GIT升级公告（GitOsis->Gitlab）

GIT服务器比较：

**[Git](http://lib.csdn.net/base/28" \t "_blank" \o "Git知识库)**服务器比较有名的是gitosis和gitolite，这两个管理和使用起来稍微有些复杂，没有web页面，而gitlab则是类似于github的一个工具，github无法免费建立私有仓库，并且为了代码安全，于是在内网升级了git服务器，多方比较，选择了gitlab进行升级。

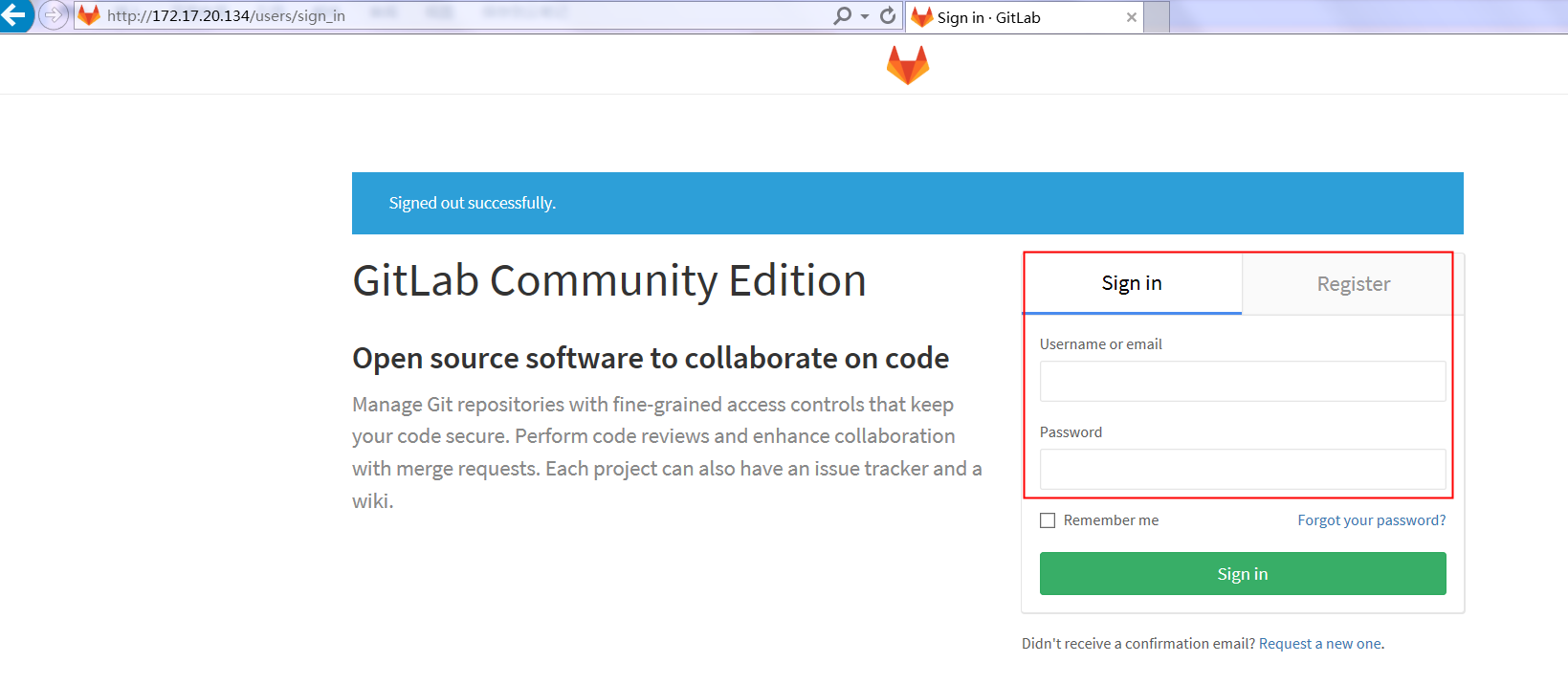
**Gitosis与Gitlab比较**

|  |  |  |
| --- | --- | --- |
|  | **Gitosis** | **Gitlab** |
| 部署系统 | CentOS7.0 | CentOS7.0 |
| web访问 | 无web端 | 有web端 |
| 协议类型 | ssh | ssh、http |
| 图形统计 | 无 | 有 |
| Merage Request | 无 | 有 |
| 用户权限级别 | R、w | Guest、Reproter、Developer、Master |
| 项目组（group） | 无 | 有 |
| 运维便捷 | 命令行 | web界面 |

三、GITLab服务器访问：

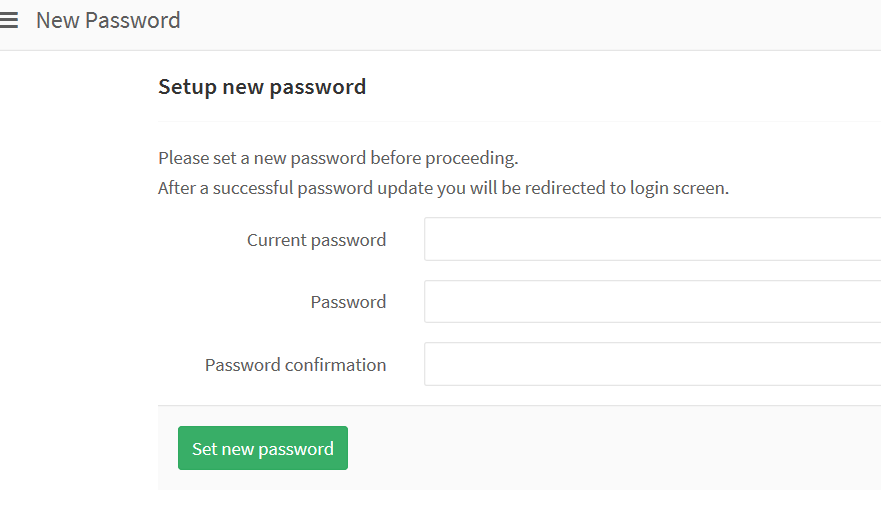
1、访问地址：

<http://172.17.20.117/users/sign_in>



输入管理员给新建的git账号和初始密码，登陆gitlab的web界面。

2、修改密码：

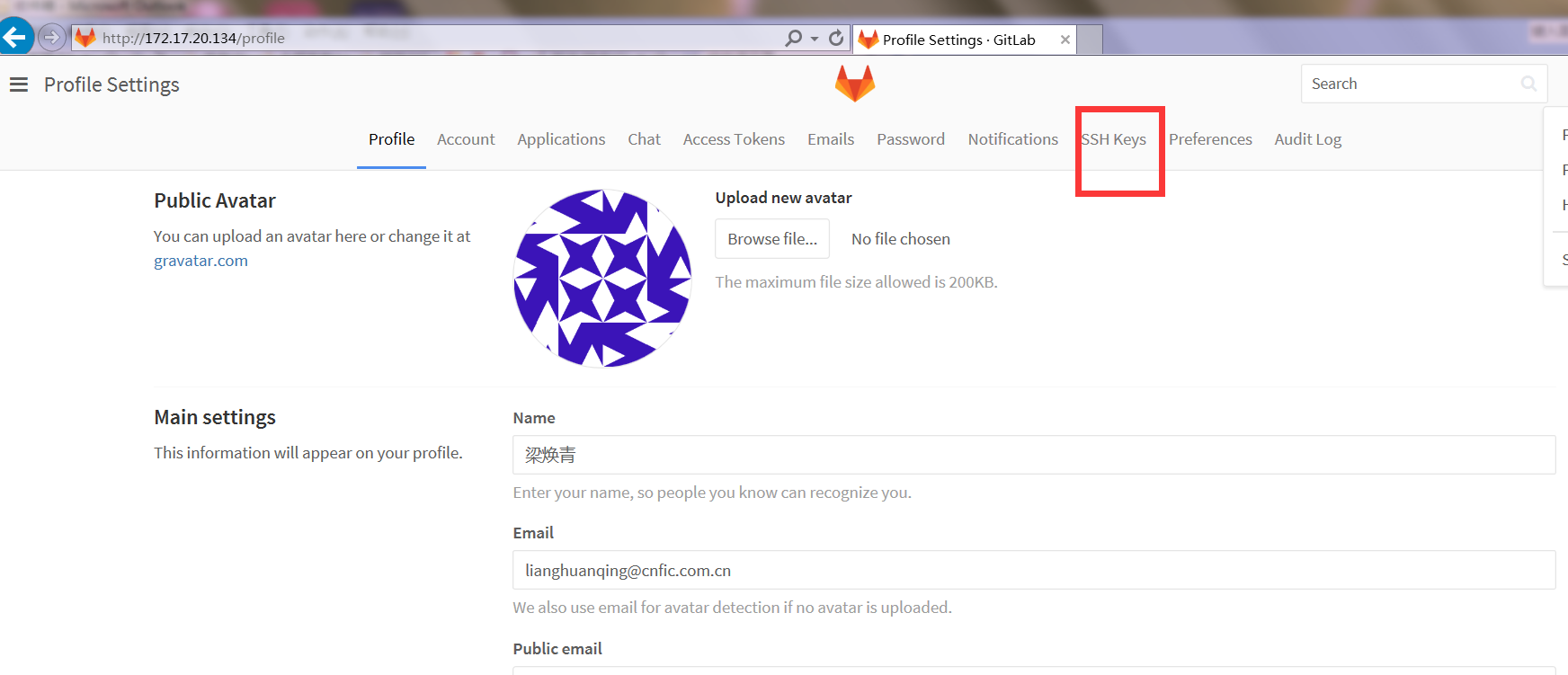


密码更新完毕之后，需要重新登陆系统，再次登陆进来可以看到自己所在的项目。

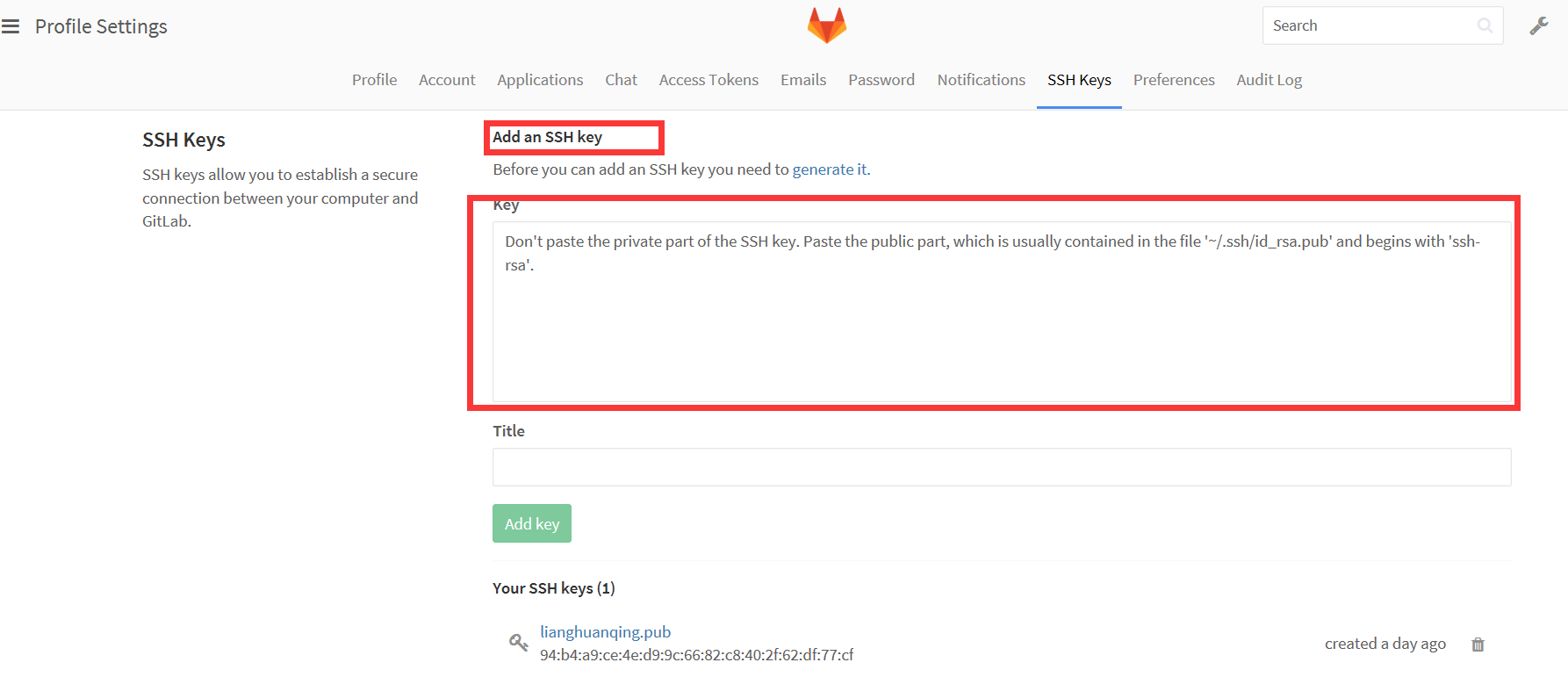
四、GIT用户添加公钥和私钥：

1、gitlab界面添加公钥

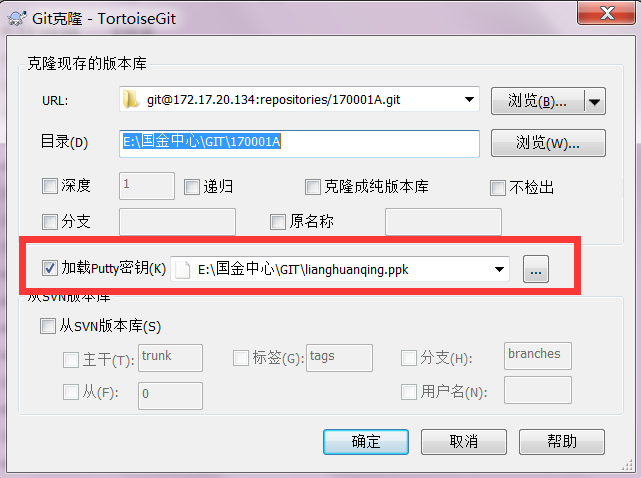




登陆系统🡪 Profile Settings🡪SSH Keys🡪 Add an SSH key,添加公钥。



2、本机保存私钥



五、获取远程仓库：

1、ssh协议：

例如：[git@172.17.20.117:repositories/appfrontend.git](mailto:git@172.17.20.117:repositories/appfrontend.git)

Git clone （pull、push）都需要用公钥、私钥认证。

2、http协议：

例如：<http://172.17.20.117/repositories/appfrontend.git>

Git clone （pull、push）不需要用公钥、私钥认证；但是需要用git用户名和密码（登陆gitlab的git账号和密码）。

六、现有本地库升级-实例操作：

场景：发人员侯志乾现在负责支付平台的项目开发工作，迁移之后如何操作。

6.1 SSH协议（同gitosis现用的协议）

**1、支付平台本地库位置：**

/home/git/git117/170004A

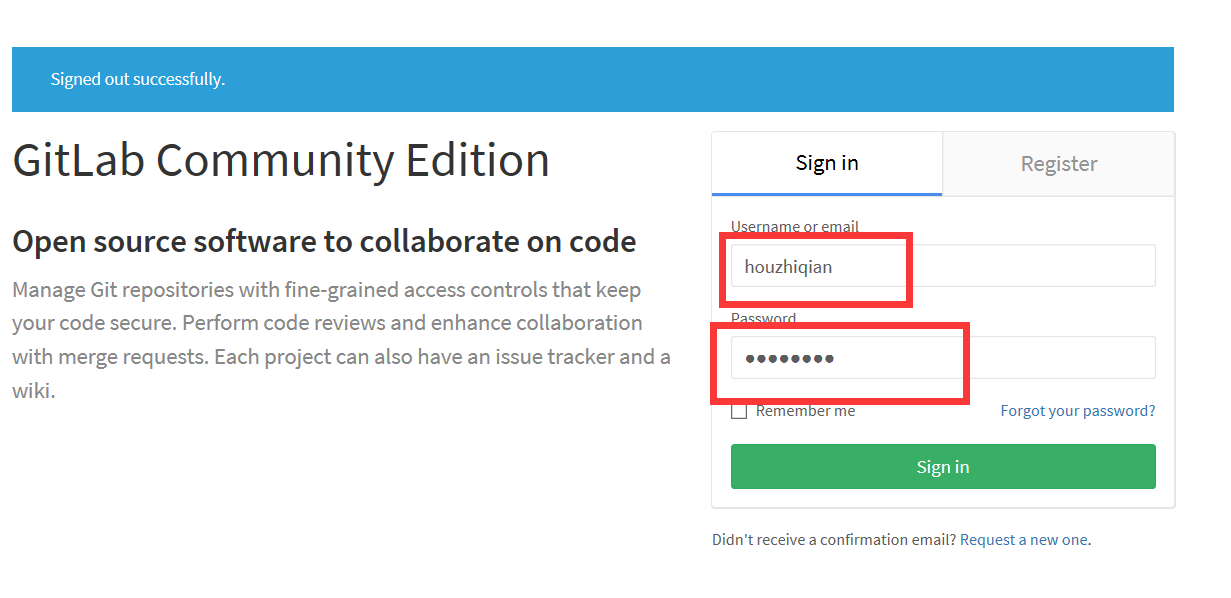
**2、支付平台远程库地址：**

[git@localhost 170004A]$ git remote -v

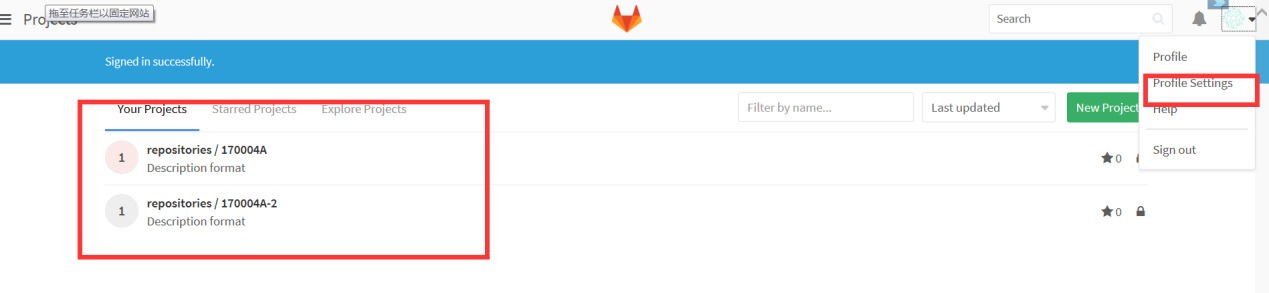
origin git@172.17.20.117:/git/repositories/170004A.git (fetch)

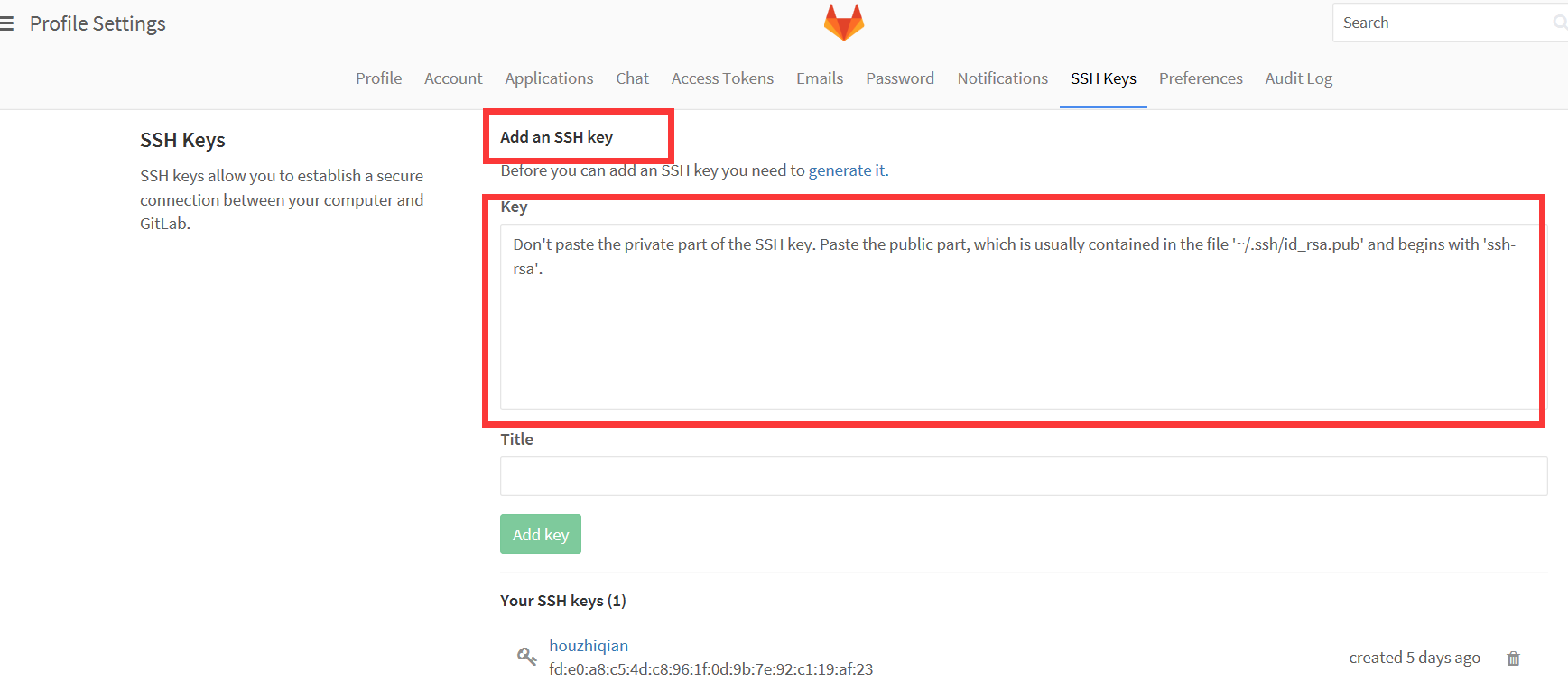
origin git@172.17.20.117:/git/repositories/170004A.git (push)

**3、用新的git用户名和密码登陆gitlab web界面,添加公钥：**



登录系统以后，可以看到自己所属项目，并从Profile Settings🡪SSH Keys🡪 Add an SSH key添加公钥：





**4、本地库保存私钥：**

这样的话，公钥与私钥进行了配对成功。

**5、检查本地库是否与远程库代码同步：**

在本地库通过git log查看提交日志和哈希值是否与远程库的提交日志和哈希值一致。

1. 结果一致，则不需要任何操作，直接进行正常工作。
2. 结果不一致，发现本地库有新增提交，则需要先push到远程库，再进行正常工作。

6.2 HTTP协议（gitlab新增的协议）

**1、支付平台本地库位置：**

/home/git/git117/170004A

**2、支付平台远程库地址：**

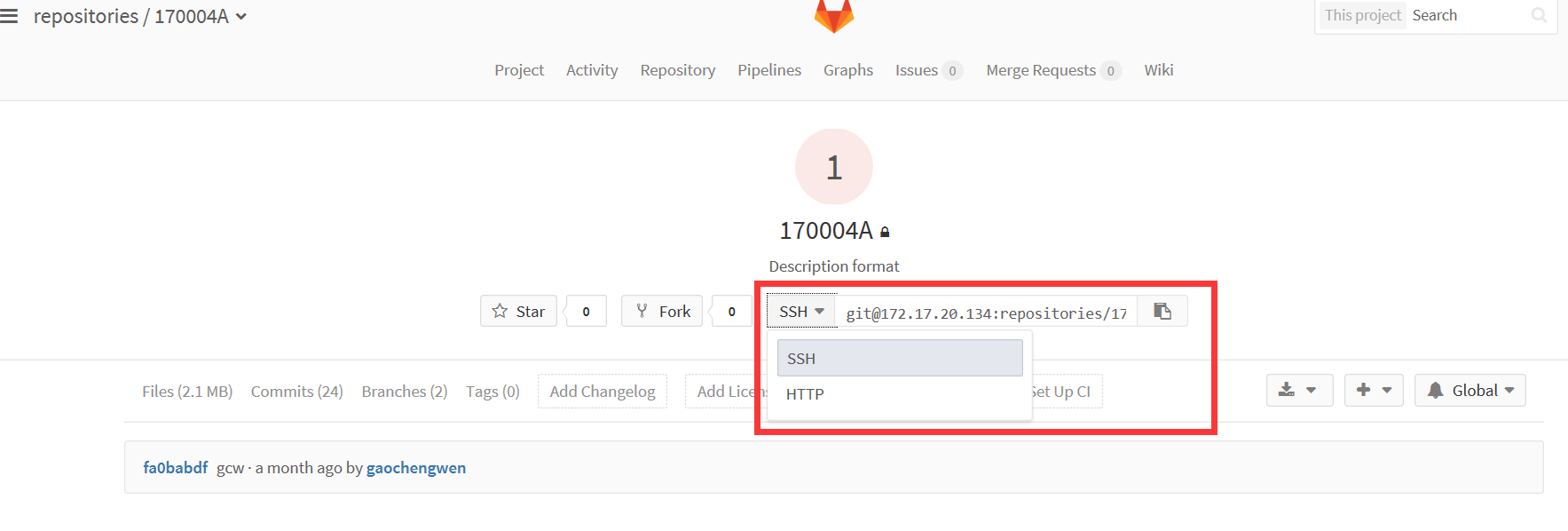
[git@localhost 170004A]$ git remote -v

origin git@172.17.20.117:/git/repositories/170004A.git (fetch)

origin git@172.17.20.117:/git/repositories/170004A.git (push)

**3、修改平台远程库协议：**

**通过gitlab web界面能看到远程库170004A支持两种协议：**

****

[git@localhost 170004A]$git remote set-url origin <http://172.17.20.134/repositories/170004A.git>

则改为了http协议的库地址：

[git@localhost 170004A]$ git remote -v

origin http://172.17.20.134/repositories/170004A.git (fetch)

origin http://172.17.20.134/repositories/170004A.git (push)

**4、检查本地库是否与远程库代码同步：**

**使用http协议进行git clone、pull和push时，是需要输入git的用户名和密码**

在本地库通过git log查看提交日志和哈希值是否与远程库的提交日志和哈希值一致。

1. 结果一致，则不需要任何操作，直接进行正常工作。
2. 结果不一致，发现本地库有新增提交，则需要先push到远程库，再进行正常工作。