Generating the SSL Keys

In order for Harbor to function properly, you'll need to set it up to use SSL. In a production environment you must use certificates from a trusted Certificate Authority. For a testing environment you can work with self-signed certificates. Here are the steps to generate the self-signed certificates.

Generate the self-signed certificates with the command:

openssl req -newkey rsa:4096 -nodes -sha256 -keyout ca.key -x509 -days 3650 -out ca.crt

Generate the signing request with the command:

openssl req -newkey rsa:4096 -nodes -sha256 -keyout 192.168.1.75 -out 192.168.1.75

Note: Substitute the above IP address with either your server domain or server IP address.

Create a configuration file for the Subject Alternate name with the command:

nano extfile.cnf

In that file, paste the following contents:

subjectAltName = IP:192.168.1.75

Again, substitute the IP address of your Harbor server for 192.168.1.75.

Generate the certificate with the command:

openssl x509 -req -days 3650 -in 192.168.1.75 -CA ca.crt -CAkey ca.key -CAcreateserial -extfile extfile.cnf -out 192.168.1.75

Once again, substitute the above IP address with either your server domain or server IP address.

Create the client certificate with the command:

openssl req -new -x509 -text -key ca.key -out ca.cert

Copy the newly-generated certificates into the proper directory with the command:

sudo cp \*.crt \*.key \*.cert /etc/docker/certs.d/192.168.1.75

Make sure to substitute the IP address above with the IP address of your Harbor server. If you find that the /etc/docker/certs.d directory doesn't exist, create it with the command:

sudo mkdir /etc/docker/certs.d

You will probably also have to create the IP address subdirectory in the same fashion.