

Лабораторная работа №2  
Курс: Защита информации

Патраков Николай

24 сентября 2015 г.

## Содержание

1	Цель работы	3
2	Ход работы	3
3	Выводы	5

## 1 Цель работы

Изучить систему контроля версий Git, освоить основные приемы работы с ней.

## 2 Ход работы

Git - распределенная система контроля версий. Это подразумевает, что каждый разработчик хранит на собственной машине полноценную копию репозитория. Пока разработчик работает в рамках собственного репозитория, все происходит в рамках обычной СКВ. Помимо этого, предусмотрены также команды для работы с удаленным репозиторием. Например, push для мержа изменений локального репозитория на удаленный и pull для обратной операции.

### **Изучить справку для основных команд**

Ниже приведен список самых используемых команд:

1. git add  
Добавление файла в репозиторий
2. git clone  
Создать локальную копию репозитория
3. git push  
Обновление удаленного репозитория
4. git pull  
Обновление локального репозитория
5. git rm  
Удаление файла из индекса
6. git diff  
Отображение внесенных в файл изменений
7. git merge  
Слияние двух и более веток разработки в одну.

### **Получение содержимого репозитория**

Для получения содержимого репозитория необходимо сделать его копию:

```
git clone https://github.com/CrookedNail/InfoSecCourse2015.git
```

### **Добавить новую папку и первого файла под контроль версий**

```
git add /latex
git add lab1.tex
```

### **Зафиксировать изменения в локальном репозитории**

Перед тем как зафиксировать изменения удалим ненужные файлы:

```
git rm myfirst.pdf
...
```

Фиксация:

```
git commit -a -m "Lab1"
```

### **Внести изменения в файл и посмотреть различия**

Для этого добавим в репозиторий файл текущего отчета, зафиксируем изменение, после чего сравним с измененной версией. Изменения можно просматривать между текущим состоянием и коммитом, между любыми двумя коммитами, также можно уточнять изменения какого файла именно нам интересны.

```
git diff a946e06 13507d8 lab2_patrakov.tex
```

### **Отменить локальные изменения**

```
git reset - -hard a946e06
```

### **Внести изменения в файл и посмотреть различия**

```
git diff lab2_patrakov.tex
```

### **Зафиксировать изменения в локальном репозитории, зафиксировать изменения в центральном репозитории**

```
git commit -a -m "testing push"
git push origin
```

### **Получить изменения из центрального репозитория**

```
git pull
```

### **Поэкспериментировать с ветками**

Создание новой ветки

```
git branch test
```

Переключение на другую ветку

```
git checkout test
```

Слияние

```
git merge test
```

Удаление ветки после слияния  
`git branch -d test`

Просмотр списка локальных веток  
`git branch -a`

### 3 Выводы

Git – это распределенная система контроля версий. Она позволяет вести разработку более эффективно не зависимости от размера проекта. Благодаря СКР не нужно беспокоиться о сохранности собственного проекта. Параллельно несколько разработчиков могут вести разработку над одним проектом, таким образом более эффективно кооперируясь. В какой то момент кто-то из разработчиков может выделиться и пробовать свои собственные решения в другой ветке. Все эти плюсы делают Git незаменимым инструментом при разработке приложений любой сложности.