# **Configure Sonatype Nexus repository with Docker and AWS EC2**

install docker

# add Repoyum-config-manager --add-repo [https://download.docker.com/linux/centos/docker-ce.repo](https://download.docker.com/linux/centos/docker-ce.repo" \t "https://perfecto25.medium.com/_blank)# installsudo yum install docker-ce

install docker-compose

sudo pip install docker-compose

make sure Docker is running

sudo systemctl start docker.service

# **Configure & Start Nexus**

on EC2 host, create a new Nexus directory, a Data directory (we can use this dir for backups), and create a new Docker-Compose file

mkdir /etc/nexus  
mkdir /etc/nexus/nexus-data  
touch /etc/nexus/docker-compose.yaml

add the following to the YAML

version: "2"services:  
 nexus:  
 image: sonatype/nexus3  
 volumes:  
 - "/etc/nexus/nexus-data:/nexus-data"  
 ports:  
 - "8081:8081"  
   
volumes:  
 nexus-data: {}

Change the Data directory permission so Docker can access it

chown -R 200 /etc/docker/nexus-data

Now start Nexus as a daemon

cd /etc/nexus  
docker-compose up -d

follow Nexus logs with

docker logs -f nexus\_nexus\_1

and check status of the Nexus container with

docker ps -a

Nexus should startup after a few minutes and you should have access to the console via [http://<ip](http://ip-of-ec2/" \t "https://perfecto25.medium.com/_blank) of your instance>:8081

default login is admin:admin123

# **Securing Nexus with HTTPS**

Add your IP to your DNS or Namehost so *repo.yourcompany.com* resolves to the public IP of the EC2 instance

If you have certificate or even a self-signed cert you can use, secure Nexus with either Apache or Nginx reverse proxy. This example shows Apache config, but Nginx is very similar

install HTTPD as well as mod\_ssl

sudo yum install httpd mod\_ssl

Once installed, configure a Nexus conf internal binary/artifact repository using

*vi /etc/httpd/conf.d/nexus.conf*

LoadModule headers\_module /usr/lib64/httpd/modules/mod\_headers.so#### NEXUS REPOProxyRequests Off  
RewriteEngine On  
ErrorLog /var/log/httpd/nexus\_error.log<VirtualHost \*:80>  
 ServerName repo.yourcompany.com  
 ServerAlias repo  
 Redirect / [https://repo.yourcompany.com](https://repo.quantitativebrokers.com/" \t "https://perfecto25.medium.com/_blank)  
</VirtualHost><VirtualHost \*:443>  
 ProxyPreserveHost On  
 ServerName repo.yourcompany.com  
 ServerAlias repo SSLEngine On   
 SSLCertificateKeyFile /etc/ssl/certs/yourcompany.key  
 SSLCertificateFile /etc/ssl/certs/yourcompany.crt  
 SSLCACertificateFile /etc/ssl/certs/yourcompanyintermdate.crt AllowEncodedSlashes NoDecode  
 ProxyPass / [http://localhost:8081/](http://localhost:8081/" \t "https://perfecto25.medium.com/_blank) nocanon  
 ProxyPassReverse / [http://localhost:8081/](http://localhost:8081/" \t "https://perfecto25.medium.com/_blank)  
 RequestHeader set X-Forwarded-Proto "https"  
</VirtualHost>

This will redirect any HTTP request to HTTPS, and then reverse proxy the 443 request to local port 8081

# **Uploading RPMs to Nexus**

once Nexus is setup, create a new YUM Hosted repository (lets call it “yumrepo”

to upload to this repo, you can use curl from cmd line

curl -v -k — user ‘admin:admin123’ — upload-file [myapp-1.0.5-x86\_64.rpm https://repo.mycompany.com/repository/yumrepo/myapp/myapp](https://repo.quantitativebrokers.com/repository/qb_yum/$folder/$binary" \t "https://perfecto25.medium.com/_blank)-1.0.5-x86\_64.rpm

# **Add Yum repo file**

add this file to any host that needs to access your repo

vi /etc/yum.repos.d/company.repo

[corp\_repo]  
name=Corp Yum Repo  
baseurl=[https://repo.company.com/repository/yumrepo](http://qbinfra02:8081/repository/qb_yum" \t "https://perfecto25.medium.com/_blank)  
enabled=1  
gpgcheck=0  
priority=1

# **Backing up Nexus data to AWS S3**

to backup all your data, simply use AWS S3 service and schedule a nightly backup

install AWS cli client

sudo pip install awscli

sign up for S3 and create a read/write account, create a new Bucket and a subdirectory that matches your EC2 hostname

example: S3://mybucket/ec2nexus

add your account credentials and region connection info

vi /root/.aws/config

[default]  
output = json  
region = us-east-1

vi /root/.aws/credenials

[default]  
aws\_access\_key\_id = ABCXYZ123  
aws\_secret\_access\_key = xxxyyyyzzzzz

add a cron to copy your Nexus-Data dir to S3

sudo crontab -e -u root

0 1 \* \* \* aws s3 sync /etc/nexus/nexus-data s3://yourBucketName/nameOfEC2host/nexus/ --exclude=\*.log

Your data will now be backed up nightly at 1am, in case Nexus fails or EC2 gets destroyed, simply rerun previous steps and copy over data from S3 to /etc/nexus/nexus-data