках и предложений.
- Франсис Пинар – множество подробных отчетов об ошибках и обсуждений.
- Карл Эйхвальдер – большая помощь с интернационализацией, макетом Makefile и многим другим.
- Джунио Хамано – предоставил поддержку Opie и HTTP- Digest аутентификации.
- Мауро Тортонеси – улучшена поддержка IPv6, добавлена поддержка систем двойного семейства. Переработан и улучшен код FTP IPv6. Поддерживал GNU Wget с 2004 по 2007 год.
- Кристофер Г. Льюис – сопровождение версии GNU WGet для Windows.
- Gisle Vanem – множество полезных исправлений и улучшений, особенно для поддержки Windows и MS-DOS.
- Ральф Вильденхьюс – предоставил исправления для преобразования Wget для использования Automake в процессе сборки, а также различные исправления ошибок.
- Стивен Шубигер – множество полезных исправлений, исправлений и улучшений. В частности, преобразование Wget для использования модулей GnuLib quotes и quoteargs, а также добавление запросов пароля на консоли через модуль GnuLib getpasswd-gnu.
- Тед Мильчарек – пожертвовал поддержку CSS.
- Сент-Ксавье – поддержка IRI (RFC 3987).
- Тим Рюсен – множество полезных патчей, особенно поддержка фаззинга и непрерывная интеграция. Сопровождающий с 2014 года.

- Даршит Шах – много полезных патчей. Поддержка сообщества на различных платформах. Сопровождающий с 2014 года.
- Люди, которые делали пожертвования на развитие, включая Брайана Гофа.

Следующие люди предоставили исправления, отчеты об ошибках/сборках, полезные предложения, услуги бета-тестирования, письма от фанатов и все остальное, что делает обслуживание таким увлекательным:

Тим Адам, Адриан Айхнер, Мартин Баер, Дитер Барон, Роджер Биман, Дэн Бергер, Т. Бхарат, Кристиан Бьер, Поль Блудов, Дэниэл Бодае, Марк Бойнс, Джон Берден, Жюльен Бути, Вандерлей Кавассен, Жиль Седок, Тим Чаррон, Ноэль Крэгг, Кристиан Чонкаш, Джон Дейли, Андреас Дамм, Амон Дэнси, Эндрю Дэвисон, Бертран Демидделаер, Александр Дергачев, Эндрю Дерябин, Ульрих Дреппер, Марк Дюпоншель, Дамир Джеко, Алан Элдридж, Ханс-Андреас Энгель, Александр Эркалович, Энди Эскильссон, Жоао Феррейра, Кристиан Френкель, Дэвид Фриц, Майк Фрайзингер, Чарльз К. Фу, ФУДЗИСИМА Сацуки, Масаси Фудзита, Ховард Гэйл, Марсель Герритс, Лембл Грегори, Ханс Гроблер, Ален Гиберт, Матье Гийом, Аарон Хоули, Йохен Хайн, Карл Хойер, Мадхусудан Хосааграхара, ХИРОСЕ Масааки, Ульф Харнхаммар, Грегор Хоффляйт, Эрик Магнус Хюльтен, Ричард Хувенирс, Йонас Йенсен, Ларри Джонс, Саймон Йозефссон, Марио Юрич, Хак Кампбьерн, Конст Каплински, Горан Кезунович, Игорь Христофоров, Роберт Кляйне, КОДЖИМА Хайме, Фила Колодны, Александр Куракос, Мартин Кремер, Сами Кранк, Джей Крелл, Σίμος Ξενιτέλλης (Симос Ксенителлис), Кристиан Лакас, Хрвое Лацко, Дэниел С. Льюарт, Николас Лихтмайер, Дэйв Лав, Александр В. Лукьянов, Томас Лусниг, Андре Майорель, Орельен Маршан Мэттью Дж. Меллон, Джордан Мендельсон, Тед Мельчарек, Роберт Миллан, Линь Же Мин, Ян Минар, Тим Муни, Кит Мур, Адам Д. Мосс, Саймон Мантон, Чарли Негиеси, Р. К. Оуэн, Джим Пэрис, Кенни Парнелл, Леонид Петров, Симона Пиунно, Эндрю Поллок, Стив Потье, Ян Пршикрыл, Марин Пургар, Чаба Радули, Кит Рефсон, Билл Ричардсон, Тайлер Риддл, Тобиас Рингстрем, Йохен Родербург, Хуан Хосе Родригес, Мацей В. Розицки, Эдвард Дж. Саболь, Хайнц Зальцманн, Роберт Шмидт, Николас Шоде, Бенно Шуленберг, Андреас Шваб, Стивен М. Шведа, Крис Сивуд, Пранаб Шеной, Деннис Смит, Тоомас Суме, Таге Стабель-Куло, Филип Стадерманн, Дэниэл Стенберг, Свен Штернбергер, Маркус Штрассер, Джон Саммерфилд, Сакачиц Сабольч, Майк Томас, Филипп Томас, Мауро Тортонеси, Дэйв Тернер, Жисл Ванем, Рабин Винсент, Рассел Винсент, Желько Врба, Чарльз Дж. Уолдман, Дуглас Э. Вегшайд, Ральф Вильденхьюс, Джошуа Дэвид Уильямс, Бенджамин Вулси, Сент-Ксавье, ЯМАЗАКИ Макото, Жасмин Зайнул, Боян Ждрня, Кристиан Циммер, Синь Цзоу.

Приношу извинения всем, кого я случайно не упомянул, и большое спасибо всем подписчикам списка рассылки Wget.

---

Следующая: [Указатель концепций](#), Предыдущая: [Приложения](#), Вверх: [Wget 1.24.5](#) [[Содержание](#)] [[Указатель](#)]

# Приложение А Копирование данного руководства

## ■ Лицензия на бесплатную документацию GNU

---

Предыдущий: [Копирование данного руководства](#) , Вверх: [Копирование данного руководства](#)  
[ [Содержание](#) ][ [Индекс](#) ]

## А.1 Лицензия на бесплатную документацию GNU

Версия 1.3, 3 ноября 2008 г.

Copyright © 2000–2002, 2007–2008, 2015, 2018–2024 Бесплатно  
Фонд программного обеспечения, Inc.  
<http://fsf.org/>

Каждому разрешено копировать и распространять дословные копии.  
настоящего лицензионного документа, но его изменение не допускается.

### 0. ПРЕАМБУЛА

Цель настоящей Лицензии – сделать руководство, учебник или другой функциональный и полезный документ *свободным* в смысле свободы: гарантировать каждому эффективную свободу копировать и распространять его с модификацией или без нее, как в коммерческих, так и в некоммерческих целях. Во-вторых, настоящая Лицензия оставляет за автором и издателем возможность получить признание за свою работу, не неся при этом ответственности за изменения, внесенные другими.

Эта Лицензия представляет собой своего рода «авторское лево», что означает, что производные работы на основе документа сами по себе должны быть свободными в том же смысле. Она дополняет Стандартную общественную лицензию GNU, которая представляет собой лицензию с авторским левом, предназначенную для свободного программного обеспечения.

Мы разработали эту Лицензию для использования ее в руководствах к свободным программам, поскольку бесплатному программному обеспечению нужна свободная документация: к свободной программе должны прилагаться руководства, предоставляющие те же свободы, что и само программное обеспечение. Но эта Лицензия не ограничивается руководствами по программному обеспечению; его можно использовать для любой текстовой работы, независимо от ее тематики и от того, опубликована ли она в виде печатной книги. Мы рекомендуем эту Лицензию главным образом для работ, целью которых является инструкция или ссылка.

### 1. ПРИМЕНИМОСТЬ И ОПРЕДЕЛЕНИЯ

Настоящая Лицензия распространяется на любое руководство или другую работу на любом носителе, содержащую размещенное правообладателем уведомление о том, что она может распространяться в соответствии с условиями настоящей Лицензии. Такое уведомление предоставляет действующую во всем мире безвозмездную лицензию на неограниченный срок действия на использование данного произведения на условиях, изложенных в настоящем документе. «Документ» ниже относится к любому такому руководству или работе. Любой представитель общественности является лицензиатом и к нему обращаются на «вы». Вы принимаете лицензию, если копируете, изменяете или распространяете произведение способом, требующим разрешения в соответствии с законом об авторском праве.

«Модифицированная версия» Документа означает любую работу, содержащую Документ или его часть, скопированную дословно или с изменениями и/или переведенную на другой язык.

«Вторичный раздел» – это поименованное приложение или основной раздел Документа, который касается исключительно отношений издателей или авторов Документа к общей теме Документа (или к связанным с ней вопросам) и не содержит ничего, что могло бы напрямую относиться к в рамках этой общей темы. (Таким образом, если Документ частично является учебником по математике, Вторичный раздел не может объяснять какую-либо математику.) Отношения могут быть вопросом исторической связи с предметом или связанными с ним вопросами или юридическими, коммерческими, философскими, этическими вопросами. или политическая позиция по отношению к ним.

«Неизменяемые разделы» – это определенные Второстепенные разделы, названия которых обозначены как названия Неизменяемых разделов в уведомлении о том, что Документ выпущен по настоящей Лицензии. Если раздел не соответствует приведенному выше определению вторичного, его нельзя обозначать как инвариантный. Документ может содержать ноль Неизменяемых разделов. Если в Документе не указаны какие-либо Неизменяемые разделы, то их нет.

«Тексты на обложке» – это определенные короткие отрывки текста, которые указаны как «Тексты на передней обложке» или «Тексты на задней обложке» в уведомлении о том, что Документ выпущен под настоящей Лицензией. Текст на передней обложке может содержать не более 5 слов, а текст на задней обложке – не более 25 слов.

«Прозрачная» копия Документа означает машиночитаемую копию, представленную в формате, спецификация которого доступна широкой публике, которая подходит для непосредственного редактирования документа с помощью обычных текстовых редакторов или (для изображений, состоящих из пикселей) стандартной краски. программы или (для рисунков) какой-либо широко доступный редактор рисунков, который подходит для ввода в форматировщики текста или для автоматического перевода в различные форматы, подходящие для ввода в форматировщики текста.

Копия, сделанная в другом формате файла «Прозрачный», разметка или отсутствие разметки которого предназначена для предотвращения или предотвращения последующего изменения читателями, не является «Прозрачной». Формат изображения не является прозрачным, если он используется для значительного объема текста. Копия, которая не является «Прозрачной», называется «Непрозрачной».

Примеры подходящих форматов для прозрачных копий включают простой ASCII без разметки, входной формат Texinfo, входной формат LaTeX, SGML или XML с использованием общедоступного DTD, а также соответствующий стандарту простой HTML, PostScript или PDF, предназначенный для модификации человеком. Примеры форматов прозрачных изображений включают PNG, XCF и JPG. К непрозрачным форматам относятся проприетарные форматы, которые могут читаться и редактироваться только собственными текстовыми процессорами, SGML или XML, для которых DTD и/или инструменты обработки обычно недоступны, а также машинно-генерируемые HTML, PostScript или PDF, создаваемые некоторыми текстовыми процессорами для только для целей вывода.

«Титульный лист» для печатной книги означает сам титульный лист, а также такие последующие страницы, которые необходимы для разборчивого размещения материала, требуемого настоящей Лицензией, для размещения на титульном листе. Для произведений в форматах, не имеющих титульного листа как такового, «Титульный лист» означает текст рядом с наиболее заметным местом названия произведения, предшествующий началу основной части текста.

«Издатель» означает любое физическое или юридическое лицо, которое распространяет копии Документа среди общественности.

Раздел «Под названием XYZ» означает именованный подраздел Документа, заголовок которого либо в точности соответствует XYZ, либо содержит XYZ в круглых скобках после текста, который переводит XYZ на другой язык. (Здесь XYZ означает конкретное имя раздела, упомянутое ниже, например «Благодарности», «Посвящения», «Подтверждения» или «История».) «Сохранить название» такого раздела при изменении документа означает, что он согласно этому определению остается разделом «Под названием XYZ».

Документ может включать отказ от гарантийных обязательств рядом с уведомлением о том, что настоящая Лицензия распространяется на Документ. Настоящие Отказ от гарантийных обязательств считаются включенными в настоящую Лицензию посредством ссылки, но только в отношении отказа от гарантий: любые другие последствия, которые могут иметь эти Отказ от гарантийных обязательств, являются недействительными и не влияют на смысл настоящей Лицензии.

## 2. ДОСТАЛЬНОЕ КОПИРОВАНИЕ



Вы можете копировать и распространять Документ на любом носителе, как в коммерческих, так и в некоммерческих целях, при условии, что настоящая Лицензия, уведомления об авторских правах и уведомление о лицензии, в котором говорится, что данная Лицензия применима к Документу, воспроизводятся во всех копиях и что вы не добавляете никаких других условий. положениям настоящей Лицензии. Вы не можете использовать технические меры, чтобы препятствовать или контролировать чтение или дальнейшее копирование копий, которые вы делаете или распространяете. Тем не менее, вы можете принять компенсацию в обмен на копии. Если вы распространяете достаточно большое количество копий, вы также должны соблюдать условия раздела 3.

Вы также можете предоставлять копии на условиях, указанных выше, и можете публично демонстрировать копии.

### 3. КОПИРОВАНИЕ В КОЛИЧЕСТВЕ

Если вы публикуете печатные копии (или копии на носителях, которые обычно имеют печатные обложки) Документа, числом более 100, и в уведомлении о лицензии на Документ требуются тексты обложки, вы должны вложить копии в обложки, которые ясно и разборчиво содержат все эти Тексты на обложке: тексты на передней обложке на передней обложке и тексты на задней обложке на задней обложке. Обе обложки также должны четко и разборчиво идентифицировать вас как издателя этих копий. На передней обложке должно быть указано полное название, причем все слова заголовка должны быть одинаково заметными и видимыми. Вы можете дополнительно добавить на обложки другой материал. Копирование с изменениями, ограниченными обложками, при условии, что они сохраняют название Документа и удовлетворяют этим условиям, в других отношениях может рассматриваться как дословное копирование.

Если требуемые тексты для любой обложки слишком объемны и не помещаются в них разборчиво, вам следует поместить первые из перечисленных (столько, сколько влезет) на саму обложку, а остальные продолжить на соседних страницах.

Если вы публикуете или распространяете непрозрачные копии Документа, число которых превышает 100, вы должны либо включить машиночитаемую прозрачную копию вместе с каждой непрозрачной копией, либо указать в каждой непрозрачной копии местоположение в компьютерной сети, из которого общая сетевая Используя public, имеет доступ к загрузке с использованием общедоступных сетевых протоколов полной прозрачной копии Документа без дополнительных материалов. Если вы используете последний вариант, вы должны принять разумные меры, когда вы начинаете массовое распространение непрозрачных копий, чтобы гарантировать, что эта прозрачная копия останется доступной в указанном месте в течение как минимум одного года после последнего распространения вами копии. Непрозрачная копия (напрямую или через ваших агентов или розничных продавцов) этого издания для публичного доступа.

Просьба, но не обязательно, чтобы вы связались с авторами Документа задолго до распространения большого количества копий, чтобы дать им возможность предоставить вам обновленную версию Документа.

#### 4. МОДИФИКАЦИИ

Вы можете копировать и распространять Модифицированную версию Документа в соответствии с условиями разделов 2 и 3 выше, при условии, что вы выпускаете Модифицированную версию именно под этой Лицензией, при этом Модифицированная версия выполняет роль Документа, тем самым лицензируя распространение и изменение Модифицированную версию всем, у кого есть ее копия. Кроме того, в Модифицированной версии необходимо выполнить следующие действия:

1. Используйте на титульном листе (и на обложках, если таковые имеются) заголовок, отличный от названия Документа и предыдущих версий (которые должны, если таковые имеются, быть указаны в разделе «История» Документа). Вы можете использовать то же название, что и в предыдущей версии, если первоначальный издатель этой версии дает разрешение.
2. Укажите на Титульном листе в качестве авторов одно или несколько физических или юридических лиц, ответственных за авторство изменений в Измененной версии, а также не менее пяти основных авторов Документа (всех его основных авторов, если их число менее пяти), если только они не освободят вас от этого требования.
3. Укажите на титульном листе имя издателя Измененной версии в качестве издателя.
4. Сохраните все уведомления об авторских правах на Документ.
5. Добавьте соответствующее уведомление об авторских правах для ваших изменений рядом с другими уведомлениями об авторских правах.
6. Включите сразу после уведомлений об авторских правах уведомление о лицензии, дающее публичное разрешение на использование Модифицированной версии в соответствии с условиями настоящей Лицензии, в форме, показанной в Приложении ниже.
7. Сохраните в этом уведомлении о лицензии полные списки Неизменяемых разделов и обязательных сопроводительных текстов, приведенные в уведомлении о лицензии к Документу.
8. Включите неизменную копию настоящей Лицензии.
9. Сохраните раздел под названием «История», сохраните его название и добавьте к нему элемент, указывающий как минимум название, год, новых авторов и издателя Модифицированной версии, как указано на титульном

листе. Если в Документе нет раздела под названием «История», создайте его, указав название, год, авторов и издателя Документа, как указано на его титульной странице, затем добавьте элемент, описывающий Модифицированную версию, как указано в предыдущем предложении.

10. Сохраните сетевое расположение, если таковое имеется, указанное в Документе для публичного доступа к прозрачной копии Документа, а также сетевые расположения, указанные в Документе для предыдущих версий, на которых он был основан. Их можно разместить в разделе «История». Вы можете опустить сетевое местоположение для произведения, которое было опубликовано не менее чем за четыре года до самого Документа, или если первоначальный издатель версии, на которую оно ссылается, дает разрешение.
11. Для любого раздела, озаглавленного «Благодарности» или «Посвящения», сохраните название раздела и сохраните в разделе всю суть и тон каждого из приведенных в нем благодарностей и/или посвящений участников.
12. Сохраните все Неизменяемые разделы Документа в неизменном виде в их тексте и названиях. Номера разделов или их эквиваленты не считаются частью названий разделов.
13. Удалите любой раздел под названием «Одобрения». Такой раздел не может быть включен в Модифицированную версию.
14. Не меняйте название существующего раздела на «Одобрения» и не конфликтуйте по названию с каким-либо неизменным разделом.
15. Сохраняйте любые заявления об отказе от гарантийных обязательств.

Если Измененная версия включает новые вводные разделы или приложения, которые квалифицируются как второстепенные разделы и не содержат материалов, скопированных из Документа, вы можете по своему усмотрению обозначить некоторые или все эти разделы как неизменяемые. Для этого добавьте их названия в список Неизменяемых разделов в лицензионном уведомлении Модифицированной версии. Эти заголовки должны отличаться от названий других разделов.

Вы можете добавить раздел под названием «Одобрения», при условии, что он не содержит ничего, кроме одобрений вашей Модифицированной версии различными сторонами – например, заявлений о рецензировании или того, что текст был одобрен организацией в качестве авторитетного определения стандарта.

Вы можете добавить отрывок длиной до пяти слов в качестве текста на передней обложке и отрывок длиной до 25 слов в качестве текста на задней обложке в конец списка текстов на обложке в Модифицированной версии. Только один отрывок из Текста на передней обложке и один из Текста на задней обложке может быть

добавлен (или по договоренности) каким-либо одним лицом. Если Документ уже содержит сопроводительный текст для той же обложки, ранее добавленный вами или по договоренности, заключенной той же организацией, от имени которой вы действуете, вы не можете добавить еще один; но вы можете заменить старую версию с явного разрешения предыдущего издателя, добавившего старую версию.

Автор(ы) и издатель(и) Документа согласно настоящей Лицензии не дают разрешения использовать свои имена в рекламных целях, а также заявлять или подразумевать одобрение какой-либо Модифицированной версии.

## 5. ОБЪЕДИНЕНИЕ ДОКУМЕНТОВ

Вы можете объединять Документ с другими документами, выпущенными по настоящей Лицензии, в соответствии с условиями, определенными в разделе 4 выше для модифицированных версий, при условии, что вы включаете в комбинацию все неизменные разделы всех исходных документов в неизменном виде и перечисляете их все. в качестве неизменяемых разделов вашей совместной работы в уведомлении о лицензии, и что вы сохраняете все их отказы от гарантийных обязательств.

Объединенная работа должна содержать только одну копию настоящей Лицензии, и несколько идентичных Неизменяемых разделов могут быть заменены одной копией. Если существует несколько неизменяемых разделов с одинаковым названием, но разным содержанием, сделайте заголовок каждого такого раздела уникальным, добавив в конце его в круглых скобках имя первоначального автора или издателя этого раздела, если оно известно, или же уникальный номер. Внесите такие же изменения в названия разделов в списке Неизменяемых разделов в уведомлении о лицензии на объединенное произведение.

При объединении необходимо объединить любые разделы под названием «История» в различных исходных документах, образуя один раздел под названием «История»; аналогичным образом объедините любые разделы под названием «Благодарности» и любые разделы под названием «Посвящения». Вы должны удалить все разделы под названием «Подтверждения».

## 6. КОЛЛЕКЦИИ ДОКУМЕНТОВ

Вы можете создать коллекцию, состоящую из Документа и других документов, выпущенных по настоящей Лицензии, и заменить отдельные копии настоящей Лицензии в различных документах единственной копией, включенной в коллекцию, при условии соблюдения правил настоящей Лицензии для дословное копирование каждого из документов во всех остальных отношениях.

Вы можете извлечь один документ из такой коллекции и распространять его индивидуально в соответствии с настоящей Лицензией при условии, что вы вставите копию настоящей Лицензии в извлеченный документ и будете следовать

настоящей Лицензии во всех других отношениях, касающихся дословного копирования этого документа.

## 7. АГРЕГИРОВАНИЕ С НЕЗАВИСИМЫМИ РАБОТАМИ

Компиляция Документа или его производных с другими отдельными и независимыми документами или произведениями в объеме носителя или носителя для распространения называется «агрегатом», если авторские права, возникающие в результате компиляции, не используются для ограничения законных прав. пользователей компиляции сверх того, что позволяют отдельные работы. Когда Документ включен в совокупность, настоящая Лицензия не распространяется на другие произведения в совокупности, которые сами по себе не являются производными от Документа.

Если требование к Сопроводительному тексту раздела 3 применимо к этим копиям Документа, то, если Документ составляет менее половины всего совокупности, Сопроводительные тексты Документа могут быть размещены на обложках, которые заключают Документ в совокупность, или электронный эквивалент обложки, если Документ составлен в электронной форме. В противном случае они должны появиться на печатных обложках, заключающих в себе весь агрегат.

## 8. ПЕРЕВОД

Перевод считается своего рода модификацией, поэтому вы можете распространять переводы Документа в соответствии с условиями раздела 4. Замена Неизменяемых разделов переводами требует специального разрешения от их правообладателей, но вы можете включать переводы некоторых или всех Неизменяемых разделов в дополнение к оригинальным версиям этих неизменных разделов. Вы можете включить в Документ перевод настоящей Лицензии и всех лицензионных уведомлений, а также любых заявлений об отказе от гарантий при условии, что вы также включите оригинальную английскую версию этой Лицензии и оригинальные версии этих уведомлений и отказов от ответственности. В случае возникновения разногласий между переводом и оригинальной версией настоящей Лицензии или уведомления или отказа от ответственности первоначальная версия будет иметь преимущественную силу.

Если раздел документа озаглавлен «Благодарности», «Посвящения» или «История», требование (раздел 4) о сохранении его названия (раздел 1) обычно требует изменения фактического названия.

## 9. ПРЕКРАЩЕНИЕ

Вы не имеете права копировать, изменять, sublicenzировать или распространять Документ, за исключением случаев, прямо предусмотренных настоящей Лицензией. Любая попытка иным образом скопировать, изменить, sublicenzировать или

распространить его является недействительной и автоматически прекращает ваши права по настоящей Лицензии.

Однако если вы прекратите все нарушения настоящей Лицензии, ваша лицензия от конкретного правообладателя будет восстановлена (а) временно, если и до тех пор, пока правообладатель явным и окончательным образом не прекратит действие вашей лицензии, и (б) навсегда, если правообладатель не откажется уведомить вас о нарушении разумными способами в течение 60 дней после прекращения.

Более того, ваша лицензия от конкретного правообладателя восстанавливается навсегда, если правообладатель уведомит вас о нарушении каким-либо разумным способом. Это первый раз, когда вы получаете уведомление о нарушении настоящей Лицензии (для любой работы) от этого правообладателя. и вы устраните нарушение в течение 30 дней после получения уведомления.

Прекращение действия ваших прав по этому разделу не прекращает действие лицензий сторон, получивших от вас копии или права по настоящей Лицензии. Если ваши права были прекращены и не восстановлены окончательно, получение копии некоторых или всех тех же материалов не дает вам никаких прав на их использование.

## 10. БУДУЩИЕ ПЕРЕСМОТРЫ НАСТОЯЩЕЙ ЛИЦЕНЗИИ

Фонд свободного программного обеспечения может время от времени публиковать новые, исправленные версии Лицензии свободной документации GNU. Такие новые версии будут схожи по духу с настоящей версией, но могут отличаться в деталях для решения новых проблем или проблем. См. <http://www.gnu.org/copyleft/>.

Каждой версии Лицензии присвоен отличительный номер версии. Если в Документе указано, что к нему применяется конкретная пронумерованная версия настоящей Лицензии «или любая более поздняя версия», у вас есть возможность следовать положениям и условиям либо этой указанной версии, либо любой более поздней версии, которая была опубликована (не как проект) Фонда свободного программного обеспечения. Если в Документе не указан номер версии настоящей Лицензии, вы можете выбрать любую версию, когда-либо опубликованную (не как черновик) Фондом свободного программного обеспечения. Если в Документе указано, что доверенное лицо может решать, какие будущие версии настоящей Лицензии можно использовать, публичное заявление этого доверенного лица о принятии версии на постоянной основе дает вам право выбрать эту версию для Документа.

## 11. РЕЛИЦЕНЗИРОВАНИЕ

«Сайт массового многоавторского сотрудничества» (или «Сайт ММС») означает любой сервер Всемирной паутины, который публикует произведения, защищенные авторским правом, а также предоставляет всем желающим удобные возможности для

редактирования этих работ. Примером такого сервера является общедоступная вики, которую может редактировать каждый. «Массовое сотрудничество нескольких авторов» (или «ММС»), содержащееся на сайте, означает любой набор охраняемых авторским правом произведений, опубликованных таким образом на сайте ММС.

«CC-BY-SA» означает лицензию Creative Commons Attribution-Share Alike 3.0, опубликованную Creative Commons Corporation, некоммерческой корпорацией с основным местом деятельности в Сан-Франциско, Калифорния, а также будущие версии этой лицензии с авторским правом. лицензия, опубликованная той же организацией.

«Включение» означает публикацию или переиздание Документа полностью или частично как часть другого Документа.

ММС «имеет право на повторное лицензирование», если он лицензируется в соответствии с настоящей Лицензией, и если все произведения, которые были впервые опубликованы под этой Лицензией где-либо кроме этой ММС, и впоследствии полностью или частично включены в ММС, (1) не имели сопроводительные тексты или неизменяемые разделы, и (2) были, таким образом, включены до 1 ноября 2008 г.

Оператор сайта ММС может повторно опубликовать ММС, содержащийся на сайте в соответствии с CC-BY-SA, на том же сайте в любое время до 1 августа 2009 г. при условии, что ММС имеет право на повторное лицензирование.

## ДОПОЛНЕНИЕ: Как использовать данную Лицензию для ваших документов

Чтобы использовать настоящую Лицензию в написанном вами документе, включите в документ копию Лицензии и поместите следующие уведомления об авторских правах и лицензии сразу после титульной страницы:

Авторское право (C)    год    , ваше имя .

Разрешено копировать, распространять и/или изменять этот документ.

в соответствии с условиями лицензии GNU Free Documentation License версии 1. или любую более позднюю версию, опубликованную Фондом свободного программного без неизменяемых разделов, текстов на передней обложке и задней обложки

Тексты. Копия лицензии включена в раздел под названием ``GNU

Лицензия на бесплатную документацию».



Если у вас есть неизменяемые разделы, тексты на передней обложке и тексты на задней обложке, замените «на...тексты». соответствовать этому:

с неизменяемыми разделами, перечисляющими *их названия* , *список* текстов на передней обложке и тексты на задней обложке быть *список* .

Если у вас есть неизменяемые разделы без сопроводительных текстов или какая-либо другая комбинация из трех, объедините эти два варианта в соответствии с ситуацией.

Если ваш документ содержит нетривиальные примеры программного кода, мы рекомендуем выпускать эти примеры параллельно под выбранной вами лицензией свободного программного обеспечения, например GNU General Public License, чтобы разрешить их использование в свободном программном обеспечении.

Предыдущий: [Копирование этого руководства](#) , Вверх: [Wget 1.24.5](#) [ [Содержание](#) ][ [Индекс](#) ]

## Указатель концепций

Перейти к:

.

#

А

Б

В

Г

Д

Е

Ж

З

И

К

Л

М

Н

О

П

Р

С

Т

У

Ф

Ш

Х

Х

Индексная запись	Раздел
.	
расширение .css	<a href="#">Параметры HTTP</a>
расширение .html	<a href="#">Параметры HTTP</a>
.листинг файлов, удаление	<a href="#">Параметры FTP</a>
.netrc	<a href="#">Запускowej файл</a>
.wgetrc	<a href="#">Запускowej файл</a>
#	
<a href="#">#wget</a>	<a href="#">Интернет-релейный чат</a>
A	



принимать каталоги	Ограничения на основе каталогов
принимать суффиксы	Типы файлов
принимать подстановочные знаки	Типы файлов
добавить в журнал	Параметры журнала и входного файла
аргументы	Вызов
аутентификация	Варианты загрузки
аутентификация	Параметры HTTP
аутентификация	Параметры HTTP
учетные данные аутентификации	Варианты загрузки

Б

резервное копирование преобразованных файлов	Параметры рекурсивного поиска
резервное копирование файлов	Варианты загрузки
ограничение пропускной способности	Варианты загрузки
база для относительных ссылок во входном файле	Параметры журнала и входного файла
привязать адрес	Варианты загрузки
привязать DNS-адрес	Варианты загрузки
отчеты об ошибках	Сообщить об ошибках
ошибки	Сообщить об ошибках

С

кэш	Параметры HTTP
кэширование DNS-запросов	Варианты загрузки
чехол складывается	Рекурсивные параметры принятия/отклонения
DNS-адрес клиента	Варианты загрузки
IP-адрес клиента	Варианты загрузки
затирание, файл	Варианты загрузки
командная строка	Вызов
комментарии, HTML	Параметры рекурсивного поиска
тайм-аут подключения	Варианты загрузки

Контент при ошибке	Параметры HTTP
Содержание-Расположение	Параметры HTTP
Content-Encoding, выберите	Параметры HTTP
Content-Length, игнорировать	Параметры HTTP
продолжить поиск	Варианты загрузки
продолжить поиск	Варианты загрузки
участники	Авторы
конвертация ссылок	Параметры рекурсивного поиска
печенье	Параметры HTTP
куки, загрузка	Параметры HTTP
печенье, сохранение	Параметры HTTP
куки, сеанс	Параметры HTTP
вырезать каталоги	Параметры каталога

Д

отлаживать	Параметры журнала и входного файла
имя страницы по умолчанию	Параметры HTTP
удалить после получения	Параметры рекурсивного поиска
каталоги	Ограничения на основе каталогов
каталоги, исключить	Ограничения на основе каталогов
каталоги, включающие	Ограничения на основе каталогов
ограничения каталога	Ограничения на основе каталогов
префикс каталога	Параметры каталога
DNS-кеш	Варианты загрузки
DNS IP-адрес, клиент, DNS	Варианты загрузки
DNS IP-адрес, клиент, DNS	Варианты загрузки
DNS-сервер	Варианты загрузки
Тайм-аут DNS	Варианты загрузки
точечный стиль	Варианты загрузки
скачиваю несколько раз	Варианты загрузки

И

ЭГДС	Параметры HTTPS (SSL/TLS)
энтропия, определяющая источник	Параметры HTTPS (SSL/TLS)
Примеры	Примеры
исключить каталоги	Ограничения на основе каталогов
выполнить команду wgetrc	Основные параметры запуска

Ф

FDL, Лицензия свободной документации GNU	Лицензия на бесплатную документацию GNU
функции	Обзор
имена файлов, ограничить	Варианты загрузки
права доступа к файлам	Параметры FTP
заполнение кэша прокси	Параметры рекурсивного поиска
переходить по FTP-ссылкам	Рекурсивные параметры принятия/отклонения
следующие ftp-ссылки	FTP-ссылки
следующие ссылки	Следующие ссылки
заставить HTML	Параметры журнала и входного файла
FTP-аутентификация	Параметры FTP
FTP-пароль	Параметры FTP
отметка времени FTP	Внутренние метки времени FTP
FTP-пользователь	Параметры FTP

Г

подстановка, переключение	Параметры FTP
---------------------------	---------------

ЧАС

возьми это	Сигналы
заголовок, добавить	Параметры HTTP
хосты, охватывающие	Охватывающие хосты
HSTS	Параметры HTTPS (SSL/TLS)
HTML- комментарии	Параметры рекурсивного поиска

HTTP-пароль	Параметры HTTP
HTTP-реферер	Параметры HTTP
HTTP-метка времени	Внутренние метки времени HTTP
HTTP-пользователь	Параметры HTTP

я

поддержка идентификатора	Варианты загрузки
игнорировать регистр	Рекурсивные параметры принятия/отклонения
игнорировать длину	Параметры HTTP
включать каталоги	Ограничения на основе каталогов
неполные загрузки	Варианты загрузки
неполные загрузки	Варианты загрузки
постепенное обновление	Отметка времени
index.html	Параметры HTTP
входной файл	Параметры журнала и входного файла
вход-металинк	Параметры журнала и входного файла
Интернет-релейный чат	Интернет-релейный чат
призывая	Вызов
IP-адрес, клиент	Варианты загрузки
IPv6	Варианты загрузки
IRC	Интернет-релейный чат
ири поддержка	Варианты загрузки

к

Keep-Alive, выключение	Параметры HTTP
держат бадхэш	Параметры журнала и входного файла

л

Последняя версия	Распределение
------------------	---------------

<a href="#">ограничить пропускную способность</a>	<a href="#">Варианты загрузки</a>
<a href="#">конвертация ссылок</a>	<a href="#">Параметры рекурсивного поиска</a>
<a href="#">ссылки</a>	<a href="#">Следующие ссылки</a>
<a href="#">список</a>	<a href="#">Списки рассылки</a>
<a href="#">загрузка файлов cookie</a>	<a href="#">Параметры HTTP</a>
<a href="#">локальная кодировка</a>	<a href="#">Варианты загрузки</a>
<a href="#">расположение wgetrc</a>	<a href="#">Расположение Wgetrc</a>
<a href="#">журнальный файл</a>	<a href="#">Параметры журнала и входного файла</a>

М

<a href="#">список рассылки</a>	<a href="#">Списки рассылки</a>
<a href="#">Metalink-индекс</a>	<a href="#">Параметры журнала и входного файла</a>
<a href="#">металинк-через-http</a>	<a href="#">Параметры журнала и входного файла</a>
<a href="#">зеркальное отображение</a>	<a href="#">Очень продвинутое использование</a>

Н

<a href="#">нет родителя</a>	<a href="#">Ограничения на основе каталогов</a>
<a href="#">бесшумный</a>	<a href="#">Варианты загрузки</a>
<a href="#">ноуп</a>	<a href="#">Вызов</a>
<a href="#">количество попыток</a>	<a href="#">Варианты загрузки</a>

О

<a href="#">компенсировать</a>	<a href="#">Варианты загрузки</a>
<a href="#">операционные системы</a>	<a href="#">Портативность</a>
<a href="#">синтаксис опции</a>	<a href="#">Синтаксис опций</a>
<a href="#">Другие методы HTTP</a>	<a href="#">Параметры HTTP</a>
<a href="#">выходной файл</a>	<a href="#">Параметры журнала и входного файла</a>
<a href="#">обзор</a>	<a href="#">Обзор</a>

п

реквизиты страницы	Параметры рекурсивного поиска
пассивный FTP	Параметры FTP
пароль	Варианты загрузки
Пауза	Варианты загрузки
Постоянные соединения, отключение	Параметры HTTP
портативность	Портативность
ПОЧТА	Параметры HTTP
Предпочтительное Местоположение	Параметры журнала и входного файла
индикатор прогресса	Варианты загрузки
прокси	Прокси
прокси	Варианты загрузки
прокси	Параметры HTTP
аутентификация прокси	Параметры HTTP
заполнение прокси	Параметры рекурсивного поиска
пароль прокси	Параметры HTTP
прокси-пользователь	Параметры HTTP

вопрос

тихий	Параметры журнала и входного файла
квота	Варианты загрузки

р

случайное ожидание	Варианты загрузки
случайность с указанием источника	Параметры HTTPS (SSL/TLS)
ставка, лимит	Варианты загрузки
тайм-аут чтения	Варианты загрузки
рекурсия	Рекурсивная загрузка
рекурсивная загрузка	Рекурсивная загрузка
перенаправить	Параметры HTTP
перенаправление вывода	Расширенное использование

<a href="#">реферер, http</a>	<a href="#">Параметры HTTP</a>
<a href="#">отклонять каталоги</a>	<a href="#">Ограничения на основе каталогов</a>
<a href="#">отклонять суффиксы</a>	<a href="#">Типы файлов</a>
<a href="#">отклонять подстановочные знаки</a>	<a href="#">Типы файлов</a>
<a href="#">относительные ссылки</a>	<a href="#">Относительные ссылки</a>
<a href="#">удаленное кодирование</a>	<a href="#">Варианты загрузки</a>
<a href="#">сообщать об ошибках</a>	<a href="#">Сообщить об ошибках</a>
<a href="#">необходимые изображения, скачивание</a>	<a href="#">Параметры рекурсивного поиска</a>
<a href="#">Возобновление загрузки</a>	<a href="#">Варианты загрузки</a>
<a href="#">Возобновление загрузки</a>	<a href="#">Варианты загрузки</a>
<a href="#">повторные попытки</a>	<a href="#">Варианты загрузки</a>
<a href="#">повторные попытки, ожидание между</a>	<a href="#">Варианты загрузки</a>
<a href="#">получение</a>	<a href="#">Рекурсивная загрузка</a>
<a href="#">исключение роботов</a>	<a href="#">Исключение роботов</a>
<a href="#">robots.txt</a>	<a href="#">Исключение роботов</a>

с

<a href="#">образец wgetrc</a>	<a href="#">Пример Wgetrc</a>
<a href="#">сохранение файлов cookie</a>	<a href="#">Параметры HTTP</a>
<a href="#">безопасность</a>	<a href="#">Вопросы безопасности</a>
<a href="#">Обслуживание сервера</a>	<a href="#">Исключение роботов</a>
<a href="#">ответ сервера, распечатать</a>	<a href="#">Варианты загрузки</a>
<a href="#">ответ сервера, сохранить</a>	<a href="#">Параметры HTTP</a>
<a href="#">сеансовые файлы cookie</a>	<a href="#">Параметры HTTP</a>
<a href="#">обработка сигнала</a>	<a href="#">Сигналы</a>
<a href="#">охват хостов</a>	<a href="#">Охватывающие хосты</a>
<a href="#">указать конфигурацию</a>	<a href="#">Параметры журнала и входного файла</a>
<a href="#">паук</a>	<a href="#">Варианты загрузки</a>
<a href="#">SSL</a>	<a href="#">Параметры HTTPS (SSL/TLS)</a>
<a href="#">SSL-сертификат</a>	<a href="#">Параметры HTTPS (SSL/TLS)</a>
<a href="#">Центр сертификации SSL</a>	<a href="#">Параметры HTTPS (SSL/TLS)</a>
<a href="#">Тип SSL-сертификата, укажите</a>	<a href="#">Параметры HTTPS (SSL/TLS)</a>
<a href="#">SSL-сертификат, проверьте</a>	<a href="#">Параметры HTTPS (SSL/TLS)</a>

SSL CRL, список отзыва сертификатов	Параметры HTTPS (SSL/TLS)
Протокол SSL, выберите	Параметры HTTPS (SSL/TLS)
PIN-код открытого ключа SSL	Параметры HTTPS (SSL/TLS)
стартовая позиция	Варианты загрузки
запускать	Запусковой файл
файл запуска	Запусковой файл
суффиксы, принять	Типы файлов
суффиксы, отклонить	Типы файлов
символические ссылки, получение	Параметры FTP
синтаксис опций	Синтаксис опций
синтаксис wgetrc	Синтаксис Wgetrc

T

рекурсивная обрезка на основе тегов	Рекурсивные параметры принятия/отклонения
отметка времени	Отметка времени
использование меток времени	Использование отметок времени
тайм-аут	Варианты загрузки
тайм-аут, подключиться	Варианты загрузки
тайм-аут, DNS	Варианты загрузки
тайм-аут, читай	Варианты загрузки
отметка времени	Отметка времени
пытается	Варианты загрузки
Имена доверенных серверов	Параметры HTTP
типы файлов	Типы файлов

B

отсоединить	Варианты загрузки
обновление архивов	Отметка времени
URL-адрес	Формат URL-адреса
Синтаксис URL-адреса	Формат URL-адреса
использование, отметка времени	Использование отметок времени
пользователь	Варианты загрузки



В

различный  
подробный

Различный  
Параметры журнала и входного  
файла

В

ждать  
подожди, случайный  
ожидание между повторными попытками  
ВАРК  
Веб-сайт  
Wget как паук  
wgetrc  
команды wgetrc  
расположение wgetrc  
синтаксис wgetrc  
подстановочные знаки, принять  
подстановочные знаки, отклонить  
Имена файлов Windows

Варианты загрузки  
Варианты загрузки  
Варианты загрузки  
Параметры HTTPS (SSL/TLS)  
Веб-сайт  
Варианты загрузки  
Запускowej файл  
Команды Wgetrc  
Расположение Wgetrc  
Синтаксис Wgetrc  
Типы файлов  
Типы файлов  
Варианты загрузки

Икс

xattr

Параметры журнала и входного  
файла

Перейти к:

. #  
А Б В Г Д Е Ж З И К Л М Н О П Р С Т  
У Ф Ш Х Х

Сноски

(1)

Если у тебя есть .netrc-файл в вашем домашнем каталоге, там также будет искать пароль.

## (2)

В качестве дополнительной проверки Wget просмотрит заголовок Content-Length и сравнит размеры; если они не совпадают, удаленный файл будет загружен независимо от отметки времени.