

微信：



微博：



全国统一咨询电话：400-0088-518

1、课程名称：标签管理



2、知识点

- 1、 如果要想学习本章之前请一定已经很好的掌握了分支的各个知识，以及熟练的应用；
- 2、 在本次课程之中将为大家讲解标签的创建与管理；
- 3、 使用 GPG 工具进行标签的加密处理。

3、具体内容

3.1、创建标签

标签可以简单的理解为属于分支定义的别名，分支本身都会进行指针的配置（分支都会指向某一个 commit id），但是标签却是一个固定的内容，可以说，标签永远指向一个 commit id。

微信：



微博：



全国统一咨询电话：400-0088-518

标签定义

- 标签可以标记某一提交点；
- 定义标签：
 - 为master定义标签： `git tag v1.0`；
 - 为指定的commitid定义标签： `git tag v0.7 commitid`；

优拓教育 (www.yootk.com) & 魔乐科技 (www.mldnjava.cn) 联合出品

现在加入说已经存在有两个分支。

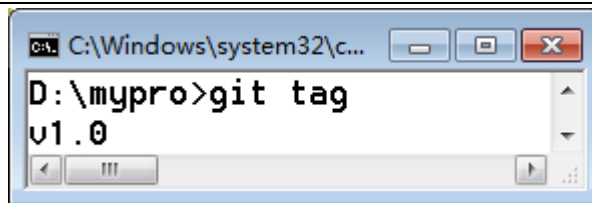
范例：为 master 定义一个标签

- master 是程序最终的发布版本，可以说，master 就是完整的开发程序。

```
git tag v1.0
```

范例：察看所有的标签

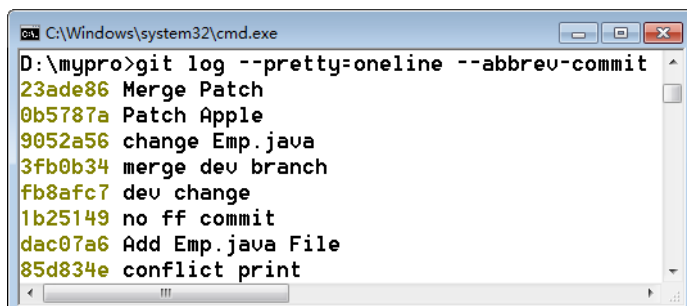
```
git tag
```



但是这个标签也只是针对于当前的 master 分支所打上的标签，可是在一个项目之中一定会有许多的提交点，那么如果要为之前的提交点打上标签该怎么做呢？只需要找到 commit id 即可。

范例：查询日志

```
git log --pretty=oneline --abbrev-commit
```



微信：



微博：



全国统一咨询电话：400-0088-518

在这些日志上都会存在有 commit id，那么下面针对一个 commit id 进行标签的定义。

范例：为“9052a56”设置标签

```
git tag v0.6 9052a56
```

```
C:\Windows\system32\cmd.exe
D:\mypro>git tag v0.6 9052a56

D:\mypro>git tag
v0.6
v1.0
```

但是此时只是显示一些很简单的标签信息并不是很好，所以下面希望可以察看标签的完整内容。

范例：察看标签的完整信息

```
git show v0.6
```

```
C:\Windows\system32\cmd.exe
D:\mypro>git show v0.6
commit 9052a565e93112815e250d05ab6e85d937 Author: yootk <mldnqa@163.com>
Date: Mon Jan 5 13:17:09 2015 +0800

    change Emp.java

diff --git a/Emp.java b/Emp.java
index 419e076..052a225 100644
--- a/Emp.java
+++ b/Emp.java
@@ -1,5 +1,7 @@
 import java.io.* ;
+import java.util.* ;
     private Integer empno ;
     private String ename ;
+    private Date hiredate ;
+
 }
\ No newline at end of file
```

在之前所建立的标签实际上都没有设置注释信息，那么秒年可以定义一些注释信息。

范例：添加标签时设置注释

```
git tag -a v0.7 -m "test version" 0b5787a
```

随后再次查询标签的详细信息。

```
C:\Windows\system32\cmd.exe
D:\mypro>git show v0.7
tag v0.7
Tagger: yootk <mldnqa@163.com>
Date: Mon Jan 5 16:12:47 2015 +0800

test version

commit 0b5787ac3dd38e7f8f8b54c35e8d0676c3bf0802
Author: yootk <mldnqa@163.com>
```

微信：



微博：



全国统一咨询电话：400-0088-518

3.2、利用 GPG 实现标签加密

标签创建完成之后如果开发者只希望可以由自己来修改标签，那么就必须进行加密的处理，而可以使用 GPG 工具实现加密操作。下载地址：<https://www.gnupg.org/>

GPG工具

➤ 下载地址：<https://www.gnupg.org>



优拓教育 (www.yootk.com) & 魔乐科技 (www.mldnjava.cn) 联合出品

在 GPG 工具出现之前，有一个 PGP 的开发工具（收费），所以后来由 GNU 开发出了一个新的 PGP 工具的替代品，而这个名称就是 GnuPG，后来简称为 GPG 工具。

网盘地址：<http://pan.baidu.com/s/1hq3TeEK>

3.2.1、GPG 工具的基本使用

如果拿到了 GPG 工具，那么用户可以直接进行安装。而 GPG 工具的所有操作都可以在命令行方式下完成。

范例：测试安装是否成功

```
gpg -help
```

微信：



微博：



全国统一咨询电话：400-0088-518

```
C:\Windows\system32\cmd.exe
D:\>gpg -help
gpg (GnuPG) 2.0.26 (Gpg4win 2.2.3)
libgcrypt 1.6.2
Copyright (C) 2013 Free Software Foundation, Inc.
License GPLv3+: GNU GPL version 3 or later <http://gnu.org/licenses/gpl.html>
This is free software: you are free to change and redistribute it.
There is NO WARRANTY, to the extent permitted by law.

Home: C:\Users\yootk\AppData\Roaming\gnupg
Supported algorithms:
Pubkey: RSA, RSA, RSA, ELG, DSA
Cipher: IDEA, 3DES, CAST5, BLOWFISH, AES, AES192, AES256, TWOFISH,
        CAMELLIA128, CAMELLIA192, CAMELLIA256
Hash: MD5, SHA1, RIPEMD160, SHA256, SHA384, SHA512, SHA224
Compression: Uncompressed, ZIP, ZLIB, BZIP2

Syntax: gpg [options] [files]
Sign, check, encrypt or decrypt
Default operation depends on the input data
```

如果出现以上的提示信息则表示 gpg 安装成功的，但是下面通过一个简单的文件加密和解密处理，来为大家说明一下 GPG 工具的使用。

1、 生成自己的密钥。

```
gpg --gen-key

C:\Windows\system32\cmd.exe - gpg --gen-key
D:\>gpg --gen-key
gpg (GnuPG) 2.0.26; Copyright (C) 2013 Free Software Foundation, Inc.
This is free software: you are free to change and redistribute it.
There is NO WARRANTY, to the extent permitted by law.

gpg: keyring `C:/Users/yootk/AppData/Roaming/gnupg/secring.gpg` created
gpg: keyring `C:/Users/yootk/AppData/Roaming/gnupg/pubring.gpg` created
Please select what kind of key you want:
  (1) RSA and RSA (default)
  (2) DSA and Elgamal
  (3) DSA (sign only)
  (4) RSA (sign only)
Your selection? _
```

这些是一些版权声明，随后会询问用户使用是什么样的加密算法，现在使用 RSA 算法（默认的选择）。

```
C:\Windows\system32\cmd.exe - gpg --gen-key
RSA keys may be between 1024 and 4096 bits long.
What keysize do you want? (2048)
```

随后会询问用户的密钥长度（理论上密钥越长，那么安全性就越高）默认的密钥长度为 2048 位。

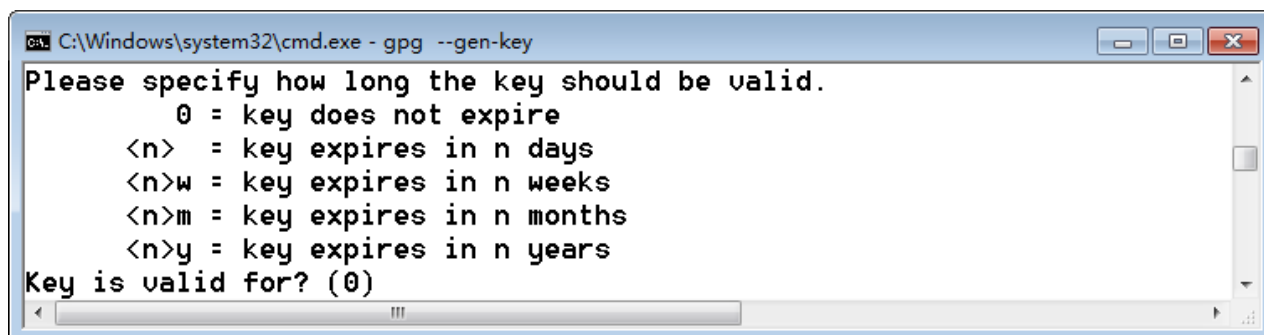
微信：



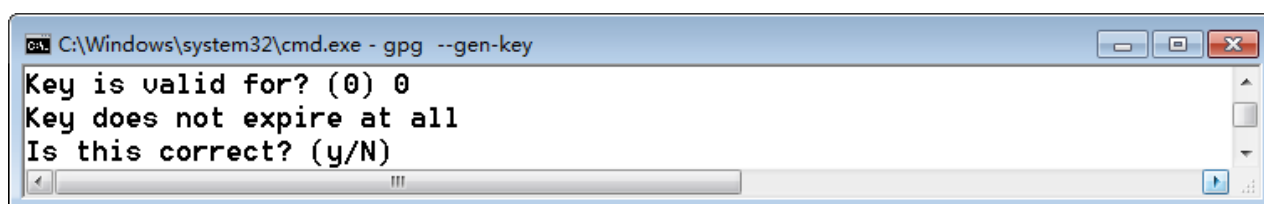
微博：



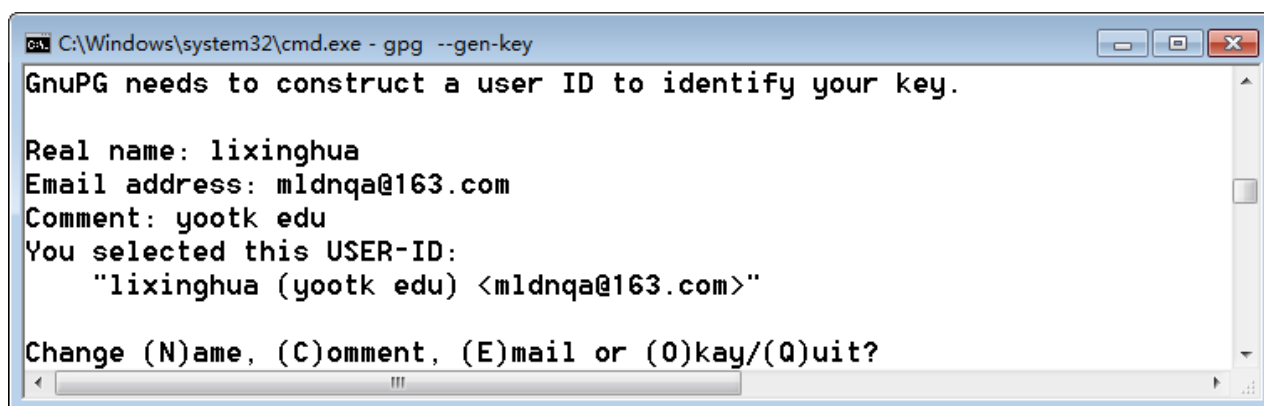
全国统一咨询电话：400-0088-518



随后会出现一个界面，询问用户密钥的有效期，本次选择密钥为永不过期。

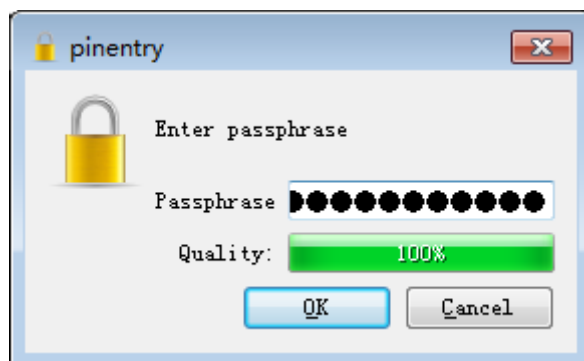


本次输入“y”表示确认以上的配置信息，那么随后会让用户输入自己的个人信息。



而后接着就会询问你是否需要进行修改，由于信息已经确认过了，所以本次直接输入“o”。

随后会出现一个提示框，主要的目的是让用户设置一个私钥的密码，为了防止有人恶意修改而提供的，本次输入的密码为“www.yootk.com”



那么这个时候就会出现一系列的提示信息了。

微信：



微博：



全国统一咨询电话：400-0088-518

```
C:\Windows\system32\cmd.exe
disks) during the prime generation; this gives the random number
generator a better chance to gain enough entropy.
gpg: C:/Users/yootk/AppData/Roaming/gnupg/trustdb.gpg: trustdb created
gpg: key 57E54640 marked as ultimately trusted
public and secret key created and signed.

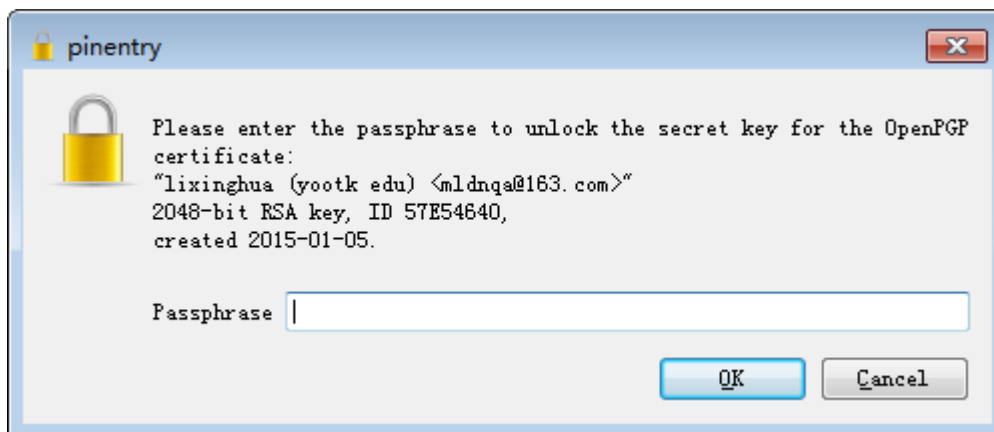
gpg: checking the trustdb
gpg: 3 marginal(s) needed, 1 complete(s) needed, PGP trust model
gpg: depth: 0 valid: 1 signed: 0 trust: 0-, 0q, 0n, 0m, 0f, 1u
pub 2048R/57E54640 2015-01-05
    Key fingerprint = C53B 6D79 BE94 CA3E 3E21 4D3E 0DAA C700 57E5 4640
uid [ultimate] lixinghua (yootk edu) <mldnqa@163.com>
sub 2048R/B8670B50 2015-01-05
```

那么在这里面有几个重要的信息：

- 密钥号：57E54640；
- 用户 ID：lixinghua (yootk edu) <mldnqa@163.com>（真实姓名 注释 邮箱）

2、实际上为了日后的管理方便，最好在生成一张撤消的整数，为的是以后如果密钥过期可以通过服务器进行撤消。

```
gpg --gen-revoke 57E54640
```



这个时候会自动的生成以下的密钥信息。

```
C:\Windows\system32\cmd.exe
-----BEGIN PGP PUBLIC KEY BLOCK-----
Version: GnuPG v2
Comment: A revocation certificate should follow

iQEkBCABAgAQBQJUqkudBx0AeW9udGsACgkQDarHAFf1RkBbIgf+KMOCJFdEck0r
3Uq4RUR0M4WQ9f0tPlp+P0mPdPdZu6eX4gek2sAPbncH255oZkyPRJ+opWjjXRg0
nTrKCcMAXjmQf1nF10I+Apt1vIDUhmuytY5unGMTKme2Y6RtOULf1Fmjziq0GIR
Ju3WH0+XFzjgFEonIr1X3vTUwt+HUXJMcTSESTX/UQduYiPi6A6Fth4ptKiZlawC
Gg9TH2X1omuDsLcGP+MiFa9IYsoc0x+RSuhudPIRbvo07sPYF0WHGzROTDALbNX
qU67JLKru+a14gUF2JiKZcRpI3RTq/JQiScqWx6Xji3EWJDYHZKeidyH8KI60cpS
rFFPB7XSwa==
=itsn
-----END PGP PUBLIC KEY BLOCK-----
```

3、列出所有的密钥。

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

微信：



微博：



全国统一咨询电话：400-0088-518

```
C:\Windows\system32\cmd.exe
D:\>gpg --list-keys
C:/Users/yootk/AppData/Roaming/gnupg/pubring.gpg
-----
pub 2048R/57E54640 2015-01-05
uid [ultimate] lixinghua (yootk edu) <mldnqa@163.com>
sub 2048R/B8670B50 2015-01-05
```

4、实现简单的文件加密与解密处理。

- 现在在 D 盘下有一个 demo.txt 的文件；

世界，你好！
hello world
www.yootk.com

- 为了安全起见在进行传输的时候应该为其进行加密处理。
- 实现加密处理

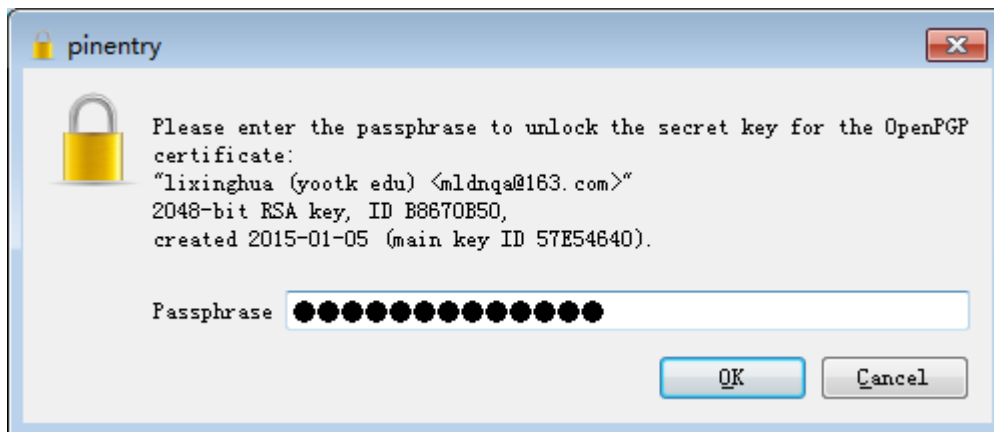
```
gpg --recipient 57E54640 --output demo.en.txt --encrypt demo.txt
```

此时的 demo.txt 的文件已经加密到了 demo.en.txt 文件之中。

- 进行加密文件的解密操作

```
gpg demo.en.txt
```

此时要进行解密处理，而后会提示用户输入的之前的操作密码。



```
C:\Windows\system32\cmd.exe
2048-bit RSA key, ID B8670B50, created 2015-01-05 (main key ID 57E5
gpg: encrypted with 2048-bit RSA key, ID B8670B50, created 2015-01-
"lixinghua (yootk edu) <mldnqa@163.com>"
gpg: demo.en.txt: unknown suffix
Enter new filename [demo.txt]: demo.de.txt
```

所以，GPG 的工具实际上使用起来都是很容易的。

微信：



微博：



全国统一咨询电话：400-0088-518

3.2.2、使用 GPG 生成标签

在操作之前一定要保证你的电脑上已经配置好了 GPG 工具。

现在已经有了一个生成的密钥，名称为“57E54640”，所以下面进行加密标签，标签的加密只需要增加一个“-s”的参数即可。

范例：使用 GPG 加密标签

```
git tag -s v0.9 -m "gpg handle tag" 23ade86
```

```
C:\Windows\system32\cmd.exe
D:\mypro>git tag -s v0.9 -m "gpg handle tag" 23ade86
gpg: skipped "yootk <mldnqa@163.com>": secret key not available
gpg: signing failed: secret key not available
error: gpg failed to sign the data
error: unable to sign the tag
```

但是此时发现并不能够进行加密处理。我们需要在加密时使用一个指定签名，所以实际上的标签的加密处理应该输入如下的指令完成。

```
git tag -u "lixinghua" -s v0.9 -m "gpg handle tag" 23ade86
```

```
C:\Windows\system32\cmd.exe
D:\mypro>git tag -u "lixinghua" -s v0.9 -m "gpg handle tag" 23ade86

You need a passphrase to unlock the secret key for
user: "lixinghua (yootk edu) <mldnqa@163.com>"
2048-bit RSA key, ID 57E54640, created 2015-01-05
```

所以此时，其他的开发者是不可能修改此标签的。

范例：查询 v0.9 标签

```
git show v0.9
```

微信：



微博：



全国统一咨询电话：400-0088-518

```
C:\Windows\system32\cmd.exe
D:\mypro>git show v0.9
tag v0.9
Tagger: yootk <mldnqa@163.com>
Date:   Mon Jan 5 16:44:48 2015 +0800

gpg handle tag
-----BEGIN PGP SIGNATURE-----
Version: GnuPG v1.4.13 (MingW32)

iQEcbAABAgAGBQJUqk8AAAOJEA2qxwBX5UZAYW0H/0bjJT+EYdS46GSTkQUL13pX
THRtkvW3sI1BuHdTff5p8MmRsQ1CXQw/9UE2ZErU1Wcz6I9rZMJzYY7Tk+JJU2yQ
ksD3o304MqM0Wo3fU9DdJi7J3EeY5IQm61uqPaqIGfnWP/I+DgfuLELCHaF/Y0e8
RTWrQwg0mWtXfq3nFUfmYiUzMvEFRxmLBhCYi20QK1vQt2BBDGn8eQ03Q3ojDUAY
n2cRcZ36NhwaIe0f2iyUt8jT0qi+hPzn5JfsmfPkC4Gbq1Pj0UHK1o4RW6NujS
rKMmz5yXGuqgDZDsW8AmH3sQQ8yQQv9/Lf1/x1BZxTfxX1rrvQC0BDCILHzr+9A=
=odZN
-----END PGP SIGNATURE-----
```

那么此时就是使用加密处理后的标签操作。

3.3、标签管理

1、 删除标签

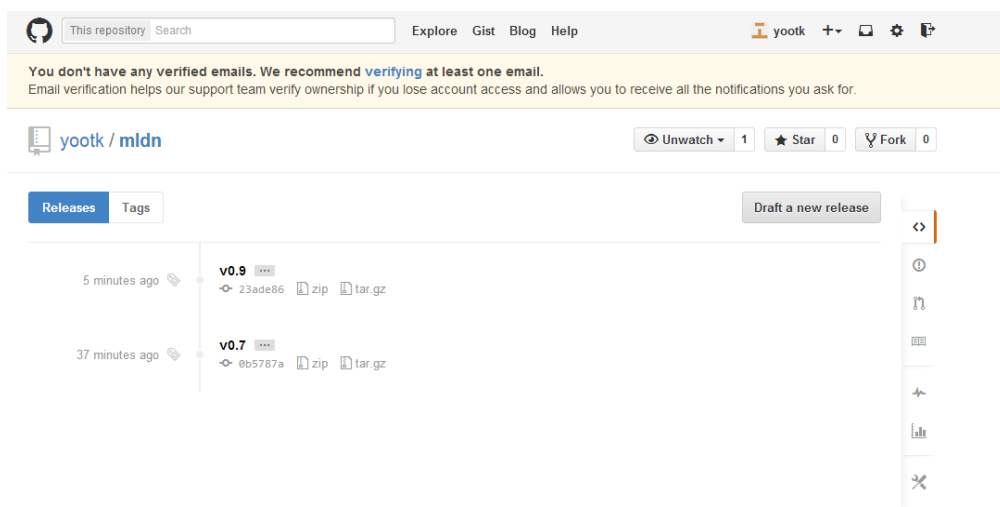
```
git tag -d v1.0
```

这样的标签只是进行了本地的标签删除，而标签也可以提交到服务器上去。

2、 将标签推送到服务器上：

```
git push origin v0.9
git push origin v0.7
```

推送完成之后可以进行标签的察看。



3、 但是这样进行一个个的标签推送很麻烦，所以下面可以一次性推送多个标签过去：

```
git push origin -tags
```

微信：



微博：



全国统一咨询电话：400-0088-518

那么此时本地所有未推送的标签都被推送到了服务器端上。

4、 删除本地某一个标签：

```
git tag -d v0.7
```

5、 但是此时在服务器端上依然会存在有此标签，所以可以继续删除远程标签：

```
git push origin :refs/tags/v0.7
```

这个时候就可以很好的实现了标签管理。

4、总结

标签实际上就是起了一个别名，有些人不喜欢起别名，那么就按照传统的方式开发。

我们可以利用 GPG 工具建立加密的安全标签。

