# **CONFIGURING NEXUS WITH SSL USING NGNIX AS REVERSE PROXY BY LETSENCRYPT**

We want [Nexus repository manager](https://www.sonatype.com/nexus-repository-sonatype) (which it’s default port is 8081). Also, want [Nginx](http://nginx.org/) as a reverse proxy to serve Nexus on port 443 (Yes, we want SSL, so we use [let’s encrypt](http://letsencrypt.org/) )



## 

## 

## 

## 

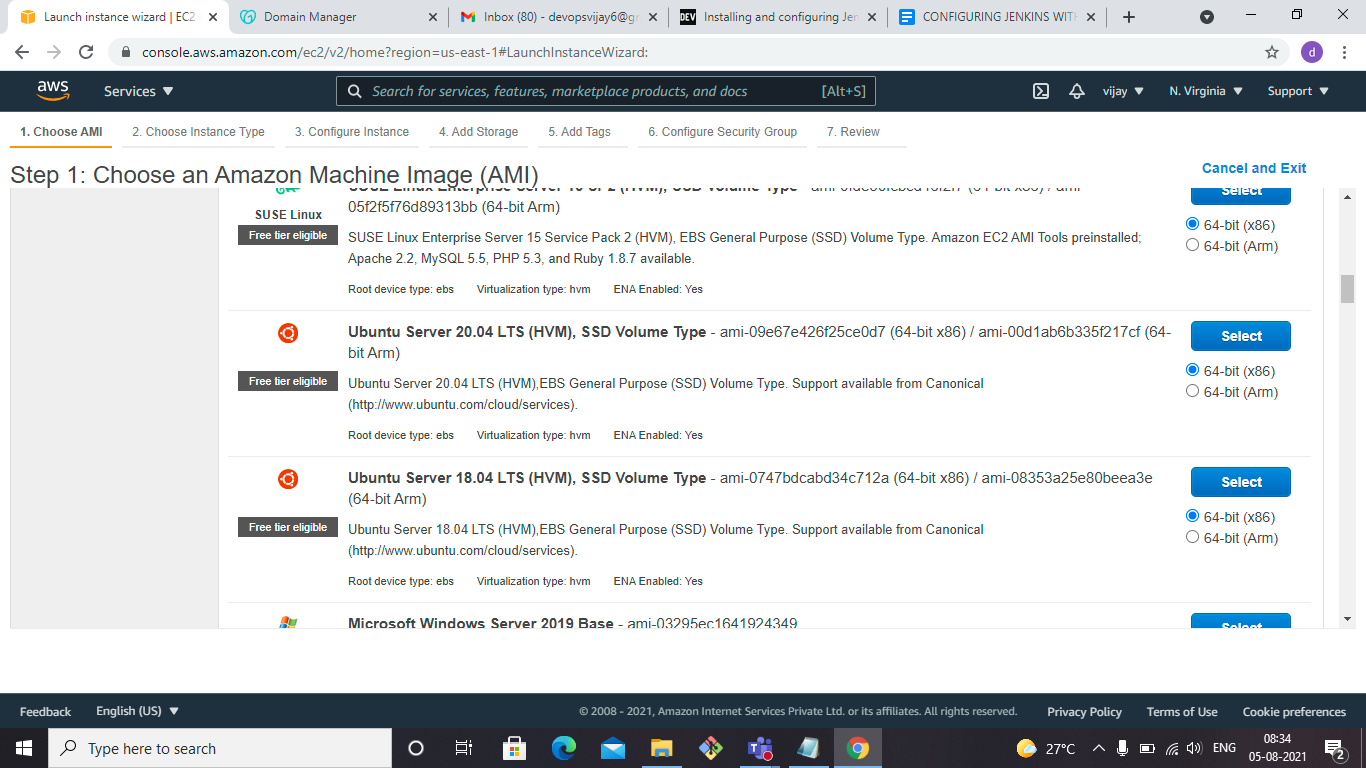
## 

## 

## **#1 First Step - Launching Instance:-**

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

### **Selecting the AMI**



### **Type of Instance**

In the next step you need to select an Instance Type, you need to select above t2.medium Instance(4gb ram and 2vcpu’s) according to your needs.

### 

### **Configure Instance**

Configure the instance to suit your requirements.

### **Selecting the storage**

Free tier eligible customers can get up to 30 GB of EBS General Purpose (SSD) or Magnetic storage.I have gone with default 8GB

### **Adding Tags**

A tag consists of a case-sensitive key-value pair. For example, you could define a tag with key = Name and value = Nexus-server.

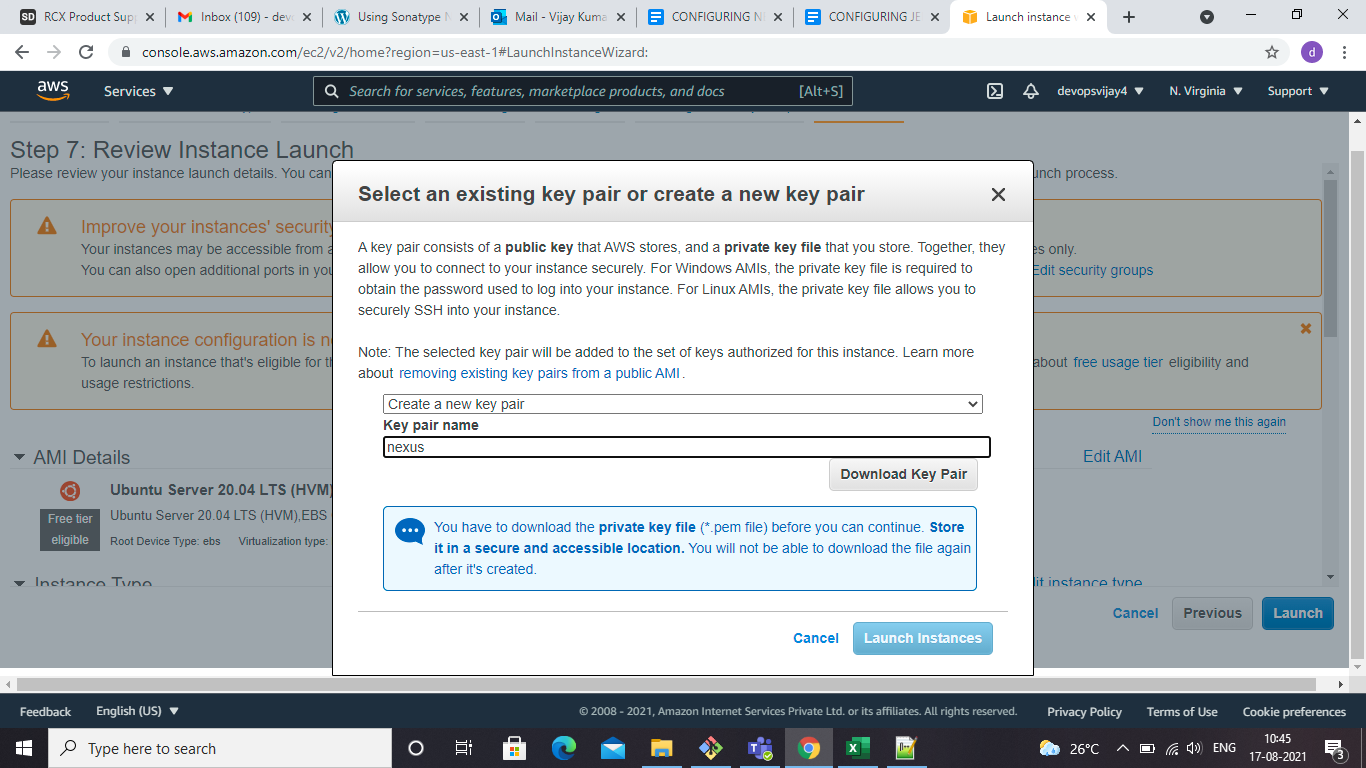
### **Security Group**

### Here set the firewall rules, to allow access to your instance

Add ports 22,8081,80,443

### **Review and launch**

In the last step you can review the total configuration and set the key pair to connect through SSH to your instance.



## 

## **#2 Second Step-connect the instance via SSH using Git Bash**

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

## You can use this command in your terminal to access the instance via SSH:

ssh -i "nexus.pem" [ubuntu@ec2-3-223-140-206.compute-1.amazonaws.com](mailto:ubuntu@ec2-3-223-140-206.compute-1.amazonaws.com)

## **#3 Fourth Step - Installing Nexus**

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

Execute the below commands in your Server

sudo -i

apt update -y

apt install openjdk-8-jre-headless -y

sudo apt-get install nginx -y

cd /opt

wget <https://download.sonatype.com/nexus/3/latest-unix.tar.gz>

tar -xvf latest-unix.tar.gz

mv nexus-3.33.0-01/ nexus

useradd -d /home/nexus -s "/bin/bash" -c "comment" -m nexus

echo 'nexus ALL=(ALL) NOPASSWD: ALL' >> /etc/sudoers

chown -R nexus:nexus /opt/nexus

chown -R nexus:nexus /opt/sonatype-work

chown -R nexus:nexus /opt/sonatype-work

touch /opt/nexus/bin/nexus.rc

echo 'run\_as\_user="nexus"' >> /opt/nexus/bin/nexus.rc

sudo ln -s /opt/nexus/bin/nexus /etc/init.d/nexus

**Install Certbot:-**

Certbot is a letsencrypt client, we need to download.certbot can automatically configure NGINX for SSL/TLS.The Certbot packages on your system come with a cron job or systemd timer that will renew your certificates automatically before they expire.

sudo add-apt-repository ppa:certbot/certbot

sudo apt-get update

sudo apt-get install python3-certbot-nginx

## **#4 Fourth Step-Godaddy entry for Nexus Server**

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

My Domain name is Registered in GODADDY.so,

- Create a DNS record that associates your domain name and your server’s public IP address.

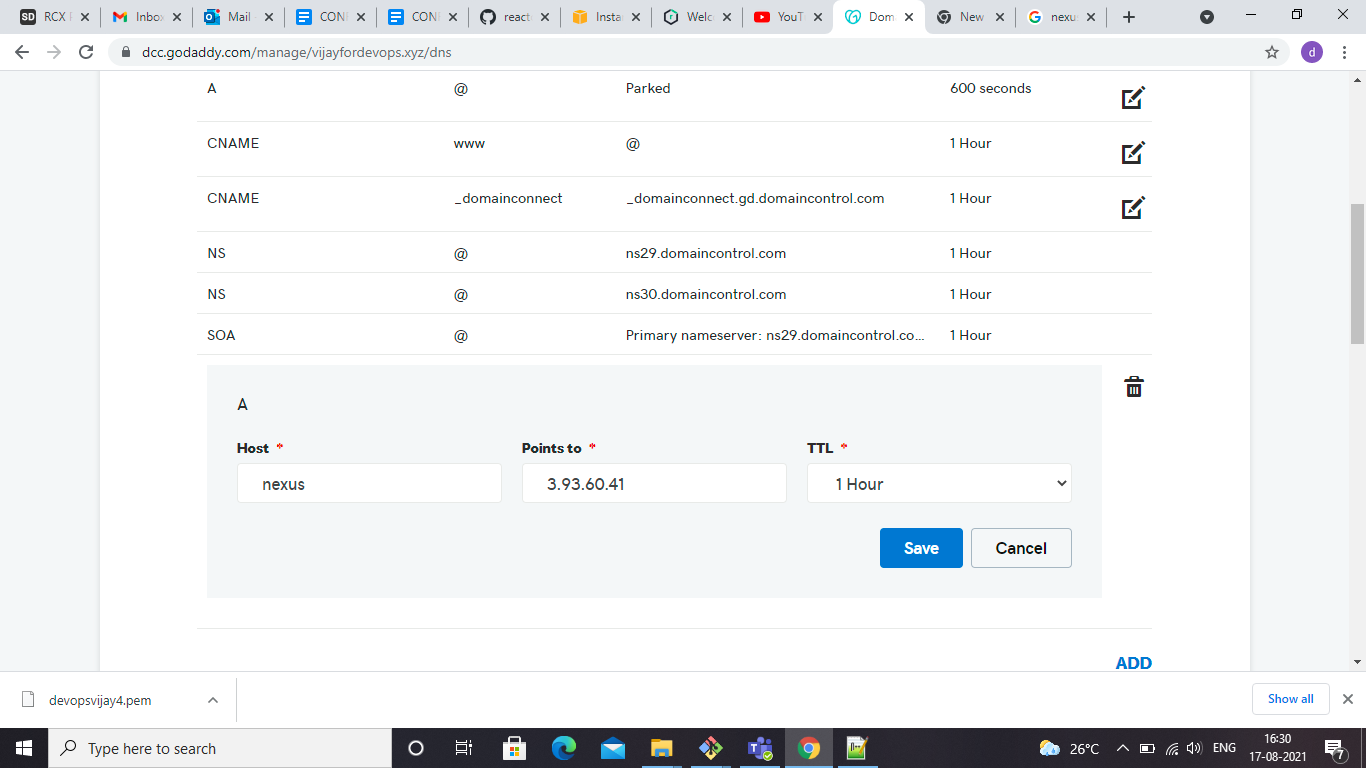
- Add "A" record for subdomain in DNS manage record on Godaddy console.

Go to DNS Management,click on Add

type:- A

host:- nexus

points to:- <Nexus server’s public-ip>

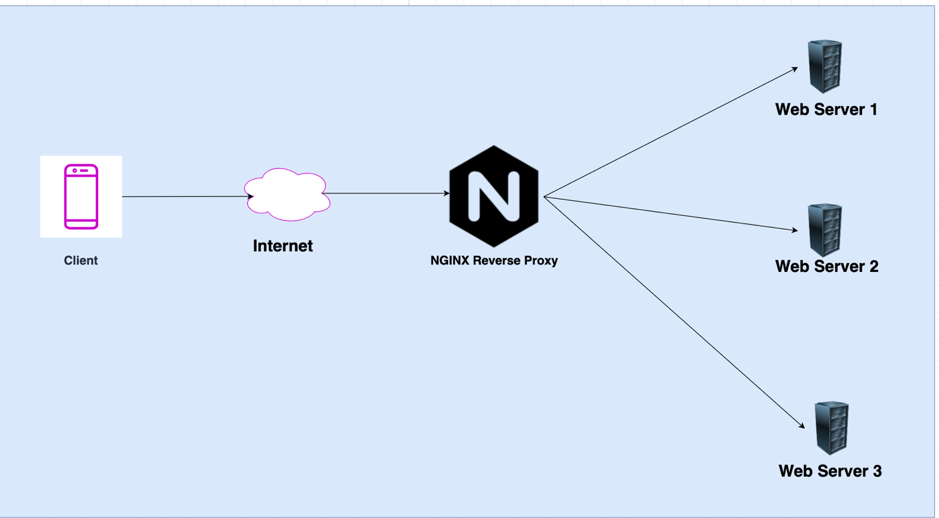


## **#5 Sixth Step - Nginx configuration with SSL**

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

- In situations where you have existing web sites on your server, you may find it useful to run Nexus behind [Nginx](https://nginx.org/), so that you can bind Nexus to the part of a bigger website that you may have.

- When a request arrives for certain URLs, Nginx becomes a proxy and further forward that request to Nexus, then it forwards the response back to the client.



go to

cd /etc/nginx/sites-available

make backup of default file:-

cd /etc/nginx/sites-available/

sudo mv default default.backup

sudo vi default (paste the below content in the default file)

server {

listen 80;

server\_name nexus.vijayfordevops.xyz;

location / {

proxy\_set\_header Host $host:$server\_port;

proxy\_set\_header X-Real-IP $remote\_addr;

proxy\_set\_header X-Forwarded-For $proxy\_add\_x\_forwarded\_for;

proxy\_set\_header X-Forwarded-Proto $scheme;

# Fix the "It appears that your reverse proxy set up is broken" error.

proxy\_pass http://127.0.0.1:8081;

proxy\_read\_timeout 90;

proxy\_redirect http://127.0.0.1:8081 https://nexus.vijayfordevops.xyz;

# proxy\_redirect http:// https://;

# Required for new HTTP-based CLI

proxy\_http\_version 1.1;

proxy\_request\_buffering off;

}

}

sudo nginx -t --(for verifying syntax)

sudo systemctl restart nginx

su - nexus

/etc/init.d/nexus start

## **#6 Sixth Step-Install SSL Certificates by using Letsencrypt**

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

sudo certbot --nginx -n -d nexus.vijayfordevops.xyz --email devopsvijay6@gmail.com --agree-tos --redirect --hsts

(it is NGINX plug‑in for certbot to generate certificates)

## How Let’s Encrypt Works:-

Before issuing a certificate, Let’s Encrypt validates ownership of your domain. The Let’s Encrypt client, running on your host, creates a temporary file (a token) with the required information in it. The Let’s Encrypt validation server then makes an HTTP request to retrieve the file and validates the token, which verifies that the DNS record for your domain resolves to the server running the Let’s Encrypt client.

Deploying Certificate to VirtualHost /etc/nginx/sites-enabled/default

Redirecting all traffic on port 80 to ssl in

/etc/nginx/sites-enabled/default

Note:-

certbot can automatically configure NGINX for SSL/TLS

You can open sudo vi /etc/nginx/sites-available/default file

and check ssl certificates are automatically configured by Nginx

listen 443 ssl; # managed by Certbot

ssl\_certificate /etc/letsencrypt/live/nexus.vijayfordevops.xyz/fullchain.pem; # managed by Certbot

ssl\_certificate\_key /etc/letsencrypt/live/nexus.vijayfordevops.xyz/privkey.pem; # managed by Certbot

include /etc/letsencrypt/options-ssl-nginx.conf; # managed by Certbot

ssl\_dhparam /etc/letsencrypt/ssl-dhparams.pem; # managed by Certbot

## **#7 Sixth Step-Check HTTPS**

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

goto

internet explorer browser and Type

[https://nexus.vijayfordevops.xyz](https://jenkins.vijayfordevops.xyz)

