

NGINX - The fastest web server

NGINX is open source software for web serving, reverse proxying, caching, load balancing, media streaming, and more. It started out as a web server designed for maximum performance and stability. In addition to its HTTP server capabilities, NGINX can also function as a proxy server for email (IMAP, POP3, and SMTP) and a reverse proxy and load balancer for HTTP, TCP, and UDP servers.

Event-driven, asynchronous architecture, high-performance

Used by many websites such as Dropbox, Netflix, Gitlab and Zynga

As a software-only all-in-one load balancer, web server, API gateway, and reverse proxy that is designed for cloud-native architectures, NGINX helps accelerate IT infrastructure and application modernization efforts

Nginx is built to offer **low memory usage** and high concurrency. Rather than creating new processes for each web request, Nginx uses an asynchronous, event-driven approach where requests are handled in a single thread.

Some common features seen in Nginx include:

- Reverse proxy with caching
- IPv6
- Load balancing
- FastCGI support with caching
- WebSockets
- Handling of static files, index files, and auto-indexing
- TLS/SSL with SNI

Installing nginx on ubuntu18

1. `sudo apt update`
2. `sudo apt install nginx`
3. `sudo ufw allow 'Nginx HTTP'`
4. Check status of nginx
 - a. `sudo systemctl status nginx`
5. In Browser access below URL
 - a. `http://<AWS_VM_HOST_Public_IP>`

You should see the default Nginx landing page:

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.

Managing the NGINX processes:

1. To stop your web server, type:
 - `sudo systemctl stop nginx`
2. To start the web server when it is stopped, type:
 - `sudo systemctl start nginx`
3. To stop and then start the service again, type:
 - `sudo systemctl restart nginx`
4. If you are simply making configuration changes, Nginx can often reload without dropping connections. To do this, type:
 - `sudo systemctl reload nginx`
5. 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`
6. To re-enable the service to start up at boot, you can type:
 - `sudo systemctl enable nginx`

Setup "file browser" mode on freshly installed nginx server:

1. Edit default config for nginx:
`sudo vim /etc/nginx/sites-available/default`
2. Add following to config section:

```
location /logs {
    alias /home/ubuntu/logs/;
    autoindex on;
}
```
3. Create folder and sample file there:
`mkdir -p /home/ubuntu/logs/release_2021_1.0_20211003`
`ls -la >/home/ubuntu/logs/testcase_001.log`
4. Restart nginx
`sudo systemctl restart nginx`
5. Check result:
`http://<AWS_VMHOST_PUBLIC_IP>/logs/`

Copying testlogs to this AWS Nginx host using SCP:

```
scp -i ~/<your_aws_key>.pem <src_file>
ubuntu@<AWS_vmhost_PublicIp>:/home/ubuntu/logs/release_2021_1.0_10032021
```