**文件操作与隐写**

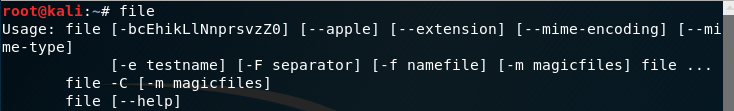
文件类型识别

文件类型（图片、压缩包、流量包、word、Excel、ppt，PDF....)

怎么知道是一张图片：后缀名是.jpg  .png...

a、File命令

（这边说到的命令或者是工具，其实都是叫工具，说命令只是说这个工具没有图形化界面，但本身就是一个工具而已，这个工具是命令行的）



在Linux系统中使用该命令

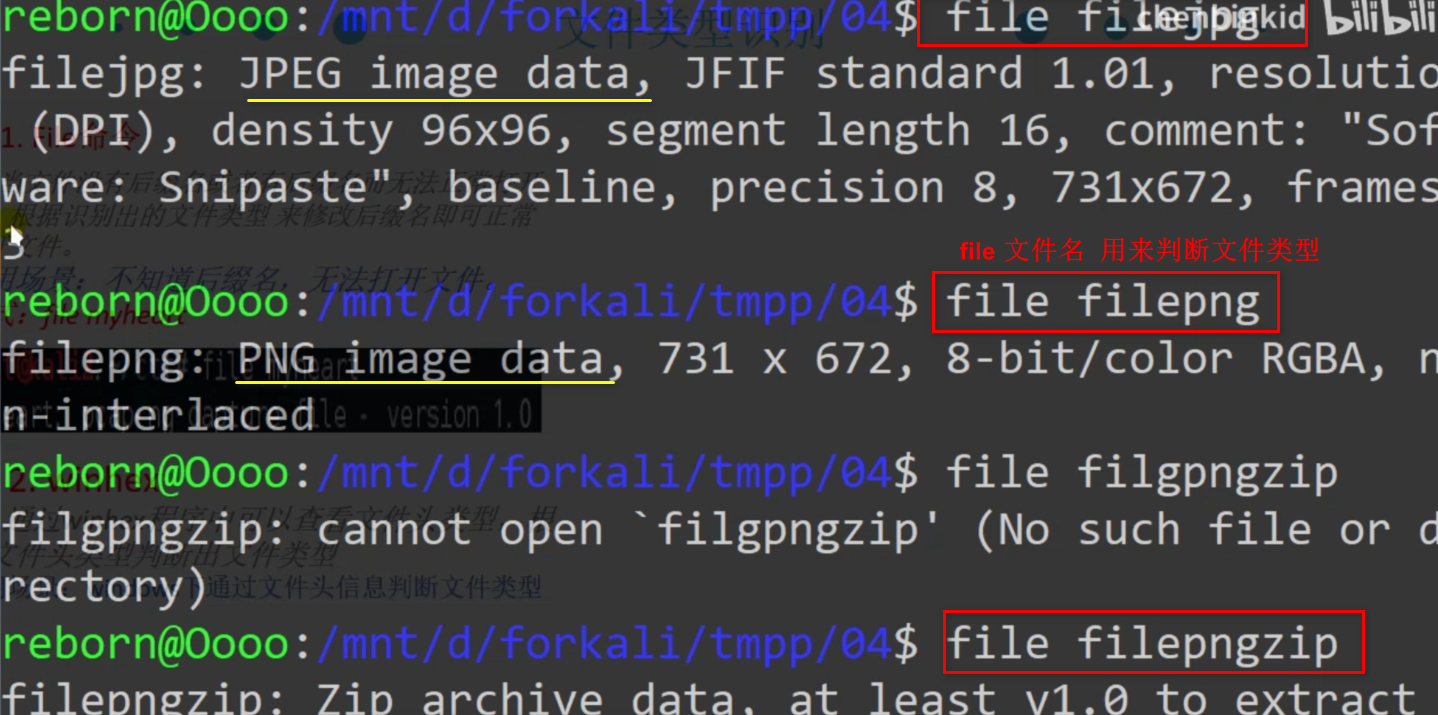
命令格式：file 文件名

当文件没有后缀名或者有后缀名但是无法正常打开时，根据识别出的文件类型来修改后缀名即可正常打开文件。

使用场景：不知道后缀名，无法打开文件

格式：file myheart





 工具是怎么识别这个文件的类型呢？



 去剪切这个文件头，解析它，zip检测到504B0304.是根据文件头的字段来识别这个文件是什么类型的

b、winhex工具

使用winhex程序可以查看文件的头类型，根据文件头类型可以判断出文件类型

使用场景：windows系统下通过文件头信息判断文件类型

hex：十六进制    winhex其实是一个十六进制文件的编辑器

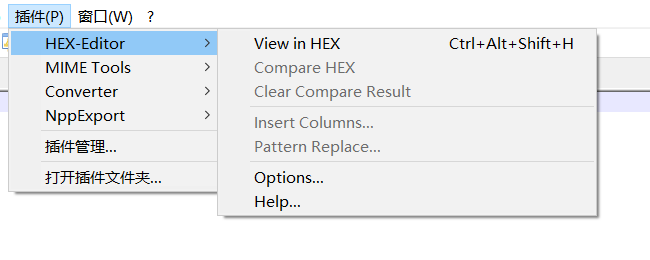
010editor（推荐）也是一个十六进制编辑器

notepad++  有一个hex插件，notepad++可以高亮代码！

三个工具

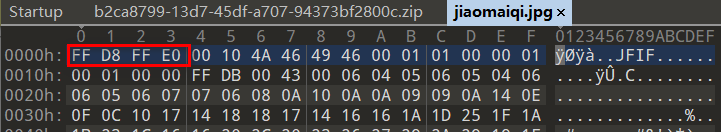
一张图片用notepad++打开是一串乱码。有一个插件——》HEX-editor——》View in HEX查看它的十六进制



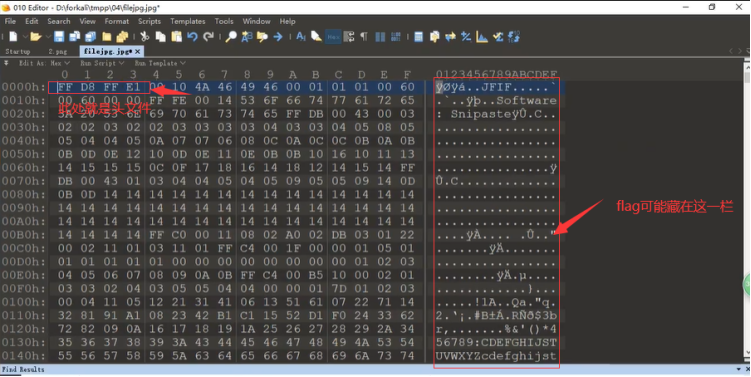




演示：用010editor打开：



 flag值可能写在右边的ASCII码中



文件头破坏了怎么识别？？

做一个修复 ↓

c、文件头残缺/错误

（可能要用到的工具：010editor 16进制编辑器）

文件头部残缺或者头部字段错误无法打开正常文件，针对这种情况，使用winhex程序添加相应的文件头，针对头部字段错误，可以找一个相同类型的文件进行替换

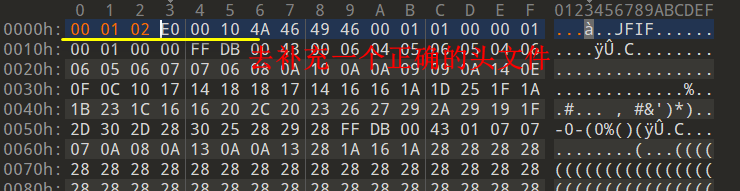
使用场景：文件头部残缺或者头部字段错误无法打开正常文件

格式：file 文件名

如果文件无法打开，使用该命令解析，显示data信息，则表示该文件头部残缺或者错误，此时就可以用16进制编辑器在第一行补充正确的文件头

会显示这是一个data文件

 010editor 16进制编辑器去修复



文件分离操作

一张图片可能包含多少信息呢？

图片种子：下载一张图片也就几百kb，其实它蕴藏的信息有一两个G

正常的隐写就是：你看到的是一张图片，它可能是由好几张图片拼成的。甚至不只是图片，一张图片里面还有压缩包，还有txt，还有PDF 还有其他文件

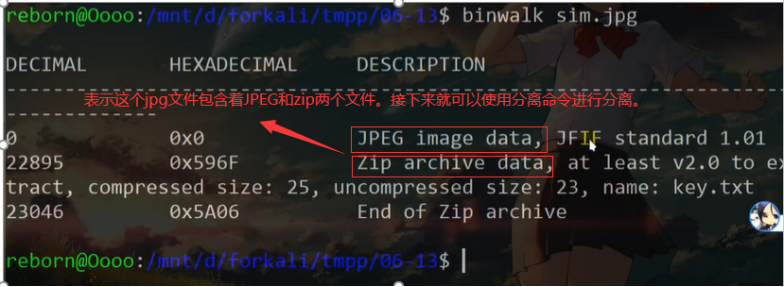
　　　　　　　　出题人把很多文件融合成一张图片——》文件分离

a. binwalk工具

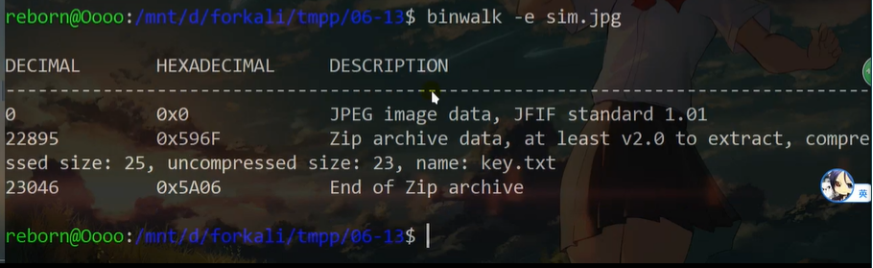
该工具是Linux下用来分析和分离文件的工具，可以快速分辨文件是否有多个文件合并而成，并将文件分离。如果分离成功会在目标文件的目录中生成一个形如\_文件名\_extracted的文件目录，目录中会有分离后的文件。

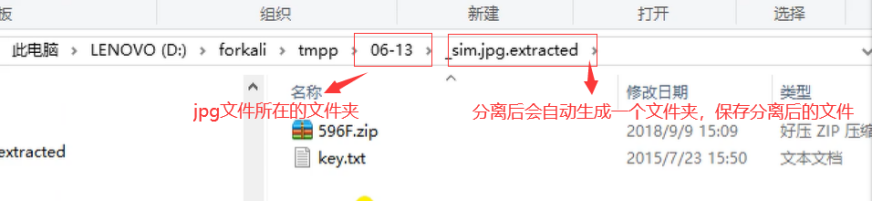
格式：

分析文件 ：binwalk filename.后缀



分离文件：binwalk -e filename





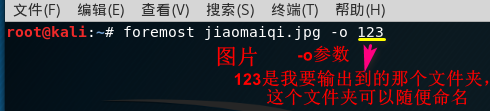
 binwalk特点是遇到压缩包会帮我们自动解压

key.txt就是我们的flag

b. foremost

如果用binwalk无法分离出文件，可以使用该工具，将目标文件复制到kali中，成功执行后会将分离后的文件保存在源文件的新文件夹中。

命令格式：foremost 文件名.后缀 -o 自定义保存分离文件的文件夹名     （用-o 把它输出到一个目录下o就是out）

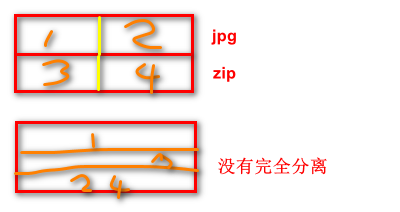
会生成一个名称为123的文件夹

 foremost是一个比较中规中矩的一个分离软件，不像binwalk（看到压缩包会帮您解压），

foremost不会，foremost会帮您分类，告诉你哪个文件夹是jpg，哪个文件夹是zip，会帮你分类。甚至分离完还会给你写一个日志文件

c. dd（半自动化分离文件工具）

当上面两个工具分离出错或者无法自动分离时，就需要dd来手动实现分离



命令格式：

dd if=源文件 of=目标文件名 bs= count= skip= 开始分离的字节

dd if=源文件 of=输出文件   bs=5（我要取前面五个字符，块就为5 ）  count=1（去1块） skip= 开始分离的字节

参数说明；

if = file  输入文件名 缺省为标准输入

of = 分离后的文件名       分离后的文件名

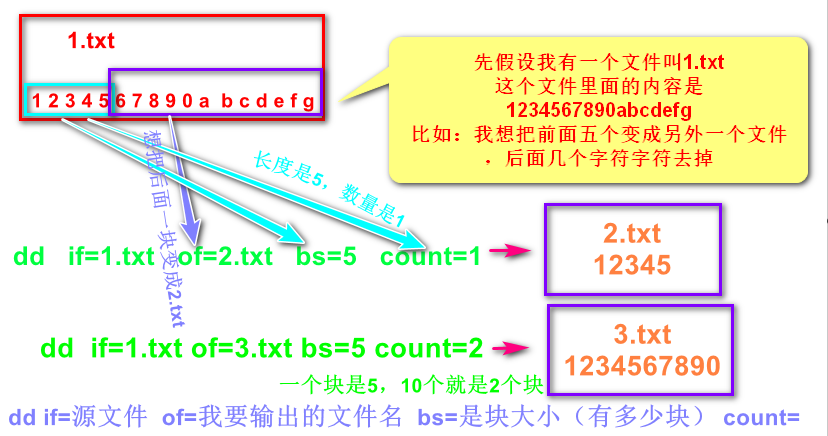
bs=i 将文件分为多个块，一个块的大小为字节为i（bs=bytes）

count=j 从头开始取j块

skip-k

命令表示：截取源文件，开始字段为i X k，截取字段的长度为i X j  当skip省略时，截取从0开始，截取字段的长度为i X j

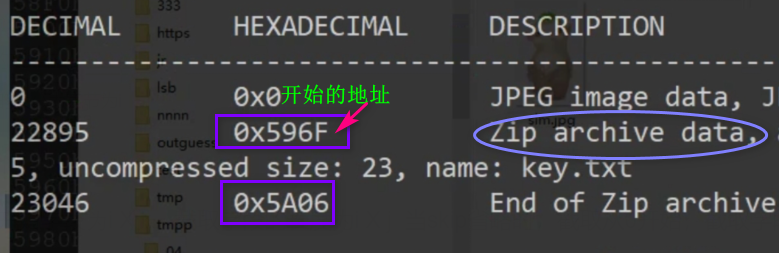
skip=blocks  从输入文件开头跳过blocks个块后再开始复制



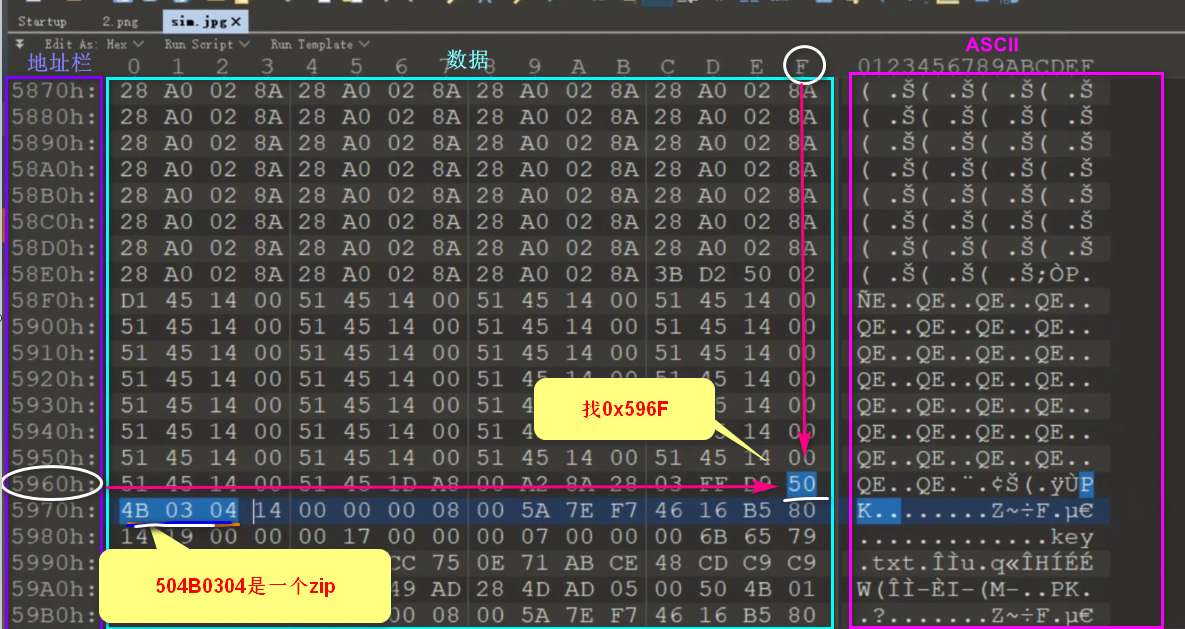


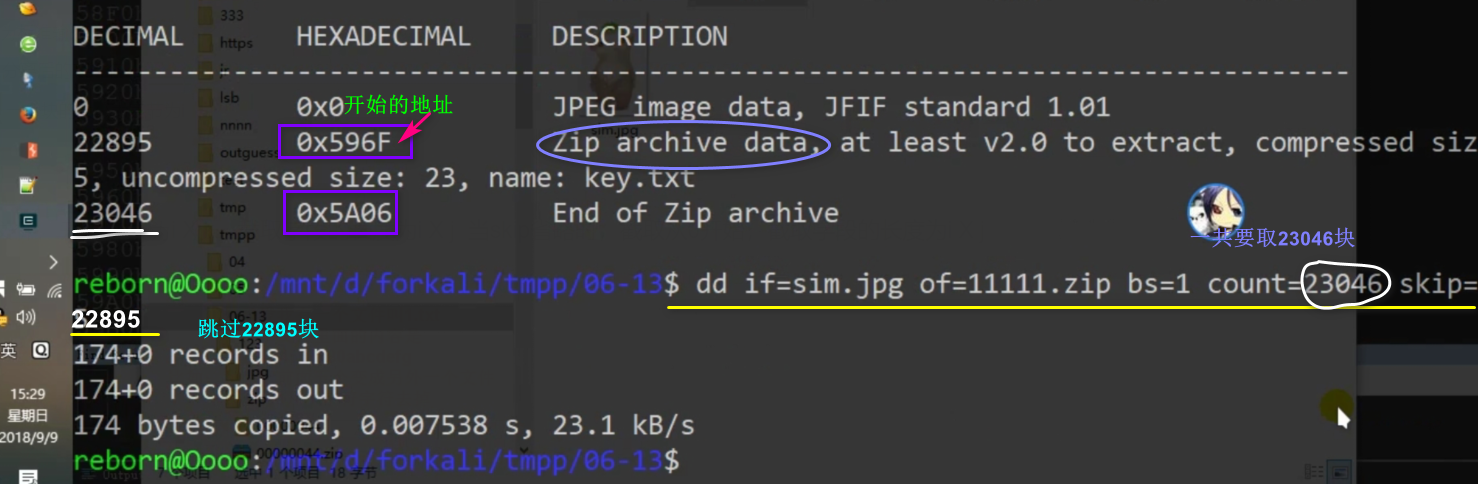
实例：

用16进制编辑器打开图片，怎么定位到地址呢？



 怎么找到0x596F？



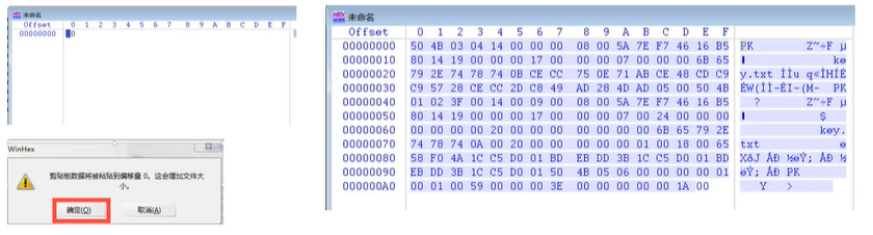


d. winhex

除了用dd外，还可以使用过winhex实现手动分离，将目标文件拖入winhex中，找到要分离的部分，点击复制即可

使用场景：windows下利用winhex程序对文件进行手动分离

例如：新建一个文件，文件大小1byte,在文件开头位置点击粘贴，弹出提示框选否，确定，将文件保存为想要的后缀即可。

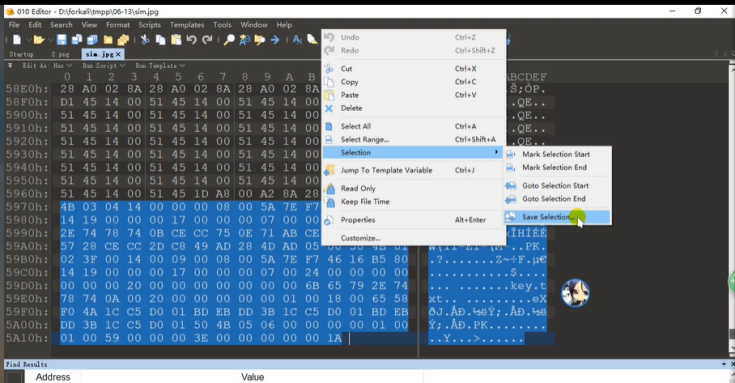


e. 010deitor（推荐）

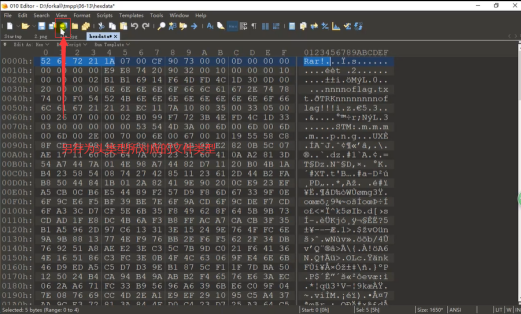
用法1：选择需要分离的文件在源文件中的内容，按图中所示选择选项即可分离出源文件包含的文件。

将某块区域文件保存的方式如下：

1、010Editor打开文件——》选中右键——》selection——》Save Selection



用法2：将16进制的文件导入，然后另存为对应的文件类型



文件合并操作

（题目给出多个文件，我们需要将这些文件按顺序合并，校验md5等确定是否合并正确，一般在打开合并的文件后就可以得到flag）

a. Linux下的文件合并

使用场景：Linux下通常对文件名类似的文件进行批量合并

命令格式：cat 合并的文件1 合并的文件2  … >合并后输出的文件名

IMG_280

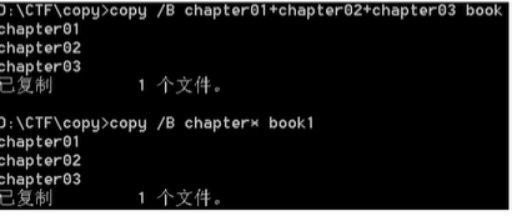
完整性检测（计算文件的md5等等）：md5sum 文件名

IMG_281

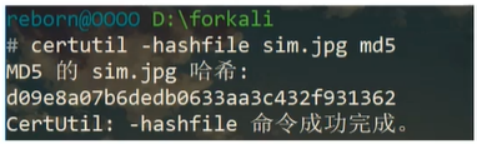
b. windows下的文件合并

使用场景：Windows下，通常要对文件名类似的文件进行批量合并

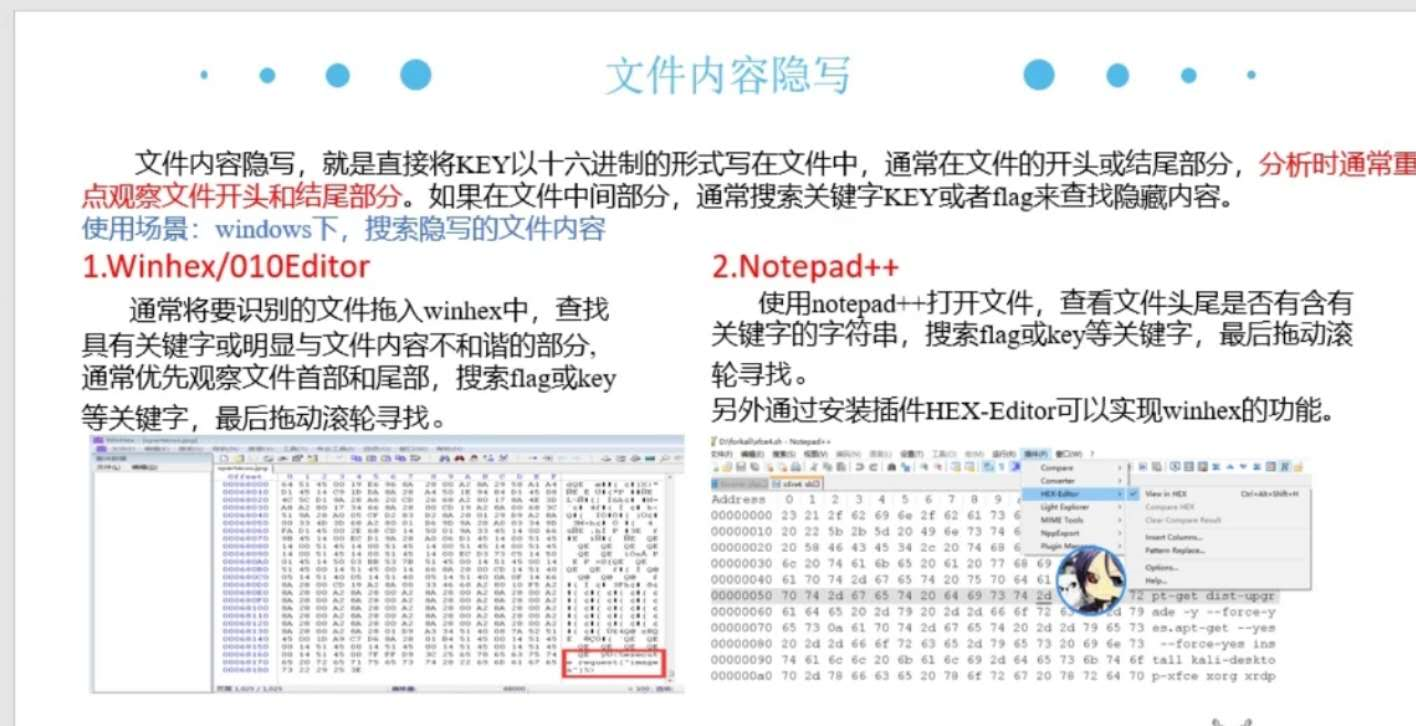
格式：copy /B 合并的文件1+合并的文件2+… 输出的文件名



完整性检测（计算文件的md5等等） ：certutil -hashfile 文件名 md5



文件内容的隐写



题目给一个或多个文件，我们需要用工具打开，例如notpad++，然后使用查找功能搜索fiag或者key等关键字，以得到题目线索。

文件可能还需要使用16进制编辑器打开，然后再进行查找等操作