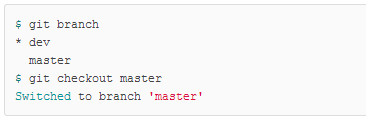
Lesson5

发布一个版本时，我们通常先在版本库中打一个标签，这样，就唯一确定了打标签时刻的版本。将来无论什么时候，取某个标签的版本，就是把那个打标签的时刻的历史版本取出来。所以，标签也是版本库的一个快照。

Git的标签虽然是版本库的快照，但其实它就是指向某个commit的指针（跟分支很像对不对？但是分支可以移动，标签不能移动），所以，创建和删除标签都是瞬间完成的。

**创建标签**

首先，切换到需要打标签的分支上：



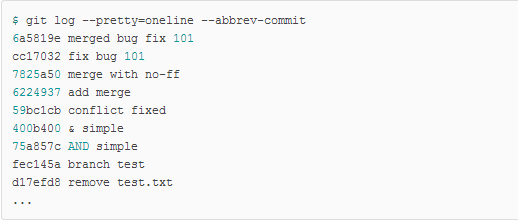
然后，敲命令git tag <name>就可以打一个新标签：



可以用命令git tag查看所有标签：



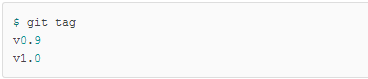
默认标签是打在最新提交的commit上的。如若想打上之前的日期的标签的话，方法是找到历史提交的commit id，然后打上就可以了：



比方说要对add merge这次提交打标签，它对应的commit id是6224937，敲入命令：



再用命令git tag查看标签：



注意，标签不是按时间顺序列出，而是按字母排序的。可以用git show <tagname>查看标签信息：



可以看到，v0.9确实打在add merge这次提交上。

还可以创建带有说明的标签，用-a指定标签名，-m指定说明文字：



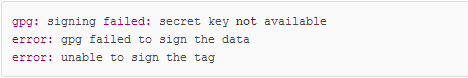
用命令git show <tagname>可以看到说明文字：



还可以通过-s用私钥签名一个标签：



签名采用PGP签名，因此，必须首先安装gpg（GnuPG），如果没有找到gpg，或者没有gpg密钥对，就会报错：



如果报错，请参考GnuPG帮助文档配置Key。

用命令git show <tagname>可以看到PGP签名信息：



用PGP签名的标签是不可伪造的，因为可以验证PGP签名。验证签名的方法比较复杂，这里就不介绍了。

小结

* 命令git tag <name>用于新建一个标签，默认为HEAD，也可以指定一个commit id；
* git tag -a <tagname> -m "blablabla..."可以指定标签信息；
* git tag -s <tagname> -m "blablabla..."可以用PGP签名标签；
* 命令git tag可以查看所有标签。

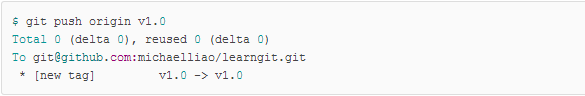
**操作标签**

如果标签打错了，也可以删除：



因为创建的标签都只存储在本地，不会自动推送到远程。所以，打错的标签可以在本地安全删除。

如果要推送某个标签到远程，使用命令git push origin <tagname>：



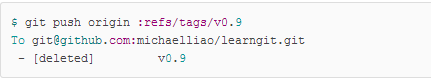
或者，一次性推送全部尚未推送到远程的本地标签：



如果标签已经推送到远程，要删除远程标签就麻烦一点，先从本地删除：



然后，从远程删除。删除命令也是push，但是格式如下：



要看看是否真的从远程库删除了标签，可以登陆GitHub查看。

### 小结

* 命令git push origin <tagname>可以推送一个本地标签；
* 命令git push origin --tags可以推送全部未推送过的本地标签；
* 命令git tag -d <tagname>可以删除一个本地标签；
* 命令git push origin :refs/tags/<tagname>可以删除一个远程标签。