md5sum

计算和校验文件报文摘要的工具程序

补充说明

md5sum命令 采用MD5报文摘要算法（128位）计算和检查文件的校验和。一般来说，安装了Linux后，就会有md5sum这个工具，直接在命令行终端直接运行。

MD5算法常常被用来验证网络文件传输的完整性，防止文件被人篡改□MD5 全称是报文摘要算法□Message-Digest Algorithm 5□□此算法对任意长度的信息逐位进行计算，产生一个二进制长度为128位（十六进制长度就是32位）的“ 指纹 ”（或称“ 报文摘要 ”），不同的文件产生相同的报文摘要的可能性是非常非常之小的。

语法

```
md5sum(选项)(参数)
```

选项

```
-b□二进制模式读取文件；
-t或--text□把输入的文件作为文本文件看待；
-c□从指定文件中读取MD5校验和，并进行校验；
--status□验证成功时不输出任何信息；
-w□当校验不正确时给出警告信息。
```

参数

文件：指定保存着文件名和校验和的文本文件。

实例

使用 md5sum 生成密码

另一种获取可用作密码的随机字符串的方法是计算 MD5 校验值！校验值看起来确实像是随机字符串组合在一起，我们可以用作密码。确保你的计算源是个变量，这样的话每次运行命令时生成的校验值都不一样。比如 date □date 命令 总会生成不同的输出。

```
[root@localhost ~]# date | md5sum
6a43f2c246cdc3e6a3592652f831d186  -
```

生成一个文件insert.sql的md5值：

```
[root@localhost ~]# md5sum insert.sql
bcda6cb5c704664f989703ac5a88f112  insert.sql
```

检查文件testfile是否被修改过：

首先生成md5文件：

```
md5sum testfile > testfile.md5
```

检查：

```
md5sum testfile -c testfile.md5
```

如果文件没有变化，输出应该如下：

```
forsort: OK
```

此时，md5sum命令返回0。

如果文件发生了变化，输出应该如下：

```
forsort: FAILED
md5sum: WARNING: 1 of 1 computed checksum did NOT match
```

此时，md5sum命令返回非0。

这里，检查用的文件名随意。如果不想有任何输出，则`md5sum testfile --status -c testfile.md5`，这时候通过返回值来检测结果。

检测的时候如果检测文件非法则输出信息的选项:

```
md5sum -w -c testfile.md5
```

输出之后，文件异常输出类似如下：

```
md5sum: testfile.md5: 1: improperly formatted MD5 checksum line
md5sum: testfile.md5: no properly formatted MD5 checksum lines found
```

这里，testfile.md5只有一行信息，但是我认为地给它多加了一个字符，导致非法。如果md5文件正常那么-w有没有都一样。