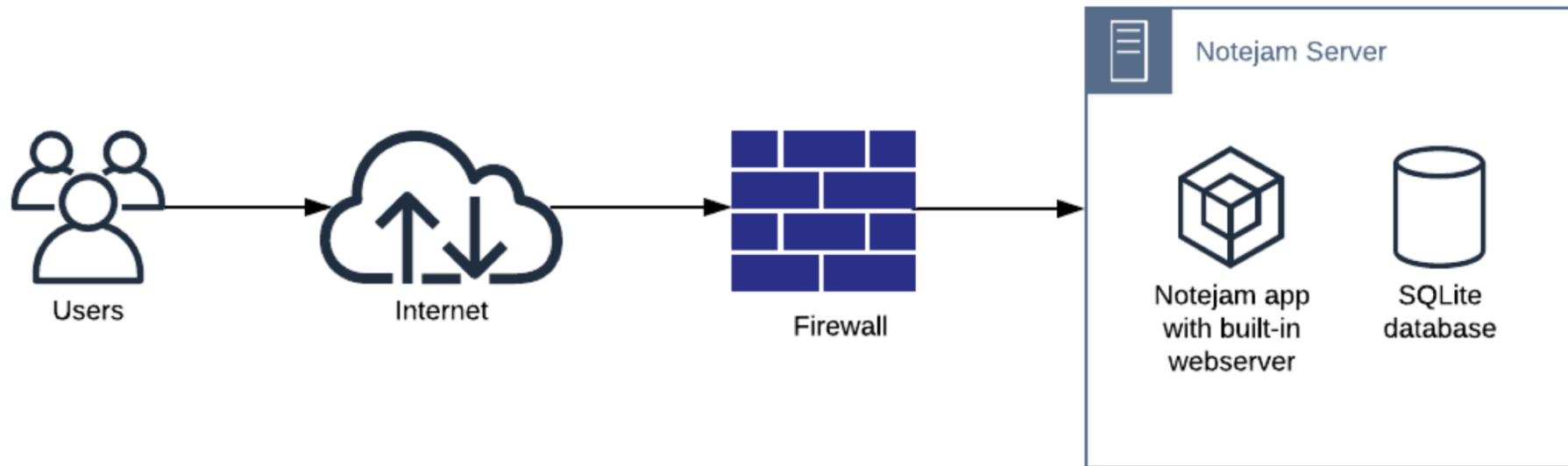
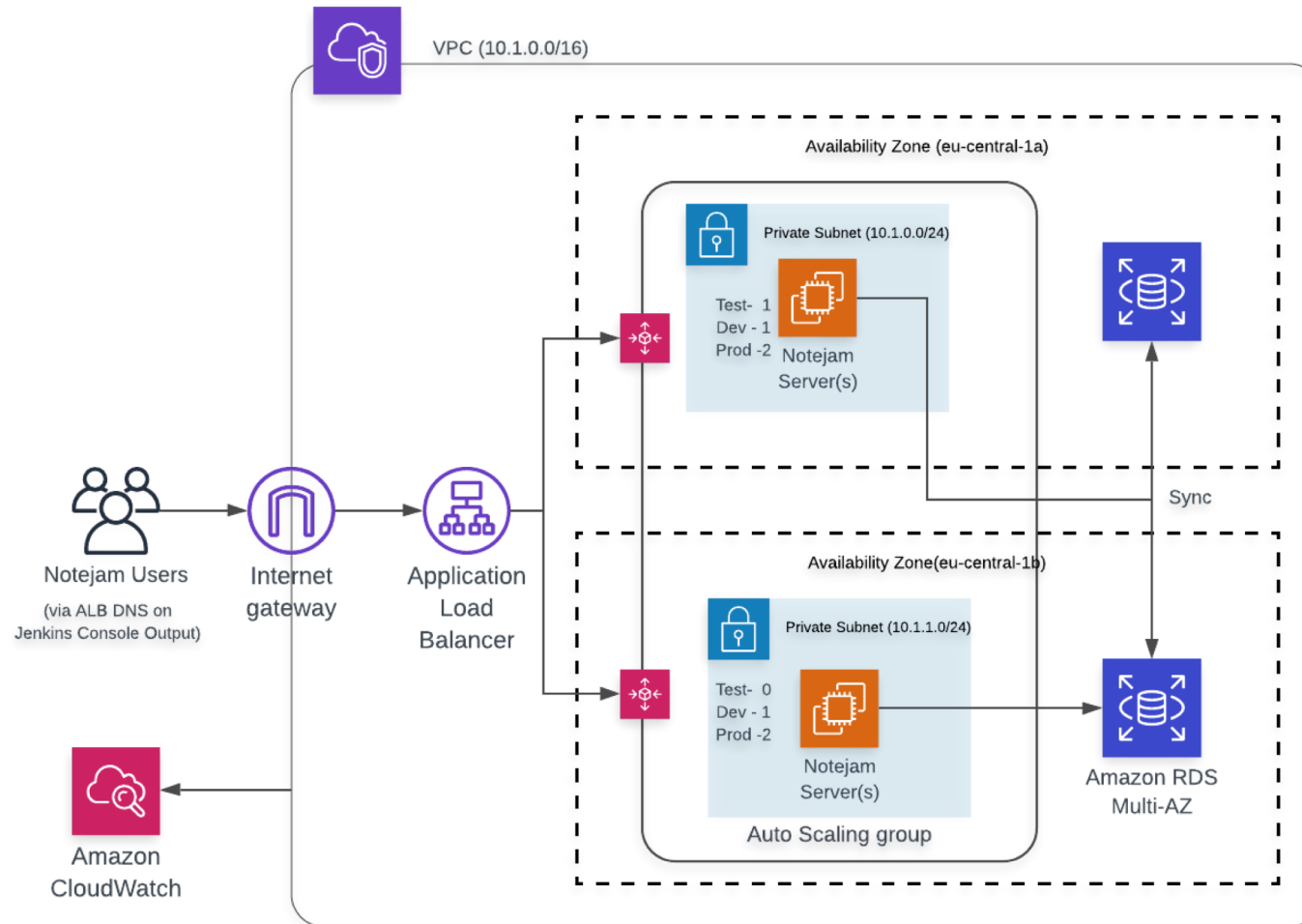


Notejam Migration to AWS

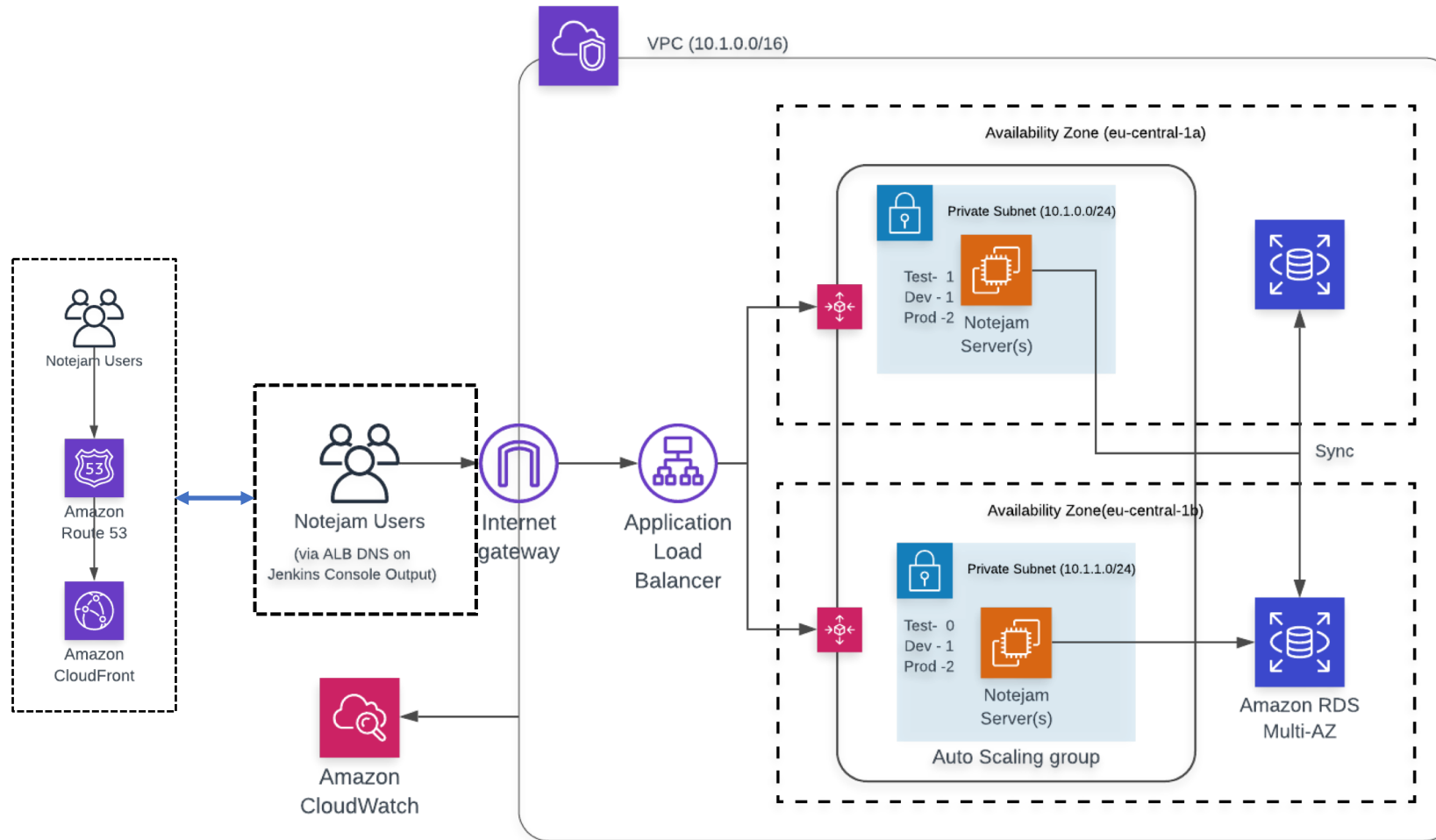
Legacy Architecture



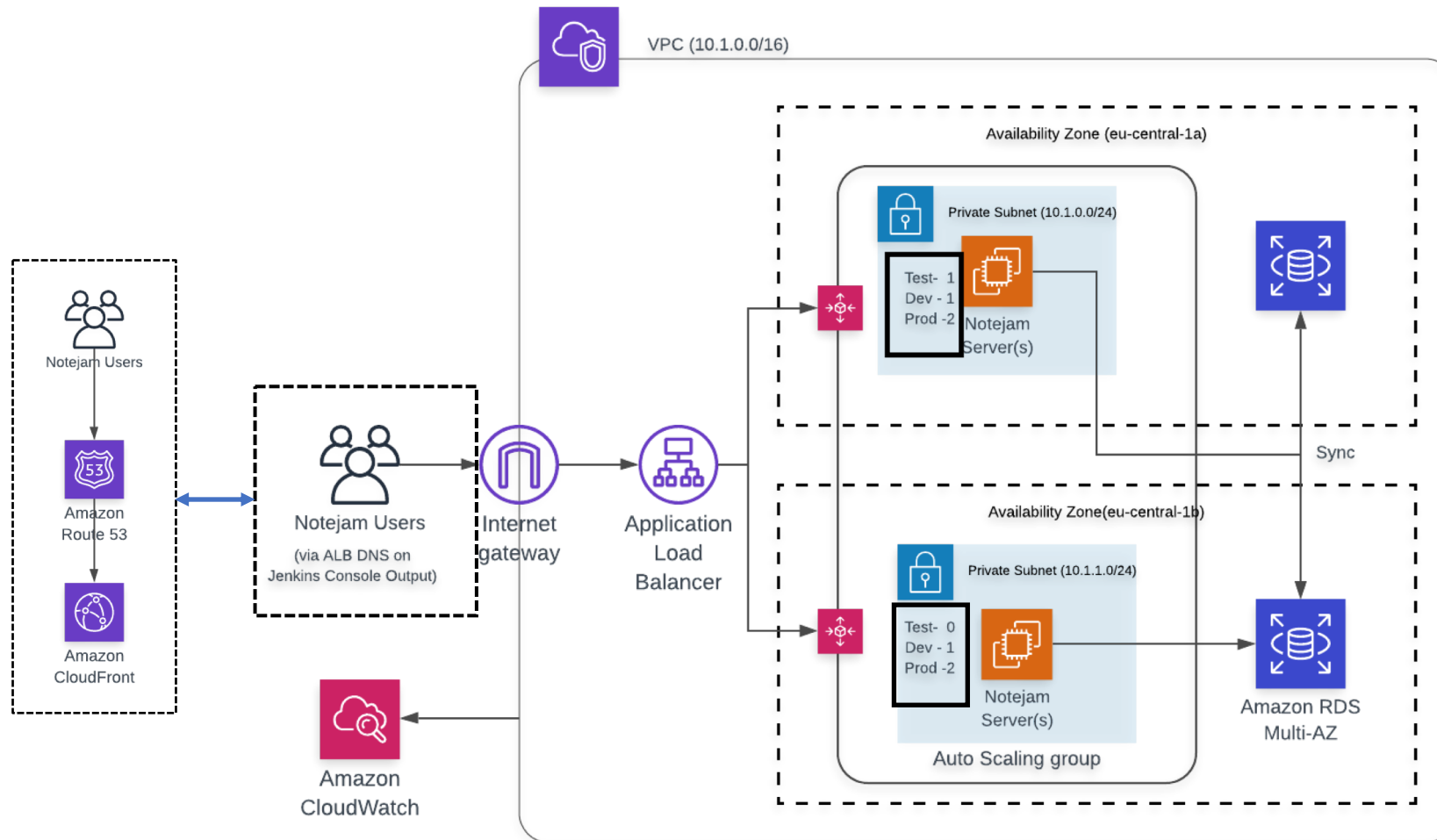
AWS Cloud architecture : General



AWS Cloud architecture : Domain Name



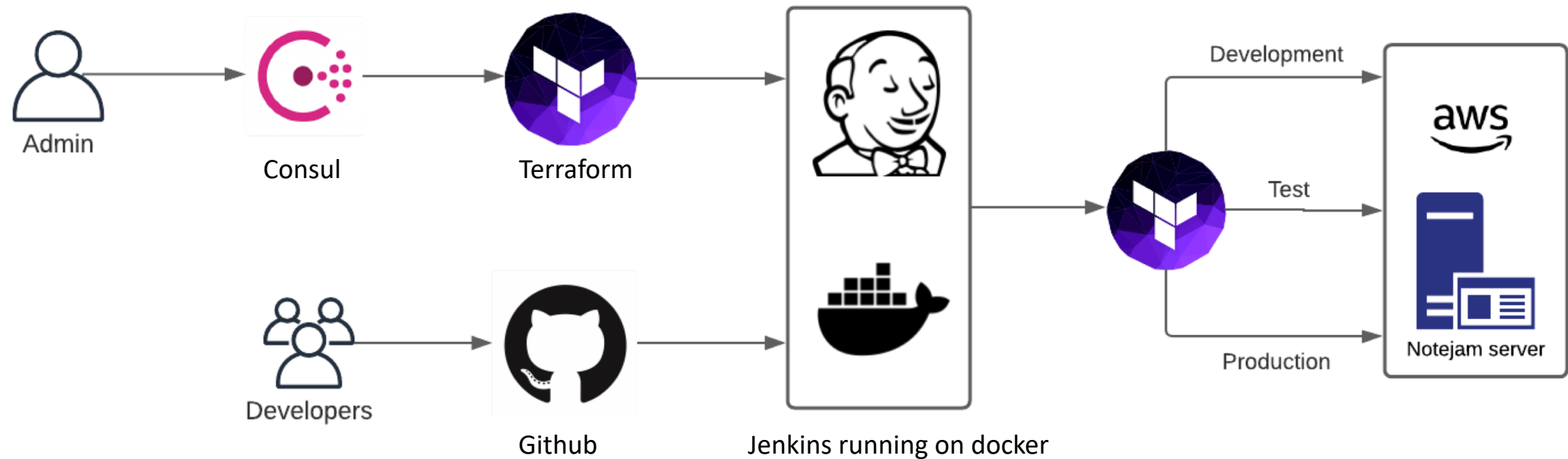
AWS Cloud architecture : Test/Dev/Prod



Benefits over the legacy architecture

- Highly Available - Scalable
- Reliable
- Secure
- Cost Optimized
- Automated Operations/Deployment

CI/CD Pipeline (Implemented and Working!)



Github link: <https://github.com/ankushbraj/Ntjm-deployment>

Usage:

- Clone repo on local computer (Mac or Linux)
- Make sure Terraform, Consul and Docker is pre-installed
- Consul Setup:
 - Go to Consul folder and `mkdir data`
 - Launch consul server instance and Run Consul in the background
 - `consul agent -bootstrap -config-file="config/consul-config.hcl" -bind="127.0.0.1"`
 - Open a new terminal and go again to consul folder
 - Generate bootstrap token : `consul acl bootstrap`
 - Export token : `export CONSUL_HTTP_TOKEN=TokenID`
 - Run Terraform commands
 - `terraform init`
 - `terraform plan -out consul.tfplan`
 - `terraform apply consul.tfplan`

Jenkins setup

- Create three tokens for Dev, test and prod
 1. `consul acl token create -policy-name=development -description="Jenkins development"`
 2. `consul acl token create -policy-name=test -description="Jenkins test"`
 3. `consul acl token create -policy-name=production -description="Jenkins production"`
- Create a Jenkins Container
 - `docker pull jenkins/jenkins:lts`
 - `docker run -p 8080:8080 -p 50000:50000 -d -v jenkins_home:/var/jenkins_home --name jenkins jenkins/jenkins:lts`
 - `docker logs jenkins`
 - Copy the admin password to <http://127.0.0.1:8080>
- Rest is to install Terraform plugin, setup Jenkins with Rep Url, link Jenkins file and pass the 3 Jenkins token, AWS creds in the global creds and you should be good to go! 😊
- Please [Uncheck lightweight checkout](#), Caused a ton of problems!!! :D
- You can destroy the provisioned infrastructure by clicking abort in final Jenkins stage!