Setup and configure Vault Server on Docker

Follow the steps given below for setting up vault server on Docker

**Step 1. Create and configure an EC2 instance**

* In this case I’m using a Amazon Linux2 instance
* Follow this [link](https://docs.aws.amazon.com/AWSEC2/latest/UserGuide/EC2_GetStarted.html) to create and connect to Amazon EC2
* Install docker and docker-compose on EC2
* Create directories vault-server && vault-init

**### AMAZON LINUX 2**yum install dockersudo curl -L "<https://github.com/docker/compose/releases/download/1.28.6/docker-compose-$(uname> -s)-$(uname -m)" -o /usr/local/bin/docker-composesudo chmod +x /usr/local/bin/docker-composesudo ln -s /usr/local/bin/docker-compose /usr/bin/docker-composedocker-compose --versionmkdir {vault-server,vault-init}

**Step 2. Create IAM Role and attach to EC2**

* For this post I’m using **DynamoDBFullAccess** Policy but you can use more granular one.
* Create KMS Key and grant usage to EC2 Role.

**Step 3. Create Vault configuration file as below**

copy this file into vault-server folder

We are using DynamoDB for storage backend and AWS KMS for auto unseal.

**Step 4. Create Python script as below to initialize vault server and store unseal keys/ root token into secrets manger**

copy this file into vault-init folder

**Step 5. Build Docker image**

We are going to build two docker containers:

* Vault-Server (This runs the actual Vault Engine)
* Vault-init utility (This is a utility container to initialize vault server and store unseal keys in AWS Secrets Manager)

Copy these two docker files to their respective folders.

Vault-Server Container

Vault-Init Container

**Step 6. Create docker compose file as below**

Now your folder structure and files should looks something like below:

.  
├── compose.yml  
├── vault-init  
│ ├── Dockerfile  
│ └── vault\_init.py  
└── vault-server  
 ├── base.hcl  
 └── Dockerfile

You can now run below command to bring up the containers:

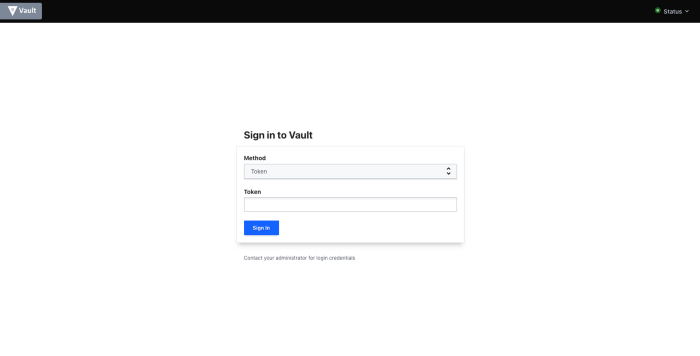
## Make sure you are running it from root folder where you have compose.yml  
**docker-compose up**  
 ## You can use this if you want to rebuild the containersdocker-**docker-compose up --build**## Finally this will bring down the containers and remove orphans  
**docker-compose down -v --remove-orphans**

**Step 7.** Now you can access vault console using the following URL

<http://127.0.0.1:8200/ui> OR[http://EC2\_IP:8200/ui](http://ec2_ip:8200/ui)

You should see something like below after access the ui:

https://miro.medium.com/max/60/1*-KuiapZPQre3rzcktVt_Hw.png?q=20



Use the root token from the secrets manager to sign in.

Conclusion

Thank you for reading. I hope you have found this useful. In the next post we will cover integrating Okta with vault and also steps to run vault on Fargate.