

PGP 文件加解密

更新于 2013-10-01 02:09:19

UEANER

简介

PGP(Pretty Good Privacy)，是一个基于 RSA 公匙的文件加密算法，通常邮件内容采用此方式加密，可以对邮件，文件，文件夹，磁盘等加密。也是较流行的一种文件加密算法。

使用

一般 Linux 系统都会默认安装 [gnupg](#) 加密软件，它是 OpenPGP 标准的实现。

生成密钥对

```
$ gpg --gen-key
```

按照提示选择 RSA -> 有效期 -> 填一个名称（email 和备注可选） -> 确认生成 -> 输入个密码 -> 等待...生成完成。

此命令运行完毕会生成公匙和私匙一对。

查看密钥

查看公匙

LINUX热门文章

- Linux 下编译安装 PHP 5.6
- 使用 adb sideload 手动更新 andro...
- Linux 下使用 adb 和 fastboot 命令
- VIM编写markdown文档
- Linux 下编译安装 Python 3.4
- GitHub 无法登录，提交
- Linux 下 YUM 安装 PHP 5.5
- openSUSE 12.3 使用 IBUS 中文输入法

热门标签

adb	android	fastboot
phalconphp	中文分词	编译
Apache	markdown	mmseg
virtualbox	代码	bash
chmsee	Git	GitHub
GnuPG	gpg	mysql
nginx	percona	PGP

本文目录

[简介](#)

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

简写：

```
$ gpg -k
```

查看私匙

```
$ gpg --list-secret-keys
```

简写：

```
$ gpg -K
```

导入密匙

导入公匙

```
$ gpg --import <public.asc>
```

导入私匙

```
$ gpg --allow-secret-key-import --import <secret.asc>
```

如果你没有生成过私匙可以直接使用 `--import` 导入。

导出密匙

导出公匙

生成密匙对

查看密匙

导入密匙

导出密匙

删除密匙

核对指纹

密匙签名

文件加密

文件解密

常见问题

```
$ gpg [-a] --export [Uid] > [public.asc]
```

加 `-a` (`--armor` 的简写) 参数，导出为 `ascii` 文本，便于存储或发给合作方等。默认为二进制。

或者：

```
$ gpg -o [public.asc] -a --export <Uid>
```

`-o` 为 `--output` 的简写。

导出私匙

```
$ gpg -a --export-secret-keys [Uid] > [secret.asc]
```

或者：

```
$ gpg -o [secret.asc] -a --export-secret-keys <Uid>
```

如果更换了机器，可以拿着这对公匙和私匙直接导入到另一台机器继续使用。

删除密匙

删除公匙

```
$ gpg --delete-keys <Uid>
```

删除私匙

```
$ gpg --delete-secret-keys <Uid>
```

核对指纹

查看指纹

```
$ gpg --fingerprint
```

核对

```
$ gpg --fingerprint <Uid>
```

密匙签名

如果你信任某个公匙，可以对公匙进行签名，以避免在使用此公匙进行加密时需要手工确认。

```
$ gpg --sign-key <Uid>
```

文件加密

```
$ gpg -o <encryptFile.gpg> -r <Uid> -e <originalFile>
```

`-o`: --output 输出文件，`-r`: --recipient 选择使用哪个公钥，`-e`: --encrypt 加密。不使用 `-a` 参数，默认生成二进制文件。

文件解密

```
$ gpg -o <outputFile> --passphrase <password> -d <decryptFile.gpg>
```

`--passphrase` 后跟生成密钥时设置的密码，避免手工输入密码。

如果你有多个私匙，使用 `-u <Uid>` 指定使用哪个私匙解密。

常见问题

1) 加密的文件对方无法解密？

虽然说各个平台（GnuPG 或 JAVA、PHP 等语言的 [PGP](#) 模块）的实现是遵照 OpenPGP 的标准，但具体实现和版本上还是有差异，你可以尝试着添加 `--disable-mdc`（加密时不使用变动检测码）参数，或更换（升级或降低）GnuPG 版本的方式，来适应不同平台的加解密。

如果你是和银行对接，那么就使用下面的命令吧：

```
/usr/local/gnupg1.4.9/bin/gpg --disable-mdc -o <outputFile> -r <Uid> -  
-passphrase <password> -e <decryptFile.gpg>
```

命令很长啊，相信你可以看出和上面命令的不同。

2) 运行 `gpg --gen-key` 产生 `can't connect to /root/.gnupg/S.gpg-agent` 错误？`

尝试运行以下命令先：

```
gpg-agent --daemon --use-standard-socket
```

3) 编译 gnupg，产生 `gpgkeys_curl-gpgkeys_curl.o error 1` 错误？

尝试使用以下命令编译：

```
./configure --prefix=/usr/local/gnupg1.4.9 --without-libcurl  
make && make install
```

不会影响加解密结果。

另附 Linux / Mac OS X / Windows 系统下 gpg 下载地址：

Linux: <ftp://ftp.gnupg.org/gcrypt/gnupg/>

Mac OS X: <https://gpgtools.org/>

Windows: <ftp://ftp.gnupg.org/gcrypt/binary/gnupg-w32cli-1.4.9.exe>

<ftp://ftp.gnupg.org/gcrypt/binary/libiconv-1.9.1.dll.zip>

<http://gpg4win.org/download.html>

转载请注明出处。

本文地址：<http://blog.aboutc.net/linux/55/pgp-file-encryption-and-decryption>

评论：**1**

姓名 *

邮箱

网站

☐ 如果有后续评论，请用邮件通知我。

发布

ueaner • 2013-10-14 23:30:59

编辑key为可信，避免加密时手动输入Y：

```
# gpg --edit-key
```

```
gpg> trust
```

```
...
```

```
5 = I trust ultimately
```

```
...
```

```
Your decision? 5
```

```
Do you really want to set this key to ultimate trust? (y/N) y
```

```
gpg> quit
```

