**Automate** **the installation and configuration of Nginx + Certbot** as part of your GitHub Actions workflow, using the same appleboy/ssh-action step where you SSH into your EC2. Essentially, you’ll just add commands to:

1. Install Nginx and Certbot (if not already installed)
2. Create the Nginx reverse proxy config for your app
3. Reload Nginx
4. Issue the SSL certificate with Certbot

**# --- Install Nginx & Certbot ---**

if ! command -v nginx >/dev/null 2>&1; then

echo "Nginx not found. Installing..."

sudo yum install -y nginx

sudo systemctl enable nginx

sudo systemctl start nginx

fi

if ! command -v certbot >/dev/null 2>&1; then

echo "Certbot not found. Installing..."

sudo yum install -y certbot python3-certbot-nginx

fi

# --- Configure Nginx reverse proxy ---

NGINX\_CONF="/etc/nginx/conf.d/incidenttracker.conf"

sudo tee $NGINX\_CONF > /dev/null <<EOF

server {

listen 80;

server\_name incidenttracker.example.com;

location / {

proxy\_pass http://127.0.0.1:3000;

proxy\_set\_header Host \$host;

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;

}

}

EOF

sudo nginx -t

sudo systemctl reload nginx

# --- Issue SSL certificate ---

sudo certbot --nginx -d incidenttracker.example.com --non-interactive --agree-tos -m your@email.com

# --- Test auto-renewal ---

sudo certbot renew --dry-run

echo "===== Configuring Nginx reverse proxy ====="

NGINX\_CONF="/etc/nginx/conf.d/incidenttracker.conf"

sudo tee $NGINX\_CONF > /dev/null <<EOF

# Redirect HTTP to HTTPS

server {

listen 80;

server\_name inctrack.space www.inctrack.space;

return 301 https://\$host\$request\_uri;

}

# HTTPS server block

server {

listen 443 ssl;

server\_name inctrack.space www.inctrack.space;

ssl\_certificate /etc/letsencrypt/live/inctrack.space/fullchain.pem;

ssl\_certificate\_key /etc/letsencrypt/live/inctrack.space/privkey.pem;

include /etc/letsencrypt/options-ssl-nginx.conf;

ssl\_dhparam /etc/letsencrypt/ssl-dhparams.pem;

location / {

proxy\_pass http://127.0.0.1:3000;

proxy\_set\_header Host \$host;

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;

}

}

EOF

**✅ How this works in your workflow**

* Whenever you push to main, GitHub Actions SSHs into EC2.
* It installs Nginx/Certbot if missing.
* Sets up your reverse proxy config.
* Reloads Nginx and issues SSL certificates automatically.
* Ensures auto-renewal works.

sudo nginx -t

sudo systemctl reload nginx

echo "===== Issuing SSL certificate ====="

sudo certbot --nginx -d inctrack.space -d www.inctrack.space --non-interactive --agree-tos -m chamodigunathilaka2018@gmail.com

sudo certbot renew --dry-run

#############################################

**How to verify**

1. **Check SSL certificate on EC2**

sudo certbot certificates

You should see your domain listed with valid start/end dates.

1. **Check Nginx config**

sudo nginx -t

sudo systemctl status nginx

Ensure it’s **active (running)**.

1. **Open your browser**  
   Go to:

https://inctrack.space

* You should see your app served over HTTPS.
* If the browser shows “Not Secure” → it means SSL wasn’t applied correctly, probably due to DNS or Certbot issues.

**Common issues**

* **DNS not propagated** → make sure incidenttracker.example.com points to the EC2 public IP.
* **Port 443 blocked** → check EC2 Security Group allows inbound TCP 443.
* **Certbot failed** → check logs:

sudo cat /var/log/letsencrypt/letsencrypt.log