



SEARCH

Type and hit enter...

HOME   BACKUP TOOLS   COMMAND LINE UTILITIES   DATABASE   DEVOPS   MOBILE   PROGRAMMING   SECURITY   VIRTUALIZATION

KEEP IN TOUCH

 FACEBOOK

 LINKEDIN

 EMAIL

 RSS

Home > Command line utilities > How To Install Perl Modules On Linux

COMMAND LINE UTILITIES ♦ LINUX ♦ OPENSOURCE ♦ PACKAGE MANAGEMENT ♦ PERL ♦ PROGRAMMING

# How To Install Perl Modules On Linux

Written by Sk | January 30, 2020 | 36.5k views

 0 comment           

This brief guide explains how to install Perl modules on Linux from [CPAN](#) (Comprehensive **P**erl **A**rchive **N**etwork) repository. As of writing this guide, there are **185,128 Perl modules** available in CPAN. Many programs written in Perl programming language depends on certain perl modules to perform a specific task. For example, the other day I was testing **Sysadmin-util** which provides a collection of [useful tools for Linux/Unix sysadmins](#). When I test a specific tool called **multi-ping**, I encountered with the following error:

```
$ ./multi-ping google.com
The required Net::DNS module is missing. Aborting.
```

In such cases, you need to find and install the missing perl module in order to use that program.

## Install Perl Modules On Linux

There are many tools available to install and Perl modules. We are going to try two tools namely **cpan** and **cpanm**. It is worth mentioning that Many modules on CPAN requires the recent version of **Perl version 5.8** or above.

Please ensure that you have installed '**make**' package on your Linux distribution. 'make' is an essential tool for building perl Modules.

If you don't install 'make', you might encountered with an error something like below:

```
Can't configure the distribution. You probably need to have 'make'.
```

'make' package is available in the default repositories most Linux distributions.

```
$ sudo pacman -S make
```

On **Debian, Ubuntu, Linux Mint**:

```
$ sudo apt install make
```

On **Fedora**:

```
$ sudo dnf install make
```

On **RHEL, CentOS**:

```
$ sudo yum install make
```

On **SUSE/openSUSE**:

```
$ sudo zypper install make
```

Let us go ahead and install perl modules.

## Install Perl modules using cpan

**cpan** is a command line client for CPAN repository and is distributed with all Perl editions by default.

To install a Perl module, for example **Net::DNS**, enter into cpan shell using command:

```
$ sudo cpan
```

And type the following command from cpan prompt to install the module:

```
install Net::DNS
```

Once the module is installed, type '**exit**' to return back to your shell.

You can also directly install the module from the Terminal using command:

```
$ sudo cpan Net::DNS
```

## Install Perl modules using Cpanminus

**Cpanminus** or **cpanm** is a cpan client to get, unpack, build and install modules from CPAN repository. It is a standalone, dependency-free script that requires zero-configuration. Many experienced Perl developers prefer cpanm over cpan.

cpanminus can be installed in many ways.

### 1. Using Perl:

This website uses cookies to improve your experience. By using this site, we will assume that you're OK with it.

Accept

Read More



SEARCH

KEEP IN TOUCH

 FACEBOOK

 LINKEDIN

 EMAIL

 RSS

To install latest cpanm version on your Linux system, just run:

```
$ curl -L https://cpanmin.us | perl - --sudo App::cpanminus
```

This command will install cpanm system-wide and install cpanm binary to your bin directory like **/usr/local/bin**.

**2. Using distribution's package manager:**

cpanm is also available in the default repositories of several Linux distributions. It is stable version, but bit old.

To install cpanminus on Arch Linux and its variants, run:

```
$ sudo pacman -S cpanminus
```

On Debian, Ubuntu, Linux Mint:

```
$ sudo apt install cpanminus
```

On CentOS:

```
$ sudo yum install perl-App-cpanminus
```

**3. Manual installation:**

Alternatively, you can manually download latest cpanm binary and put it in your \$PATH like below.

```
$ curl -L https://cpanmin.us/ -o cpanm
```

```
$ chmod +x cpanm
```

```
$ sudo mv cpanm /usr/local/bin/cpanm
```

After installing cpanm, you can install any Perl modul, for example Net::DNS, by running the following command from your Terminal:

```
$ sudo cpanm Net::DNS
```

**Sample output:**

```
--> Working on Net::DNS
Fetching http://www.cpan.org/authors/id/N/NL/NLNETLABS/Net-DNS-1.21.tar.gz ...
Configuring Net-DNS-1.21 ... OK
==> Found dependencies: Digest::HMAC
--> Working on Digest::HMAC
Fetching http://www.cpan.org/authors/id/G/GA/GAAS/Digest-HMAC-1.03.tar.gz ...
```

SEARCH

KEEP IN TOUCH

-  FACEBOOK
-  LINKEDIN
-  EMAIL
-  RSS

```
Configuring Digest-HMAC-1.03 ... OK
Building and testing Digest-HMAC-1.03 ... OK
Successfully installed Digest-HMAC-1.03
Building and testing Net-DNS-1.21 ... OK
Successfully installed Net-DNS-1.21
2 distributions installed
```

Install missing Perl modules using distribution's package manager

Many Perl modules are available as packages, so you can install it using your distribution's package manager.

On Debian, Ubuntu:

```
$ apt-cache search 'perl$' | grep Net::DNS
```

Output:

```
libnet-dns-zonefile-fast-perl - fast BIND-style zonefile parser on top of Net::DNS
```

As you can see, the Net::DS module is provided by "libnet-dns-zonefile-fast-perl", so let us install it using command:

```
$ sudo apt install libnet-dns-zonefile-fast-perl
```

To find the missing module on Arch Linux, run:

```
$ pacman -Ss '^perl-' | grep Net::DNS
```

And install the missing module using 'pacman' command.

List installed Perl modules

To list the installed Perl modules, use 'perldoc' command:

```
$ perldoc perllocal
```

Sample output:

```
Thu Jan 30 10:45:11 2020: "Module" Digest::HMAC
* "installed into: /usr/local/share/perl/5.26.1"

* "LINKTYPE: dynamic"

* "VERSION: 1.03"

* "EXE_FILES: "

Thu Jan 30 10:47:41 2020: "Module" Net::DNS
```

SEARCH

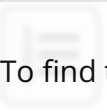
KEEP IN TOUCH

f FACEBOOK

in LINKEDIN

✉ EMAIL

📡 RSS



```
* "LINKTYPE: dynamic"
```

```
* "VERSION: 1.21"
```

```
* "EXE_FILES: "
```

```
Thu Jan 30 10:48:54 2020: "Module" Digest::BubbleBabble
```

```
* "installed into: /usr/local/share/perl/5.26.1"
```

```
* "LINKTYPE: dynamic"
```

```
* "VERSION: 0.02"
```

```
* "EXE_FILES: "
```

SEARCH

KEEP IN TOUCH

 FACEBOOK

 LINKEDIN

 EMAIL

 RSS

Another way to list the installed Perl modules is to use 'instmodsh' command like below.

```
$ instmodsh
```

You will see the following output:

```
Available commands are:
l - List all installed modules
m <module> - Select a module
q - Quit the program
cmd?
```

In **cmd** prompt, type '**l**' to list the modules.

```
Installed modules are:
App::cpanminus
Digest::BubbleBabble
Digest::HMAC
Module::Build
Net::DNS
Perl
cmd?
```

Please note that the above two commands will list the modules installed with cpan. There could be many modules installed either manually or pre-installed with your Linux distribution.

To find all installed Perl modules, run:

```
$ cpan -l
```

Or,

```
$ cpan -a
```

Uninstall Perl modules

The Perl modules can be easily removed using cpanm using command:

```
$ sudo cpanm --uninstall Net::DNS
```

Type y and hit ENTER to remove the module long with all configuration files.

```
Net::DNS contains the following files:

/usr/local/man/man3/Net::DNS.3pm
/usr/local/man/man3/Net::DNS::Domain.3pm
/usr/local/man/man3/Net::DNS::DomainName.3pm
[...]
/usr/local/share/perl/5.26.1/Net/DNS/Text.pm
/usr/local/share/perl/5.26.1/Net/DNS/Update.pm
/usr/local/share/perl/5.26.1/Net/DNS/ZoneFile.pm

Are you sure you want to uninstall Net::DNS? [y] y
```

Hope this helps.



Thanks for stopping by!

Help us to help you:

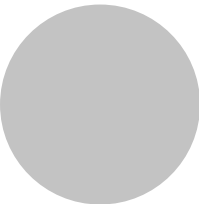
- Subscribe to our Email Newsletter : [Sign Up Now](#)
- Support OSTechNix : [Donate Via PayPal](#)
- Download free E-Books and Videos : [OSTechNix on TradePub](#)
- Connect with us: [Reddit](#) | [Facebook](#) | [Twitter](#) | [LinkedIn](#) | [RSS feeds](#)

Have a Good day!!

- CLI
- COMPREHENSIVE PERL ARCHIVE NETWORK
- CPAN
- CPANM
- CPANMINUS
- LINUX
- PERL
- PERL MODULES

0 comment

[0](#)



SK

Senthilkumar Palani (aka SK) is the Founder and Editor in chief of OSTechNix. He is a Linux/Unix enthusiast and FOSS supporter. He lives in Tamilnadu, India.

- 
- 
- 
- 
- 

Previous post

Some Useful Tools For Linux System Admins

Next post

How To Manually Add Messages To Linux System Log Files

This website uses cookies to improve your experience. By using this site, we will assume that you're OK with it.

Accept

Read More



YOU MAY ALSO LIKE

How To Search The Arch Wiki Website From...

March 3, 2017

How To Save Linux Command Output To An...

January 20, 2020

Different Ways To List Directory Contents Without Using...

April 1, 2019

LEAVE A COMMENT

Your Comment

Name\*

Email\*

☐ Save my name, email, and website in this browser for the next time I comment.

\* By using this form you agree with the storage and handling of your data by this website.

SUBMIT

This site uses Akismet to reduce spam. [Learn how your comment data is processed.](#)

Home > Command line utilities > Some Useful Tools For Linux System Admins

COMMAND LINE UTILITIES > LINUX > LINUX ADMINISTRATION > LINUX COMMANDS > OPENSOURCE > PERL > UTILITIES

Some Useful Tools For Linux System Admins

Written by Sk | **Published:** January 28, 2020 | **Last Updated on** July 26, 2020 | 4.9k views

0 comment

1

This guide provides a list of useful tools for Linux system admins. This list doesn't include any complex programs (like Ansible, Puppet) that requires a steep learning curve to setup and use in production. Most of the tools given here are just scripts and are very easy to install and use in real time. If you're a budding Linux/Unix sysadmin who has limited experience, make use of these programs to get the job done easily and quickly. These tools may not be useful all the time, but some of them might help.

About Sysadmin-util

Sysadmin-util is a collection of scripts that helps the novice sysadmins and users to perform various tasks.

This website uses cookies to improve your experience. By using this site, we will assume that you're OK with it.

Accept Read More

SEARCH

Type and hit enter...

KEEP IN TOUCH

FACEBOOK

LINKEDIN

EMAIL

RSS

Sysadmin-util consists of the following scripts:

1. ago

2. chronic

3. cidr2ip

4. collapse

5. dupes

6. empty-dir

7. expand-ipv6

8. flush-firewall

9. graphite\_send

10. ipaddr

11. maybe

12. mk-passwd-hash

13. multi-ping

14. mysql-slave-check

15. pyhttpd

16. randpass

17. since

18. splay

19. ssh-auth-types

20. ssh-test

21. ssl-expiry-date

22. timeout

23. until-error

24. until-success

25. when-down

26. when-up

27. which-shell

28. with-lock

SEARCH

KEEP IN TOUCH

- f

FACEBOOK
- in

LINKEDIN
- ✉

EMAIL
- 📡

RSS

Let us go ahead and see how to install and use some of these tools with examples.

## Install sysadmin-util On Linux

Like I already said, the installation of all tools listed here is trivial. In fact, there is no need to install. Just clone its repository and start using the tools in no time.

```
$ git clone https://github.com/skx/sysadmin-util.git
```

The above command will clone all the contents of Sysadmin-util repository in the current directory. Cd into that directory and run any tools of your choice. It's that simple!

## Sysadmin-util Provides the following useful tools for Linux system admins

### Ago

This tool helps you to find how long ago a file or directory was modified in human-readable format.

Make sure you're in sysadmin-util directory:

```
$ cd sysadmin-util
```

To find when was a file or directory was modified, just mention its path like below.

```
$ ./ago <path-to-file/directory>
```

Examples:

```
$ ./ago /home/sk/sysadmin-util/  
/home/sk/sysadmin-util/ 23 hours ago
```

```
$ ./ago /etc/  
/etc/ 3 days ago
```



As you can see, **/home/sk/sysadmin-util** is modified **23 hours ago** and **/etc/** directory is modified **3 days ago**.

Related read:

- [How To Monitor File Changes Using fswatch In Linux](#)

SEARCH

KEEP IN TOUCH

- f

FACEBOOK
- in

LINKEDIN
- ✉

EMAIL
- 📡

RSS

## Chronic

Chronic tool runs a command quietly unless it fails i.e. it runs a command and hide STDOUT and STDERR if it completes successfully. It is useful for [cron jobs](#). Instead of trying to keep the command quiet, and having to deal with mails containing accidental output when it succeeds, and not verbose enough output when it fails, you can just run it verbosely always, and use chronic to hide the successful output.

Example:

When creating a new cron job, instead of using the following line;

```
0 1 * * * backup >/dev/null 2>&1
```

you can use:

```
0 1 * * * chronic backup
```

## Cidr2ip

It converts CIDR blocks into their constituent IP addresses.

Example:

```
$ ./cidr2ip 192.168.225.0/24
192.168.225.0
192.168.225.1
192.168.225.2
192.168.225.3
192.168.225.4
192.168.225.5
[...]
```

## Collapse

The collapse tool removes blank lines and lines that contains white-space from the given file(s).

Example:

```
$ echo -e "Linux is not an OS.\n It is a Kernel. \n\nGnu/Linux is the OS.\n\nLinux is not an OS.\nIt is a Kernel.\nGnu/Linux is the OS."
```

## Dupes

The dupes tool will report the files that are identical. It helps you to find duplicate files that contains same contents recursively via SHA1 hash.

Example:

```
$ ./dupes
```

SEARCH

KEEP IN TOUCH

-  FACEBOOK
-  LINKEDIN
-  EMAIL
-  RSS

```
./git/logs/refs/remotes/origin/HEAD
./git/logs/refs/heads/master
```

Related read:

- [How To Find And Delete Duplicate Files In Linux](#)

Empty-dir

This tool will test whether the given directory is empty or not.

Example:

The following example shows whether the given directory **/home/sk/ostechnix** is empty or not.

```
$ if ./empty-dir /home/sk/ostechnix; then echo "It is empty" ; fi
It is empty
```

If the directory is not empty, you will not see any output.

Expand-ipv6

This tool expands the given abbreviated/compressed IPv6 addresses to their full-form. It can be useful when setting up DNS entries.

Example:

```
$ ./expand-ipv6 fe80::a00:27ff:feff:d2e0
fe80:0000:0000:0a00:27ff:feff:d2e0
```

Multi-ping

It is a multi-protocol ping wrapper. It is used to test the connectivity of a remote host, regardless of whether it is an IPv6 or IPv4 host. Meaning - if the remote uses IPv4, it invokes 'ping' command to test the connectivity. If the remote host uses IPv6, it will then invoke 'ping6' command.

The multi-ping utility requires **Net::DNS** perl module. If you haven't install it yet, refer the following guide.

- [How To Install Perl Modules On Linux](#)

Now ping any domain using multi-ping tool as shown below.

Example:

```
$ ./multi-ping google.com
Host google.com - 216.239.36.117 - alive
Host google.com - 216.239.32.117 - alive
Host google.com - 216.239.38.117 - alive
Host google.com - 216.239.34.117 - alive
Host google.com - 2001:4860:4802:34:0:0:0:75 - alive
```

Related read:

SEARCH

KEEP IN TOUCH

 FACEBOOK

 LINKEDIN

 EMAIL

 RSS

- [Ping Multiple Servers And Show The Output In Top-like Text UI](#)
- [Prettytyping – Make The Output Of Ping Command Prettier And Easier To Read](#)

## Pyhttpd

It is simple Python HTTP server that allows you to instantly setup a basic file server.

Examples:

```
$ ./pyhttpd 8080
Serving HTTP on 0.0.0.0 port 8080 ...
```

If you want to allow the server to be bound to localhost-only, rather than listening on all interfaces, run:

```
$ ./pyhttpd 127.0.0.1:8080
Serving HTTP on 127.0.0.1 port 8080 ...
```

### Related read:

- [How to Setup A Basic File server Using simpleHTTPserver](#)

## Randpass

As the name says, the randpass utility is used to generate a random password from command line.

Examples:

```
$ ./randpass
Gb39KTBu
```

To generate a password with specific length, use **-n** flag.

```
$ ./randpass -n 15
Xa7ZMA38tf0t702
```

### Related read:

- [Different Ways To Generate A Strong Password In Linux](#)

## Since

It shows any new content since the last time a file was read. It is useful for tracking log files.

Example:

Remove the contents of apt history.log file:

```
$ ./since /var/log/apt/history.log >/dev/null
```

SEARCH

KEEP IN TOUCH

-  FACEBOOK
-  LINKEDIN
-  EMAIL
-  RSS

```
$ sudo apt remove vim
```

Now check the content of the history.log file:

```
$ ./since /var/log/apt/history.log

Start-Date: 2020-01-28 11:52:59
Commandline: apt remove vim
Requested-By: sk (1000)
Remove: ubuntu-server:amd64 (1.417.3), vim:amd64 (2:8.0.1453-1ubuntu1.1)
End-Date: 2020-01-28 11:53:05
```

The 'since' tool will only display the newly added contents in a file since the last time it was executed. If there are no contents added, it this script exits silently and you will not see any output.

### Ssl-expiry-date

It displays the expiry date of the SSL certificate of the given domain or host.

Examples:

```
$ ./ssl-expiry-date google.com
google.com
Expires: Mar 31 15:47:12 2020 GMT
Days: 63
```

You can also specify multiple domains with space-separated:

```
$ ./ssl-expiry-date google.com ostechnix.com
google.com
Expires: Mar 31 15:47:12 2020 GMT
Days: 63
ostechnix.com
Expires: Oct 9 12:00:00 2020 GMT
Days: 254
```

To display only the number of days remaining on the certificate, use **-d** option:

```
$ ./ssl-expiry-date -d google.com ostechnix.com
google.com: 63
ostechnix.com: 254
```

### Timeout

It allows the user to run a command for a specific interval and kill it.

Example:

```
$ ./timeout -t 10 top
```

SEARCH

KEEP IN TOUCH

-  FACEBOOK
-  LINKEDIN
-  EMAIL
-  RSS



SEARCH

KEEP IN TOUCH

 FACEBOOK

 LINKEDIN

 EMAIL

 RSS

As you can see, the above command kills the 'top' command after 10 seconds.

Kill a command after 10 minutes and 10 seconds:

```
$ ./timeout -t 10:10 top
```

Kill a command after 10 hours, 10 minutes and 10 seconds:

```
$ ./timeout -t 10:10:10 top
```

Please note that **timeout utility is also available by default as part of GNU coreutils** package. So probably won't need to the timeout program from sysadmin-uti package.

Related read:

- [How To Run A Command For A Specific Time In Linux](#)
- [How To Find The Execution Time Of A Command Or Process In Linux](#)
- [How To Run A Linux Command Every X Seconds Forever](#)
- [How To Add Linux Commands To The Queue And Execute Them One By One](#)

There are few more tools available, such as **Until-error** & **Until-success** - repeat commands until it fails/succeeds when down & when up - waits till a host is down/up - must close shell - find whether the



current host is slave or not, **which-shell** - find the shell we're running under, etc.

Refer the project's GitHub page to learn more details of all tools.

- [Sysadmin-util GitHub Repository](#)

## Other useful tools for Linux sysadmins and noobs

Apart from Sysadmin-util, there are many other tools available that might be useful for Linux/Unix sysadmins as well as the newbies. We already have covered some of them.

### 1. Cockpit - Web-based system Monitoring tool

**Cockpit** is free, open source, server administration tool that allows you to easily monitor and administer Linux servers (single or multiple) via a web browser. It helps the system admins to do simple administration tasks, such as starting containers, administrating storage, configuring network, inspecting logs and so on. Check the following link to install and use Cockpit in Linux.

- [Cockpit – Monitor And Administer Linux Servers Via Web Browser](#)

### 2. Netutils-linux - Network Troubleshooting And Performance Tuning Tools For Linux

**Netutils-linux** is a collection of useful utilities written in Python that can be used to simplify Linux network troubleshooting and performance tuning. To know more about these tools, refer the following guide.

- [A Collection Of Utilities To Simplify Linux Network Troubleshooting And Performance Tuning](#)

### 3. Moreutils - more useful Unix utilities

If you're a Linux sysadmin, you will definitely heard about **GNU core utilities**. It comes pre-installed in all Linux distributions. But what about **Moreutils**? The moreutils is a collection of useful Unix utilities which are not included by default in the Unix-like operating systems. These utilities are often helpful for performing various operations on your Linux server easily and quickly. More details can be found in the below link.

- [Moreutils – A Collection Of More Useful Unix Utilities](#)

### 4. Bash-Snippets - Useful BASH Scripts For Heavy Commandline Users

**Bash-Snippets** is a collection of useful BASH scripts for heavy commandline users who live in Terminal all day. Want to check the weather of a place where you live? There is a script to check it quickly. Wondering what is a specific stock price? There is a script for it too. Feel bored? You can watch some YouTube videos. All from commandline!! You don't need to install any heavy, resource-intensive GUI applications. Click the following link to know how to use these scripts.

- [A Collection Of Useful BASH Scripts For Heavy Commandline Users](#)

### 5. Debian-goodies - Command line tools for Debian and derivatives

**Debian-goodies** is yet another collection of useful utilities for DEB-based systems. Like moreutils, Debian-goodies also provides some additional useful commands which are not available by default. Using these tools, the sysadmins can do quite number of tasks, such as;

- which programs are consuming more disk space,
- which services need to be restarted after updating the system,
- search for a file matching a pattern in a package,
- list the installed packages based on the search string and a lot more.

Check the following link to get to know more about these goodies.

- [Debian-goodies – A Set Of Useful Utilities For Debian And Ubuntu Users](#)

### 6. Some good alternatives to man pages

This website uses cookies to improve your experience. By using this site, we will assume that you're OK with it.

Accept

Read More



SEARCH

KEEP IN TOUCH

FACEBOOK

LINKEDIN

EMAIL

RSS

**Man pages are great!** There is no doubt about it. But, many man pages are comprehensive and lack in useful examples. You might have to go through the entire man page to find the detail of a certain option or flag. This is really time-consuming task when you wanted to learn a practical example of a specific Unix command using man pages. Not anymore! I know a few good alternatives to man pages which are focused on mostly examples, skipping all other comprehensive text parts. Read the below to link to learn about them.

- [Good Alternatives To Man Pages Every Linux User Needs To Know](#)

## 7. Cli-fyi - A Command line tool to retrieve information about IP, Email, Domain and more

**Cli.Fyi** is a command line query tool to fetch information about IPs, Emails, Domains, Crypto currencies, media/url, UTC date/time, country and programming language etc. You can fetch all these details either from commandline or browser as described in the following link.

- [Cli.Fyi – A CLI Tool To Fetch Information About IPs, Emails, Domains And Lots More](#)

## 8. ExplainShell - A web-based tool that explains each argument of a Linux command

ExplainShell is a great resource for those who wanted to learn Linux command line arguments and options quickly and easily, without having to refer man pages. It breaks down the given command and explains what each part of a Linux command does. All you need to do is just copy and paste a Linux command in ExplainShell website, and it will instantly display what each part of a Linux command means.

- [ExplainShell – Find What Each Part Of A Linux Command does](#)

## 9. Kmdr - Get CLI commands explanation in your Terminal

**Kmdr** is similar to ExplainShell but with some additional features. The ExplainShell helps you to learn Linux commands only. But what if you want to learn other CLI commands, for example Python? You won't find explanation of Python commands in ExplainShell. This is where Kmdr comes in rescue. Kmdr provides explanation for a lot of CLI commands including **ansible, conda, docker, git, go, kubectI, mongo, mysql, npm, ruby gems, vagrant** and hundreds of other programs such as those built into **bash**. Sounds good? Great! Check the following link to learn to use Kmdr.

- [Kmdr – Display CLI Commands Explanation In Terminal](#)

Like I already said, all of the aforementioned tools may not be useful all the time. Try one at a time and find if helps for your day-to-day job in any way.

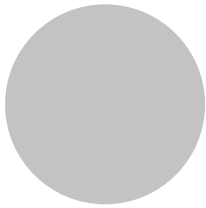
These tools are not just for sysadmins, but for anyone who wants to improve their Linux command line skill-sets in general. I will keep adding more useful tools and applications in this list. If you feel any other tool should be added in this list, feel free to let me know in the comment section below. I will check and update the list accordingly.

Thanks for stopping by!

### Help us to help you:

- Subscribe to our Email Newsletter : [Sign Up Now](#)
- Support OSTechNix : [Donate Via PayPal](#)
- Download free E-Books and Videos : [OSTechNix on TradePub](#)
- Connect with us: [Reddit](#) | [Facebook](#) | [Twitter](#) | [LinkedIn](#) | [RSS feeds](#)

Have a Good day!!



SK

Senthilkumar Palani (aka SK) is the Founder and Editor in chief of OSTechNix. He is a Linux/Unix enthusiast and FOSS supporter. He lives in Tamilnadu, India.



SEARCH

KEEP IN TOUCH

 FACEBOOK

 LINKEDIN

 EMAIL

 RSS

Previous post

**Resume Partially Transferred Files Over SSH Using Rsync**

Next post

**How To Install Perl Modules On Linux**

## YOU MAY ALSO LIKE

**ppl – A Command Line Address book For...**

May 29, 2017

**Reptyr – Move A Running Process From One...**

July 8, 2022

**Setup Headless Virtualization Server Using KVM In Ubuntu...**

July 4, 2018



## LEAVE A COMMENT

Your Comment

Name\*

Email\*

☐ Save my name, email, and website in this browser for the next time I comment.

\* By using this form you agree with the storage and handling of your data by this website.

SUBMIT

This site uses Akismet to reduce spam. [Learn how your comment data is processed.](#)





Select Month ▼

1

How To Fix  
Error On Utk  
August 6, 2020

2

SEARCH

Fix "Sub-pro  
returned ar  
Ubuntu  
June 22, 2020

3

KEEP IN TOUCH

How To Swi  
Multiple PH  
Ubuntu  
FACEBOOK  
August 9, 2018

in LINKEDIN

✉ EMAIL

📡 RSS

OSTechNix (Open Source, Technology, Nix\*) regularly publishes the latest news, how-to articles, tutorials and tips & tricks about free and opensource software and technology.



AboutContact UsPrivacy PolicySitemapTerms and Conditions

OSTechNix © 2022. All Rights Reserved. This site is licensed under [CC BY-NC 4.0](#).

