THE GEEK STUFF

Linux | DB | Open Source | Web

≡ MENU

Linux Crontab: 15 Awesome Cron Job Examples

by SATHIYAMOORTHY *on* JUNE 11, 2009

An experienced Linux sysadmin knows the importance of running the routine maintenance jobs in the background automatically.

Linux Cron utility is an effective way to schedule a routine background job at a specific time and/or day on an on-going basis.

This article is part of the on-going Productivity Tips For Geeks series. In this article, let us review 15 awesome examples of crontab job scheduling.

Linux Crontab Format

```
MIN HOUR DOM MON DOW CMD
```

Table: Crontab Fields and Allowed Ranges (Linux Crontab Syntax)

Field	Description	Allowed Value
MIN	Minute field	0 to 59
HOUR	Hour field	0 to 23

DOM	Day of Month	1-31
MON	Month field	1-12
DOW	Day Of Week	0-6
CMD	Command	Any command to be executed.

1. Scheduling a Job For a Specific Time

The basic usage of cron is to execute a job in a specific time as shown below. This will execute the Full backup shell script (full-backup) on **10th June 08:30 AM**.

Please note that the time field uses 24 hours format. So, for 8 AM use 8, and for 8 PM use 20.

```
30 08 10 06 * /home/ramesh/full-backup
```

- **30** – 30th Minute
- **08** – 08 AM
- **10** – 10th Day
- **06** – 6th Month (June)
- * – Every day of the week

2. Schedule a Job For More Than One Instance (e.g. Twice a Day)

The following script take a incremental backup twice a day every day.

This example executes the specified incremental backup shell script (incremental-backup) at 11:00 and 16:00 on every day. The comma separated value in a field specifies that the command needs to be executed in all the mentioned time.

```
00 11,16 * * * /home/ramesh/bin/incremental-backup
```

- **00** – 0th Minute (Top of the hour)
- **11,16** – 11 AM and 4 PM
- * – Every day
- * – Every month

- ***** – Every day of the week

3. Schedule a Job for Specific Range of Time (e.g. Only on Weekdays)

If you wanted a job to be scheduled for every hour with in a specific range of time then use the following.

Cron Job everyday during working hours

This example checks the status of the database everyday (including weekends) during the working hours 9 a.m – 6 p.m

```
00 09-18 * * * /home/ramesh/bin/check-db-status
```

- **00** – 0th Minute (Top of the hour)
- **09-18** – 9 am, 10 am,11 am, 12 am, 1 pm, 2 pm, 3 pm, 4 pm, 5 pm, 6 pm
- ***** – Every day
- ***** – Every month
- ***** – Every day of the week

Cron Job every weekday during working hours

This example checks the status of the database every weekday (i.e excluding Sat and Sun) during the working hours 9 a.m – 6 p.m.

```
00 09-18 * * 1-5 /home/ramesh/bin/check-db-status
```

- **00** – 0th Minute (Top of the hour)
- **09-18** – 9 am, 10 am,11 am, 12 am, 1 pm, 2 pm, 3 pm, 4 pm, 5 pm, 6 pm
- ***** – Every day
- ***** – Every month
- **1-5** -Mon, Tue, Wed, Thu and Fri (Every Weekday)

4. How to View Crontab Entries?

View Current Logged-In User's Crontab entries

To view your crontab entries type crontab -l from your unix account as shown below.

```
ramesh@dev-db$ crontab -l

@yearly /home/ramesh/annual-maintenance

*/10 * * * * /home/ramesh/check-disk-space



[Note: This displays crontab of the current logged in user]
```

View Root Crontab entries

Login as root user (su – root) and do crontab -l as shown below.

```
root@dev-db# crontab -l
no crontab for root
```

Crontab HowTo: View Other Linux User's Crontabs entries

To view crontab entries of other Linux users, login to root and use **-u {username} -l** as shown below.

```
root@dev-db# crontab -u sathiya -l

@monthly /home/sathiya/monthly-backup

00 09-18 * * * /home/sathiya/check-db-status
```

5. How to Edit Crontab Entries?

Edit Current Logged-In User's Crontab entries

To edit a crontab entries, use crontab -e as shown below. By default this will edit the current logged-in users crontab.

```
ramesh@dev-db$ crontab -e

@yearly /home/ramesh/centos/bin/annual-maintenance

*/10 * * * * /home/ramesh/debian/bin/check-disk-space

~
```

```
"/tmp/crontab.XXXXyjWkHw" 2L, 83C



[Note: This will open the crontab file in Vim editor for editing.

Please note cron created a temporary /tmp/crontab.XX... ]
```

When you save the above temporary file with :wq, it will save the crontab and display the following message indicating the crontab is successfully modified.

```
~

"crontab.XXXXyjWkHw" 2L, 83C written

crontab: installing new crontab
```

Edit Root Crontab entries

Login as root user (su – root) and do crontab -e as shown below.

```
root@dev-db# crontab -e
```

Edit Other Linux User's Crontab File entries

To edit crontab entries of other Linux users, login to root and use **-u {username} -e** as shown below.

```
root@dev-db# crontab -u sathiya -e

@monthly /home/sathiya/fedora/bin/monthly-backup

00 09-18 * * * /home/sathiya/ubuntu/bin/check-db-status

~

~

~

"/tmp/crontab.XXXXyjWkHw" 2L, 83C
```

6. Schedule a Job for Every Minute Using Cron.

Ideally you may not have a requirement to schedule a job every minute. But understanding this example will will help you understand the other examples

mentioned below in this article.

```
* * * * * CMD
```

The * means all the possible unit — i.e every minute of every hour through out the year. More than using this * directly, you will find it very useful in the following cases.

- When you specify */5 in minute field means every 5 minutes.
- When you specify 0-10/2 in minute field mean every 2 minutes in the first 10 minute.
- Thus the above convention can be used for all the other 4 fields.

7. Schedule a Background Cron Job For Every 10 Minutes.

Use the following, if you want to check the disk space every 10 minutes.

```
*/10 * * * * /home/ramesh/check-disk-space
```

It executes the specified command check-disk-space every 10 minutes through out the year. But you may have a requirement of executing the command only during office hours or vice versa. The above examples shows how to do those things.

Instead of specifying values in the 5 fields, we can specify it using a single keyword as mentioned below.

There are special cases in which instead of the above 5 fields you can use @ followed by a keyword — such as reboot, midnight, yearly, hourly.

Table: Cron special
keywords and its
meaning

Keyword	Equivalent
@yearly	0 0 1 1 *
@daily	0 0 * * *
@hourly	0 * * * *

@reboot	Run at startup.

8. Schedule a Job For First Minute of Every Year using @yearly

If you want a job to be executed on the first minute of every year, then you can use the **@yearly** cron keyword as shown below.

This will execute the system annual maintenance using annual-maintenance shell script at 00:00 on Jan 1st for every year.

```
@yearly /home/ramesh/red-hat/bin/annual-maintenance
```

9. Schedule a Cron Job Beginning of Every Month using @monthly

It is as similar as the @yearly as above. But executes the command monthly once using **@monthly** cron keyword.

This will execute the shell script tape-backup at 00:00 on 1st of every month.

```
@monthly /home/ramesh/suse/bin/tape-backup
```

10. Schedule a Background Job Every Day using @daily

Using the **@daily** cron keyword, this will do a daily log file cleanup using cleanup-logs shell scriptat 00:00 on every day.

```
@daily /home/ramesh/arch-linux/bin/cleanup-logs "day started"
```

11. How to Execute a Linux Command After Every Reboot using @reboot?

Using the **@reboot** cron keyword, this will execute the specified command once after the machine got booted every time.

```
@reboot CMD
```

12. How to Disable/Redirect the Crontab Mail Output using MAIL keyword?

By default crontab sends the job output to the user who scheduled the job. If you want to redirect the output to a specific user, add or update the MAIL variable in the crontab as shown below.

```
ramesh@dev-db$ crontab -l

MAIL="ramesh"


@yearly /home/ramesh/annual-maintenance

*/10 * * * * /home/ramesh/check-disk-space


[Note: Crontab of the current logged in user with MAIL variable]
```

If you wanted the mail not to be sent to anywhere, i.e to stop the crontab output to be emailed, add or update the MAIL variable in the crontab as shown below.

```
MAIL=""
```

13. How to Execute a Linux Cron Jobs Every Second Using Crontab.

You cannot schedule a every-second cronjob. Because in cron the minimum unit you can specify is minute. In a typical scenario, there is no reason for most of us to run any job every second in the system.

14. Specify PATH Variable in the Crontab

All the above examples we specified absolute path of the Linux command or the shell-script that needs to be executed.

For example, instead of specifying /home/ramesh/tape-backup, if you want to just specify tape-backup, then add the path /home/ramesh to the PATH variable in the crontab as shown below.

```
ramesh@dev-db$ crontab -l


PATH=/bin:/sbin:/usr/bin:/usr/sbin:/home/ramesh


@yearly annual-maintenance
*/10 * * * * check-disk-space


[Note: Crontab of the current logged in user with PATH variable]
```

15. Installing Crontab From a Cron File

Instead of directly editing the crontab file, you can also add all the entries to a cron-file first. Once you have all thoese entries in the file, you can upload or install them to the cron as shown below.

```
ramesh@dev-db$ crontab -l
no crontab for ramesh


$ cat cron-file.txt
@yearly /home/ramesh/annual-maintenance
*/10 * * * * /home/ramesh/check-disk-space


ramesh@dev-db$ crontab cron-file.txt


ramesh@dev-db$ crontab -l
@yearly /home/ramesh/annual-maintenance
*/10 * * * * /home/ramesh/check-disk-space
```

Note: This will install the cron-file.txt to your crontab, which will also remove your old cron entries. So, please be careful while uploading cron entries from a cron-file.txt.

Additional Cron Tutorials

- [Cron Vs Anacron: How to Setup Anacron on Linux (With an Example)](#)
- [How to Run Cron Every 5 Minutes, Seconds, Hours, Days, Months](#)
- [6 Linux Crontab Command Examples](#)
- [How To Install, Edit, or Remove Cron Jobs in Batch Mode](#)
- [Understand at, atq, atrm, batch Commands using 9 Examples](#)

Awesome Linux Articles

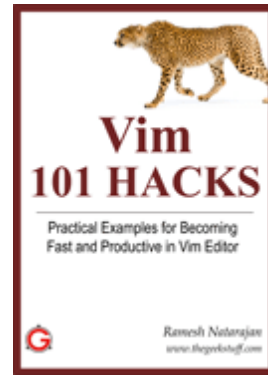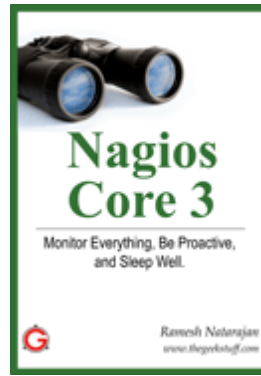Following are few awesome **15 examples** articles that you might find helpful.

- [Mommy, I found it! — 15 Practical Linux Find Command Examples](#)
- [15 Examples To Master Linux Command Line History](#)
- [Unix LS Command: 15 Practical Examples](#)
- [Get a Grip on the Grep! – 15 Practical Grep Command Examples](#)

Like 5   Add your comment

If you enjoyed this article, you might also like..

1. 50 Linux Sysadmin Tutorials
2. 50 Most Frequently Used Linux Commands (With Examples)
3. Top 25 Best Linux Performance Monitoring and Debugging Tools
4. Mommy, I found it! – 15 Practical Linux Find Command Examples
5. Linux 101 Hacks 2nd Edition eBook **Free**

- [Awk Introduction – 7 Awk Print Examples](#)
- [Advanced Sed Substitution Examples](#)
- [8 Essential Vim Editor Navigation Fundamentals](#)
- [25 Most Frequently Used Linux IPTables Rules Examples](#)
- [Turbocharge PuTTY with 12 Powerful Add-Ons](#)

Tagged as: /etc/cron.allow, /etc/cron.d, /etc/crontab, /var/spool/cron, Cron examples, Crontab PHP, Crontab Script, Crontab

Tutorial, Learn crontab, Linux Crond Daemon, Practical Cron example, Productivity Tips For Geeks, Schedule job in Linux

Comments on this entry are closed.

PREVIOUS COMMENTS

prasad

December 30, 2014, 2:47 am

it is very nice ...Thanks Ramesh.

∞

Rucha

January 1, 2015, 12:20 pm

Great post!! Very helpful.Thanks

∞

leslie Satenstein

January 11, 2015, 8:52 pm

I have a crontab skeleton that I use for new system setups.
Here it is.

If you copy and paste, you will find that I have reminders added to the top of the file for the first few lines. To properly view it, use a fixed size font.

```
############################################################
#
# string meaning
# —— ——-
# @reboot Run once, at startup.
# @yearly Run once a year, "0 0 1 1 *".
# @annually (same as @yearly)
# @monthly Run once a month, "0 0 1 * *".
# @weekly Run once a week, "0 0 * * 0".
# @daily Run once a day, "0 0 * * *".
# @midnight (same as @daily)
# @hourly Run once an hour, "0 * * * *".
#
#mm hh Mday Mon Dow CMD # minute, hour, month-day month DayofW CMD
#......................................Minute of the hour
#| ...............................Hour in the day (0..23)
#| | ........................Day of month, 1..31 (mon,tue,wed)
#| | | .................Month (1.12) Jan, Feb.. Dec
#| | | | ........day of the week 0-6 7==0
#| | | | | |command to be executed
#V V V V V V
15 3 * * * find $HOME -name core 2>/dev/null | xargs rm -f >/dev/null 2>&1
0 11 12 1 * mailx leslie%Happy Birthday!%Time for lunch in an hour.
15 10 * * 1,5 /usr/sbin/fstrim -a #trim for ssd
* * */4 * * find /home/leslie -name core 2>/dev/null | xargs rm -f /dev/null 2>&1
```

∞

---

leslie Satenstein

January 14, 2015, 10:06 pm

Add to the above, to run a job on the last day of the month.
#to run something on the last day of the month (/root/test)
36 14 28-31 * * [`date -d tomorrow +%d` -eq '01'] && /root/test

∞

---

kalyan

January 21, 2015, 5:46 am

Hello @natarajan:: i have a reqirement..we are using weblogic APP Serevr 10.3.6 on RHEL.i want to clear the "tmp" folder present in my (domians/servers/tmp).Not as the systems tmp..so hot to write a cron job cmd/script to clear tmp data once a day....tell me .Thanks in Advance...

∞

Muhammad Nouman

February 11, 2015, 11:08 pm

I think the value of weekday should be 1-7 rather 0-6 or this differs from linux flavours.. I am using centos 6.x

Regards,

Nouman

∞

Muhammad Nouman

February 11, 2015, 11:12 pm

Just got the answer after posting questions and here I am answering my own question:

0 and 7 stand for Sunday.

Hence, the numbers from 1 to 6 are fixed to Monday.. Saturday.

∞

Vijay Kumar Kanta

February 16, 2015, 4:45 am

Murthy garu, the article is a wonderful piece of resource for Linux geeks.

∞

Pradeep

March 3, 2015, 7:58 am

How to test if cronjob is working. I have setup the following cronjob:

* 03 * * * root /home/pi/SafetyLINK/Anchor/upgrade

* 03 * * * root /home/pi/SafetyLINK/Anchor/upgrade/upgrade

"upgrade" is a executable file with permission 777.

But its not executing. How can I test if cronjob is working or nor.

∞

Amit gupta

March 13, 2015, 1:28 pm

Hi Everyone…

It' awesome post….good job guys..

I have below query…

I created a bash script(recycle.sh) …I want to run it on alternate Sunday at 23:00

Can someone share the crontab command for this..

Could u plz help me here ..Thanks..

∞

aisha

May 12, 2015, 8:15 pm

Thanks !!!

∞

nani

June 26, 2015, 5:03 pm

Nice Material for who are learning to the basics of the Linux job scheduling
How executing an certain time period of the jobs

thanks bro

∞

Leslie

July 5, 2015, 9:59 am

Here are a few hints about how to run every alternate Sunday

The date function Date "+%U" Returns the "week number" in the year, with Sunday as the first day of the week. If you take the residue of the week number by division by two, the job will run every second Sunday. There is only one problem however, at year end, some years have 53 weeks. Therefore you have to modify the script to handle a 53 week year.
This is what I mean
Wed date= 2015-12-30 dow=3 Week Number=53
Thu date= 2015-12-31 dow=4 Week Number=53
Fri date= 2016-01-01 dow=5 Week Number=53 << Not week 1
Sat date= 2016-01-02 dow2=6 Week Number=53
Here below is a script for you to modify

```
#!/bin/bash
# this script is titled evenodd.sh
# Replace echo 0 with your command for even weeks"
# and Replace echo 1 with your command for odd weeks"
# This will work well except for last week of December 2015
# Last week of December 2015 is week 53.
#
Week=$(date +%U)
ODD=$( expr $Week % 2)
if [ $ODD == 1 ]; then
echo "1 Replace the next line with your second week Sunday job"
/Path/to/your/recycle.sh
else
```

echo "0 Week is Even and use th next line for your second week Sunday Job"
#Path/to/your/recycle.sh
fi

Your crontab entry will have

0 23 * * sun oddeven.sh

∞

Giacomo

July 6, 2015, 7:35 am

Hi, how are you?
I have a question. I would like to use cron to set up a backup on the cloud every
hour. But what happens if a backup has not finished to upload the files and the
next one is launched? For example, the backup at 13:00 starts to upload a large
file and is still uploading at 14:00. But at this moment the next backup is
launched. What happens? Cron knows that the previous command has not
finished or he launch it anyways?
Thanks 🙄

∞

---

vikas

July 10, 2015, 3:04 am

Hi All, I am a pure windows guy. I need to investigate some solaris servers and
need to send Cron jobs details to management obviously that should be easily
readable.So any way to get the easily formatted, readable output eg:tabular,html
report etc..

∞

---

rizwan

July 20, 2015, 1:59 am

how to run a php script using cron for every 5 mins

∞

Gayu

July 24, 2015, 12:51 am

@rizwan:

to run script in every 5 minutes- */5 * * * * PHPscriptpath

@Ramesh–Very useful post 😊

∞

KennyW

July 28, 2015, 2:53 am

Great Post, thanks..

∞

martha

September 8, 2015, 4:29 am

hi, i wanted to create a scheduled backup automatically. I wanted to use cron. Anyone help me where to start and what i should do?

Thanks.

∞

Sagar

October 17, 2015, 6:37 am

Its really good

∞

Vineet

October 20, 2015, 10:33 pm

0 8 * * * /usr/local/bin/perl cust_bugs.pl

It is suppose to run 8 am Pecific time but it's running 1 am pecific time
I checked server time, which is set as per pecific time.
What is wrong here?

∞

mongrel

October 29, 2015, 6:20 pm

This has to be the best linux help guide I've seen on the webz.
Good work sir.

∞

Meg

December 4, 2015, 1:37 pm

Very useful post. Thank you for your effort!

∞

uttam

January 13, 2016, 1:58 am

How to run crontab on the last day of every month

∞

Sufian

February 4, 2016, 2:27 pm

Could you show me any cron job with shell script step by step. I did
crontab -e
*/10 * * * * /tmp/test1.sh

Escape

:wq!

But, it is not working!

Could you show me all the lines, which are required to successfully run the crontab. I saw online that it is must to put #!/bin/sh, but I don't know where

Another, I have to execute also [chmod +x /tmp/test1.sh

∞

Leslie Satenstein

February 8, 2016, 7:55 pm

Yes for the chmod for test1.sh What are the read/write/execute parameters for test1.sh contents? Also, check ownerships

∞

Tory

February 21, 2016, 1:54 pm

Could you tell me what I did wrong in this crontab?

0 0 * * 2 su – ldmosley -c "VAR1=`date +%U`;VAR2=`expr $VAR1 % 2`; if [$VAR2=1]; then Server-Health.sh > health_`date +%F_%T`.txt; fi"

∞

Leslie Satenstein

February 22, 2016, 10:33 pm

several points

you want to tun as root, creat the crontab entry using the root account.

su – asks for root's password. You cannot provide a password if there is a task without a terminal

You must include the path to ldmosley. When it runs as root, ldmosley may not be in root's path statement.

Why dont you create a script for all that bash scripting and put the script in /usr/local/bin

Then your crontab line will be much easier to maintain.

∞

---

manjit

March 21, 2016, 3:30 am

normally i would skip thanking the creator of the article. Not this time.

This is very good articulate stuff . so simple anyone can understand. Appreciate the author for sharing the knowledge and helping people worldwide

∞

Satyanarayana Reddy Vakiti

April 26, 2016, 12:53 pm

Hi Experts,

I need to schedule the "executable" jar to run as when ever a file get dropped in the same folder as of that "executable jar". Could you please help me out on how to do it in Linux.

Thanks in advance.

BR,
Satya.

∞

Yahya

May 3, 2016, 5:37 am

Hi Ramesh,

can you please advice how to create cron-entry for this timing

23:30 to 11:30 am

∞

ramesh

June 20, 2016, 12:12 am

@yahya u can set cronjob like this as bellow

30 11-23 * * * /cmd

∞

Leslie Satenstein

June 23, 2016, 7:33 pm

23:30 to 11:30am on hour:30

23:30 to midnight
30 23* * *
#midnight to 11:30am
30 0-11 * * *

∞

W.Lego

July 6, 2016, 3:18 pm

One important thing you left out... how to add comment to the crontab – unless
the author assumes that everyone reading this knows to apply the same comment
syntax as a shell script.

∞

Yamkela Kutu

January 10, 2017, 2:28 am

Very helpful, learnt a lot in a space of 10 min. Thank you

∞

---

apurva

January 26, 2017, 10:13 pm

Very useful post. Thanks alot 🙂 appreciate it.

∞

Kailash

March 7, 2017, 4:15 am

I have the script file – TEST-NFT3750-CALL-SCRIPT.sh

crontab -e (Contains the following)
PATH=/usr/kerberos/bin:/usr/local/bin:/bin:/usr/bin:/usr/kerberos/bin:/usr/local/bin:/bin:/usr/bin:/sbin:/usr/sbin:/usr/local/sbin:/home/ypokhrel/bin:/home/ypokhrel
(ypokhrel -is the user)

* * * * * /home/ypokhrel/TEST-NFT3750-CALL-SCRIPT.sh >> /tmp/script.log

This is not working. Could someone help here.

∞

Don Chaplin

April 12, 2017, 11:16 am

Can a start time of 8:00 am for the date range of August 31st through September 4th be represented by cron?

∞

Leslie Satenstein

May 11, 2017, 7:50 pm

Don, the answer is yes

Cron handles tasks that can be launched between midnight to 23.59 the same day. If you want to cross midnight with a job stream, enter the start time on the start date and create a second entry for a job to be started on the following date. If your "before midnight job" crosses the midnight time, and you must wait for it to finish, you could add a last step to it. That last step can trigger the job that starts immediately after the preceding job ends.

∞

Leslie Satenstein

May 11, 2017, 7:58 pm

Don, re 8am job for August, please please read the first few lines of the example crontab listing above.

Check the second entry to the link
http://forums.fedoraforum.org/showthread.php?p=1697840#post1697840

∞

Leslie Satenstein

May 11, 2017, 9:03 pm

Ever want to run something every other week? What is a simple way to insure that irrespective of the year number, your code will always run every second week. The small script below, returns a 1 for an odd week, and 0 for an even week, based on the count of the number of weeks since 1 Jan 1970

————————————————————————-
#!/bin/bash
return a 1 if the week number is odd
return a 0 if the week number is even
You can add one to week number to have
the reverse result, 0 if week is odd
#
The Linux/Unix date function with %s returns the

number of seconds since midnight 1 Jan 1970 (birth of Unix)

Author: Leslie S May 11, 2017

#

s=$(date +%s) #seconds since 1 jan 1970

secsperweek=604800 #60*60*24*7

two=2 #used to get remainder

#

weeks=$(($s / $secsperweek)) #weeks since 1 jan 1970

#echo "weeks since 1 Jan 1970 $weeks"

evenodd=$(($weeks % $two))

if [$((evenodd == 1))];then

echo "Weeknumber is odd, returning $evenodd "

else

echo "weeknumber is even, returning 0"

fi

exit $evenodd

————————————————————————

The above calculates 1(odd) for 2017/05/11

For a job that is to run every other week, use the above as an entry in your crontab.

∞

suresh

February 4, 2019, 4:53 am

how to write a cron expression for every 6 hour with a specific time start . suppose i want to start at 9.00 am and it will be scheduled every 6 hour from 9..00 am

∞

PREVIOUS COMMENTS

Next post: PPM Install Error: no suitable installation target found for package

Previous post: 4 Easy Steps To Upgrade Linksys Wireless Router Firmware

EBOOKS

Free
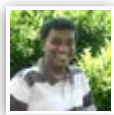
 Linux 101 Hacks 2nd Edition eBook - Practical Examples to Build a Strong Foundation in Linux

Bash 101 Hacks eBook - Take Control of Your Bash Command Line and Shell Scripting

Sed and Awk 101 Hacks eBook - Enhance Your UNIX / Linux Life with Sed and Awk

Vim 101 Hacks eBook - Practical Examples for Becoming Fast and Productive in Vim Editor

Nagios Core 3 eBook - Monitor Everything, Be Proactive, and Sleep Well

The Geek Stuff
16,196 followers

Follow Page

Share

POPULAR POSTS

15 Essential Accessories for Your Nikon or Canon DSLR Camera

12 Amazing and Essential Linux Books To Enrich Your Brain and Library

50 UNIX / Linux Sysadmin Tutorials

50 Most Frequently Used UNIX / Linux Commands (With Examples)

How To Be Productive and Get Things Done Using GTD

30 Things To Do When you are Bored and have a Computer

Linux Directory Structure (File System Structure) Explained with Examples

Linux Crontab: 15 Awesome Cron Job Examples

Get a Grip on the Grep! – 15 Practical Grep Command Examples

Unix LS Command: 15 Practical Examples

15 Examples To Master Linux Command Line History

Top 10 Open Source Bug Tracking System

Vi and Vim Macro Tutorial: How To Record and Play

Mommy, I found it! -- 15 Practical Linux Find Command Examples

15 Awesome Gmail Tips and Tricks

15 Awesome Google Search Tips and Tricks

RAID 0, RAID 1, RAID 5, RAID 10 Explained with Diagrams

Can You Top This? 15 Practical Linux Top Command Examples

Top 5 Best System Monitoring Tools

Top 5 Best Linux OS Distributions

How To Monitor Remote Linux Host using Nagios 3.0

Awk Introduction Tutorial – 7 Awk Print Examples

How to Backup Linux? 15 rsync Command Examples

The Ultimate Wget Download Guide With 15 Awesome Examples

Top 5 Best Linux Text Editors

Packet Analyzer: 15 TCPDUMP Command Examples

The Ultimate Bash Array Tutorial with 15 Examples

3 Steps to Perform SSH Login Without Password Using ssh-keygen & ssh-copy-id

Unix Sed Tutorial: Advanced Sed Substitution Examples

UNIX / Linux: 10 Netstat Command Examples

The Ultimate Guide for Creating Strong Passwords

6 Steps to Secure Your Home Wireless Network

Turbocharge PuTTY with 12 Powerful Add-Ons

CATEGORIES

Linux Tutorials

Vim Editor

Sed Scripting

Awk Scripting

Bash Shell Scripting

Nagios Monitoring

OpenSSH

IPTables Firewall

Apache Web Server

MySQL Database

Perl Programming

Google Tutorials

Ubuntu Tutorials

PostgreSQL DB

Hello World Examples

C Programming

C++ Programming

DELL Server Tutorials

Oracle Database

VMware Tutorials

ABOUT THE GEEK STUFF

My name is **Ramesh Natarajan**. I will be posting instruction guides, how-to, troubleshooting tips and tricks on Linux, database, hardware, security and web. My focus is to write articles that will either teach you or help you resolve a problem. Read more about Ramesh Natarajan and the blog.

CONTACT US

Email Me : Use this Contact Form to get in touch me with your comments, questions or suggestions about this site. You can also simply drop me a line to say hello!.

Follow us on Twitter

Become a fan on Facebook

SUPPORT US

Support this blog by purchasing one of my ebooks.

Bash 101 Hacks eBook

Sed and Awk 101 Hacks eBook

Vim 101 Hacks eBook

Nagios Core 3 eBook