Nginx Proxy Manager Reverse Proxy Server by Code Creator

Customer Quick Start Guide for AWS Marketplace

This guide helps you launch, access, and begin using your Nginx Proxy Manager Reverse Proxy Server by Code Creator on Amazon EC2.

1. What this server provides

This AMI provides a ready to launch Nginx Proxy Manager server for managing reverse proxy hosts, web application routing, and SSL certificates from a browser based admin interface. It is designed for customers who want a faster path to hosting and routing self hosted applications without manually writing Nginx configuration files.

- Nginx Proxy Manager web admin interface on port 81.
- HTTP proxy traffic on port 80 and HTTPS proxy traffic on port 443.
- Docker Compose based deployment with MariaDB for application data.
- First boot automation that initializes the stack and writes customer access instructions for the new instance public IP.
- Helper commands for URL display, status checks, logs, and restart operations.

2. Required AWS security group ports

Open these inbound ports for the instance. For stronger security, restrict SSH and the admin UI to your trusted IP address whenever possible.

Port	Protocol	Purpose	Recommended source
22	TCP	SSH access	Your trusted IP address
80	TCP	HTTP proxy traffic and Lets Encrypt HTTP validation	0.0.0.0/0 when serving public web traffic
443	TCP	HTTPS proxy traffic	0.0.0.0/0 when serving public web traffic
81	TCP	Nginx Proxy Manager admin UI	Your trusted IP address

3. Connect to the instance

Connect to your instance using your Amazon private key and the **ubuntu** user.

Example SSH command. Replace the key file and public IP address with your own values:

```
ssh -i your-key.pem ubuntu@YOUR_INSTANCE_PUBLIC_IP
```

4. Read the first login file

After the instance launches, the first boot setup creates a customer instruction file on the server. View it with this command:

```
cat /home/ubuntu/FIRST_LOGIN.txt
```

The file shows your current public IP based URLs and the initial Nginx Proxy Manager login information.

5. Show the access URLs

Use the URL helper command to display the admin UI and proxy traffic URLs for the current EC2 public IP address:

```
codecreator-npm-url
```

Typical access paths are:

- Admin UI: http://YOUR_INSTANCE_PUBLIC_IP:81
- HTTP proxy traffic: http://YOUR_INSTANCE_PUBLIC_IP
- HTTPS proxy traffic: https://YOUR_INSTANCE_PUBLIC_IP

6. Log in to Nginx Proxy Manager

Open the admin UI in your browser:

```
http://YOUR_INSTANCE_PUBLIC_IP:81
```

Use the initial Nginx Proxy Manager login:

Field	Value
Email	admin@example.com
Password	changeme

Important: change the default admin email and password immediately after first login. Write down the new password because you will need it for future logins.

7. Point a domain or subdomain to the instance

For normal SSL certificate issuance, use a real domain or subdomain. Create a DNS A record that points your domain or subdomain to the EC2 instance public IP address.

- Example hostname: app.example.com
- DNS record type: A
- DNS record value: YOUR_INSTANCE_PUBLIC_IP

A raw public IP address is useful for basic access testing, but public SSL certificate issuance normally requires a domain or subdomain that resolves to the server.

8. Create your first proxy host

1. Log in to Nginx Proxy Manager at http://YOUR_INSTANCE_PUBLIC_IP:81.
2. Go to Hosts, then Proxy Hosts.
3. Choose Add Proxy Host.
4. Enter your domain or subdomain in Domain Names.

5. Enter the forward hostname or IP address for the application you want to expose.
6. Enter the forward port used by the target application.
7. Save the proxy host and test the domain in a browser.

9. Request an SSL certificate

1. Open the proxy host you created.
2. Go to the SSL tab.
3. Select Request a new SSL Certificate.
4. Enter a valid email address for certificate notifications.
5. Enable Force SSL if you want visitors redirected to HTTPS.
6. Agree to the certificate terms and save.

Make sure ports 80 and 443 are open in the security group and that the domain points to this instance public IP before requesting the certificate.

10. Useful server commands

Run these commands from the Ubuntu shell after connecting with SSH.

Command	Purpose
cat /home/ubuntu/FIRST_LOGIN.txt	Show the generated first login guide.
codecreator-npm-url	Show admin UI and proxy URLs.
codecreator-npm-status	Show Docker, container, port, and local admin UI status.
codecreator-npm-logs	Show recent application logs.
codecreator-npm-restart	Restart the Nginx Proxy Manager stack.

11. Basic troubleshooting

Check service status:

```
codecreator-npm-status
```

View logs:

```
codecreator-npm-logs
```

Restart the stack:

```
codecreator-npm-restart
```

If the admin UI does not load, confirm that the instance security group allows TCP 81 from your IP address. If your domain does not work, confirm that DNS points to the instance public IP and that ports 80 and 443 are open.

12. Recommended instance sizing

The best instance size depends on traffic volume, number of proxy hosts, and whether the server handles many SSL certificates or high request volume.

Use case	Recommended instance	Notes
Testing or small personal use	t3.small	Good for evaluation and light proxy traffic.
Small business or multiple apps	t3.medium	Recommended default for most customers.
Higher traffic or many proxy hosts	t3.large or m6i.large	Use when routing more applications or serving heavier traffic.

Recommended root volume: 30 GB gp3 or larger.

13. Important security notes

- Change the default admin login immediately after first login.
- Restrict SSH port 22 to trusted IP addresses.
- Restrict admin UI port 81 to trusted IP addresses whenever possible.
- Keep ports 80 and 443 open only when this server is intended to serve public web traffic.
- Use strong passwords and keep a secure record of the new Nginx Proxy Manager administrator credentials.