

## 一、 Windows 需安装 GitBash(下载后一路下一步即可安装成功)

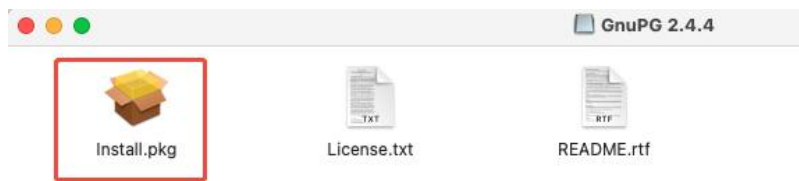
<http://192.144.220.147:9000/bin/Git-2.42.0.2-64-bit.exe>

## 二、 Mac 需要安装 gunpg

### 1、gunpg 下载

<http://192.144.220.147:9000/bin/GnuPG-2.4.4.dmg>

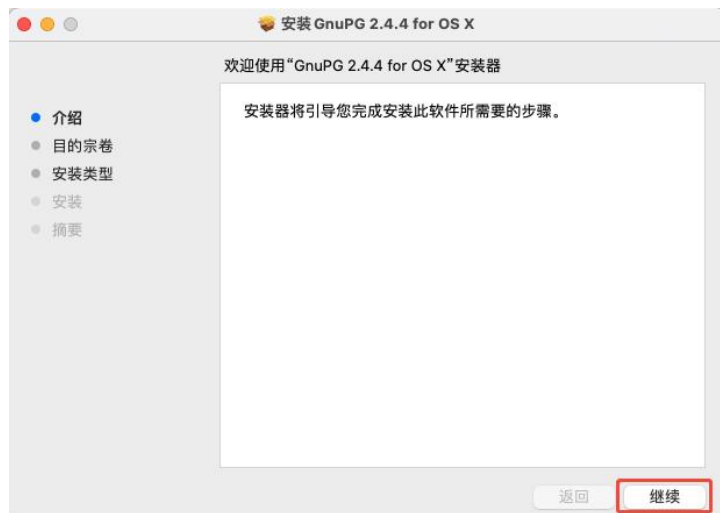
### 2、打开如下图



### 3、如遇阻止，请按下图操作



### 4、接下来一路下一步



### 三、 生成密钥


`gpg --quick-generate-key 改为你的邮箱@duole.com rsa2048 sign,encrypt,auth 0y`

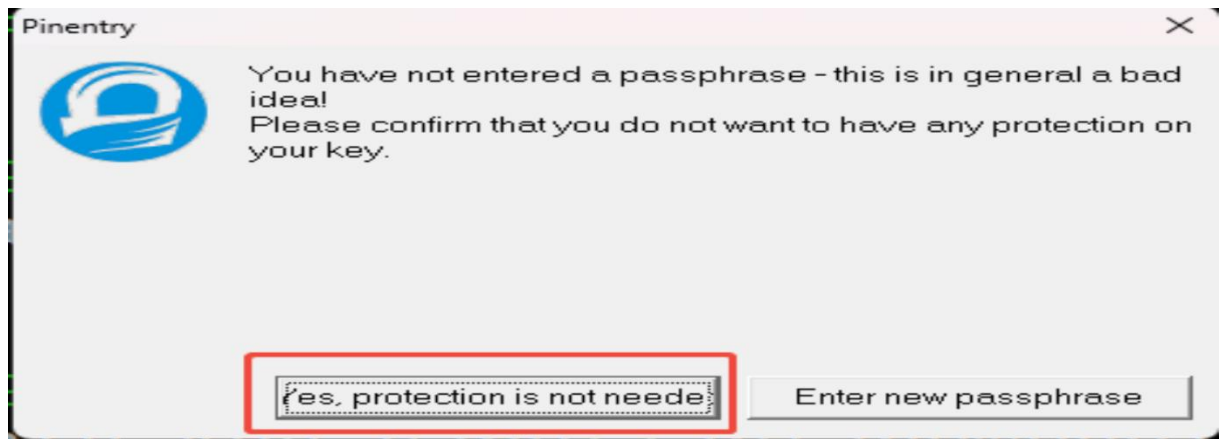
```
$ gpg --quick-generate-key yangtianfang@duole.com rsa2048 sign,encrypt,auth 0y
We need to generate a lot of random bytes. It is a good idea to perform
some other action (type on the keyboard, move the mouse, utilize the
disks) during the prime generation; this gives the random number
generator a better chance to gain enough entropy.
gpg: key 0600813F118698C4 marked as ultimately trusted
gpg: directory '/c/Users/younger/.gnupg/openpgp-revocs.d' created
gpg: revocation certificate stored as '/c/Users/younger/.gnupg/openpgp-revocs.d/B526587B7DA61603461764030600813F118698C4.rev'
public and secret key created and signed.

pub   rsa2048 2024-12-28 [SCEA]
      B526587B7DA61603461764030600813F118698C4
uid
      yangtianfang@duole.com
```

tips: 密码这里建议为空

```
younger@LAPTOP-SI4CG:~$ gpg -k
$ gpg --quick-generate-key yangtianfang@duole.com 2048 0
gpg: Key generation
gpg: Key generation
$ gpg --quick-generate-key yangtianfang@duole.com rsa2048 sign,encrypt,auth 0y
We need to generate a lot of random bytes. It is a good idea to perform
some other action (type on the keyboard, move the mouse, utilize the
disks) during the prime generation; this gives the random number
generator a better chance to gain enough entropy.
```





#### 四、查看密钥

```
gpg --list-keys
```

```
$ gpg --list-keys
/c/Users/younger/.gnupg/pubring.kbx
-----
pub      rsa2048 2024-12-28 [SCEA]
          B526587B7DA61603461764030600813F118698C4
uid      [ultimate] yangtianfang@duole.com
```

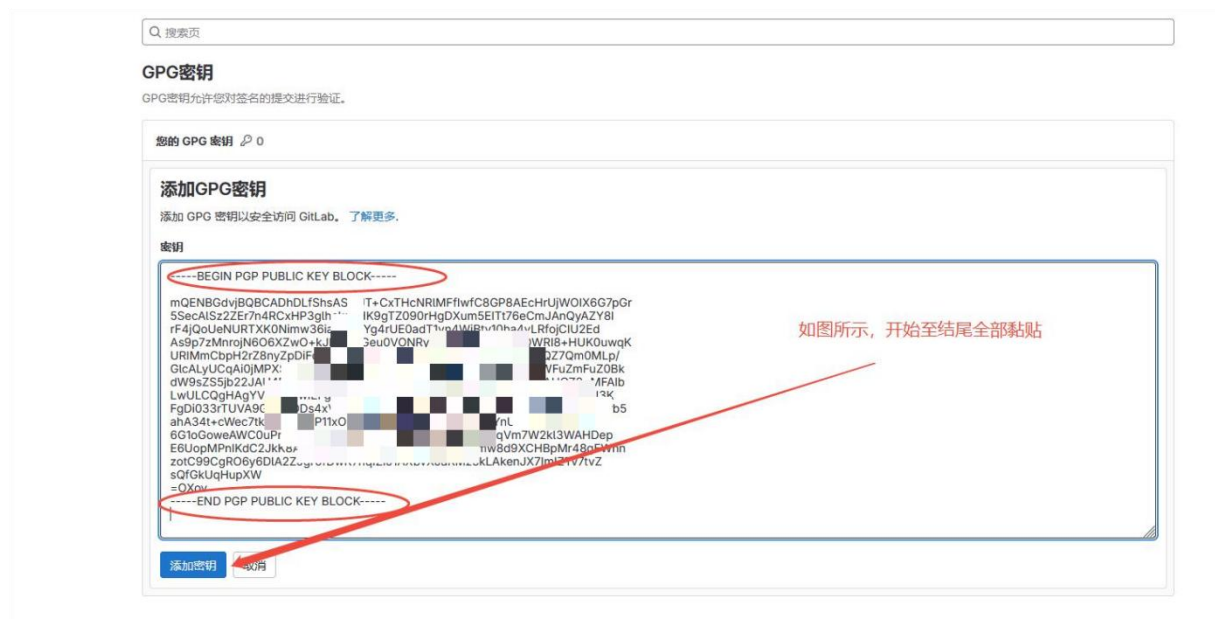
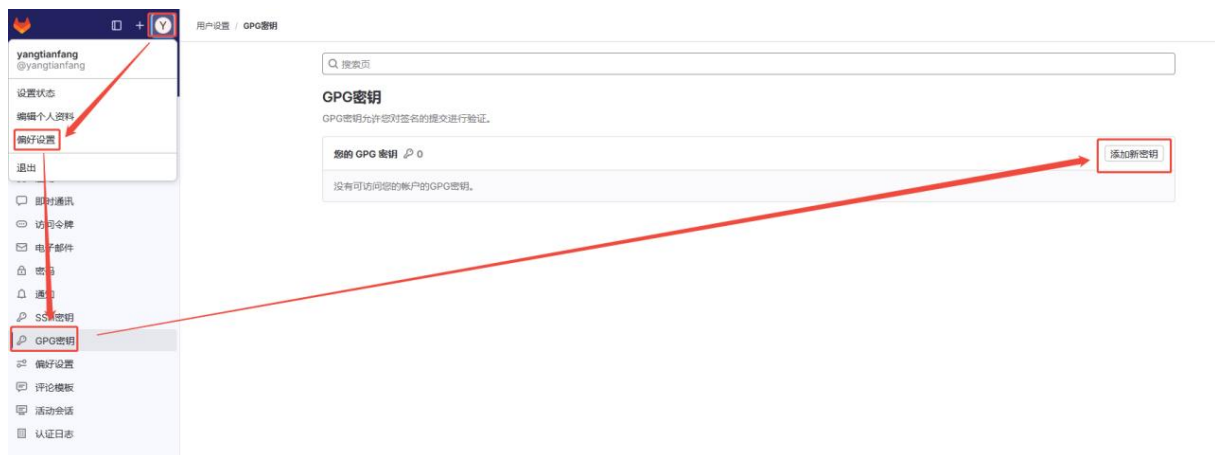
#### 五、导出公钥

```
gpg --export --armor 改为你的邮箱@duole.com
```

```
$ gpg --export --armor yangtianfang@duole.com
-----BEGIN PGP PUBLIC KEY BLOCK-----

mQENBGdvjBQBCADhDLfShsASCUT+CxTHcNRlMFfiwfc8GP8AEChRujWOIX6G7pGr
5SecAlsz2ZEr7n4RCxHP3glhcksuHK9gTZ090rHgDXum5EITt76eCmJAnQyAZY8I
rF4jQoU...
As9p7zMnrojN6n6x7wn-...
URIMmCbph2...
GlcALyUCq...
dw9sZS5jb...
LwULCQgHAgY...
FgDi033rTUVAg...
ahA34t+cWec7tkcTSL...
6G1oGoweAWC0uPr69wWA+cIqmi/sxId7dzvb1RhTG0tjy7sjqVm7w2kl3WAHDep
E6UopMPnIKdC2J...
zotC99CgRO6y6D1A2ZJgi...
sqfGkUqHupXw
=OXov
-----END PGP PUBLIC KEY BLOCK-----
```

## 六、打开 gitlab 网站，添加密钥



## 七、接下来我们在返回客户端进行调配哈

tips: 以下修改全局生效（如想单个仓库生效请在项目目录中执行把--global 改为--local）

```
git config --global user.email 改为你的邮箱@duole.com
```

```
git config --global user.signingkey 改为你的邮箱@duole.com
```

```
git config --global commit.gpgsign true
```

```

younger@LAPTOP-SI4CGGRB MINGW64 ~/Desktop/yunwei (master)
$ git config --global user.email yangtianfang@duole.com

younger@LAPTOP-SI4CGGRB MINGW64 ~/Desktop/yunwei (master)
$ git config --global user.signingkey yangtianfang@duole.com

younger@LAPTOP-SI4CGGRB MINGW64 ~/Desktop/yunwei (master)
$ git config --global commit.gpgsign true

```

## 八、验证结果

验证配置是否成功，可以检查以下位置的状态

The top screenshot shows the repository settings page for a repository named 'aaaa'. The 'Auto DevOps' section is highlighted, indicating that the CI/CD configuration is successful. The 'Commit Signing' section shows that the repository is signed by 'yangtianfang' with a green '已验证' (Verified) status.

The bottom screenshot shows the commit history page for the same repository. The commit history is filtered by the 'master' branch. The first commit, created by 'yangtianfang' 4 hours ago, is highlighted with a green '已验证' (Verified) status, indicating that the commit signing is successful.