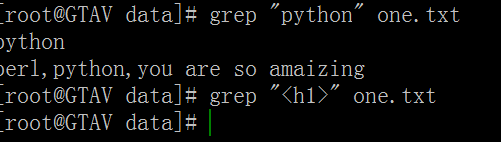
# Linux三剑客

## Grep

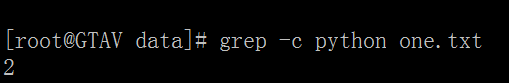
这个命令全称为是**Global Regular Expression Print，即全局正则表达式打印。它使用强大的正则表达式匹配相应的文本。是Linux中最强大的命令之一**

Grep + 过滤文本 +（参数）+ 要过滤的文件

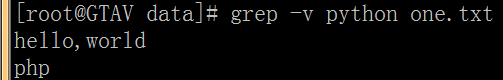
返回的是过滤后的结果，如果筛选到，则返回筛选到的文本。否则，返回空值。



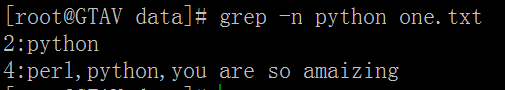
参数-c，只返回匹配到的行数



参数-v， 返回不匹配的内容



参数-n，返回匹配到的行号

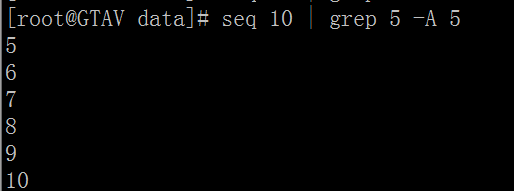


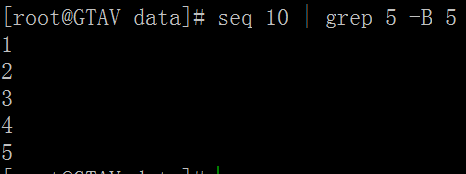
三个参数：

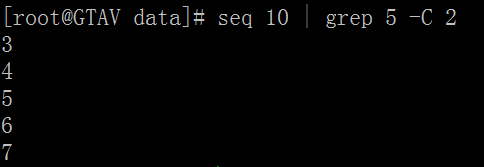
-A n：显示匹配的这一行，并且显示这一行之后的n行

-B n：显示匹配的这一行，并显示这一行之前的n行

-C n：显示匹配的这一行，并显示这一行前后的各n行







# Sed

Sed命令，是一个Linux三剑客第二个，同样用来替换，查找。

Sed命令主要是对每一行进行处理，增加，修改，删除，选取的操作。文件内容没有改变，除非使用重定向输出。它更多的用于编辑内容以筛选出需要的内容然后进行编辑和输出

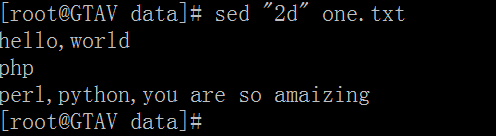
格式为：set 参数 “command” 文件名

-n 使用安静模式。默认的，每次sed命令执行，所有的来自STDIN的字符都会被打印一次，使用这个参数可以只打印选取的结果

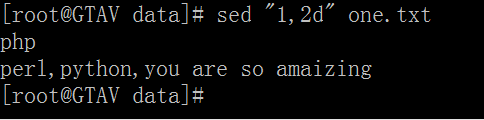
常用command:

d : 忽略，用来忽略指定的行

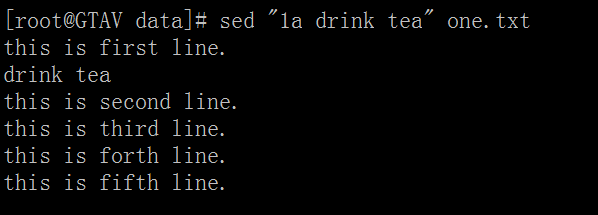
忽略第二行：



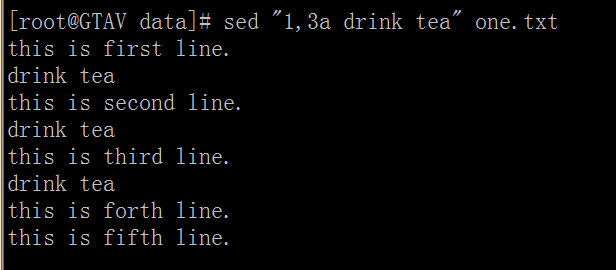
返回除了第一行到第二行之外的内容：



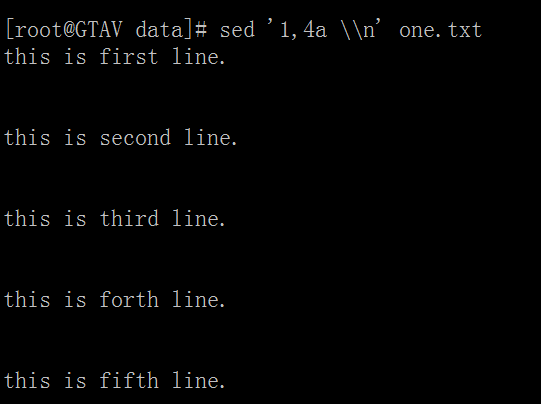
增加指定内容到某一行的后面



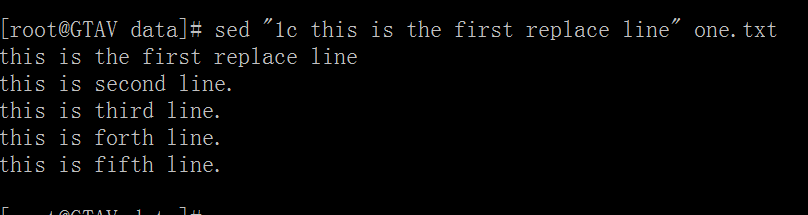
增加多个同样的内容到多行，在一到三行里增加drink tea



可以增加换行符来让每一行之间有空间

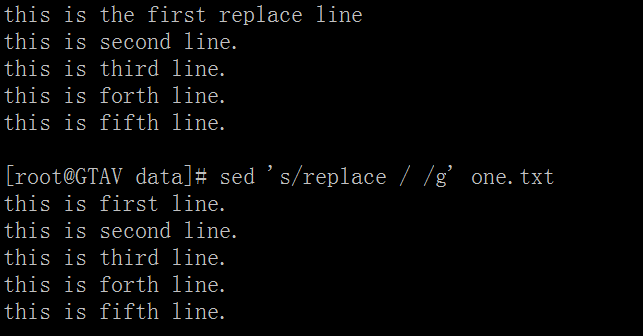


使用c来替换成目标内容



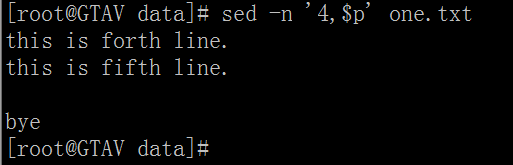
替换字符串: ‘s/要替换的字符串/替换成什么字符串/g’，使用-i来替换源文件，否则只在当前回话中输出替换后的文本，源文件不变。

可以使用///来作为分割符，也可以使用# ，@等，

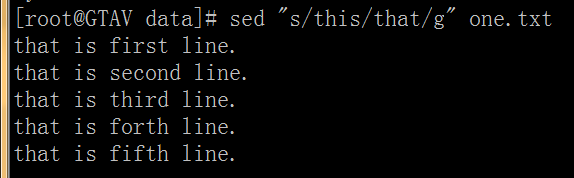


把replace替换为空字符串

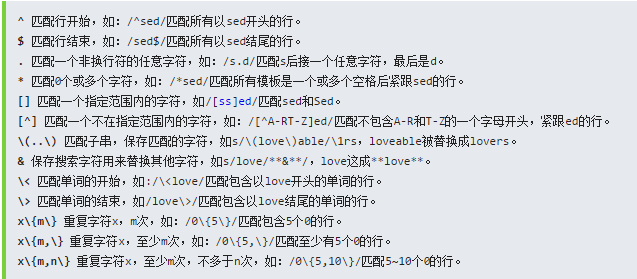
筛选指定的行到底部最后一行



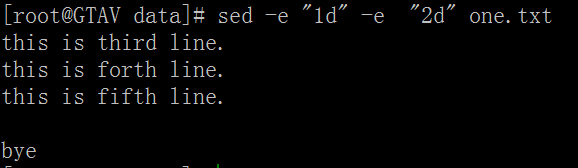
替换文件中的所有指定的内容



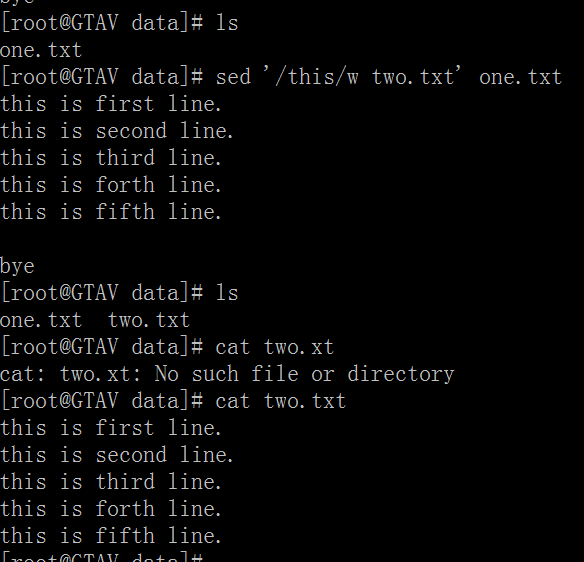
可以使用正则表达式来筛选替换的内容，支持的正则表达式如下：



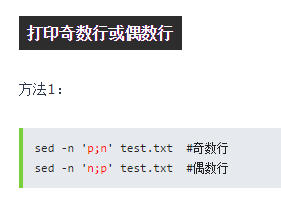
-e用来在一行执行多个命令



W，写入命令，sed ‘/要写入的字符串/w file’ finename ，把filname中的和匹配的字符串能匹配到的行写入到file中。



打印奇数行和偶数行



# Linux三剑客之AWK

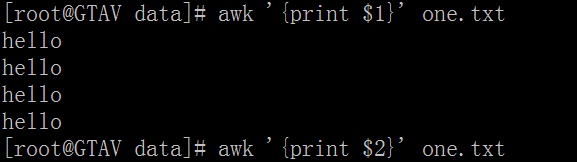
Awk是Linux中最强大的文本处理工具之一。相比于grep文本的查找，sed对行的操作，AWK更多的是对数据进行详细的分析并生成报告，它可以把数据按行分析，然后按照分隔符进行切片，切开的部分进行分析处理等。它是Linux现有的最强大的数据处理引擎之一

Awk现在已经成为一种新的编程语言：awk语言--样式扫描与处理语言。它可以实现正则表达式的匹配，样式装入，流控制，进程控制等，甚至内置的条件语句也非常强大。Awk有三个版本，awk，nawk，gawk，通常所指的是gawk，即gun版本。

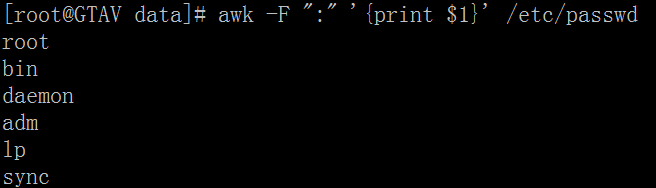
基本用法：awk =+ ‘{pattern + action}’ filename：pattern是模式，可以是正则表达式，也可以是awk内置的函数。Action是要执行的动作

筛选第一列，然后输出

筛选文本的每行的第一列，分隔符为空格

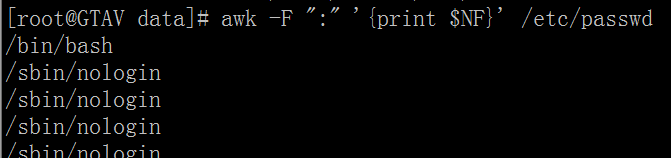


使用-F来强制指定分隔符



筛选最后一列，使用$NF，不能使用双引号包含{}。$0表示整个文本

$NF-1表示倒数第二列



Awk是有一系列常量，NR表示行号，$NF是表示读到的最后一行

