Terraform Advanced

Hackathon!! ☺

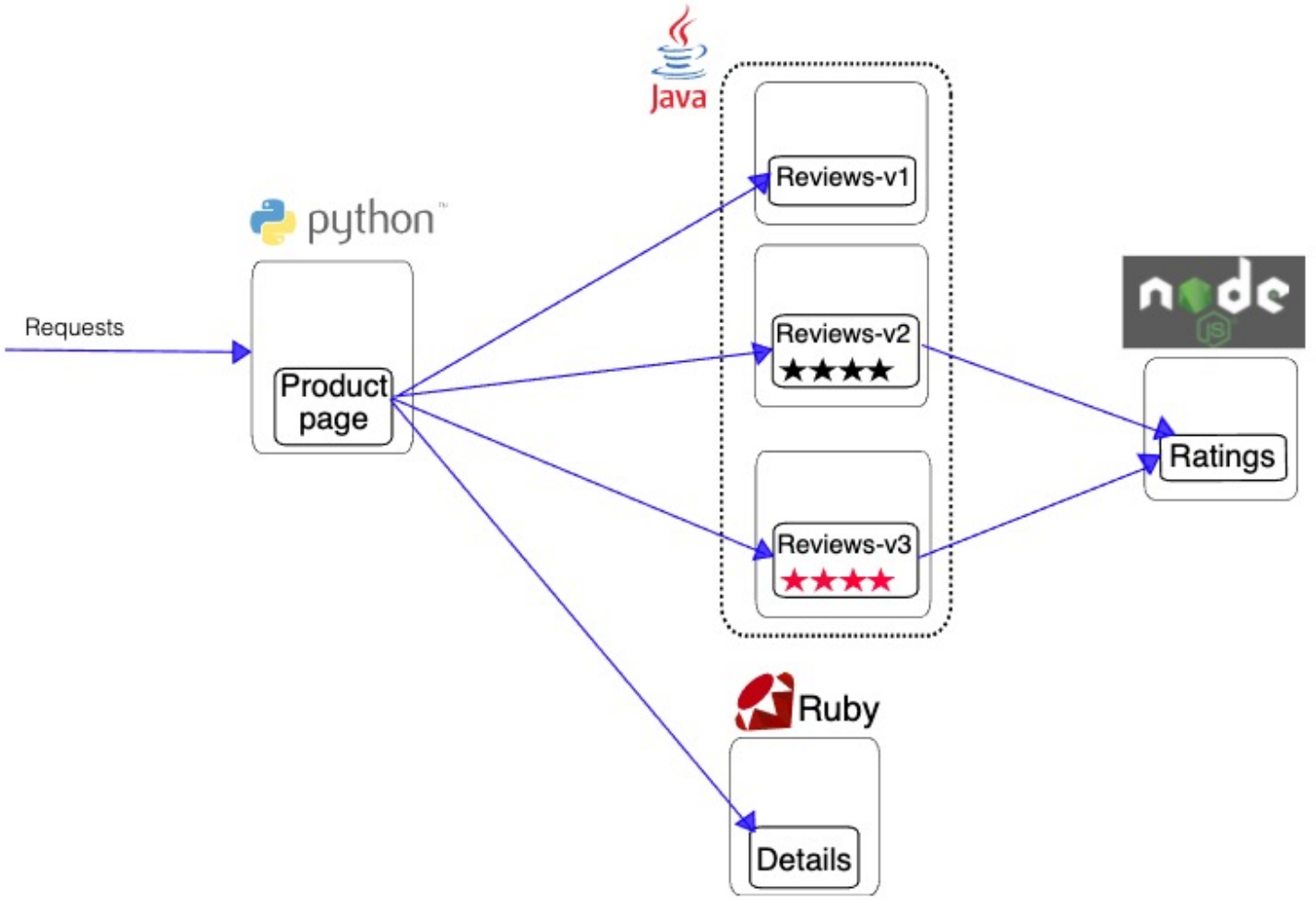
Develop Intelligence

Before we start …

Develop Intelligence

- We'll break in groups
- You have to know the basics of AWS
- There's a time cap of 3.5 hours + presentations
- You have freedom to create solution as you want
- You don't have to build the application, only the infrastructure
- There's no wrong or right, each group will explain their solution at the end
- I have a solution that I'll share and review at the end (if time allowed)
- I'll play the role of the developer that knows a little bit of AWS too
- Let's get into the details …

Bookinfo Application



Bookinfo Application

- One container per VM
- You have to use the ASG + ALB + LT combo
- Each service will run as a Docker container in a VM
- You have to pass the endpoints of all the services to the main service
- For service discovery, you have to use a private hosted zone in R53
  - Something like reviews.dev.tfadvanced.com pointing to the reviews service
- The product service (main) has to be publicly accessible

Terraform Requirements

Develop Intelligence

- Modules, modules, modules
- Once you have service running, the rest should be easy
- You should be able to deploy each service independently
- You should have more than one environment (i.e. dev and prod)
- Don’t pre-built AMIs, instead make use of the user_data attribute
- Use Terragrunt (optional, but necessary!)

Continous Delivery (Optional)

- If you have time, create a CI/CD pipeline for each module
- You could create a pipeline for the live environments as well
- Feel free to use GitLab Pipelines, GitHub Actions, or Jenkins
- The ide should be to showcase you can do a change and promote it

Details Service

- Runs on port 9080
- It has a /health endpoint for health check purposes
- Docker image: docker.io/istio/examples-bookinfo-details-v1:1.16.2
- You can test it like this
  - http://{HOST}:9080/health
  - http://{HOST}:9080/details/101

Details Service (Amazon Linux v2)

```
#!/bin/bash
sudo yum update -y
sudo yum install -y docker
sudo service docker start

sudo docker run -d -p 9080:9080 --restart on-failure
docker.io/istio/examples-bookinfo-details-v1:1.16.2
```

Ratings Service

- Runs on port 9080
- It has a /health endpoint for health check purposes
- You need to specify the following environment variables
  - SERVICE_VERSION='v3'
- Docker image: docker.io/istio/examples-bookinfo-ratings-v1:1.16.2
- You can test it like this
  - http://{HOST}:9080/health
  - http://{HOST}:9080/ratings/101

Ratings Service (Amazon Linux v2)

```
#!/bin/bash
sudo yum update -y
sudo yum install -y docker
sudo service docker start

sudo docker run -d -p 9080:9080 --env SERVICE_VERSION='v3' --restart
on-failure docker.io/istio/examples-bookinfo-ratings-v1:1.16.2
```

Reviews Service

- Runs on port 9080
- It has a /health endpoint for health check purposes
- You need to specify the following environment variables
  - RATINGS_HOSTNAME='{ALB_DNS_NAME}'
- Docker image: docker.io/istio/examples-bookinfo-reviews-v3:1.16.2
- You can test it like this
  - http://{HOST}:9080/health
  - http://{HOST}:9080/reviews/101

Products Service

- Runs on port 9080
- It has a /health endpoint for health check purposes
- You need to specify the following environment variables
  - RATINGS_HOSTNAME='{ALB_DNS_NAME}'
  - DETAILS_HOSTNAME='{ALB_DNS_NAME}'
  - REVIEWS_HOSTNAME='{ALB_DNS_NAME}'
- Docker image: docker.io/istio/examples-bookinfo-productpage-v1:1.16.2
- You can test it like this
  - http://{HOST}:9080/health
  - http://{HOST}:9080/

General Recommendations

Develop Intelligence

- Terraform documentation site is going to be of huge help!
- Feel free to use modules to reduce the HCL code
- Terragrunt will help you to DRY, try to use it
- Start simple, then continue improving
- Start with a service that has no dependencies
- Deploy and test service by service, then make it secure (private)
- Use Amazon Linux AMI, it’s going to be easier to SSH into if needed
- Go back to all the labs we’ve done, simply put all the pieces together!

Final Thoughts!

Develop Intelligence

- Have fun, this is another learning exercise
- Try to solve the problems with your team first, then call me
- It doesn't matter if you don't finish, I'm interested in what you learn
- When the time is over, each team will go back to present the solution

Good luck! ☺ … I'll be around if you need me