



# Shell scripting (BASH) : How to create temporary random file name

March 23, 2005 in [Linux](#), [Shell scripting](#), [Tips](#), [UNIX](#) last updated April 19, 2017

Sometimes you need to create a temporary file in your shell script. There are various methods exist to create a random temporary file name. It is quite useful if your application/shell scripting needs temporary unique file names.



Live There with  
Airbnb - Sign up now

From bedrooms to  
apartments to villas - Airbnb  
has it all.

[airbnb.com](https://airbnb.com)

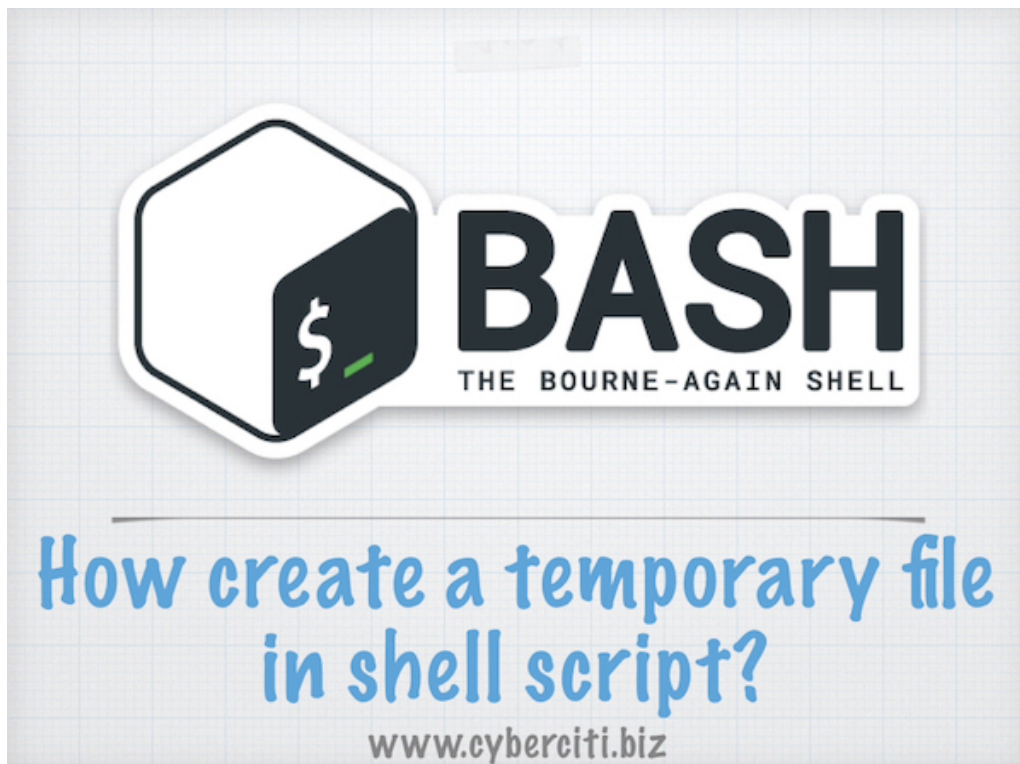


Fig.01: How create a temporary file in linux unix bash/ ksh /zsh shell script

## Method # 1 Use of mktemp or tempfile utility

As the name suggest, both of the following commands create a unique temporary file or directory. Just type the `mktemp` at the shell prompt to create it a temp file or dir:

```
$ mktemp
```

Sample outputs

```
/tmp/tmp.Le4jmo6TrH
```

OR

```
$ tempfile
```

Sample outputs:

```
/tmp/file1wqzc0
```

Please note that the `tempfile` command is deprecated; you should use always use `mktemp` command instead. So to create a temp file:

```
tfile=$(mktemp /tmp/foo.XXXXXXXXXX)
echo "a file: $tfile"
```

To create a temp dir:

```
tdir=$(mktemp -d /tmp/foo.XXXXXXXXXX)
echo "a direcotry: $tdir"
```

## How to make a directory

Make a unique temporary directory instead of a file using `-d` option. The syntax is:

```
$ mktemp -d
```

## A shell script example

```
#!/bin/bash
s="https://server1.cyberciti.biz/?Download=ips-v4&Format=text"
f=$(mktemp /tmp/myscript.XXXXXX)
wget -q -O $f $s
echo "IPv4 address downloaded to '$f'.."
echo "Processing..."
#
# logic to do something on $f here
#
# Delete the temp file
rm -f "$f"
```

Rest of the following methods are insecure and do not use them in production. They are here for historical reasons only.

## Method #2: Use \$RANDOM bash shell variable

At shell prompt type the command:

```
$ echo $RANDOM
```

Sample outputs:

```
13354
```

You can use it as follows:

```
file="/tmp/myscript.$RANDOM"
echo "Working on temp $file ..."
echo "Deleting $file ..."
rm -f "$file"
```

## Method # 3 Use of \$\$ variable

This is old and classic method. \$\$ shell variable returns the current running process this can be use to create unique temporary file as demonstrated in following script:

```
vi random2.bash
```

```
#!/bin/bash
#
TFILE="/tmp/${basename $0}.$$$.tmp"
ls &gt; $TFILE
echo "See directory listing in $TFILE"
```

Save the script and execute as follows:

```
$ chmod +x random2.bash
$ ./ random2.bash
```

Use this method if your script needs only ONE temporary file.

 14 comment

**surender** July 24, 2007 at 6:03 am

how can i record the log in the log file in shell script?

like during running whatever the user is seeing, that should be recorded in log file.

**nixCraft** July 24, 2007 at 7:52 am

Use [logger command](#)

**Artem Nosulchik** October 5, 2007 at 8:17 am

createtempfiles.bash is missing... But article is still useful :)

**nixCraft** October 5, 2007 at 11:10 am

Artem,

Thanks for the heads up! The post has been updated.

**Antti Kaihola** September 24, 2008 at 8:25 am

```
$ tempfile -d
```

does **not** create a temporary directory. Instead, it requires a directory as an argument and creates a temporary file inside that directory.

**alO** August 12, 2015 at 10:28 am

MKTEMP(1)	User Commands	MKTEMP(1)
...		
	-d, --directory	
	create a directory, not a file	
So your comment seems to be incorrect.		
To specify directory to create a file in use either		
	--tmpdir[=DIR]	
	interpret TEMPLATE relative to DIR. If DIR is not specified, use \$TMPDIR	
	if set, else /tmp. With this option, TEMPLATE must not be an absolute name. Unlike	
	with -t, TEM-	
	PLATE may contain slashes, but mktemp creates only the final component.	
or		
	-p DIR use DIR as a prefix; implies -t [deprecated]	

**tnt2br** October 30, 2008 at 6:46 pm

Thank's. Obrigado ! Very good tutorial !

**Anuj Aggarwal** March 3, 2009 at 10:27 am

Hi

useful posts.

But i need to have filenames in sequence eachtime when i run the script in which i am creating the file.

**CWS** April 30, 2013 at 11:09 pm

This is very necro, but for others who find this rather high ranking search and want to create some “temp” files in sequence, you could do something like:

```
BASE=$(mktemp)
for a in {0..5}; do
cp $BASE $BASE.$a
done
```

That said, I really don't understand why you'd care if they were in sequence.

**Hai Vu** May 21, 2009 at 5:08 pm

This article is very useful: it helps me solve my problem. Thank you.

**David** September 7, 2010 at 9:49 pm

After reading <http://www.linuxsecurity.com/content/view/full/115462/151/>, I think some of your examples may be vulnerable to symlink attacks.

\$RANDOM (once in the filename) or \$\$ alone are not enough because it may be possible for an attacker to create symlinks for all filenames.

**Yuki Matsukura** September 12, 2011 at 11:56 am

Great summary. Thank you.  
It helps writing shell script!

**Wesley** April 19, 2013 at 8:15 am

```
mktemp /path/of/directory/filename.XXXXXX
```

The more X's you add the more random characters it adds.

**Hal** September 24, 2013 at 9:04 pm

Suggest that you list the preferred method first rather than last.

Tagged as: [file names](#), [mktemp command](#), [temporary file](#), [tmpfile command](#)

@2000-2017 nixCraft. All rights reserved.

Hosted by [Linode](#)

DNS & CDN by [Cloudflare](#) and [StackPath](#)

Designed and Developed by  [Prospect One](#)











