Nginx Install Journal

This document journals starting from scratch and setting up a plain html/css website for http and https on GCP Debian 10 instance with nginx. The next step, which is found in a different document, is to get nginx to reverse proxy django.

**Machine Setup**

GCP instance, ‘reasoning-technology-server-1’, 1vCPU Intel Broadwell, 3.75GB memory, 30 Gig disk. (The machine configuration may be changed on the console later by first stopping the machine, and then editing on the ‘Instance Details’ panel.)

Setting the correct static IP for the instance

Recover an existing one: VPC network -> External IP address

Make a new one: Instance Details -> edit -> network interface

I gave reasoning-technology-server-1 our ‘ ip-4’ (35.194.71.194).

DNS A records adjusted for reasoningtechnology.com on Godaddy. This would be done later if they had pointed to another site that worked.

Debian 10. Was a toss up with Centos, but Marc had difficulty getting the version of Python needed on Centos, so thought I would try Debian. (Debian is also notorious for being behind on versions for packages …)

# apt update

# apt upgrade -y

**Package install nginx**

# apt install nginx

Received an error that it was already installed. Indeed:

> ps -ef | grep nginx

root 25598 1 0 11:40 ? 00:00:00 nginx: master process /usr/sbin/nginx -g daemon on; master\_process

on;

www-data 25599 25598 0 11:40 ? 00:00:00 nginx: worker process

thomas\_+ 25930 25923 0 11:54 pts/0 00:00:00 grep nginx

Going to reasoningtechnology.com hoping for a hello page but got “refused to connect”.

**Server run control commands**

# systemctl enable|start|restart|status|stop nginx

Enabling, rebooting, starting, and still get ‘refused to connect’. ‘Status’ shows it is running with no complaints.

# systemctl status -l 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 2020-02-26 12:04:44 UTC; 1h 23min ago

Docs: man:nginx(8)

Process: 3388 ExecStartPre=/usr/sbin/nginx -t -q -g daemon on; master\_process on; (code=exited, status=0/SUCCESS\

)

Process: 3389 ExecStart=/usr/sbin/nginx -g daemon on; master\_process on; (code=exited, status=0/SUCCESS)

Main PID: 3390 (nginx)

Tasks: 2 (limit: 4374)

Memory: 2.8M

CGroup: /system.slice/nginx.service

├─3390 nginx: master process /usr/sbin/nginx -g daemon on; master\_process on;

└─3391 nginx: worker process

Feb 26 12:04:44 reasoning-technology-server-1 systemd[1]: Starting A high performance web server and a reverse pro\

xy server...

Feb 26 12:04:44 reasoning-technology-server-1 systemd[1]: nginx.service: Failed to parse PID from file /run/nginx.\

pid: Invalid argument

Feb 26 12:04:44 reasoning-technology-server-1 systemd[1]: Started A high performance web server and a reverse prox\

y server.

It gives an error, “Failed to parse PID from file /run/nginx”. (note: <https://www.cloudinsidr.com/content/heres-fix-nginx-error-failed-read-pid-file-linux/>)

# umask

0022

# mkdir /etc/systemd/system/nginx.service.d

# printf "[Service]\nExecStartPost=/bin/sleep 0.1\n" > /etc/systemd/system/nginx.service.d/override.conf

# systemctl daemon-reload

# systemctl restart nginx

And indeed the failed to parse PID error is now gone, but we still get a ‘refused to connect’.

/var/log/nginx/error.log shows no errors.

**Installing UFW firewall**

See the file here ‘ufw install’. UFW is not installed by default in Debian 10, but I added the install as part of ‘new sytem setup’.

# **ufw status**

Status: active

To Action From

-- ------ ----

WWW Full ALLOW Anywhere

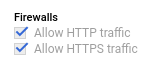
22/tcp ALLOW Anywhere

WWW Full (v6) ALLOW Anywhere (v6)

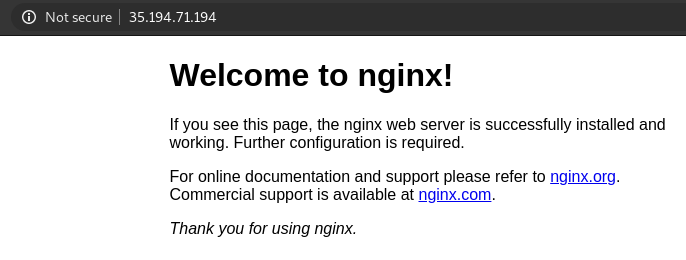
22/tcp (v6) ALLOW Anywhere (v6)

Still have ‘www.reasoningtechnology.com refused to connect’. Stands to reason I suppose that *adding* a firewall wouldn’t open the site up.

Checking the google instance details to see that http and https are allowed. Yep, already allowed.



Maybe the mistake is to use reasoningtechnology.com instead of just the IP address. Maybe it isn’t recognized as a virtual host. Yes indeed, when going to the http:35.194.71.194 in chrome the welcome panel appears:



**Conventional Configuration**

I will start with a conventional website, then add the SSL certificates. After that I plan to take a snapshot of the GCP instance, and work on the duplicate machine to get the django portion running. Once that is running I’ll reassign the static IP to the django working machine.

**Installing a Test Page for Reasoning Technology**

(Note: <https://www.tecmint.com/install-nginx-with-server-blocks-on-debian-10/>)

Create an index.html test page.

> pwd

/var/www/html/reasoningtechnology.com

> ls -l index.html

-rw-r--r-- 1 thomas\_lynch thomas\_lynch 124 Feb 26 16:15 index.html

> cat index.html

<html>

<head>

<title>Welcome to RT</title>

</head>

<body>

<h1>Success!</h1>

</body>

</html>

Create file and link

# cd /etc/nginx/sites-available/

# ls -l reasoningtechnology.com

-rw-r--r-- 1 root root 349 Feb 26 16:26 reasoningtechnology.com

# cat reasoningtechnology.com

server {

listen 80;

listen [::]:80;

root /var/www/html/reasoningtechnology.com;

index index.html index.htm index.nginx-debian.html;

# sic, should be 'domain-name'

server\_name reasoningtechnology.com www.reasoningtechnology.com;

location / {

try\_files $uri $uri/ =404;

}

# cd /etc/nginx/sites-enabled/

root@reasoning-technology-server-1:/etc/nginx/sites-available# ls -l reasoningtechnology.com

-rw-r--r-- 1 root root 349 Feb 26 16:26 reasoningtechnology.com

# nginx -t

… ok

# systemctl restart nginx

Going to <http://reasoningtechnology.com> shows the test page. However, <https://reasoningtechnology.com> gives the ‘failed to connect’ message. That is because there is no port 443 in the server block.

Uploaded the flat website. RT is now back online, albeit in a limited sense.

Adding an SSL certificate

(note <https://www.digitalocean.com/community/tutorials/how-to-secure-nginx-with-let-s-encrypt-on-debian-10>)

I had to change the html resident directory name from reasoning-technology-host to reasoningtechnology.com to meet the requirements of the SSL install scripts. I have now reflected this change in this journal and in the notes.

# apt install python3-acme python3-certbot python3-mock python3-openssl python3-pkg-resources python3-pyparsing python3-zope.interface

# apt install python3-certbot-nginx

# systemctl reload nginx

# sudo certbot --nginx -d reasoningtechnology.com -d www.reasoningtechnology.com

Saving debug log to /var/log/letsencrypt/letsencrypt.log

Plugins selected: Authenticator nginx, Installer nginx

Obtaining a new certificate

Performing the following challenges:

http-01 challenge for reasoningtechnology.com

http-01 challenge for www.reasoningtechnology.com

Waiting for verification...

Cleaning up challenges

Deploying Certificate to VirtualHost /etc/nginx/sites-enabled/reasoningtechnology.com

Deploying Certificate to VirtualHost /etc/nginx/sites-enabled/reasoningtechnology.com

Please choose whether or not to redirect HTTP traffic to HTTPS, removing HTTP access.

- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -

1: No redirect - Make no further changes to the webserver configuration.

2: Redirect - Make all requests redirect to secure HTTPS access. Choose this for

new sites, or if you're confident your site works on HTTPS. You can undo this

change by editing your web server's configuration.

- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -

Select the appropriate number [1-2] then [enter] (press 'c' to cancel): **2**

Redirecting all traffic on port 80 to ssl in /etc/nginx/sites-enabled/reasoningtechnology.com

Redirecting all traffic on port 80 to ssl in /etc/nginx/sites-enabled/reasoningtechnology.com

- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -

Congratulations! You have successfully enabled [https://reasoningtechnology.com](https://reasoningtechnology.com/)

and https://www.reasoningtechnology.com

You should test your configuration at:

<https://www.ssllabs.com/ssltest/analyze.html?d=reasoningtechnology.com>

<https://www.ssllabs.com/ssltest/analyze.html?d=www.reasoningtechnology.com>

- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -

**IMPORTANT NOTES:**

- Congratulations! Your certificate and chain have been saved at:

/etc/letsencrypt/live/reasoningtechnology.com/fullchain.pem

Your key file has been saved at:

/etc/letsencrypt/live/reasoningtechnology.com/privkey.pem

Your cert will expire on 2020-05-26. To obtain a new or tweaked

version of this certificate in the future, simply run certbot again

with the "certonly" option. To non-interactively renew \*all\* of

your certificates, run "certbot renew"

- If you like Certbot, please consider supporting our work by:

Donating to ISRG / Let's Encrypt: <https://letsencrypt.org/donate>

Donating to EFF: <https://eff.org/donate-le>

Ok, the site is working without the django components, (so no signup or login) ..