nginx ----- https server configuration

Yes, you can configure HTTPS for an IP address without a domain name. The process is similar, and you can use a self-signed certificate or obtain a free SSL/TLS certificate from Let's Encrypt.

Here are the steps using a self-signed certificate:

Generate a Self-Signed Certificate:

You can generate a self-signed certificate using OpenSSL. Run the following commands:

-------------------------sudo openssl req -x509 -nodes -days 365 -newkey rsa:2048 -keyout /etc/nginx/certs/self-signed.key -out /etc/nginx/certs/self-signed.crt

This will create a self-signed certificate (self-signed.crt) and private key (self-signed.key) in the /etc/nginx/certs/ directory.

Configure Nginx for HTTPS:

Edit your Nginx configuration file:

------------------------sudo nano /etc/nginx/sites-available/default

It seems that the /etc/nginx/certs/ directory does not exist in the current directory listing. In this case, you can create the directory and then run the openssl command to generate the self-signed certificate. Here are the steps:

Create the Certs Directory:

Run the following command to create the /etc/nginx/certs/ directory:

-----------sudo mkdir -p /etc/nginx/certs/

Set Permissions:

Ensure that the user running the openssl command has the necessary permissions to write to the /etc/nginx/certs/ directory. You can change the ownership of the directory to the user with the following command:

-----------------------chmod 777 -R \*

Now, run the openssl command again to generate the self-signed certificate:

------------------------------ sudo openssl req -x509 -nodes -days 365 -newkey rsa:2048 -keyout /etc/nginx/certs/self-signed.key -out /etc/nginx/certs/self-signed.crt

This should generate the self-signed certificate and key in the /etc/nginx/certs/ directory.

After completing these steps, you should be able to use the generated self-signed certificate for configuring HTTPS in Nginx. Remember to update your Nginx configuration with the correct paths to the certificate and key files.

sudo service nginx restart------------

server {

listen 80 default\_server;

listen [::]:80 default\_server;

root /var/www/html;

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

server\_name 3.6.225.138;

location / {

proxy\_pass http://localhost:5000;

proxy\_http\_version 1.1;

proxy\_set\_header Upgrade $http\_upgrade;

proxy\_set\_header Connection 'upgrade';

proxy\_set\_header Host $host;

proxy\_cache\_bypass $http\_upgrade;

}

}

server {

listen 443 ssl default\_server;

listen [::]:443 ssl default\_server;

ssl\_certificate /etc/nginx/certs/self-signed.crt;

ssl\_certificate\_key /etc/nginx/certs/self-signed.key;

gzip off;

root /var/www/html;

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

server\_name 3.6.225.138;

location / {

proxy\_pass http://localhost:5000;

proxy\_http\_version 1.1;

proxy\_set\_header Upgrade $http\_upgrade;

proxy\_set\_header Connection 'upgrade';

proxy\_set\_header Host $host;

proxy\_cache\_bypass $http\_upgrade;

}

}

----------------nginx -t

----------------sudo service nginx restart