**Процесс добавления нового репозитория git и группы разработчиков имеющих доступ к репозиторию на базе FreeBSD 9.2**

1) Заходим на сервер с установленным git`ом под root`ом

**login: root**

**password: telecardsvn**

В случае успешной авторизации должно появиться приглашение вида:

**oissvnserver#**

2) Заходим в папку git`а, где находятся все репозитории:

**#cd /usr/home/git/repositories/**

Если вы зашли на сервер под root`ом, то при вводе команды:

**#ls**

появится список из конфигурационных файлов git`а и самих репозиториев, приблизительный вывод показан ниже:

*.cshrc .mailrc gitosis-admin.git*

*.login .profile project.git*

*.login\_conf .rhosts test.git*

*.mail\_aliases .shrc*

Папки с расширение ".git" как раз и являются репозиториями git`a.

3) Создаем папку под репозиторий:

**#mkdir my\_repo\_name.git**

Проверяем наличие созданной папки командой "ls"

4)Заходим в папку, созданную под репозиторий и инициализируем там репозиторий:

**#cd my\_repo\_name.git**

**#git init --bare --shared**

Должно появиться сообщение вида:

Initialized empty shared Git repository in /usr/home/git/repositories/my\_repo \_name.git/

5) Теперь необходимо скопировать репозиторий и добавить в него хотя бы один файл, чтобы удаленный компьютер не ругался на пустой репозиторий:

**#cd /usr/home/git/tmp** (либо в любую другую временную папку)

**#git clone /usr/home/git/repositories/my\_repo\_name.git/ my\_repo\_name**

Появится сообщение вида:

Cloning into 'my\_repo\_name' ... done.

warning: You appear to have cloned an empty repository

После этого заходим в скопированную папку:

**#cd my\_repo\_name/**

Теперь создаем файл и пушим его на сервер:

**#echo "test" > test**

**#git add test**

**#git commit -a -m "initial branch"**

**#git push origin master**

Если все произошло успешно, то создастся начальная ветка git`а и репозиторий возможно будет использовать в дальнейшем.

На данном этапе создание репозитория успешно завершено, теперь необходимо добавить группу разработчиков.

6) Проверяем, где находится git-shell:

**#which git-shell**

/usr/local/bin/git-shell - Запоминаем вывод (Может отличаться)

7)Проверяем добавлен ли git-shell (строка выше) в /etc/shells

#**cat /etc/shells**

Если путь к git-shell не указан в файле, то необходимо его добавить:

**#echo /usr/local/bin/git-shell >> /etc/shells**

8) Создание нового пользователя для доступа к git репозиторию my\_repo\_name:

**#adduser**

Появиться диалоговое окно, которое будет просить вас вводить данные

Username: my\_repo1 (вписываем имя пользователя)

Full name: my\_repo1 (вписываем имя пользователя)

UID (Leave empty for default): (Оставляем пустым и нажимаем Enter)

Login group [my\_repo1]: my\_repo1 (Вписываем имя группы или оставляем пустым, имя группы должно совпадать с username, иначе новая группа не создастся)

Login group is my\_repo1. Invite my\_repo1 into other groups? []: (Оставляем пустым и нажимаем Enter)

Login class [default]: (Оставляем пустым и нажимаем Enter)

Shell [sh]: /usr/local/bin/git-shell (Вписываем адрес, который добавлен в /etc/shells)

Home directory [/home/my\_repo1]:/home/git/repositories/ (Вписываем полный путь к git-репозитроиям)

Home directory permissions (Leave empty for default): (Оставляем пустым и нажимаем Enter)

Use password-based authentication? [yes]: yes (Пишем yes)

Use an empty password? (yes/no) [no]: no (Пишем no)

Use a random password? (yes/no) [no]: no (Пишем no)

Enter password: pass (Вводим пароль для учетной записи)

Enter password again: pass (Повторяем пароль для учетной записи)

Lock out the account after creation? [no] : (Оставляем пустым и нажимаем Enter)

OK? (yes/no): yes (Пишем yes)

adduser: INFO: Successfully added (my\_repo1) to the user database.

Add another user? (yes/no): no (Пишем no)

После данной операции будет создан пользователь my\_repo1.

9) Разрешаем доступ пользователю my\_repo1 доступ по ssh

**#echo "AllowUsers my\_repo1" >> /etc/ssh/sshd\_config**

Проверяем:

**#cat /etc/ssh/sshd\_congig**

Перезапускаем службу sshd:

**#service sshd restart**

10) Изменяем владельца и группы владельца репозитория my\_repo\_name.git:

**#chown -R my\_repo1:my\_repo1 /usr/home/git/repositories/my\_repo\_ name.git**

Проверяем:

**#ls -l /usr/home/git/repositories/**

Должна быть запись такого рода:

drwxrwxr-x 7 my\_repo1 my\_repo1 ... my\_repo\_name.git

11) Даем полный доступ владельцу и группе репозитория my\_repo\_name.git

#**chmod 770 /usr/home/git/repositories/my\_repo\_name.git**

и устанавливаем в этой папке наследование прав для группы

#**chmod +s** **/usr/home/git/repositories/my\_repo\_name.git**

Проверяем:

**#ls -l /usr/home/git/repositories/**

Должна быть запись такого рода:

drwsrws--- 7 my\_repo1 my\_repo1 ... my\_repo\_name.git

12) Теперь переходим на сторону клиента. необходимо, чтобы на персональном компьютере под управлением windows было установлен сам клиент git и желательно для удобства tortoisergit. Скачать их можно по следующим ссылкам:

git: http://git-scm.com/downloads на данной странице необходимо выбрать git для windows

tortoisegit: https://code.google.com/p/tortoisegit/

Сначала устанавливаем git, потом tortoise git. После установки tortoisegit необходимо добавить в настройки tortoisegit имя автора и его почту. Для этого кликаем в проводнике на пустом месте правой кнпкой мышки, выбираем tortoisegit -> settings, выбрать вкладку Git и в поле Name ввести ваше имя, а поле email ваш электронный адресс. Если эти действия не проделать, то при первом использовании tortoisegit он сам выведем форму для ввода user`a и email

После установки заходим в любую пустую папку, кликаем на пустом месте правой кнопокой мыши и в Popupmenu выбираем "Git clone" (рис.1):

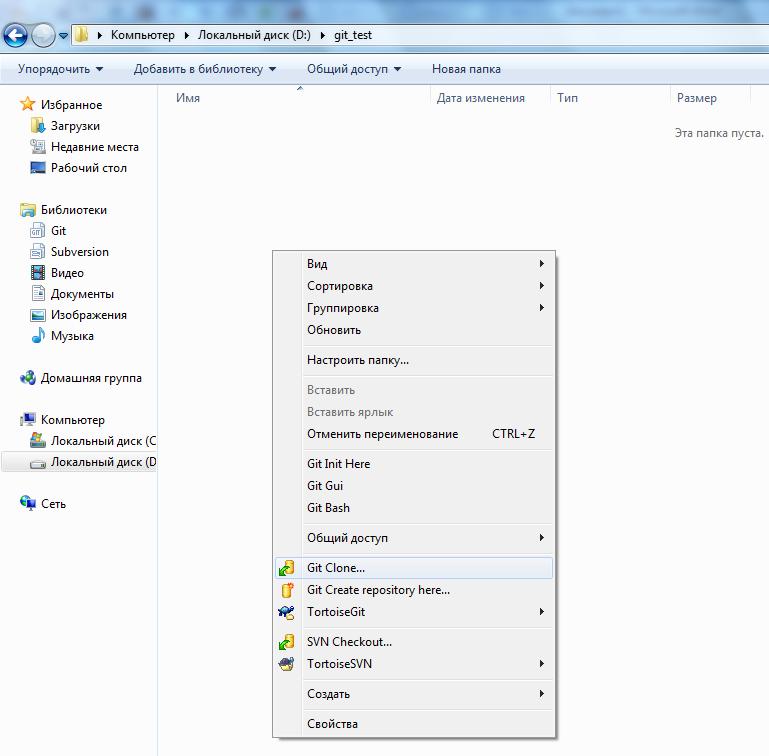


Рисунок 1

После этого появится следующее диалогове окно, показанное на рис.2. В поле URL необходимо ввести следующую строчку:

my\_repo1@192.168.20.252:my\_repo\_name.git

my\_repo1 - пользователь, которого мы добавили на сервере,

192.168.20.252 - IP адресс сервера,

my\_repo\_name.git - имя созданного нами репозитория,

после того как ввели данную строчку - нажимаем ОК.

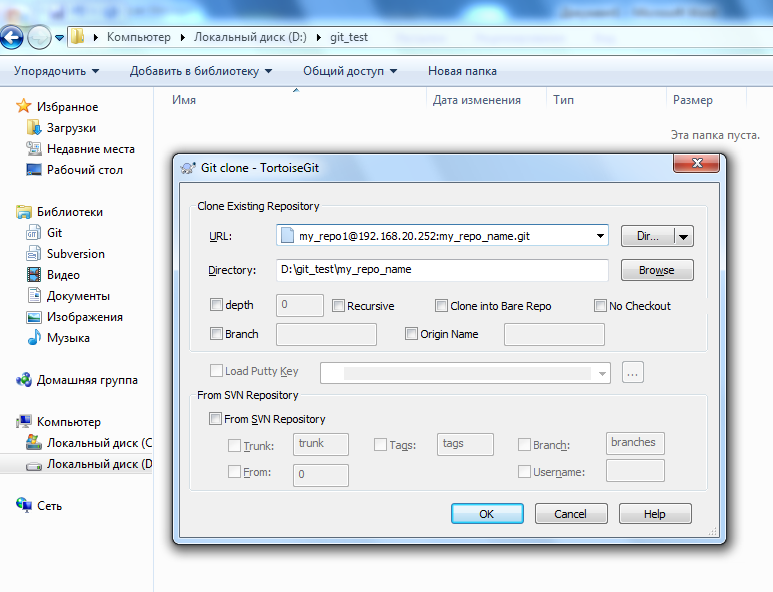


Рисунок 2

После нажатия ОК появится 2 окна (рис 3):

- окно статуса показывающий процесс копирования репозитория на локальную машину

- окно запрашивающее пароль для учетной записи my\_repo1

Необходимо ввести пароль, который вводился при создании нового пользователя, в данном случае pass.

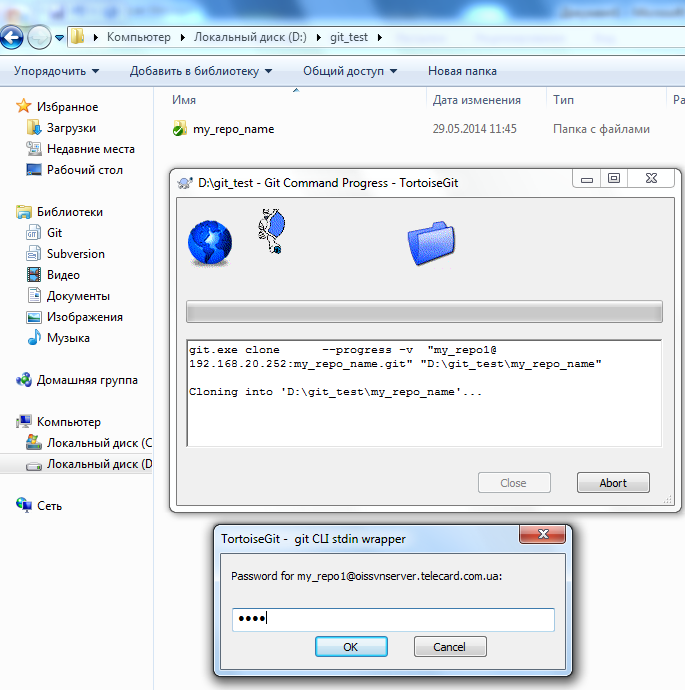


Рисунок 3

В случае, усли настройка цчетной записи и git-сервера была успешна, то появится следующее окно (рис.4)

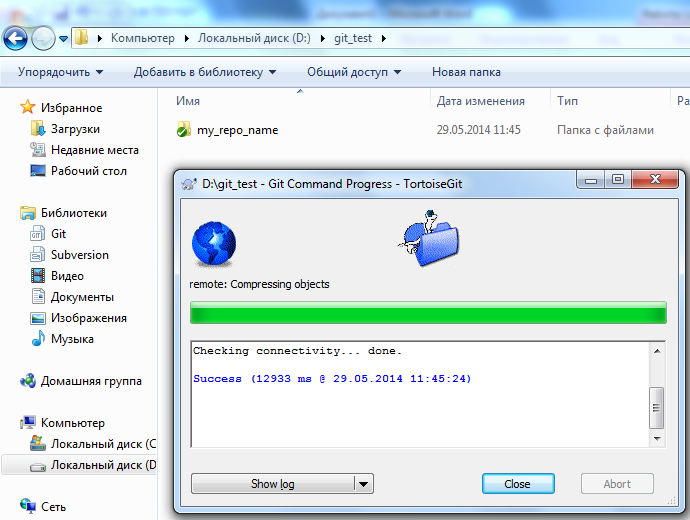


Рисунок 4

Зайдя в папку my\_repo\_name можно увидеть созданный ранее файл test.

таким образом проведя все вышеописанные шаги мы получили новый репозиторий для группы пользователей, однако на данный момент создан только один пользователь в группе. Чтобы добавить пользователей в созданную нами группу my\_repo1 переходим снова на сервер.

13) Добавление новых пользователей в группу разработчиков.

Создание еще одного пользователя для доступа к git репозиторию my\_repo\_name:

**#adduser**

Появиться диалоговое окно, кторое будет просить васс вводить данные

Username: my\_repo2 (вписываем имя пользователя)

Full name: my\_repo2 (вписываем имя пользователя)

UID (Leave empty for default): (Оставляем пустым и нажимаем Enter)

Login group [my\_repo2]: my\_repo1 (Вписываем имя группы, которую мы создали вместе с первым пользователем my\_repo1)

Login group is my\_repo1. Invite my\_repo1 into other groups? []: (Оставляем пустым и нажимаем Enter)

Login class [default]: (Оставляем пустым и нажимаем Enter)

Shell [sh]: /usr/local/bin/git-shell (Вписываем адресс, который добавлен в /etc/shells)

Home directory [/home/my\_repo1]:/home/git/repositories/ (Вписываем полный путь к git-репозитроиям)

Home directory permissions (Leave empty for default): (Оставляем пустым и нажимаем Enter)

Use password-based authentication? [yes]: yes (Пишем yes)

Use an empty password? (yes/no) [no]: no (Пишем no)

Use a random password? (yes/no) [no]: no (Пишем no)

Enter password: pass (Вводим пароль для учетной записи)

Enter password again: pass (Повтрояем пароль для учетной записи)

Lock out the account after creation? [no] : (Оставляем пустым и нажимаем Enter)

OK? (yes/no): yes (Пишем yes)

adduser: INFO: Successfully added (my\_repo2) to the user database.

Add another user? (yes/no): no (Пишем no)

После данной операции будет создан пользователь my\_repo2.

Разрешаем доступ пользователю my\_repo2 доступ по ssh

**#echo "AllowUsers my\_repo2" >> /etc/ssh/sshd\_config**

Проверяем:

**#cat /etc/ssh/sshd\_congig**

Перезапускаем службу sshd:

**#service sshd restart**

После данных операций повторяем шаг 12, однако в URL теперь записываем команду с новым пользователем:

my\_repo2@192.168.20.252:my\_repo\_name.git

P.S. Если вы используете Git Bash, то команда копирования репозитория будет выглядеть следующим образом:

git clone my\_repo2@192.168.20.252:my\_repo\_name.git D://your\_path

**Полезные ссылки:**

**1) Установка git сервера на Freebsd 7.2 c клиентами EGit на Eclipse под Windows**

**http://vermus.blogspot.com/2009/06/git-freebsd-72-c-egit-eclipse-windows.html**

**2) Git: разворачиваем репозиторий для группы разработчиков**

**http://wired-mind.info/post/190/**

**3) Установка gitosis на FreeBsd**

**http://snippets.pp.ru/article/240**

**4) Git на сервере**

**https://github.com/progit/progit/blob/master/ru/04-git-server/01-chapter4.markdown**

**5) Git — начинаем работать**

**http://blog.perlover.com/2011/09/06/git-start-up/#.U4bNJXA5PsM**