

How to display the date and time in Linux Terminal


Let's take a look at the advanced uses of the `date` command in the command line and how to use it in the shell script to perform more functions than simply displaying the time.

Surely you are no stranger to the **date** command in the Bash shell, this is the default shell in most Linux distributions and even macOS. However, in this tutorial, let's take a look at the advanced usage of the **date** command in the command line and how to use it in the shell script to perform more functions than simply showing the time time.

1. Basic Linux commands everyone needs to know
2. Basic Shell commands in Linux
3. 7 useful commands for Linux network

When running the `date` command, you will see the current date and time in your country time zone.

```
date
```

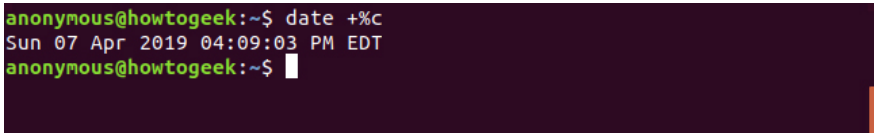


```
anonymous@howtogeek:~$ date
Sun Apr  7 16:08:45 EDT 2019
anonymous@howtogeek:~$
```

You can see the default format doesn't look right. Why is the year not printed after the date but left behind after the time zone? Fortunately, there are more than 40 ways to arrange dates so you can adjust the output as desired.

To adjust the `date` command, type **date**, space, plus (+) and option (usually letters) with the % symbol. Optional % (date and time in local format) displays the date and time in your localized standardized format. Your location is set according to the geographic information you provided when installing the operating system. It includes currency symbols, page sizes, time zones and some other standards.

```
date +% c
```



```
anonymous@howtogeek:~$ date +%c
Sun 07 Apr 2019 04:09:03 PM EDT
anonymous@howtogeek:~$
```

You can see, five hours have appeared in the position right after the date in the output.

You can change some options at the same time. A string of options is called format string. To see the date name (% A), date of the month (% d) and month name (% B), use the following command:

```
date +%A%d%B
```

```
anonymous@howtogeek:~$ date +%A%d%B
Sunday07April
anonymous@howtogeek:~$ █
```

The command is correct but the format is not very nice, it is difficult to see. To make it easier to read, you can add spaces between options but must be enclosed in quotation marks. Note, the plus sign is outside the quotation.

```
date + "%A %d %B"
```

```
anonymous@howtogeek:~$ date +"%A %d %B"
Sunday 07 April
anonymous@howtogeek:~$ █
```

You can add text to a format string like:

```
date + "Today is:%A %d %B"
```

```
anonymous@howtogeek:~$ date +"Today is: %A %d %B"
Today is: Sunday 07 April
anonymous@howtogeek:~$ █
```

Here are some options to help you edit the result format that displays the date and time.

Option to display date and time

1. **% c** : Displays the date and time in the local format, including the time zone.

```
anonymous@howtogeek:~$ date
Sun Apr  7 16:08:45 EDT 2019
anonymous@howtogeek:~$ █
```

Optional display date

1. **% D** : Display the date in month / day / year format.
2. **% F** : Displays the date according to the year format - month - day.
3. **% x** : Displays the date according to locale format.

```
anonymous@howtogeek:~$ date +%D
04/07/19
anonymous@howtogeek:~$ date +%F
2019-04-07
anonymous@howtogeek:~$ date +%x
04/07/2019
anonymous@howtogeek:~$ █
```

Optional display date

1. **% a** : Display the abbreviated date name such as Mon, Tue, Wed, etc.
2. **% A** : Display the full day name like Monday Tuesday, Wednesday, etc.
3. **% u** : Display numbers instead of days of the week like Monday = 1, Tuesday = 2, Wednesday = 3, etc.
4. **% w** : Display numbers instead of days of the week starting Sunday like Sunday = 0, Monday = 1, Tuesday = 2, etc.
5. **% d** : Display the day of the month with the leading zero (01, 02 . 09) if required.
6. **% e** : Display the day of the month, with the space before each day ('1', '2' . '9') if required. Note, single quotes are not displayed.
7. **% j** : Display the day of the year with up to two leading zeros if required.

```
anonymous@howtogeek:~$ date +%a
Sun
anonymous@howtogeek:~$ date +%A
Sunday
anonymous@howtogeek:~$ date +%u
7
anonymous@howtogeek:~$ date +%w
0
anonymous@howtogeek:~$ date +%d
07
anonymous@howtogeek:~$ date +%e
 7
anonymous@howtogeek:~$ date +%j
097
anonymous@howtogeek:~$ █
```

Week display options

1. **% U** : Display the number of weeks a year, see Sunday as the first day of the week. For example, the third week of the year, the twentieth week of the year, etc.
2. **% V** : Display the number of ISO weeks a year, see Monday as the first day of a week.
3. **% W** : Number of weeks per year, considered Monday as the first day of a week.

```
anonymous@howtogeek:~$ date +%U
14
anonymous@howtogeek:~$ date +%V
14
anonymous@howtogeek:~$ date +%W
13
anonymous@howtogeek:~$ █
```

Month display options

1. **% b** or **% h**: Displays the name of the month abbreviated as Jan, Feb, Mar, etc.
2. **% B** : Displays the full name of a month like January, February, March, etc. .
3. **% m** : Displays the number of months with the leading zero if required.

```
anonymous@howtogeek:~$ date +%b
Apr
anonymous@howtogeek:~$ date +%h
Apr
anonymous@howtogeek:~$ date +%B
April
anonymous@howtogeek:~$ date +%m
04
anonymous@howtogeek:~$ █
```

Options display year

1. **% C** : Century display has no year. For example, 2019 is 20 (20th century).
2. **% y** : Display the year in two-digit form. For example, 2019 is 19.

3. **% Y** : Displays the year with 4 digits.

```
anonymous@howtogeek:~$ date +%C
20
anonymous@howtogeek:~$ date +%y
19
anonymous@howtogeek:~$ date +%Y
2019
anonymous@howtogeek:~$ █
```

Optional display time

1. **% T** : Display time in hours: Minutes: Seconds.
2. **% R** : Display hours and minutes in Hour format: Minutes without seconds, use 24-hour clock.
3. **% r** : Display local time, using 12-hour clock and AM or PM indicator.
4. **% X** : Display local time, using 24-hour clock.

```
anonymous@howtogeek:~$ date +%T
16:15:55
anonymous@howtogeek:~$ date +%R
16:16
anonymous@howtogeek:~$ date +%r
04:16:06 PM
anonymous@howtogeek:~$ date +%X
04:16:20 PM
anonymous@howtogeek:~$ █
```

Option to display hours

1. **% H** : Display hours in the form of 00, 01, 02 . 23.
2. **% I** : Display the 12-hour clock usage time in the form of 00, 01, 02 . 12, with 0 standing ahead if required.

```
anonymous@howtogeek:~$ date +%H
16
anonymous@howtogeek:~$ date +%I
04
anonymous@howtogeek:~$ █
```

Optional display minutes

1. **% M** : Display minutes 01, 02, 03 . 59, with leading 0 if required.

```
anonymous@howtogeek:~$ date +%M
17
anonymous@howtogeek:~$ █
```

Option to display seconds

1. **% s** : Displays the number of seconds since 01/01/1970 00:00:00, starting from Unix Epoch.
2. **% S** : Display the number of seconds 01, 02, 03 . 59, with the number 0 standing ahead if required.
3. **% N** : Display nano seconds.

```
anonymous@howtogeek:~$ date +%s
1554668298
anonymous@howtogeek:~$ date +%S
21
anonymous@howtogeek:~$ date +%N
500161856
anonymous@howtogeek:~$ █
```

Option to display time zone information

1. **% z** : Displays the time difference between your time zone and UTC.
2. **%: z** : Displays the time difference between your time zone and UTC with a colon between hours and minutes. Note:: between% and z.
3. **% :: z** : Displays the time difference between your time zone and UTC with a colon between hours, minutes and seconds. Note, :: between% and z.

4. **% Z** : Displays the time zone name by letter.

```
anonymous@howtogeek:~$ date +%z
-0400
anonymous@howtogeek:~$ date +%:z
-04:00
anonymous@howtogeek:~$ date +%::z
-04:00:00
anonymous@howtogeek:~$ date +%Z
EDT
anonymous@howtogeek:~$ █
```

Options related to format

1. **% p** : Display AM or PM indicator in capital letters.
2. **% P** : Display indicator am or pm in lowercase letters.
3. **% t** : Displayed in a tab.
4. **% n** : Display to a new line.

```
anonymous@howtogeek:~$ date +%p
PM
anonymous@howtogeek:~$ date +%P
pm
anonymous@howtogeek:~$ date +"%H%t%M"
16      20
anonymous@howtogeek:~$ date +"%H%n%M"
16
21
anonymous@howtogeek:~$ █
```

Other editing options

The following options can be added to % and letters of other options to edit the display format. For example, **% - S** will remove 0 leading the one-digit seconds value.

- : The hyphen removes the 0-digit leading the one-digit value.

_ : An underscore adds a space before a one-digit number.

0 : Add 0 before the one-digit numbers.

^ : Use capital letters if possible (not all options may apply this way).

: Use the default format if possible (not all options can apply this way).

```
anonymous@howtogeek:~$ date +%m
04
anonymous@howtogeek:~$ date +%-m
4
anonymous@howtogeek:~$ date +%_m
_4
anonymous@howtogeek:~$ date +%0m
04
anonymous@howtogeek:~$ date +%^P
pm
anonymous@howtogeek:~$ date +%#Z
edt
anonymous@howtogeek:~$
```

To edit the time of a file, you can use the **-r** option. Here **-** (hyphen) replaces **%** and does not require a **+** sign.

```
date -r .bashrc
```

```
anonymous@howtogeek:~$ date -r .bashrc
Sat Apr  6 09:08:06 EDT 2019
anonymous@howtogeek:~$
```

Installing TZ allows you to change the time zone throughout a command.

```
TZ = GMT date +% c
```

```
anonymous@howtogeek:~$ date +%c
Sun 07 Apr 2019 04:32:44 PM EDT
anonymous@howtogeek:~$ TZ=GMT date +%c
Sun 07 Apr 2019 08:32:54 PM GMT
anonymous@howtogeek:~$
```

Use the date command in the script

Activate the Bash shell script to display the date and time as a small matter, create a text file with the following content and save it as `gd.sh`.

```
#!/ bin / bash

TODAY = $ (date + "Today is% A,% d of% B")
TIMENOW = $ (date + "The local time is% r")
TIME_UK = $ (TZ = BST date + "The time in the UK is% r")

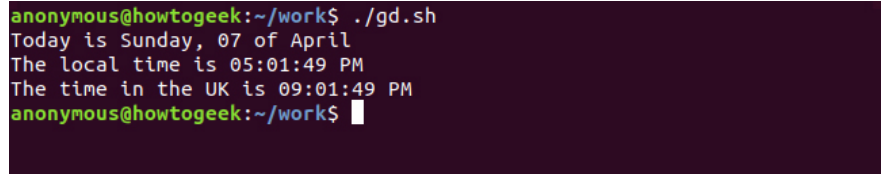
echo $ TODAY
echo $ TIMENOW
echo $ TIME_UK
```

Type the following command to set the execution permissions for the script.

```
chmod + x gd.sh
```

Run the script with this command:

```
./gd.sh
```



```
anonymous@howtogeek:~/work$ ./gd.sh
Today is Sunday, 07 of April
The local time is 05:01:49 PM
The time in the UK is 09:01:49 PM
anonymous@howtogeek:~/work$
```

You can use the date command to create a timestamp. The display script will create a folder with a timestamp as its name. It will then copy the entire text file from the current directory into it. By running this script regularly, you can take a text file image. And then it is possible to build a series of directories with different text file versions.

Note, this is not a strong backup system.

Create a text file with the following content and save it as snapshot.sh.

```
#!/ bin / bash

# obtaining the date and time
date_stamp = $(date + "% F-% H-% M-% S")

# tạo m?c th? m?c v?i tên
mkdir "$ date_stamp"

# sao chép t?p tin t? th? m?c hi?n th?i vào nó
cp * .txt "$ date_stamp"

# all done, report back and exit
echo "Text file copied to directory:" $ date_stamp
```

Type the following command to set execution permissions.

```
chmod + x snapshot.sh
```

Run the script with the following command:

```
./snapshot.sh
```

```
anonymous@howtogeek:~/work$ ls
chapter_1.txt chapter_2.txt gd.sh snapshot.sh
anonymous@howtogeek:~/work$ ./snapshot.sh
Text files copied to directory: 2019-04-07-17-10-33
anonymous@howtogeek:~/work$ ls
2019-04-07-17-10-33 chapter_1.txt chapter_2.txt gd.sh snapshot.sh
anonymous@howtogeek:~/work$ ls 2019-04-07-17-10-33/
chapter_1.txt chapter_2.txt
anonymous@howtogeek:~/work$
```

You will see the newly created folder named after the executable script date. Inside that folder is a copy of the text file.

I wish you all success!

You finished reading the article "**How to display the date and time in Linux Terminal**" edited by the [TipsMake](#) team. We hope this article has provided you with many useful tech tips and tricks. You can search for similar articles on tips and guides. Thank you for reading and for following us regularly.