

git-ftp Настройка -- Как загружать изменения на хостинг через FTP (GitFTP) - Windows

```
1git ftp -s test push
```

Что происходит

В данной заметке рассматривается использование скриптов git-ftp для загрузки изменений из локального git-репозитория на FTP-хостинг.

Программные средства

Для заливки же кода через FTP, будем использовать набор скриптов git-ftp (<https://github.com/git-ftp/git-ftp>(link is external)) - за который спасибо программисту René Moser-у.

Данную штуковину можно считать элементарным средством для [деплойа](#) на удалённый сервер.

Установка

Рассмотри установку:

- [B Windows](#)
- [B Ubuntu Linux](#)
-

Настройка параметров FTP-сервера

Переходим в папку репозитория (теперь уже вашего проекта, изменения в котором будем загружать на удаленный сервер с помощью git ftp), и открываем папку .git, в которой есть файл config:

```
1.git/config
```

-- открываем его редактором и добавляем туда реквизиты доступа к вашему FTP-серверу, например такие:

```
1[git-ftp "test"]
2  user = df836as
3  url  = "92.51.92.51/spec/public_html"
4  password = dfgd34rg34g
```

После чего файл будет выглядеть как-то так:

```
1[core]
2  bare = false
3  repositoryformatversion = 0
4  filemode = false
5  symlinks = false
6  ignorecase = true
7  logallrefupdates = true
8[git-ftp "test"]
9  user = df836as
10 url = "92.51.92.51/spec/public_html"
11 password = dfgd34rg34g
```

Что есть что:

1.test - условное название нашего ftp сервера,

2.далее указано имя пользователя FTP

3.затем указывается путь к директории проекта (в ней не обязательно должна быть папка .git - более того на сервер её лучше не закидывать , работать можно с обычным набором файлов, но .git обязательно должна быть в наборе файлов на машине с которой вы будете заливать изменения на FTP сервер), который начинается с доменного имени или ip-адреса сервера, а дальше всё зависит от того к какой именно директории дан доступ пользователю (в данном случае df836as имеет доступ к корню в котором есть папка spec, а уже в ней папка public_html, куда и подразумевается заливать проект и/или изменения).

4.92.51.92.51/spec/public_html -- в вашем случае этот путь может быть иным (зависит от того в какую именно директорию логинится ftp-пользователь - именно от неё надо писать доп. путь после ip или имени домена.)

5.Последний параметр - пароль для пользователя, указанного выше (у нас - для df836as).

Далее, после того как мы настроили параметры сервера переходим в папку с локальным репозиторием (в том же Git Bash в windows или в терминале в Linux), например:

```
1cd /D/www/spec.local
```

Ну и начинаем работать уже с git-ftp.

Работа с git-ftp

Чтобы загрузить весь проект на хостинг (первый раз) выполните:

```
1git ftp -s test init
```

Если же проект уже загружен (актуальные версии файлов уже на сайте), то выполните:

lgit ftp -s test catchup

После этого (загрузки всего репозитория или отметки о том, что он уже в актуальном состоянии) выгружаться будут только новые изменения, что (после выполнения коммита) можно сделать командой:

lgit ftp -s test push

А если подробно (с коммитом из командной строки - 3 команды):

```
1$ git add .
2$ git commit -m "Commit name"
3$ git ftp -s test push
```

Если что-то не получается, то чтобы узнать причину ошибки используйте флаг -v, например:

lgit ftp -s test catchup -v

ПРИМЕЧАНИЕ: на старых версиях возможна [такая проблема](#), лучше сразу обновить Git for Windows.

Несколько серверов

Можно указать и несколько FTP серверов для git-ftp, например (пример содержимого .git/config):

```
1[core]
2  bare = false
3  repositoryformatversion = 0
4  filemode = false
5  symlinks = false
6  ignorecase = true
7  logallrefupdates = true
8[git-ftp "test"]
9  user = df836as
10 url = "92.51.92.51/spec/public_html"
11 password = dfgd34rg34g
12[git-ftp "prod"]
13 user = b4407539_qwe
14 url = "free.get.com/"
15 password = df333Grg34g
```

Тогда для загрузки на указанный сервер prod новых изменений, соответственно, выполняем:

```
lgit ftp -s prod push
```

Вот и всё) Удачи в использовании)

Git FTP на примере в Windows

Данный пост для тех, кто уже освоил немного GIT, и хочет чтобы его жизнь стала ещё проще, а производительность выше! Т.е. делать push сразу на FTP сервер, к которому отсутствует доступ SSH.

Для работы с консолью я использовал Git Bash.

Я разделил пост на несколько этапов:

1. Установка Git-Ftp
2. Проверка установленного Git-Ftp
3. Настройка репозитория
4. Использование, загрузка на FTP сервер

1. Установка Git-Ftp:

```
$ git clone https://github.com/git-ftp/git-ftp
$ cd git-ftp
$ chmod +x git-ftp
$ cp ~/git-ftp/git-ftp /bin/git-ftp
```

2. Проверка установленного Git-Ftp:

```
$ cd ~  
$ which git-ftp  
/bin/git-ftp  
$ git-ftp --help
```

Если всё выполнилось без ошибок, то установка, прошла как и планировалось.

3. Настройка репозитория:

Применение к проекту:

```
$ cd /c/OpenServer/domains/site-name  
$ notepad .git/config
```

Добавьте в открывшийся config следующее:

```
[git-ftp "test"]  
  user = ваш_ftp_пользователь  
  url = 255.165.35.35/www/site-name.com  
  password = ваш_ftp_пароль
```

url — это Ip-адрес/путь к репозиторию на ftp-сервере

test — название конфига, который будет указывать, на какой ftp-сервер, будут загружаться обновления. Можно указать несколько.

Поставить метку, о том, что на сервер уже были загружены свежие

обновления, выполнять не обязательно. В этом случае git-ftp будет обновлять только вновь изменённые файлы.

```
$ git ftp -s test catchup
```

4. Использование, загрузка на FTP сервер:

```
$ git add .  
$ git commit -m "Commit name"  
$ git ftp -s test push
```

test — название ftp-конфига, который мы указали в настройках Git'a.

Вот в общем-то и всё. При редактировании файлов повторяйте процедуру из пункта 4, для обновления файлов на сервере.

Теги: gitftpwindows

Хабы: Git