**Setting Up SSL Redirection Using NGINX**

# **Objective:**

## The objective of this document is to provide a step-by-step guide for configuring NGINX to redirect requests from a domain to Grafana, while ensuring secure connections using SSL/TLS certificates.

Install Nginx in the ubuntu

| sudo apt update -y  sudo apt install -y nginx |
| --- |

Our next step will be to create a website.

The default HTML page is in the directory: /var/www/html

| sudo mkdir -p /var/www/html/test  sudo chown -R $USER:$USER /var/www/html/test  sudo chmod -R 755 /var/www/html/test  Sudo vi /var/www/html/test/index.html  <html>  <head>  <title>Welcome to your\_domain!</title>  </head>  <body>  <h1>Success! The your\_domain server block is working!</h1>  </body>  </html>  ~  ~ |
| --- |

**Create a new Nginx server block file :**

| sudo vi /etc/nginx/sites-available/test |
| --- |

| server {  listen 80;  listen [::]:80;  root /var/www/html/test;  index index.html index.htm index.nginx-debian.html;  server\_name test www.test.com;  location / {  try\_files $uri $uri/ =404;  }  } |
| --- |

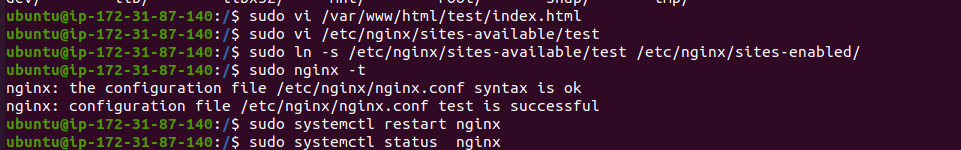
~

**Create a symbolic link to enable this configuration**

| sudo ln -s /etc/nginx/sites-available/test /etc/nginx/sites-enabled/ |
| --- |

**Test the configuration for syntax errors:**

| **sudo nginx -t** |
| --- |

****

**Incase the default page is loaded, remove the symbolic link**

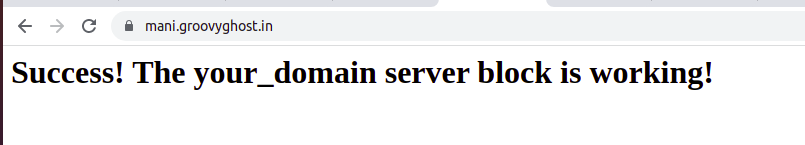
| **sudo unlink /etc/nginx/sites-enabled/default** |
| --- |

****

**Create a SSL certificate using LetsEncrypt**

| **sudo apt install certbot python3-certbot-nginx** |
| --- |

| **sudo certbot --nginx -d mani.groovyghost.in** |
| --- |

****

**Renewing the certificates:**

| **sudo certbot renew --dry-run** |
| --- |

| Crontab -e |
| --- |

| # Auto-renew let's encrypt SSL certificates 0 0 \* \* \* sudo certbot renew |
| --- |

**Recommended way to create domain in nginx is to create in the path /etc/nginx/conf.d**

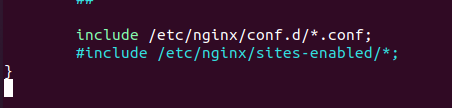
| **sudo vi /etc/nginx/conf.d/nginxtest.com.conf** |
| --- |

| **server {  listen 80;  listen [::]:80;   server\_name test.com www.test.com;   return 301 https://$host$request\_uri; }  server {  listen 443 ssl;  listen [::]:443 ssl;   root /var/www/html/test;  index index.html index.htm index.nginx-debian.html;   server\_name test.com www.test.com;   ssl\_certificate /etc/letsencrypt/live/mani.groovyghost.in/fullchain.pem;  ssl\_certificate\_key /etc/letsencrypt/live/mani.groovyghost.in/privkey.pem;   ssl\_protocols TLSv1.2 TLSv1.3;  ssl\_prefer\_server\_ciphers off;   location / {  try\_files $uri $uri/ =404;  }   # Additional SSL configuration can be added her }** |
| --- |

**I removed the domain under the site-available inorder to load the config file from the /etc/nginx/conf.d/nginx.test.com.conf**

| **cd /etc/nginx/sites-available sudo rm test systemctl restart nginx** |
| --- |

**Incase if the nginx fails to start, go to /etc/nginx/conf.d and look for http block, comment the #include the sites-enabled,**

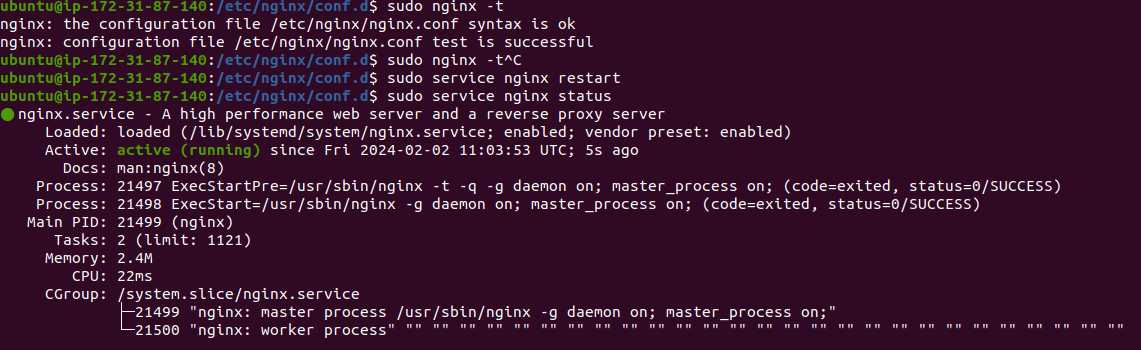
****

**Then run the configuration test in etc/nginx/conf.d/nginxtest.com.conf**

**sudo nginx -t**

**sudo service nginx restart**

**sudo service nginx status**

****

****