Deployment Documentation

Abstract

We focused on a free, and effective deployment method. Ensuring users can access the platform from wherever-ever. Specifically focusing on England, for response time, due to the physical location of the project - although it can be set up anywhere in the world. The technology used includes: Nginx, Docker, Portainer, Namecheap (hostname), DDNS & openMediaVault (Linux based OS). We discuss the method behind deployment and ensuring the project is live at https://www.ecomon.org.uk/.

Method

Docker is a platform which packages up any application into a “box” which can be ported to any system, therefore being system agnostic. We dockerized the platform, allowing it to be deployed easily.

Adam has a machine running 24/7 at his home which serves many of the projects he has done. Using this machine was the obvious choice for this project. The machine runs OpenMediaVault, a linux based server OS, with Docker sitting on top. Therefore, once the project is archived (.tar) and scp over to the server, it can be deployed easily using Docker. The server runs Portainer, a UI built for docker, making managing projects, images and containers easier.

Firstly the docker image must be built. This is done using the docker file, along with a shell script to aid the setup of django, named entrypoint.sh. Following the docker-compose.yml, a configuration file, Portainer build a stack or Docker containers, which all relate to one another. This compose file contains each “service” and the image required. Now the project is running on the local machine.

To deploy to the internet, we purchased the domain ecomon.org.uk, where the project is hosted (https://www.ecomon.org.uk/), from Namecheap. There is an obvious issue here, how does Adam’s home’s dynamic home IP update namecheap’s DNS? This is achieved using DDNS. Here the server updates namecheap’s record at a set interval, to ensure the A+ record points to the correct IP. This was achieved within the stack.

Now the domain is sorted, let's discuss how the local project is exposed to the IP. Nginx runs on this server, where incoming requests are pointed towards the port(s) of the desired endpoint based on it’s hostname. We exposed port 9765 for this Django project. Heres the flow:

User requests https://www.ecomon.org.uk/ -> sends to server -> server points to correct container

The next challenge is ensuring HTTPS is used not HTTP. Here we used a Let's Encrypt Certificate. They are free and fairly straightforward to set up. After requesting the certificate, we configured Nginx to only allow SSL. Now all requests made will default to HTTPS.

Summary

Ecomon is deployed via a self-hosted solution using technologies including: Docker, Portainer, Nginx, OpenMediaVault (linux), Namecheap and DDNS. It enforces HTTPS with a Let’s Encrypt Certificate.