DigitalOcean   ☰

⌄   ☐⁺ Subscribe   ⬆ Share   ☰ Contents ⌄

✔ How To Install Nginx on Ubuntu 16.04

Posted April 21, 2016   👁 824.2k   NGINX   UBUNTU   UBUNTU 16.04

⌃
♡
169

By: Justin Ellingwood

Not using **Ubuntu 16.04**? Choose a different version:

Introduction

Nginx is one of the most popular web servers in the world and is responsible for hosting some of the largest and highest-traffic sites on the internet. It is more resource-friendly than Apache in most cases and can be used as a web server or a reverse proxy.

SCROLL TO TOP

In this guide, we'll discuss how to get Nginx installed on your Ubuntu 16.04 server.

Prerequisites

eges

ollowing our
initial server setup guide for Ubuntu 16.04.

When you have an account available, log in as your non-root user to begin.

Step 1: Install Nginx

Nginx is available in Ubuntu's default repositories, so the installation is rather straight forward.

Since this is our first interaction with the `apt` packaging system in this session, we will update our local package index so that we have access to the most recent package listings. Afterwards, we can install `nginx`:

```
$ sudo apt-get update
$ sudo apt-get install nginx
```

After accepting the procedure, `apt-get` will install Nginx and any required dependencies to your server.

Step 2: Adjust the Firewall

Before we can test Nginx, we need to reconfigure our firewall software to allow access to the service. Nginx registers itself as a service with `ufw`, our firewall, upon installation. This makes it rather easy to allow Nginx access.

We can list the applications configurations that `ufw` knows how to work with by typing:

```
$ sudo ufw app list
```

You should get a listing of the application profiles:

```
Output
```

```
Available applications:
  Nginx Full
  Nginx HTTP
  Nginx HTTPS
```

As you can see, there are three profiles available for Nginx:

- **Nginx Full**: This profile opens both port 80 (normal, unencrypted web traffic) and port 443 (TLS/SSL encrypted traffic)

- **Nginx HTTP**: This profile opens only port 80 (normal, unencrypted web traffic)

- **Nginx HTTPS**: This profile opens only port 443 (TLS/SSL encrypted traffic)

It is recommended that you enable the most restrictive profile that will still allow the traffic you've configured. Since we haven't configured SSL for our server yet, in this guide, we will only need to allow traffic on port 80.

You can enable this by typing:

```
$ sudo ufw allow 'Nginx HTTP'
```

You can verify the change by typing:

```
$ sudo ufw status
```

You should see HTTP traffic allowed in the displayed output:

```
Output
Status: active

To                         Action      From
--                         ------      ----
OpenSSH                    ALLOW       Anywhere
Nginx HTTP                 ALLOW       Anywhere
OpenSSH (v6)               ALLOW       Anywhere (v6)
Nginx HTTP (v6)            ALLOW       Anywhere (v6)
```

Step 3: Check your Web Server

At the end of the installation process, Ubuntu 16.04 starts Nginx. The web server should already be up and running

ing:

```
$ systemctl status nginx
```

Output

```
● nginx.service - A high performance web server and a reverse proxy server
   Loaded: loaded (/lib/systemd/system/nginx.service; enabled; vendor preset: enabled)
   Active: active (running) since Mon 2016-04-18 16:14:00 EDT; 4min 2s ago
 Main PID: 12857 (nginx)
   CGroup: /system.slice/nginx.service
           ├─12857 nginx: master process /usr/sbin/nginx -g daemon on; master_process on
           └─12858 nginx: worker process
```

As you can see above, the service appears to have started successfully. However, the best way to test this is to actually request a page from Nginx.

You can access the default Nginx landing page to confirm that the software is running properly. You can access this through your server's domain name or IP address.

If you do not have a domain name set up for your server, you can learn how to set up a domain with DigitalOcean here.

If you do not want to set up a domain name for your server, you can use your server's public IP address. If you do not know your server's IP address, you can get it a few different ways from the command line.

Try typing this at your server's command prompt:

```
$ ip addr show eth0 | grep inet | awk '{ print $2; }' | sed 's/\/.*$//'
```

You will get back a few lines. You can try each in your web browser to see if they work.

An alternative is typing this, which should give you your public IP address as seen from another location on the internet:

```
$ sudo apt-get install curl
$ curl -4 icanhazip.com
```

http://server_domain_or_IP

You should see the default Nginx landing page, which should look something like this:

This page is simply included with Nginx to show you that the server is running correctly.

Step 4: Manage the Nginx Process

Now that you have your web server up and running, we can go over some basic management commands.

To stop your web server, you can type:

```
$ sudo systemctl stop nginx
```

To start the web server when it is stopped, type:

```
$ sudo systemctl start nginx
```

To stop and then start the service again, type:

```
$ sudo systemctl restart nginx
```

If you are simply making configuration changes, Nginx can often reload without dropping connections. To do this, this command can be used:

By default, Nginx is configured to start automatically when the server boots. If this is not what you want, you can disable this behavior by typing:

```
$ sudo systemctl disable nginx
```

To re-enable the service to start up at boot, you can type:

```
$ sudo systemctl enable nginx
```

Step 5: Get Familiar with Important Nginx Files and Directories

Now that you know how to manage the service itself, you should take a few minutes to familiarize yourself with a few important directories and files.

Content

- `/var/www/html` : The actual web content, which by default only consists of the default Nginx page you saw earlier, is served out of the `/var/www/html` directory. This can be changed by altering Nginx configuration files.

Server Configuration

- `/etc/nginx` : The Nginx configuration directory. All of the Nginx configuration files reside here.

- `/etc/nginx/nginx.conf` : The main Nginx configuration file. This can be modified to make changes to the Nginx global configuration.

- `/etc/nginx/sites-available/` : The directory where per-site "server blocks" can be stored. Nginx will not use the configuration files found in this directory unless they are linked to the `sites-enabled` directory (see below). Typically, all server block configuration is done in this directory, and then enabled by linking to the other directory.

- `/etc/nginx/sites-enabled/` : The directory where enabled per-site "server blocks" are stored. Typically, these are created by linking to configuration files found in the `sites-available` directory.

Server Logs

- `/var/log/nginx/access.log` : Every request to your web server is recorded in this log file unless Nginx is configured to do otherwise.

- `/var/log/nginx/error.log` : Any Nginx errors will be recorded in this log.

Conclusion

Now that you have your web server installed, you have many options for the type of content to serve and the technologies you want to use to create a richer experience.

Learn how to use Nginx server blocks here. If you'd like to build out a more complete application stack, check out this article on how to configure a LEMP stack on Ubuntu 16.04.

By: Justin Ellingwood　　　　　　　　　♡ Upvote (169)　　　⌐ Subscribe　　　⬆ Share

Introducing Projects on DigitalOcean

Sign up for our newsletter. Get the latest tutorials on SysAdmin and open source topics.   ✕

Enter your email address   Sign Up

READ MORE

Related Tutorials

How To Upgrade Nginx In-Place Without Dropping Client Connections

How To Target Your Users with Nginx Analytics and A/B Testing

How to Retrieve Let's Encrypt SSL Wildcard Certificates using CloudFlare Validation on CentOS 7

How To Create a Kubernetes 1.11 Cluster Using Kubeadm on Ubuntu 18.04

How To Develop a Node.js TCP Server Application using PM2 and Nginx on Ubuntu 16.04

34 Comments

Leave a comment...

Log In to Comment

iamkingsleyf *April 25, 2016*

0   Thanks

Sign up for our newsletter. Get the latest tutorials on SysAdmin and open source topics.   ✕

Enter your email address   Sign Up

0   Thank you , it was very useful

---

chiragnayyar *April 27, 2016*

0   Thank you for this information

---

arnsdorf *May 23, 2016*

1   Thanks for the tutorial! I followed all the tutorials for setting up a new Ubuntu 16.04 droplet, securing it
and installing the LEMP stack. Everything is up and running however I have a practical question.

I am using Dreamweaver to manage files the server. I can connect to the server in Dreamweaver and
see the files on the server. However, I am unable to upload to the server. I know the problem is due to
permission issues but I don't know the best way to resolve the problem.

Here are some points that may be important:

1. The /var/www/html directory is owned by **root:root**

2. Nginx uses the default user **www-data**

3. My user is configured with sudo access

4. As instructed in one of the tutorials, users can't login to the server with a password so it is
    configured to use an RSA key

Thank you in advance.

---

ihor.vorotnov *June 17, 2016*

0   Try your_user:www-data instead of root:root for /var/www/html recursively

---

bsy *June 23, 2016*

2   Just upgraded to 16.04 from 14.04 following this
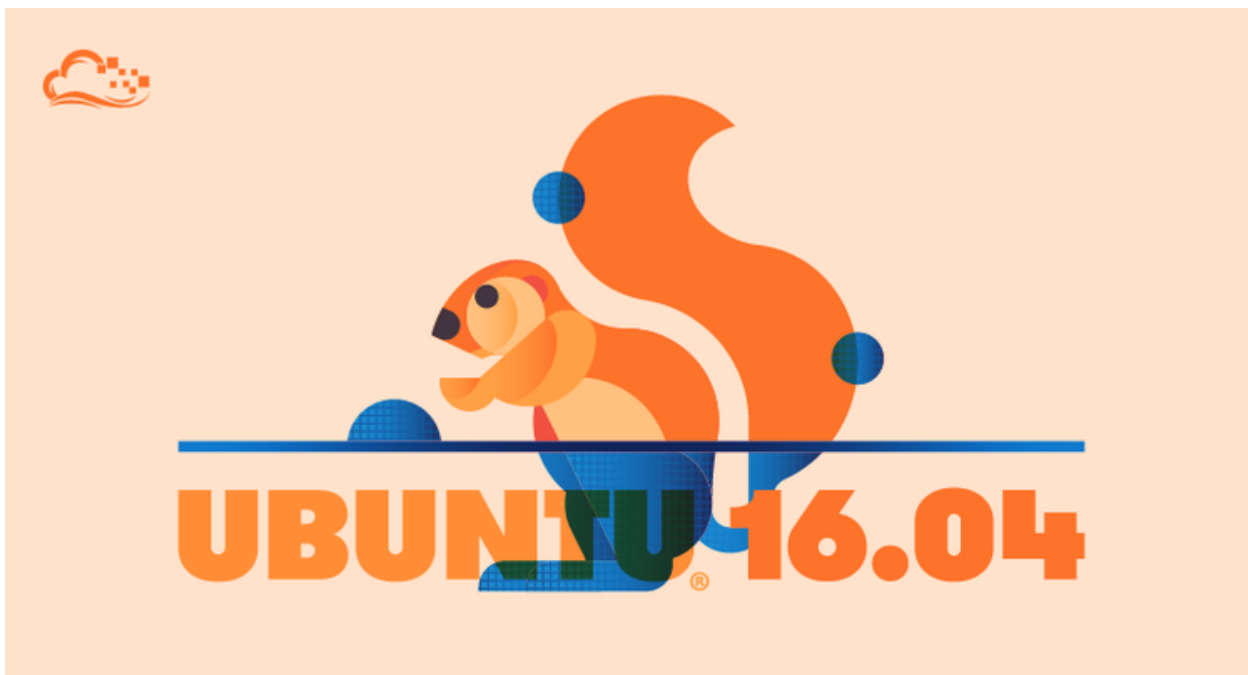https://www.digitalocean.com/community/tutorials/how-to-upgrade-to-ubuntu-16-04-lts

Had issues with installing nginx so I removed nginx and reinstalled, but now when I run status i'm getting an error:

$ sudo systemctl status nginx

How To Upgrade to Ubuntu 16.04 LTS

by Brennen Bearnes

Ubuntu's next Long Term Support release, version 16.04 (Xenial Xerus), is due to be released on April 21, 2016. This guide explains the upgrade process for systems including (but not limited to) DigitalOcean Droplets running Ubuntu 15.10.

WSoap *January 5, 2017*

1 Did you ever find a solution to this? I'm experiencing the same issue.

---

reveltas *November 20, 2016*

0 Hi Justin, please update the server blocks link at the bottom of the article, it's linking to article using Ubuntu v14, but there's another, Ubuntu v16 article.

---

jellingwood MOD *November 21, 2016*

1 @reveltas Good catch. That one wasn't available at the time this was initially written. Should be good to go now. Thanks!

Sign up for our newsletter. Get the latest tutorials on SysAdmin and open source topics.   ✕

| Enter your email address | Sign Up |

---

acenayeem *May 20, 2017*

0 i think you are right

---

jgwitkowski *October 8, 2017*

2 Mind updating documentation to account for above? It took me some time to discover this. Thanks!! @jellingwood

---

jellingwood MOD *October 9, 2017*

1 @jgwitkowski The initial server setup guide for Ubuntu 16.04 listed as a prerequisite enables the UFW firewall. This guide assumes that you've completed the steps in that guide before starting. Hope that helps!

Initial Server Setup with Ubuntu 16.04

by Mitchell Anicas

When you start a new server, there are a few steps that you should take every time to add some basic security and give you a solid foundation. In this guide, we'll walk you through the basic steps necessary to hit the ground running with Ubuntu 16.04.

heretics99 *December 10, 2017*

~~w is already~~
~~ntu 16.04'. If~~
~~ge~~ `Status:`
`inactive`".

I spent a considerable time wondering why does it still show ufw status as inactive when I was following your guide to the letter. I request you to please update your tutorial.

jonblatho *December 20, 2017*

0 Agreed. In technical writing you never really get to "assume" the user did anything. Indeed, the tutorial never explicitly states that it assumes those exact steps have been followed—only that a non-root sudoer is required. If a user already has a non-root sudoer on the server, why would they follow that link to do setup work that they already did?

As others have suggested, I'd recommend just adding that very simple step to the tutorial, or at least highlighting that the tutorial assumes the exact steps in the linked article have been followed—not suggesting that it only needs to be followed if a non-root sudoer doesn't yet exist.

JefferyZ *January 6, 2018*

0 You are right!

dennisbauszus *January 6, 2017*

2 Following this tutorial I have NGINX running and be able to restart the service from my non root user using

```
sudo service nginx restart
```

However there seems to be a race between systemd and nginx as the service status informs me that it failed to read PID from file /run/nginx.pid: Invalid argument. As if systemd was expecting the PID file to be populated before nginx had the time to create it. As a work around I create this file:

/etc/systemd/system/nginx.service.d/override.conf

With following content:

```
[Service]
ExecStartPost=/bin/sleep 0.1
```

Thereafter I need to reload the daemon.

```
systemctl daemon-reload
```

This resolves the problem. The service will now wait a moment which is enough for the PID file to be populated.

[aleemfirnas22](#) *July 27, 2017*

0

I have Apache web server already installed in droplet which is used to serve html/php based websites, but for now I need to deploy a NodeJs app, so that I think I need to go with nGinx ? How can it be done ? can both be installed at one droplet, how ?

Thanks

[profnandaa](#) *August 15, 2017*

0

Thanks for the nice tut. Just one more step also forgotten, above what was mentioned in the comment here.

```
sudo ufw allow 'OpenSSH'
```

Without this, you won't be able to access your shell again through SSH, I had to rebuild by droplet twice before realizing this :(

Make sure `sudo ufw status` gives exactly the output as put in the tutorial:
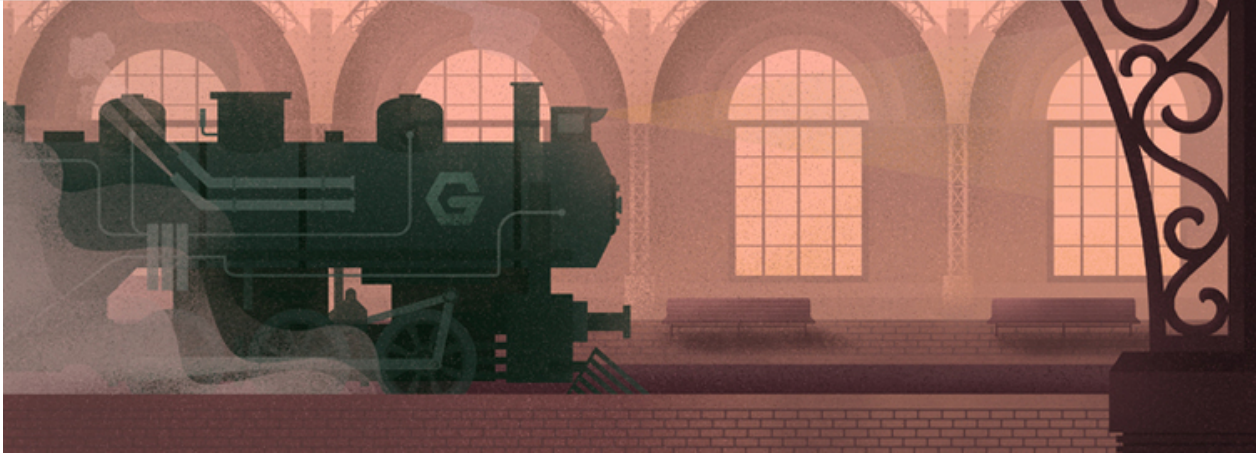
```
Status: active

To                         Action      From
--                         ------      ----
OpenSSH                    ALLOW       Anywhere
Nginx HTTP                 ALLOW       Anywhere
OpenSSH (v6)               ALLOW       Anywhere (v6)
Nginx HTTP (v6)            ALLOW       Anywhere (v6)
```

Nginx HTTP (v6)   ALLOW   Anywhere (v6)

Sign up for our newsletter. Get the latest tutorials on SysAdmin and open source topics. ✕

Enter your email address   Sign Up

How To Install Nginx on Ubuntu 16.04

by Justin Ellingwood

Nginx is one of the most popular web servers in the world and is responsible for hosting some of the largest and highest-traffic sites on the internet. It is more resource-friendly than Apache in most cases and can be used as a web server or a reverse proxy. In this guide,…

---

jellingwood MOD *August 15, 2017*

1 **@profnandaa** Hey there. Sorry to hear that you had trouble.

The command to allow OpenSSH through your firewall is included in the initial server setup guide for Ubuntu 16.04 which is a prerequisite for this guide. This article assumes that you are beginning the instructions with a server in the state that the initial set up guide leaves you in. Hope that helps for the future.

Initial Server Setup with Ubuntu 16.04

by Mitchell Anicas

When you start a new server, there are a few steps that you should take every time to add some basic security and give you a solid foundation. In this guide, we'll walk you through the basic steps necessary to hit the ground running with Ubuntu 16.04.

---

profnandaa *August 15, 2017*

0   Thanks for the callout, I'd missed that part. My bad. @jellingwood

---

francescobertel *January 12, 2018*

0   I can't believe that I just bought a droplet here, installed apache etc... trying to install NGINX and I get errors. ...why???

dpkg: error processing package nginx (--configure):
dependency problems - leaving unconfigured
No apport report written because the error message indicates its a followup error from a previous failure.

---

francescobertel *January 12, 2018*

1   you may wann say you have to stop any other serivce first no???

do

```
sudo service apache2 stop
```

before trying this procedure.

aseradyn *March 21, 2018*

0 Thank you!

yatharthg *June 5, 2018*

0 i did this and after this and installed nginx using above procedure and now my website is down. i'm not familiar with linux and i don't know what to do now.

macgyvermartins *August 20, 2017*

0 Awesome article!

abenedetti *September 2, 2017*

0 Thanks! I was missing the step to allow nginx in ufw! Solved a big problem! :)

prasanthv *November 26, 2017*

0 For folks who are setting up certs via letsencrypts certbot, you will need to open up that tls port. So if you're doing it right after, might as well select Nginx Full. Great article!

JefferyZ *January 5, 2018*

0 This is a good start! Thank you

abhisheksankhe *January 21, 2018*

0 Please help, I cannot install Nginx. My installation is One Click Wordpress. Nginx is needed for Let's Encrypt SSL.

When I do :

```
sudo apt-get install nginx
```

```
Reading package lists... Done
Building dependency tree
Reading state information... Done
nginx is already the newest version (1.10.3-0ubuntu0.16.04.2).
The following packages were automatically installed and are no longer required:
  augeas-lenses letsencrypt libaugeas0 python-augeas
Use 'sudo apt autoremove' to remove them.
0 upgraded, 0 newly installed, 0 to remove and 48 not upgraded.
3 not fully installed or removed.
After this operation, 0 B of additional disk space will be used.
Do you want to continue? [Y/n] Y
Setting up nginx-core (1.10.3-0ubuntu0.16.04.2) ...
Job for nginx.service failed because the control process exited with error code. See
invoke-rc.d: initscript nginx, action "start" failed.
● nginx.service - A high performance web server and a reverse proxy server
   Loaded: loaded (/lib/systemd/system/nginx.service; enabled; vendor preset: enabled
   Active: failed (Result: exit-code) since Sun 2018-01-21 20:09:15 UTC; 8ms ago
  Process: 17681 ExecStart=/usr/sbin/nginx -g daemon on; master_process on; (code=exi
  Process: 17676 ExecStartPre=/usr/sbin/nginx -t -q -g daemon on; master_process on;

Jan 21 20:09:13 mblive-wordpress-s-1vcpu-1gb-blr1-01 nginx[17681]: nginx: [emerg] lis
Jan 21 20:09:14 mblive-wordpress-s-1vcpu-1gb-blr1-01 nginx[17681]: nginx: [emerg] lis
Jan 21 20:09:14 mblive-wordpress-s-1vcpu-1gb-blr1-01 nginx[17681]: nginx: [emerg] lis
Jan 21 20:09:14 mblive-wordpress-s-1vcpu-1gb-blr1-01 nginx[17681]: nginx: [emerg] lis
Jan 21 20:09:14 mblive-wordpress-s-1vcpu-1gb-blr1-01 nginx[17681]: nginx: [emerg] lis
Jan 21 20:09:15 mblive-wordpress-s-1vcpu-1gb-blr1-01 nginx[17681]: nginx: [emerg] sti
Jan 21 20:09:15 mblive-wordpress-s-1vcpu-1gb-blr1-01 systemd[1]: nginx.service: Contr
Jan 21 20:09:15 mblive-wordpress-s-1vcpu-1gb-blr1-01 systemd[1]: Failed to start A hi
Jan 21 20:09:15 mblive-wordpress-s-1vcpu-1gb-blr1-01 systemd[1]: nginx.service: Unit
Jan 21 20:09:15 mblive-wordpress-s-1vcpu-1gb-blr1-01 systemd[1]: nginx.service: Faile
Hint: Some lines were ellipsized, use -l to show in full.
dpkg: error processing package nginx-core (--configure):
 subprocess installed post-installation script returned error exit status 1
dpkg: dependency problems prevent configuration of nginx:
 nginx depends on nginx-core (>= 1.10.3-0ubuntu0.16.04.2) | nginx-full (>= 1.10.3-0ub
   Package nginx-core is not configured yet.
```

```
  Package nginx-full is not installed.
  Package nginx-light is not installed.
  Package nginx-extras is not installed.
  nginx depends on nginx-core (<< 1.10.3-0ubuntu0.16.04.2.1~) | nginx-full (<< 1.10.3-
```

Sign up for our newsletter. Get the latest tutorials on SysAdmin and open source topics.   ✕

| Enter your email address | Sign Up |

```
  Package nginx-extras is not installed.

 dpkg: error processing package nginx (--configure):
  dependency problems - leaving unconfigured
 dpkg: dependency problems prevent configuration of python-certbot-nginx:
  python-certbot-nginx depends on nginx; however:
   Package nginx is not configured yet.
   Package nginNo apport report written because the error message indicates its a foll
                  No apport report written because the error message indicates its
                       x-core which provides nginx is not configured yet.

 dpkg: error processing package python-certbot-nginx (--configure):
  dependency problems - leaving unconfigured
 Errors were encountered while processing:
  nginx-core
  nginx
  python-certbot-nginx
 E: Sub-process /usr/bin/dpkg returned an error code (1)
```

⌃
♡
0   **felixbade** *January 24, 2018*

I got an error here:
$ sudo ufw allow 'Nginx HTTP'
ERROR: Need 'to' or 'from' clause

This fixed it:

https://www.digitalocean.com/community/questions/firewall-apache-error

---

⌃
♡
0   **trentbaur** *February 19, 2018*

I would really appreciate some guidance on how to reliably force nginx to take up a changed static file. This seems to be a common issue based on this ServerFault post:

https://serverfault.com/questions/549200/how-do-i-force-nginx-to-load-new-static-files

I have followed all of the suggestion without success.

I have restarted nginx, restarted my droplet, I have DELETED THE ACTUAL FILE and it is still being

andrifmsyh *March 5, 2018*

0 i follow the article carefully twice, but i never get the nginx 'welcome ...' when i try include my ip in the browser :(please help!

yatharthg *June 5, 2018*

0 After installing Nginx, how can i move my wordpress website to Nginx from Apache? Can someone please help

Copyright © 2018 DigitalOcean™ Inc.

Community   Tutorials   Questions   Projects   Tags   Newsletter   RSS 🔊

Distros & One-Click Apps   Terms, Privacy, & Copyright   Security   Report a Bug   Write for DOnations   Shop

Sign up for our newsletter. Get the latest tutorials on SysAdmin and open source topics. ✕

Enter your email address
Sign Up