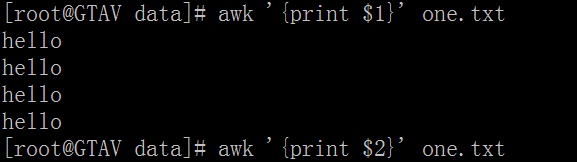
# AWK命令

## 描述:

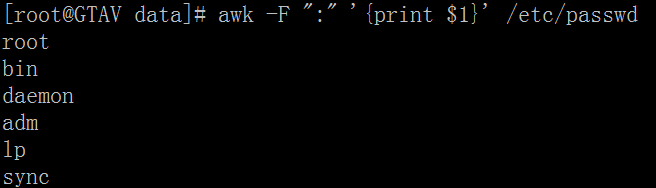
1. Awk是Linux中最强大的文本处理工具之一。相比于grep文本的查找，sed对行的操作，AWK更多的是对数据进行详细的分析并生成报告，它可以把数据按行分析，然后按照分隔符进行切片，切开的部分进行分析处理等。它是Linux现有的最强大的数据处理引擎之一
2. Awk现在已经成为一种新的编程语言：awk语言--样式扫描与处理语言。它可以实现正则表达式的匹配，样式装入，流控制，进程控制等，甚至内置的条件语句也非常强大。Awk有三个版本，awk，nawk，gawk，通常所指的是gawk，即gun版本。
3. 基本用法：awk =+ ‘{pattern + action}’ filename
4. 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
5. 因为awk已经成为一门独立的语言，这里只做基本的了解，不做深入的研究

## 参数

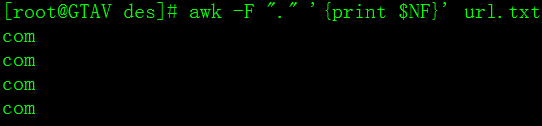
1. 筛选第一列，然后输出
2. 筛选文本的每行的第一列，分隔符为空格



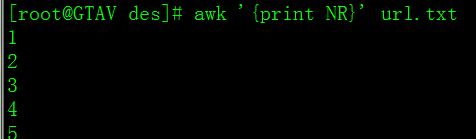
1. 使用-F来强制指定分隔符



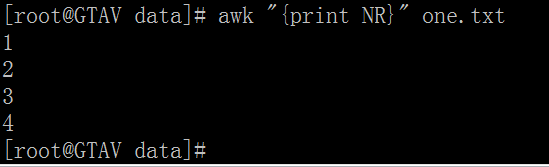
1. 筛选最后一列，使用$NF，不能使用双引号包含{}。$0表示整个文本
2. $NF-1表示倒数第二列



1. Awk是有一系列常量，
2. NR表示行号

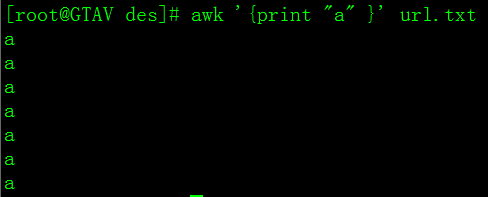


1. $0 表示当前所有行
2. $1 表示第一列，也就是第二个参数；
3. $2 表示第二列，以此类推
4. $NF是表示读到的最后一个参数，也就是最后一列



### {print var} 输出命令

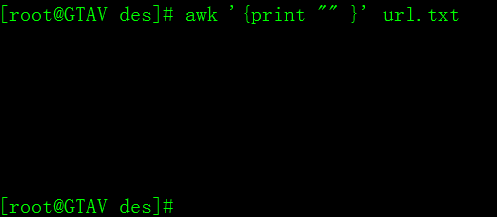
* 1. {print “a”} filename : 输出同等行数的a



* 1. {print $1;print $2} file 将每一行中第一列和第二列的内容分行输出



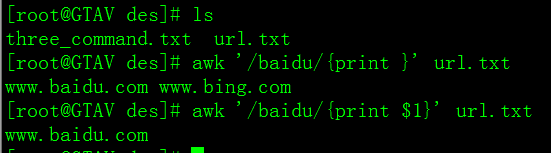
* 1. {print “”} filename : 输出和filename同样多的空行，表明这个命令是逐行处理的。



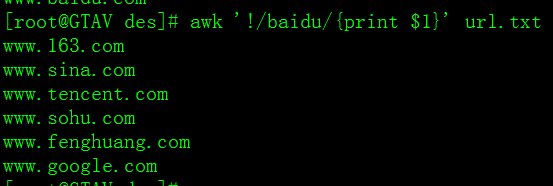
1. OFS ‘symbol’ 以symbol来分割字符串，-F用来以已有的分隔符进行分割，返回的字符串会去除分隔符。



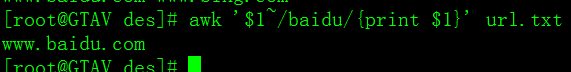
1. **//纯字符匹配，匹配baidu，在文件中搜索，输出匹配的行**



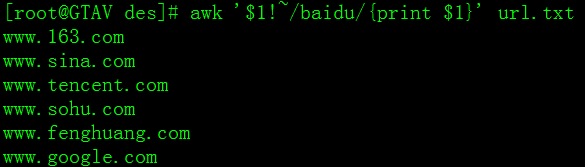
1. **!//纯字符不匹配，输出不匹配的字符**



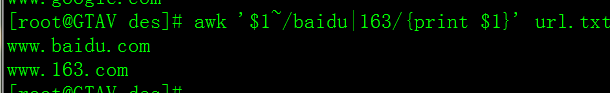
1. **~//字段值匹配，输出匹配的字符串**



1. **!~//字段值不匹配，输出不匹配的字符串**



1. **~/a1|a2/字段值匹配a1或a2**

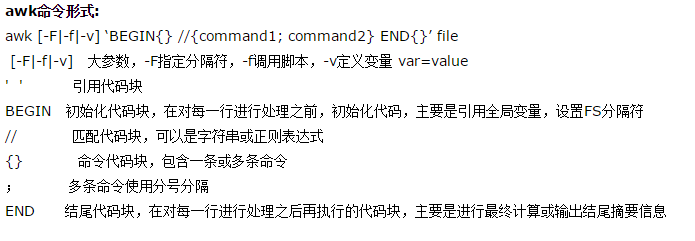


## 核心内容

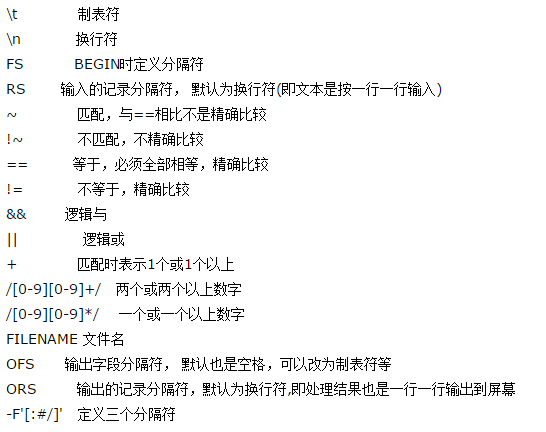
1. 这个命令最重要的部分是它对文本的处理更加精细化，可以具体到每个单词的改变，文本的迭代，条件判断等，已经成为一门独立的语言；

## Awk编程研究

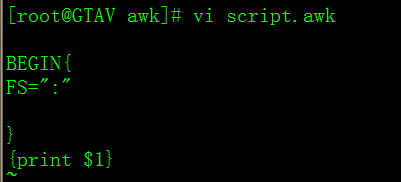
1. 下图展示了一般的awk命令的基本格式和详细讲解

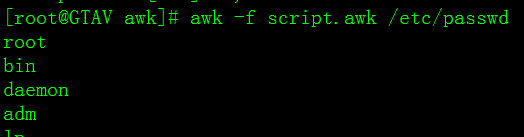


1. 一些特殊字符的使用



* 1. 可以使用-f 选项，指定脚本文件，然后运行代码，下面是一个例子
  2. 创建scripts.awk文件，后缀名必须是awk

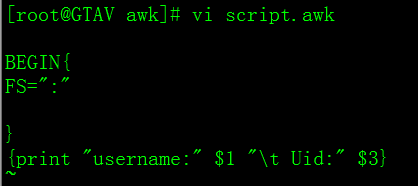
，然后执行命令

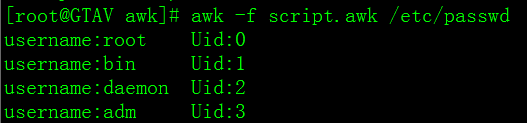


* 1. FS表示指定分隔符，它和下图的命令效果一致

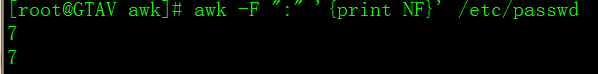


另一个例子

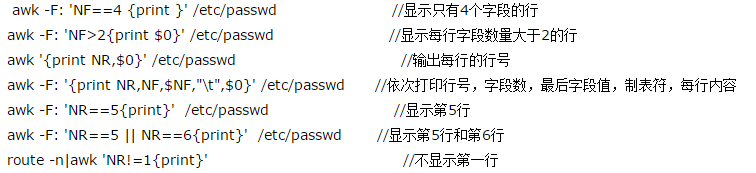




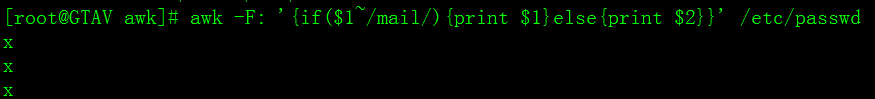
NF表示每一行有几个字段，以分隔符计算



一些其他的NF字段应用

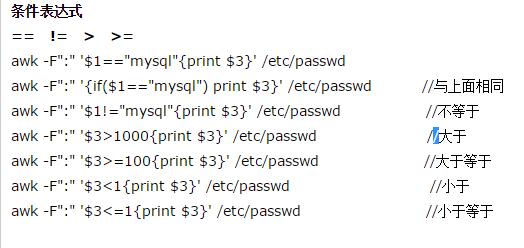


1. 使用if语句
   1. If语句必须在{}里面，用（）来包含比较的内容，下面是例子



* + 1. 解释：-F分隔符，{}中包含的是if语句，如果每行中的第一个字段是mail，则输出mail，否则，输出每行中的第二个字段，这里第二个字段是x；
    2. 也可以把else去掉，只保留if语句部分

1. 条件表达式也可以表示判断，如下图



后面的不记了，慢慢在实践中联系，不知道的查手册

## 补充：

1. 下图展示了如何使用awk的system函数，翻译为如果在passwd中寻找到root，uid大于0，则执行ping命令；

