

You have 1 free member-only story left this month. [Sign up for Medium and get an extra one](#)



Prashant Lakhera Follow

Apr 3, 2019 · 3 min read ·



Member-only



# 100 Days of DevOps — Day 52-Conditional Testing in Shell

Welcome to Day 52 of 100 Days of DevOps, Focus for today is Conditional Testing in Bash.

Today I am going to discuss test command

```
test - check file types and compare values
```

- In your script you generally see, it starts with left bracket ( [ ) when need to perform some testing.

## Exit Status

- In bash after running any command, the exit status is returned and stored in variable \$?

```
0 --> good
1 or some other code --> issue
```

## Test Integers

```
# test 1 -eq 1;echo $?
```

```
0
```

- Here we are trying to check if 1 is equal to 1 and then perform command chaining to merge two command

- As mentioned above exit status 0 mean good

```
# test 1 -eq 2;echo $?
```

```
1
```

- Exit status other than zero, in the above case, means some issue
- The same way we can test less than or greater than or not equal to

```
# less then
# test 1 -lt 2;echo $?
```

```
0
```

```
# greater then
```

```
# test 1 -gt 2;echo $?
```

```
1
```

```
# not equal to
```

```
OR
```

```
# test 1 -ne 2;echo $?
```

```
0
```

## Test Strings

- To compare string, use a single equal sign(this is different as compare to other programming languages where single equal sign is assignment operator(=))

```
# test hello = hello; echo $?
```

```
0
```

- Same way you can test not equal to by using !

```
# test hello != hello; echo $?
```

```
1
```

## Test Files

- Here I have two files(file1 and file2), where you can see file2(created at 05.15) is newer then file1(created at 05:14)

```
# ls -l file?
```

```
-rw-r--r-- 1 root root 0 Apr  3 05:14 file1
```

```
-rw-r--r-- 1 root root 0 Apr  3 05:15 file2
```

- To test if file2 is newer is than file1

```
# test file2 -nt file1;echo $?
```

```
0
```

- To check for block and character device

```
# To check block device
# test -b /dev/xvda;echo $?
```

```
0
```

```
# To check character device
# test -c /dev/ttyl;echo $?
```

```
0
```

- To check if the file exists

```
# test -e file1; echo $?
```

```
0
```

- To check if the file exists and its a regular file(use -f)

```
# mkdir file3
```

```
# test -e file3; echo $?
```

```
0
```

```
# test -f file3; echo $?
```

```
1
```

- To check if the file exists and it's non-zero

```
# test -s file2; echo $?
```

```
1
```

```
# echo "hello"> file2
```

```
# test -s file2; echo $?
```

```
0
```

- Let put together everything in the script we use in our day/today life, to check if the file exists

```
1 #!/bin/bash
2 FILE="/etc/passwd"
3 if [ -f $FILE ]; then
4     echo "The file '$FILE' exists."
5 else
6     echo "The file '$FILE' in not found."
7 fi
```

file\_exist.sh hosted with ❤️ by GitHub

[view raw](#)

- We can further improve this script by asking input from a user rather than hardcode the filename using positional parameter

```
1 #!/bin/bash
2 FILE="$1"
3 if [ -f $FILE ]; then
4     echo "The file '$FILE' exists."
5 else
6     echo "The file '$FILE' in not found."
7 fi
```

fileexist1.sh hosted with ❤️ by GitHub

[view raw](#)

- When we execute this script

```
# ./fileexist.sh testfile
```

```
The file 'testfile' in not found.
```

```
# ./fileexist.sh /etc/shadow
```

```
The file '/etc/shadow' exists.
```

- Some other functionality you can test

```
-b FILE
```

```
FILE exists and is block special
```

```
-c FILE
```

```
FILE exists and is character special
```

```
-d FILE
```

```
FILE exists and is a directory
```

```
-e FILE
```

```
FILE exists
```

```
-f FILE
```

```
FILE exists and is a regular file
```

```
-g FILE
```

```
FILE exists and is set-group-ID
```

```
-G FILE
```

```
FILE exists and is owned by the effective group ID
```

```
-h FILE
```

```
FILE exists and is a symbolic link (same as -L)
```

```
-k FILE
```

```
FILE exists and has its sticky bit set
```

```
-L FILE
```

```
FILE exists and is a symbolic link (same as -h)
```

```
-O FILE
```

```
FILE exists and is owned by the effective user ID
```

```
-p FILE
```

```
FILE exists and is a named pipe
```

```
-r FILE
```

```
FILE exists and read permission is granted
```

```
-s FILE
```

```
FILE exists and has a size greater than zero
```

```
-S FILE
```

```
FILE exists and is a socket
```

```
-t FD file descriptor FD is opened on a terminal
```

```
-u FILE
```

```
FILE exists and its set-user-ID bit is set
```

```
-w FILE
```

```
FILE exists and write permission is granted
```

```
-x FILE
```

```
FILE exists and execute (or search) permission is granted
```

Looking forward from you guys to join this journey and spend a minimum an hour every day for the next 100 days on DevOps work and post your progress using any of the below medium.

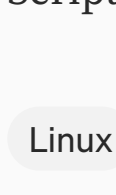
- Twitter: [@100daysofdevops](#) OR [@lakhera2015](#)
- Facebook: <https://www.facebook.com/groups/795382630808645/>
- Medium: <https://medium.com/@devopslearning>
- Slack: <https://devops-myworld.slack.com/messages/CF41EFG49/>
- GitHub Link:<https://github.com/100daysofdevops>

## Reference

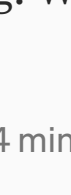
### 100 Days of DevOps — Day 0

D-day is just one day away and finally, this is a continuation of the post(I posted a month earlier)

medium.com



21



Get an email whenever Prashant Lakhera publishes.

Your email

Subscribe

By signing up, you will create a Medium account if you don't already have one. Review our [Privacy Policy](#) for more information about our privacy practices.

## More from Prashant Lakhera

Follow



AWS Community Builder, Ex-Redhat, Author, Blogger, YouTuber, RHCA, RHCDs, RHCE, Docker Certified,4XAWS, CCNA, MCP, Certified Jenkins, Terraform Certified, 1XGCP

Apr 2, 2019 Member-only

## 100 Days of DevOps — Day 51-Introduction to Bash Scripting

Welcome to Day 51 of 100 Days of DevOps, Focus for today is Bash Scripting. Wow, we have completed 50 days and in the last 50 days...

Linux · 4 min read



Share your ideas with millions of readers.

[Write on Medium](#)

Apr 1, 2019 Member-only

## 100 Days of DevOps — Day 50-Introduction to Route53 Failover

Welcome to Day 47 of 100 Days of DevOps, Focus for today is Amazon Route53 Failover On Day 49, I talked about Route53, Let extend that concept...

AWS · 4 min read



Mar 31, 2019 Member-only

## 100 Days of DevOps — Day 49-Introduction to Route53

Welcome to Day 49 of 100 Days of DevOps, Focus for today is Route53 What is AWS Route53? Amazon Route 53 is a highly available and scalable Domain Name System (DNS) web service. You can use Route 5...

AWS · 4 min read



Mar 30, 2019 Member-only

## 100 Days of DevOps — Day 48- Threat detection and mitigation at AWS

Welcome to Day 48 of 100 Days of DevOps, Focus for today is Threat detection and mitigation at AWS This 27th Wednesday, I got the chance...

AWS · 6 min read



Mar 29, 2019

## 100 Days of DevOps — Day 47-Introduction to Amazon Elastic File System (EFS)

Welcome to Day 47 of 100 Days of DevOps, Focus for today is Amazon Elastic File System (EFS) What is Amazon EFS? Amazon EFS provides...

AWS · 3 min read

