How to Install Nginx on Ubuntu 22.04 LTS

FOSS TechNix

In this article we are going to cover How to Install NGINX on Ubuntu 22.04 LTS,Adjust the Firewall in Ubuntu 22.04 LTS,Verify Nginx Installation on Ubuntu 22.04 LTS,Managing Nginx processes using systemctl,Configure Nginx Server-Blocks onUbuntu 22.04 LTS,Nginx Configuration files and Directories.

Table of Contents

What is Nginx ?

Nginx is a free, open-source, high-performance HTTP server and reverse proxy, as well as an IMAP/POP3 proxy server. Nginx is known for its stability, rich feature set, simple configuration, and low resource consumption.
Nginx is a web server that is used to serve web pages. It is a popular choice for web servers because it is fast and reliable. Nginx can be used to host websites and applications. It can also be used as a reverse proxy server.

Benefits of using Nginx

Nginx is a high-performance web server with a small footprint. It is efficient and can handle a large number of concurrent connections with ease. Nginx is also highly scalable and can be used to load balance across multiple servers.

Using Nginx can result in significant performance gains for your website or application. Nginx is able to handle more concurrent connections than other web servers, meaning that it can serve more users at the same time without slowing down. Nginx is also much more efficient than other web servers, using less memory and CPU resources.

Additionally, Nginx is highly scalable and can be used to load balance across multiple servers. This means that if you have a large website or application with heavy traffic, Nginx can distribute the load across multiple servers to ensure that your site remains responsive.

Why might someone choose to use Nginx over another web server software ?

Main reasons:

Nginx is known for its performance and efficiency. It can handle a large number of concurrent connections and has a small memory footprint, which makes it a good choice for high-traffic websites.

Nginx is also highly scalable. It can be easily configured to handle a large amount of traffic, and it can be easily scaled up or down as needed.

Nginx is also very versatile. It can be used as a reverse proxy, load balancer, and HTTP cache, which makes it a good choice for a wide range of applications.

Nginx is also easy to configure and maintain. Its configuration files are simple and straightforward, which makes it easy to manage even for users who are not experienced with web server software.

Overall, Nginx is a popular choice for web server software because of its performance, efficiency, scalability, versatility, and ease of use.

Prerequisites:-

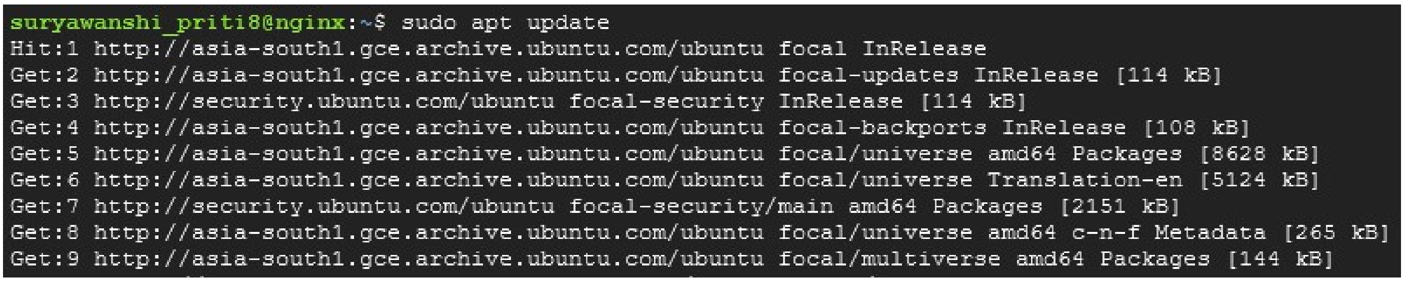
You should have create VM (ubuntu).

Have a user account with sudo privileges

Have a domain name pointing to your public server IP.

First update the system packages on Ubuntu 22.04 using below command

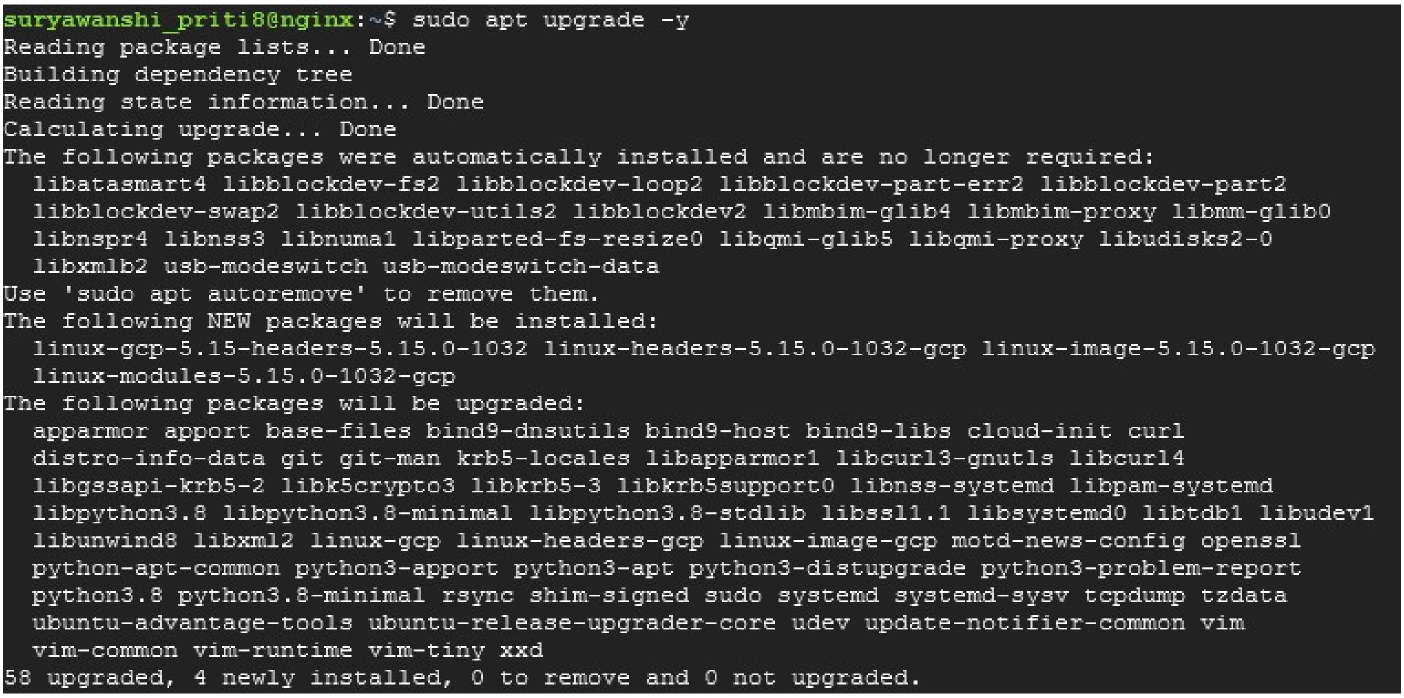
sudo apt update



```
suryawanshi_priti8@nginx:~$ sudo apt update
Hit:1 http://asia-south1.gce.archive.ubuntu.com/ubuntu focal InRelease
Get:2 http://asia-south1.gce.archive.ubuntu.com/ubuntu focal-updates InRelease [114 kB]
Get:3 http://security.ubuntu.com/ubuntu focal-security InRelease [114 kB]
Get:4 http://asia-south1.gce.archive.ubuntu.com/ubuntu focal-backports InRelease [108 kB]
Get:5 http://asia-south1.gce.archive.ubuntu.com/ubuntu focal/universe amd64 Packages [8628 kB]
Get:6 http://asia-south1.gce.archive.ubuntu.com/ubuntu focal/universe Translation-en [5124 kB]
Get:7 http://security.ubuntu.com/ubuntu focal-security/main amd64 Packages [2151 kB]
Get:8 http://asia-south1.gce.archive.ubuntu.com/ubuntu focal/universe amd64 c-n-f Metadata [265 kB]
Get:9 http://asia-south1.gce.archive.ubuntu.com/ubuntu focal/multiverse amd64 Packages [144 kB]
```

if you want to upgrade your system packages you can use below command however it is optional

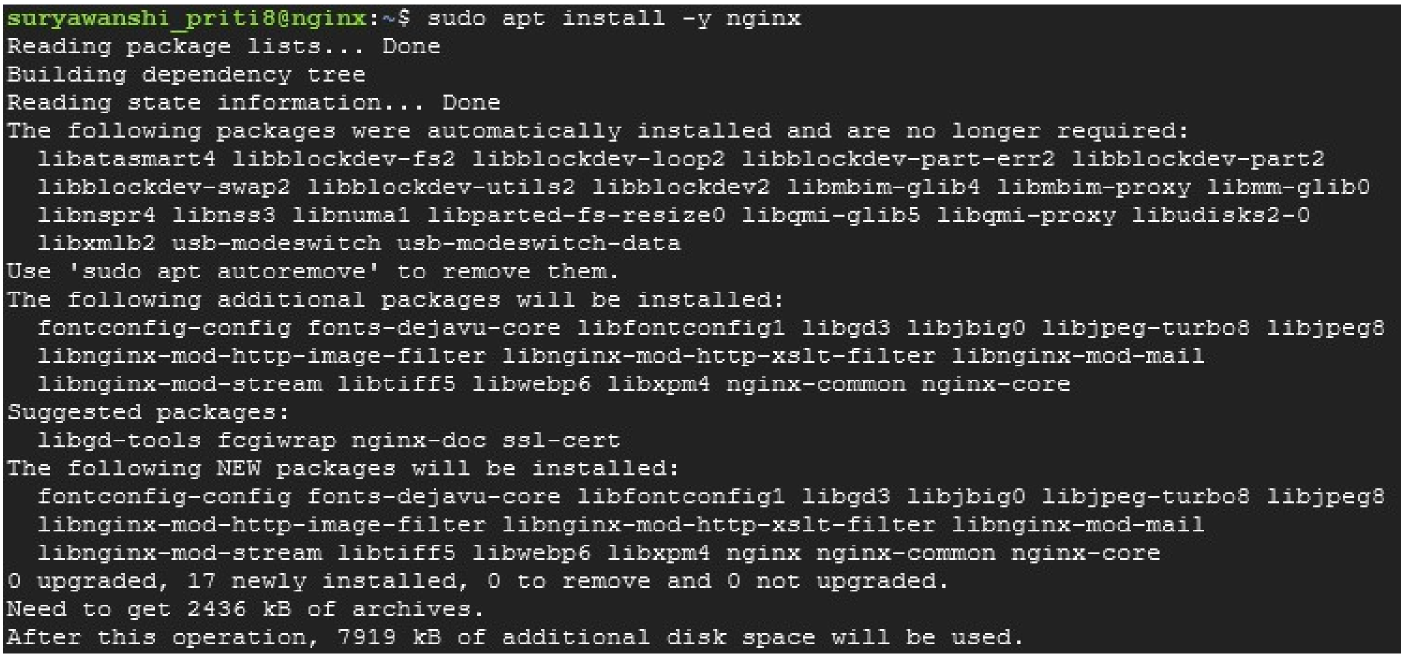
sudo apt –y upgrade



```
suryawanshi_priti8@nginx:~$ sudo apt upgrade -y
Reading package lists... Done
Building dependency tree
Reading state information... Done
Calculating upgrade... Done
The following packages were automatically installed and are no longer required:
  libatasmart4 libblockdev-fs2 libblockdev-loop2 libblockdev-part-err2 libblockdev-part2
  libblockdev-swap2 libblockdev-utils2 libblockdev2 libmbim-glib4 libmbim-proxy libmm-glib0
  libnspr4 libnss3 libnuma1 libparted-fs-resize0 libqmi-glib5 libqmi-proxy libudisks2-0
  libxmlb2 usb-modeswitch usb-modeswitch-data
Use 'sudo apt autoremove' to remove them.
The following NEW packages will be installed:
  linux-gcp-5.15-headers-5.15.0-1032 linux-headers-5.15.0-1032-gcp linux-image-5.15.0-1032-gcp
  linux-modules-5.15.0-1032-gcp
The following packages will be upgraded:
  apparmor apport base-files bind9-dnsutils bind9-host bind9-libs cloud-init curl
  distro-info-data git git-man krb5-locales libapparmor1 libcurl3-gnutls libcurl4
  libgssapi-krb5-2 libk5crypto3 libkrb5-3 libkrb5support0 libnss-systemd libpam-systemd
  libpython3.8 libpython3.8-minimal libpython3.8-stdlib libssl1.1 libsystemd0 libtdb1 libudev1
  libunwind8 libxml2 linux-gcp linux-headers-gcp linux-image-gcp motd-news-config openssl
  python-apt-common python3-apport python3-apt python3-distupgrade python3-problem-report
  python3.8 python3.8-minimal rsync shim-signed sudo systemd systemd-sysv tcpdump tzdata
  ubuntu-advantage-tools ubuntu-release-upgrader-core udev update-notifier-common vim
  vim-common vim-runtime vim-tiny xxd
58 upgraded, 4 newly installed, 0 to remove and 0 not upgraded.
```

Install NGINX on Ubuntu 22.04 LTS using apt

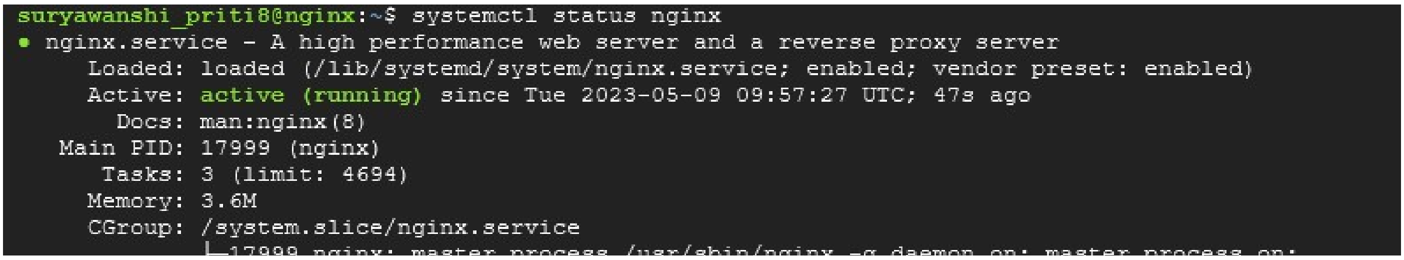
sudo apt install –y nginx



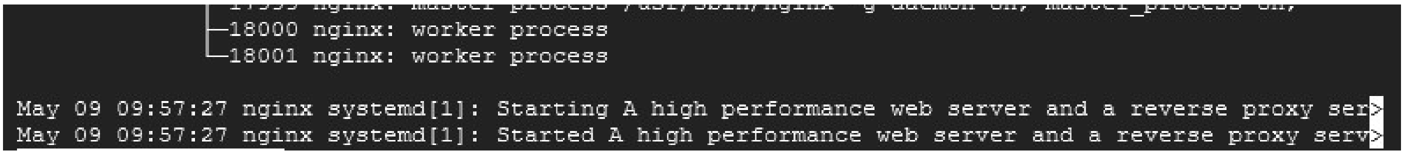
```
suryawanshi_priti8@nginx:~$ sudo apt install -y nginx
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following packages were automatically installed and are no longer required:
  libatasmart4 libblockdev-fs2 libblockdev-loop2 libblockdev-part-err2 libblockdev-part2
  libblockdev-swap2 libblockdev-utils2 libblockdev2 libmbim-glib4 libmbim-proxy libmm-glib0
  libnspr4 libnss3 libnuma1 libparted-fs-resize0 libqmi-glib5 libqmi-proxy libudisks2-0
  libxmlb2 usb-modeswitch usb-modeswitch-data
Use 'sudo apt autoremove' to remove them.
The following additional packages will be installed:
  fontconfig-config fonts-dejavu-core libfontconfig1 libgd3 libjbig0 libjpeg-turbo8 libjpeg8
  libnginx-mod-http-image-filter libnginx-mod-http-xslt-filter libnginx-mod-mail
  libnginx-mod-stream libtiff5 libwebp6 libxpm4 nginx-common nginx-core
Suggested packages:
  libgd-tools fcgiwrap nginx-doc ssl-cert
The following NEW packages will be installed:
  fontconfig-config fonts-dejavu-core libfontconfig1 libgd3 libjbig0 libjpeg-turbo8 libjpeg8
  libnginx-mod-http-image-filter libnginx-mod-http-xslt-filter libnginx-mod-mail
  libnginx-mod-stream libtiff5 libwebp6 libxpm4 nginx nginx-common nginx-core
0 upgraded, 17 newly installed, 0 to remove and 0 not upgraded.
Need to get 2436 kB of archives.
After this operation, 7919 kB of additional disk space will be used.
```

use below command to check Nginx service status using systemctl

systemctl status nginx



```
suryawanshi_priti8@nginx:~$ systemctl status nginx
● 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 Tue 2023-05-09 09:57:27 UTC; 47s ago
       Docs: man:nginx(8)
   Main PID: 17999 (nginx)
      Tasks: 3 (limit: 4694)
     Memory: 3.6M
     CGroup: /system.slice/nginx.service
             ├─17999 nginx: master process /usr/sbin/nginx -g daemon on; master_process on;
```



```
             ├─17999 nginx: master process /usr/sbin/nginx -g daemon on; master_process on;
             ├─18000 nginx: worker process
             └─18001 nginx: worker process

May 09 09:57:27 nginx systemd[1]: Starting A high performance web server and a reverse proxy ser>
May 09 09:57:27 nginx systemd[1]: Started A high performance web server and a reverse proxy serv>
```

We have covered Install Nginx on Ubuntu 22.04 LTS.

Step #2:Adjust the Firewall in Ubuntu 22.04 LTS

If you are installing Nginx on you local system then allow traffic on port 80 in ufw , if you are installing on Cloud Instance then allow port in cloud security group.

sudo ufw allow 80/tcp

OR

```
sudo ufw app list
```

you will see list of profiles

```
OutputAvailable applications:
  Nginx Full
  Nginx HTTP
  Nginx HTTPS
  OpenSSH
```

To allow traffic on port 80, enable this by typing:

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

You can verify the change by typing:

```
sudo ufw status
```

The output will indicated which HTTP traffic is allowed:

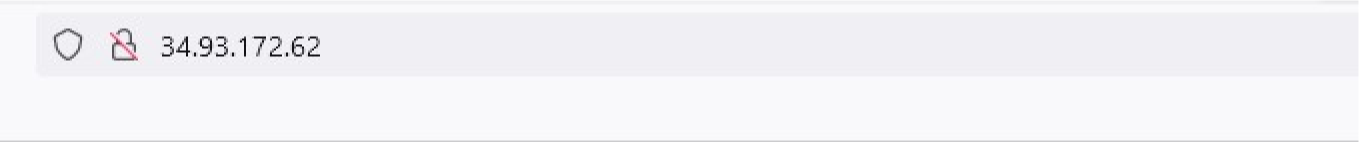
```
Output
Status: active
```

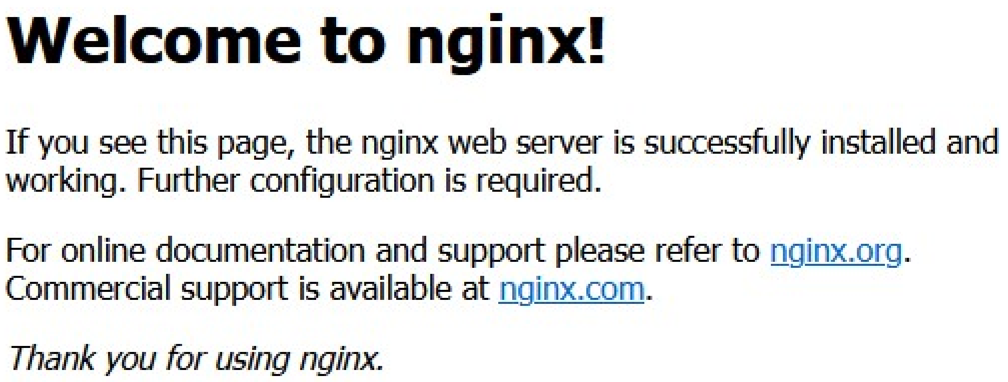
Step #3:Verify Nginx Installation on Ubuntu 22.04 LTS

To Verify Nginx was installed correctly, open a web browser and type in the address bar **http://server_ip_address**

After opening this URL you can see the Nginx default page as in the image below:

VM IP / localhost





Welcome to nginx!

If you see this page, the nginx web server is successfully installed and working. Further configuration is required.

For online documentation and support please refer to nginx.org.
Commercial support is available at nginx.com.

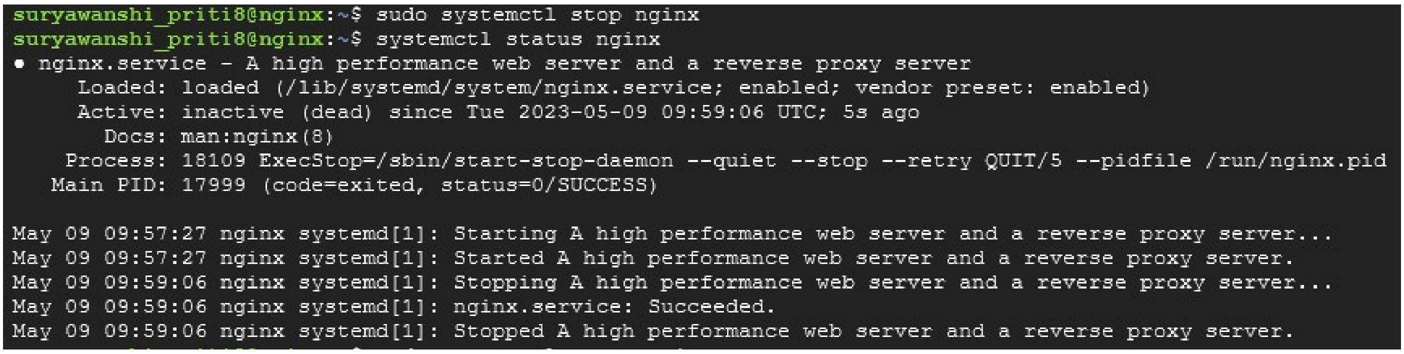
Thank you for using nginx.

Step #4:Managing Nginx processes using systemctl

Below are systemctl commands to manage Nginx web server on Ubuntu 22.04 LTS

Use below command to stop Nginx process using systemctl

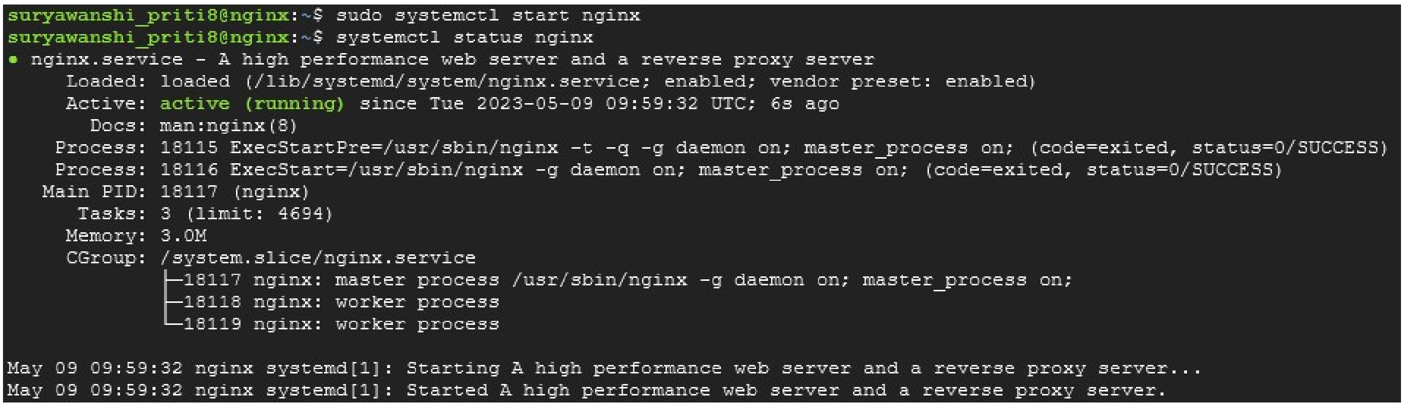
sudo systemctl stop nginx



```
suryawanshi_priti8@nginx:~$ sudo systemctl stop nginx
suryawanshi_priti8@nginx:~$ systemctl status nginx
● nginx.service - A high performance web server and a reverse proxy server
     Loaded: loaded (/lib/systemd/system/nginx.service; enabled; vendor preset: enabled)
     Active: inactive (dead) since Tue 2023-05-09 09:59:06 UTC; 5s ago
       Docs: man:nginx(8)
    Process: 18109 ExecStop=/sbin/start-stop-daemon --quiet --stop --retry QUIT/5 --pidfile /run/nginx.pid
   Main PID: 17999 (code=exited, status=0/SUCCESS)

May 09 09:57:27 nginx systemd[1]: Starting A high performance web server and a reverse proxy server...
May 09 09:57:27 nginx systemd[1]: Started A high performance web server and a reverse proxy server.
May 09 09:59:06 nginx systemd[1]: Stopping A high performance web server and a reverse proxy server...
May 09 09:59:06 nginx systemd[1]: nginx.service: Succeeded.
May 09 09:59:06 nginx systemd[1]: Stopped A high performance web server and a reverse proxy server.
```

.To start the Nginx web server when it is stopped

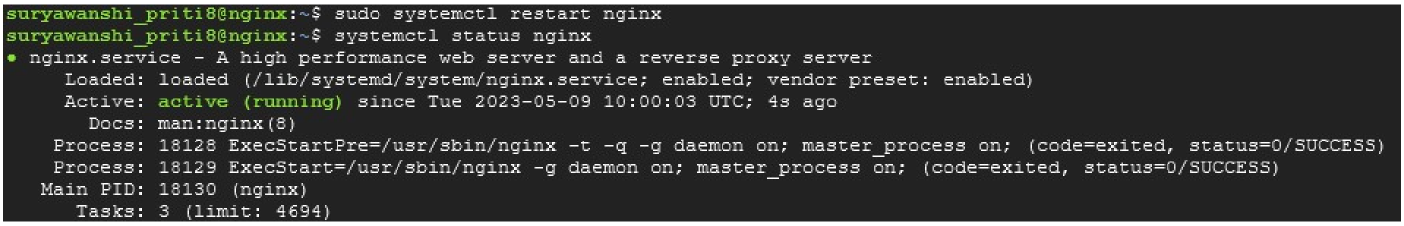
sudo systemctl start nginx



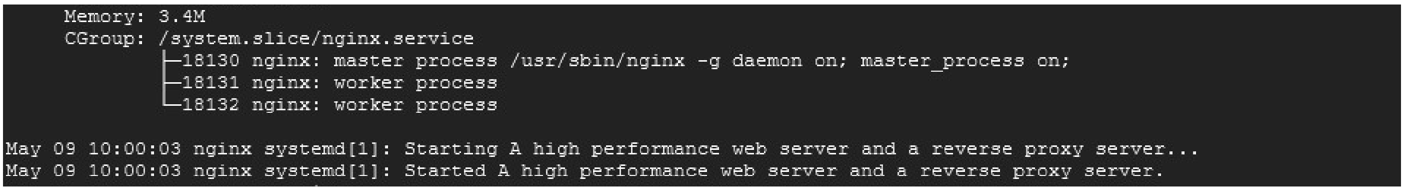
```
suryawanshi_priti8@nginx:~$ sudo systemctl start nginx
suryawanshi_priti8@nginx:~$ systemctl status nginx
● nginx.service - A high performance web server and a reverse proxy server
     Loaded: loaded (/lib/systemd/system/nginx.service; enabled; vendor preset: enabled)
     Active: active (running) since Tue 2023-05-09 09:59:32 UTC; 6s ago
       Docs: man:nginx(8)
    Process: 18115 ExecStartPre=/usr/sbin/nginx -t -q -g daemon on; master_process on; (code=exited, status=0/SUCCESS)
    Process: 18116 ExecStart=/usr/sbin/nginx -g daemon on; master_process on; (code=exited, status=0/SUCCESS)
   Main PID: 18117 (nginx)
      Tasks: 3 (limit: 4694)
     Memory: 3.0M
     CGroup: /system.slice/nginx.service
             ├─18117 nginx: master process /usr/sbin/nginx -g daemon on; master_process on;
             ├─18118 nginx: worker process
             └─18119 nginx: worker process

May 09 09:59:32 nginx systemd[1]: Starting A high performance web server and a reverse proxy server...
May 09 09:59:32 nginx systemd[1]: Started A high performance web server and a reverse proxy server.
```

To restart Nginx web server using systemctl

sudo systemctl restart nginx



```
suryawanshi_priti8@nginx:~$ sudo systemctl restart nginx
suryawanshi_priti8@nginx:~$ systemctl status nginx
● 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 Tue 2023-05-09 10:00:03 UTC; 4s ago
       Docs: man:nginx(8)
    Process: 18128 ExecStartPre=/usr/sbin/nginx -t -q -g daemon on; master_process on; (code=exited, status=0/SUCCESS)
    Process: 18129 ExecStart=/usr/sbin/nginx -g daemon on; master_process on; (code=exited, status=0/SUCCESS)
   Main PID: 18130 (nginx)
      Tasks: 3 (limit: 4694)
```



```
     Memory: 3.4M
     CGroup: /system.slice/nginx.service
             ├─18130 nginx: master process /usr/sbin/nginx -g daemon on; master_process on;
             ├─18131 nginx: worker process
             └─18132 nginx: worker process

May 09 10:00:03 nginx systemd[1]: Starting A high performance web server and a reverse proxy server...
May 09 10:00:03 nginx systemd[1]: Started A high performance web server and a reverse proxy server.
```

If you are only making configuration changes, Nginx can reload without dropping connections.

sudo systemctl reload nginx

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

sudo systemctl disable nginx

To re-enable the service to start up at boot

sudo systemctl enable nginx

Step #5:Configure Nginx Server-Blocks on Ubuntu 22.04 LTS

First Create a directory structure as shown below command

sudo mkdir -p /var/www/test.devopshint.info/html

Assign the directory's ownership to the currently logged-in user. Ensure to be logged in as a regular user rather than a sudo user, it will make it easier for us to edit or create the content in the directories.

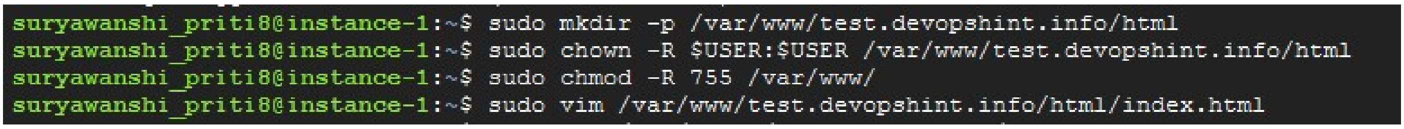
sudo chown -R $USER:$USER /var/www/test.devopshint.info/html

Grant the logged-in user the read, write and execute permissions.

sudo chmod -R 755 /var/www/

create a sample index.html file inside the domain root directory for each of our sites that will be displayed when you visit the domain in your browser.

sudo vim /var/www/test.devopshint.info/html/index.html

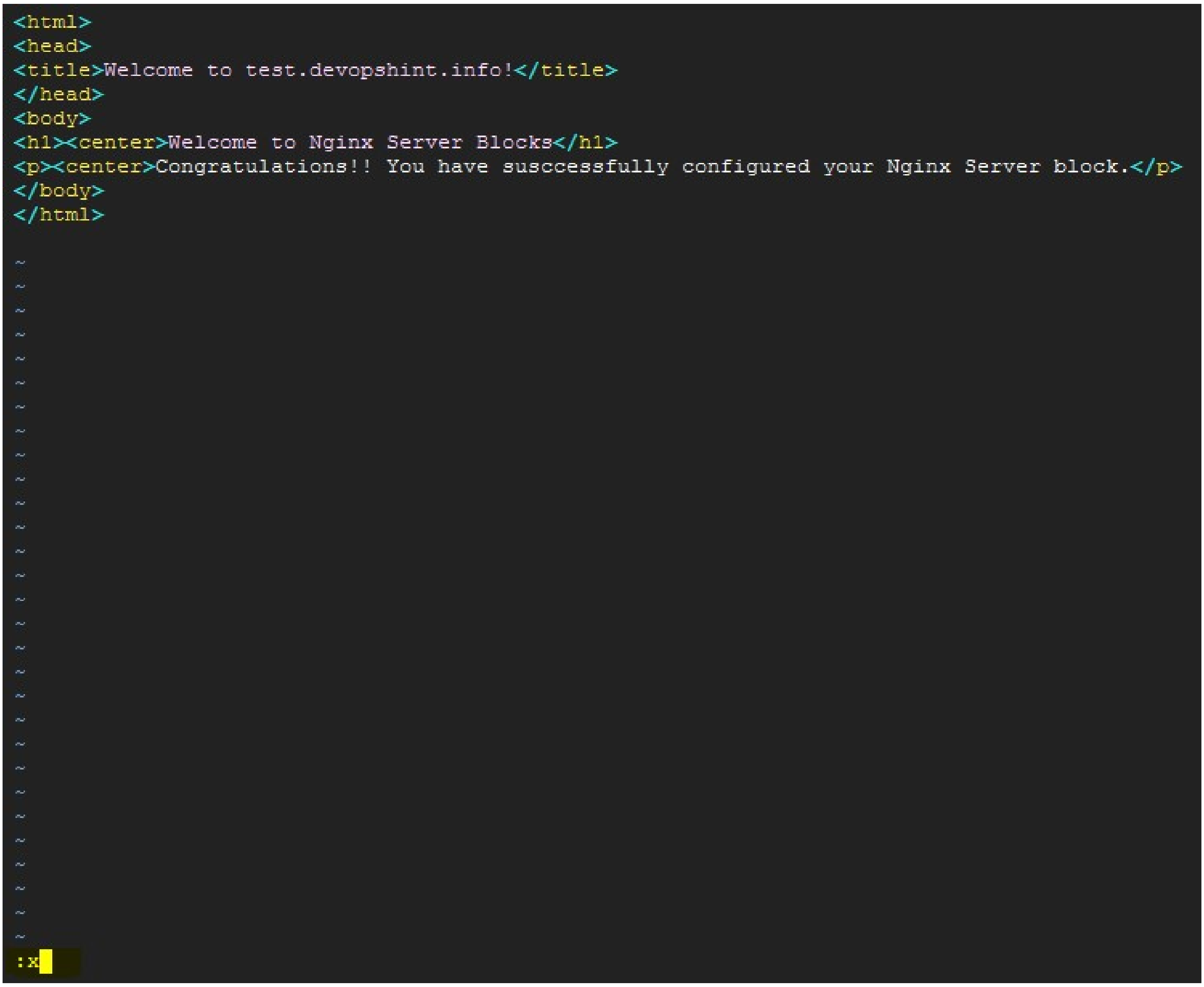


```
suryawanshi_priti8@instance-1:~$ sudo mkdir -p /var/www/test.devopshint.info/html
suryawanshi_priti8@instance-1:~$ sudo chown -R $USER:$USER /var/www/test.devopshint.info/html
suryawanshi_priti8@instance-1:~$ sudo chmod -R 755 /var/www/
suryawanshi_priti8@instance-1:~$ sudo vim /var/www/test.devopshint.info/html/index.html
```

Paste the following HTML content.

```
<html>
<head>
<title>Welcome to test.devopshint.info! </title>
</head>
<body>
<h1><center> Welcome to Nginx Server Blocks</h1>
<p><center>Congratulations!! You have susccessfully configured your Nginx Server block.</p>
```

```
</body>
</html>
```



```
<html>
<head>
<title>Welcome to test.devopshint.info!</title>
</head>
<body>
<h1><center>Welcome to Nginx Server Blocks</h1>
<p><center>Congratulations!! You have suscessfully configured your Nginx Server block.</p>
</body>
</html>

~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
:x
```

Note:- To save and exit press Esc :x+ enter

The location for Nginx servers is in the `/etc/nginx/sites-available` directory. We will therefore need to create server blocks that will serve the content in the `index.html` files we created earlier.

```
sudo vim /etc/nginx/sites-available/test.devopshint.info
```

Paste in the content below;

```
server {
listen 80;
listen [::]:80;
root /var/www/test.devopshint.info/html;
index index.html index.htm index.nginx-debian.html;
server_name test.devopshint.info;
location / {
```

try_files $uri $uri/ =404;

}

}

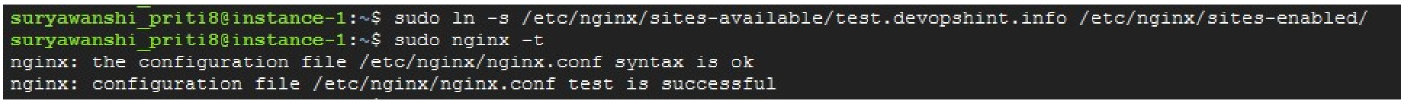
To save and exit the file press **Esc** 😡 **enter**.

Enable Nginx Server Block (Link the Nginx server block to `/etc/nginx/sites-enabled/` directory as shown to enable)

sudo ln -s /etc/nginx/sites-available/test.devopshint.info /etc/nginx/sites-enabled/

You should now confirm whether all configurations are in order.

sudo nginx –t

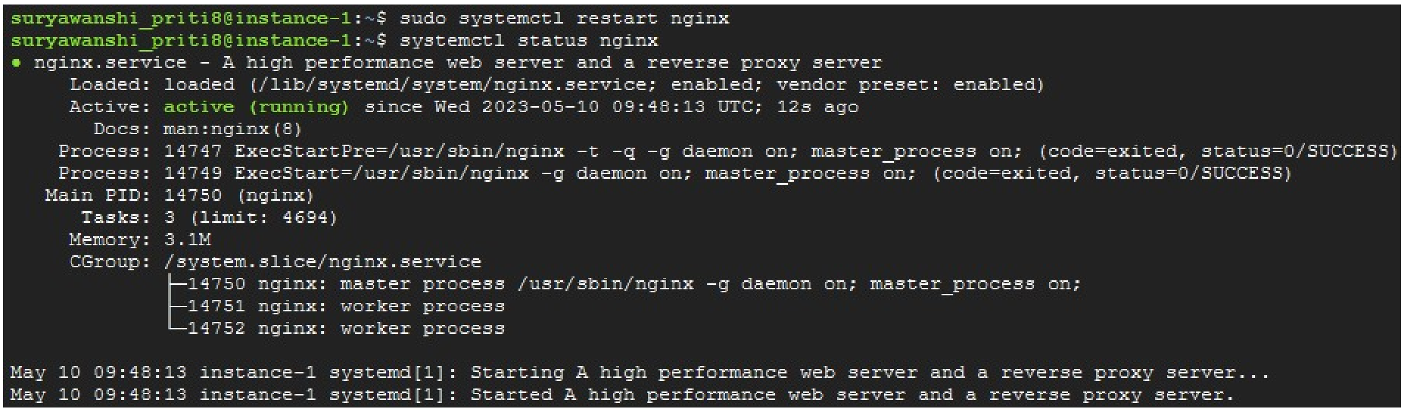


```
suryawanshi_priti8@instance-1:~$ sudo ln -s /etc/nginx/sites-available/test.devopshint.info /etc/nginx/sites-enabled/
suryawanshi_priti8@instance-1:~$ sudo nginx -t
nginx: the configuration file /etc/nginx/nginx.conf syntax is ok
nginx: configuration file /etc/nginx/nginx.conf test is successful
```

To verify changes restart Nginx service and check Nginx is running or not.

sudo systemctl restart nginx

systemctl status nginx

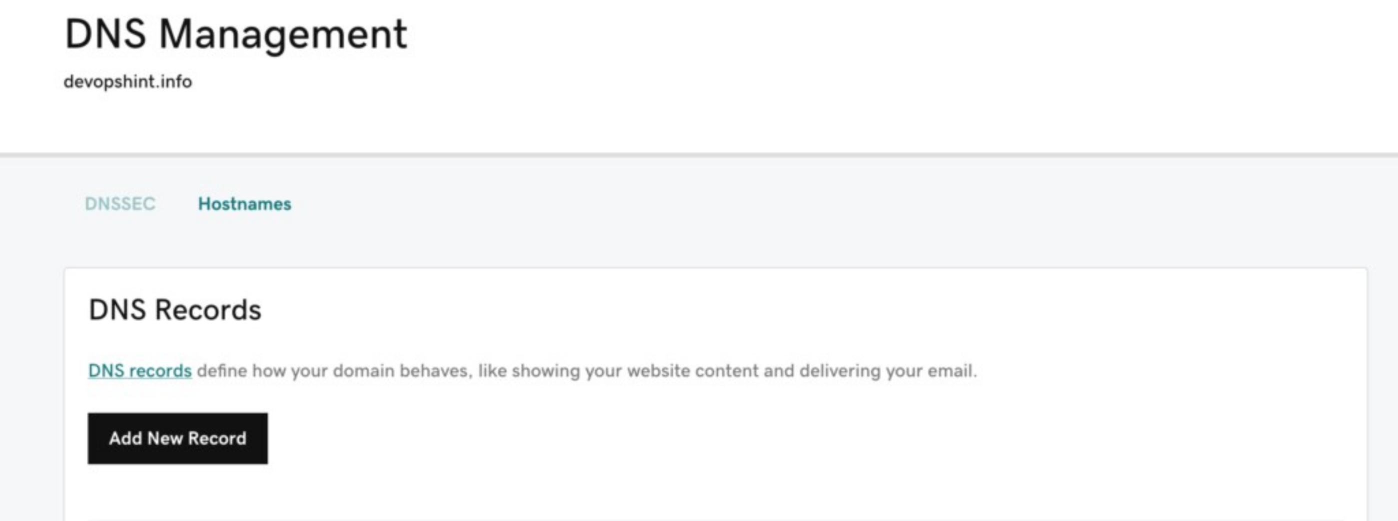


```
suryawanshi_priti8@instance-1:~$ sudo systemctl restart nginx
suryawanshi_priti8@instance-1:~$ systemctl status nginx
● nginx.service - A high performance web server and a reverse proxy server
     Loaded: loaded (/lib/systemd/system/nginx.service; enabled; vendor preset: enabled)
     Active: active (running) since Wed 2023-05-10 09:48:13 UTC; 12s ago
       Docs: man:nginx(8)
    Process: 14747 ExecStartPre=/usr/sbin/nginx -t -q -g daemon on; master_process on; (code=exited, status=0/SUCCESS)
    Process: 14749 ExecStart=/usr/sbin/nginx -g daemon on; master_process on; (code=exited, status=0/SUCCESS)
   Main PID: 14750 (nginx)
      Tasks: 3 (limit: 4694)
     Memory: 3.1M
     CGroup: /system.slice/nginx.service
             ├─14750 nginx: master process /usr/sbin/nginx -g daemon on; master_process on;
             ├─14751 nginx: worker process
             └─14752 nginx: worker process

May 10 09:48:13 instance-1 systemd[1]: Starting A high performance web server and a reverse proxy server...
May 10 09:48:13 instance-1 systemd[1]: Started A high performance web server and a reverse proxy server.
```

Nginx is running on your Ubuntu machine.

Test the Nginx server block we will test by browsing your domain name in your favorite browser.

Add **A** record in with your domain provider with your VM IP to access your server blocks using domain name

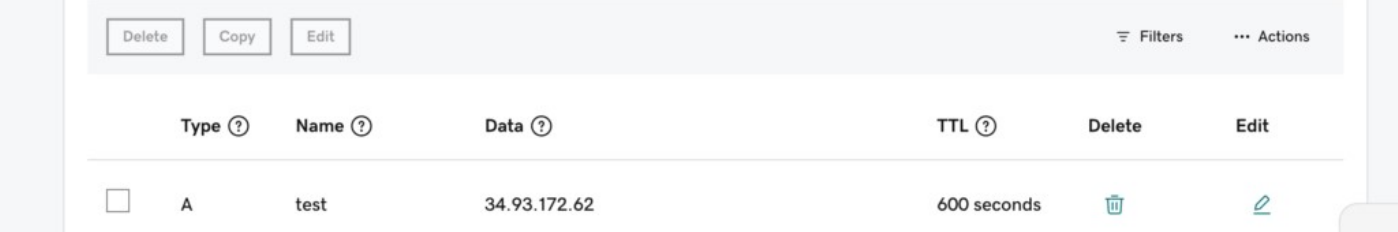


DNS Management

devopshint.info

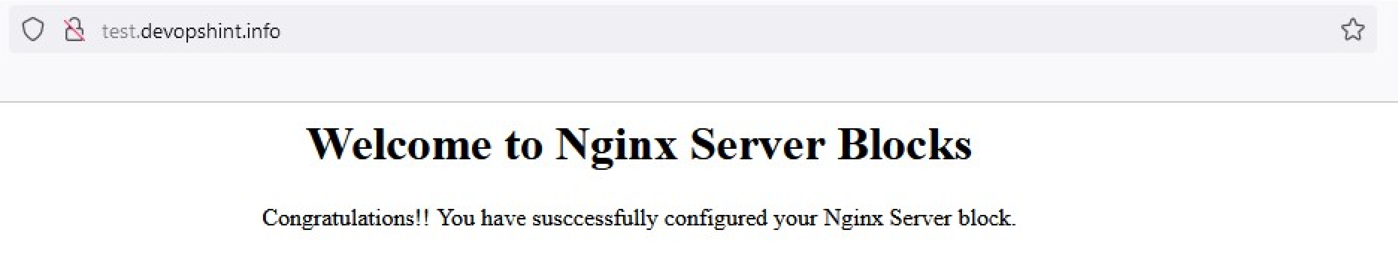
DNS Records

DNS records define how your domain behaves, like showing your website content and delivering your email.



	Type	Name	Data	TTL	Delete	Edit
□	A	test	34.93.172.62	600 seconds		

Now Access your server block domain on browser as shown below screenshot.



Welcome to Nginx Server Blocks

Congratulations!! You have susccessfully configured your Nginx Server block.

Step #6:Nginx Configuration files and Directories

Below are Important Nginx configuration files and directory which you should know

Default Nginx web content:

/var/www/html: When we install Nginx be default /var/www/html directory gets created and when you access it on browser then content loads from this directory.

Nginx Server configuration files:

`/etc/nginx`: this is Nginx configuration directory. All of the Nginx configuration files reside here.

`/etc/nginx/nginx.conf`: this is Nginx global configuration file when you modify it , it will apply to all content which used Nginx.

`/etc/nginx/sites-available/`: To enable a website, you must create a symbolic link inside the /etc/nginx/sites-enabled directory pointing to the actual vhost file in /etc/nginx/sites-available . The nginx. conf file reviews the contents of the sites-enabled directory and determines which virtual host files to include

`/etc/nginx/sites-enabled/`: The ../sites-enabled/ folder will include symlinks to the site configuration files located within **/etc/nginx/sites-available/**

`/etc/nginx/snippets`: Snippets allow you to insert raw NGINX config into different contexts of the NGINX configurations that the Ingress Controller generates.

Conclusion:

In this article we have covered How to Install NGINX on Ubuntu 22.04 LTS,Adjust the Firewall in Ubuntu 22.04 LTS,Verify Nginx Installation on Ubuntu 22.04 LTS,Managing Nginx processes using systemctl, Configure Nginx Server-Blocks onUbuntu 22.04 LTS, Nginx Configuration files and Directories.

Related Articles:

How to Install Nginx Ingress Controller Kubernetes KOPS using Helm 3

Reference:

Nginx official documentation