

date命令是显示或设置系统时间与日期。
很多shell脚本里面需要打印不同格式的时间或日期，以及要根据时间和日期执行操作。延时通常用于脚本执行过程中提供一段等待的时间。日期可以以多种格式去打印，也可以使用命令设置固定的格式。在类UNIX系统中，日期被存储为一个整数，其大小为自世界标准时间（UTC）1970年1月1日0时0分0秒起流逝的秒数。

语法

date (选项) (参数)

选项

-d<字符串>: 显示字符串所指的日期与时间。字符串前后必须加上双引号；

-s<字符串>: 根据字符串来设置日期与时间。字符串前后必须加上双引号；

-u: 显示GMT；

--help: 在线帮助；

--version: 显示版本信息。

参数

<+时间日期格式>: 指定显示时使用的日期时间格式。

日期格式字符串列表

%H	小时，24小时制（00~23）
%I	小时，12小时制（01~12）
%k	小时，24小时制（0~23）
%l	小时，12小时制（1~12）
%M	分钟（00~59）
%p	显示出AM或PM
%r	显示时间，12小时制（hh:mm:ss %p）
%s	从1970年1月1日00:00:00到目前经历的秒数
%S	显示秒（00~59）
%T	显示时间，24小时制（hh:mm:ss）
%X	显示时间的格式（%H:%M:%S）
%Z	显示时区，日期域（CST）
%a	星期的简称（Sun~Sat）
%A	星期的全称（Sunday~Saturday）
%h,%b	月的简称（Jan~Dec）
%B	月的全称（January~December）
%c	日期和时间（Tue Nov 20 14:12:58 2012）
%d	一个月的第几天（01~31）
%x,%D	日期（mm/dd/yy）
%j	一年的第几天（001~366）
%m	月份（01~12）
%w	一个星期的第几天（0代表星期天）
%W	一年的第几个星期（00~53，星期一为第一天）
%y	年的最后两个数字（1999则是99）

实例

格式化输出：

```
date +%Y-%m-%d
2009-12-07
```

输出昨天日期：

```
date -d "1 day ago" +%Y-%m-%d
2012-11-19
```

2秒后输出：

```
date -d "2 second" +%Y-%m-%d %H:%M:%S
2012-11-20 14:21.31
```

传说中的 1234567890 秒：

```
date -d "1970-01-01 1234567890 seconds" +%Y-%m-%d %H:%m:%S"
2009-02-13 23:02:30
```

普通转格式：

```
date -d "2009-12-12" +%Y/%m/%d %H:%M.%S"
2009/12/12 00:00.00
```

apache格式转换：

```
date -d "Dec 5, 2009 12:00:37 AM" +%Y-%m-%d %H:%M.%S"
2009-12-05 00:00.37
```

格式转换后时间游走：

```
date -d "Dec 5, 2009 12:00:37 AM 2 year ago" +%Y-%m-%d %H:%M.%S"
2007-12-05 00:00.37
```

加减操作：

```
date +%Y%m%d           //显示前天年月日
date -d "+1 day" +%Y%m%d //显示前一天的日期
date -d "-1 day" +%Y%m%d //显示后一天的日期
date -d "-1 month" +%Y%m%d //显示上一月的日期
date -d "+1 month" +%Y%m%d //显示下一月的日期
date -d "-1 year" +%Y%m%d //显示前一年的日期
date -d "+1 year" +%Y%m%d //显示下一年的日期
```

设定时间：

```
date -s                //设置当前时间，只有root权限才能设置，其他只能查看
date -s 20120523        //设置成20120523，这样会把具体时间设置成空00:00:00
date -s 01:01:01        //设置具体时间，不会对日期做更改
date -s "01:01:01 2012-05-23" //这样可以设置全部时间
date -s "01:01:01 20120523"   //这样可以设置全部时间
date -s "2012-05-23 01:01:01" //这样可以设置全部时间
date -s "20120523 01:01:01"   //这样可以设置全部时间
```

有时需要检查一组命令花费的时间，举例：

```
#!/bin/bash

start=$(date +%s)
nmap man.linuxde.net &> /dev/null

end=$(date +%s)
difference=$(( end - start ))
echo $difference seconds.
```