

# 如何使用隐写术保护你的机密文件

原创

darksun9972  于 2020-04-26 21:12:02 发布  308  收藏 1

分类专栏: [linux和它的小伙伴](#) 文章标签: [linux和它的小伙伴](#)

版权声明: 本文为博主原创文章, 遵循 [CC 4.0 BY-SA](#) 版权协议, 转载请附上原文出处链接和本声明。

本文链接: <https://blog.csdn.net/lujun9972/article/details/105777212>

版权



[linux和它的小伙伴](#) 专栏收录该内容

91 篇文章 0 订阅

订阅专栏

原文地址:<https://lujun9972.github.io/blog/2020/04/26/如何使用隐写术保护你的机密文件/index.html>

## 目录

- [解决方案一：使用Steghide命令](#)
  - [安装](#)
  - [解决方案](#)
- [解决方案二：使用outguess命令](#)
  - [安装](#)
  - [解决方案](#)

为了防止泄密, 我们一般会对机密文件进行加密。然而加密的文件本身就容易引起人们的好奇心, 如果密码不够强健还容易被暴力破解调。如果能让人根本注意不到机密文件的存在, 岂不美哉? 隐写术就是让你将机密文件嵌入一个普通文件中, 从而不会引起他人注意的一种技术。

## 解决方案一：使用Steghide命令

涉及到的命令: steghide

本部分以archlinux系统为例

### 安装

在archlinux上运行下面命令进行安装:

```
sudo pacman -S steghide --noconfirm
```

### 解决方案

steghide能帮你将文件隐写入图片或音频文件中, 目前它支持WAV, JPEG, AU, BMP等多种格式。

使用steghide进行隐写非常简单, 只需使用 `embed` 子命令就行了。像这样:

```
steghide embed -ef 机密文件 -cf 内嵌文件
```

比如,假设我有一个 `/tmp/secret` 文件, 我要把它隐写入 `~/Pictures/wallpaper/1.jpg` 中, 那么我可以这样操作

```
echo "The Secret word is fox" >/tmp/secret
steghide embed -ef /tmp/secret -cf ~/Pictures/wallpaper/1.jpg
```

执行 `steghide` 命令后，它会要求你连续输入两次加密机密文件的密码，当你从内嵌文件中分离机密文件时就需要输入该密码了。

当然,要是你觉得麻烦，也可以直接按两次回车。

```
lujun9972:~/ $ steghide embed -ef /tmp/secret -cf ~/Pictures/wallpaper/1.jpg
Enter passphrase:
Re-Enter passphrase:
embedding "/tmp/secret" in "/home/lujun9972/Pictures/wallpaper/1.jpg"... done
```

这样隐写就完成了。不过这里需要注意的是，隐写不会删掉你原始的机密文件，因此你需要手工删除机密文件：

```
rm /tmp/secret
```

当我们需要抽取出机密文件时,只需要使用 `extract` 子命令即可，语法如下：

```
steghide extract -sf 内嵌文件
```

比如我们来尝试还原刚才被删除的 `secret` 文件：

```
steghide extract -sf ~/Pictures/wallpaper/1.jpg
```

然后输入隐写时设置的密码后就能看到在 当前目录下 生成了机密文件 `secrete` 了

```
lujun9972:~/ $ steghide extract -sf ~/Pictures/wallpaper/1.jpg
Enter passphrase:
wrote extracted data to "secret".
lujun9972:~/ $ cat secret
The Secret word is fox
```

## 解决方案二：使用outguess命令

涉及到的命令: outguess

本部分以archlinux系统为例

### 安装

outguess在AUR仓库中，在archlinux上运行下面命令进行安装：

```
yaourt -S outguess
```

### 解决方案

使用outguess进行隐写一般要用到它的 `-d` 和 `-k` 选项。其中 `-d` 执行机密文件的路径，`-k` 指定加密的密码，当然你也可以忽略 `-k` 选项不设置密码。

```
outguess -d 机密文件 图片文件 内嵌文件
# 或者
outguess -k "密码" -d 机密文件 图片文件 内嵌文件
```

outguess就会将机密文件隐写如图片文件中，并生成新的内嵌文件

比如,我要把 `/tmp/secret` 文件隐写入 `~/Pictures/wallpaper/purple-wallpaper6.jpg` 中，那么可以这样操作

```
outguess -k "fox" -d /tmp/secret ~/Pictures/wallpaper/purple-wallpaper6.jpg /tmp/secret.jpg
```

相比 `steghide` 来说, `=outguess=` 的好处是不会要求你手工输入密码,这样便于放在脚本中使用。

当我们需要抽取出机密文件时,则使用 `-r` 选项,语法如下:

```
outguess -r 内嵌文件 机密文件
```

# 或

```
outguess -k "密码" -r 内嵌文件 机密文件
```

比如我们可以使用下面迷你够了还原 `secret` 文件:

```
outguess -k "fox" -r /tmp/secret.jpg secret
```