Tasks

1. Using Terraform, and on Oracle Cloud Infrastructure (OCI), build a hop and spoke network and all of its necessary components.
2. Provision an application VM with Nuxt.js as a frontend and Laravel as a backend.
3. Provision a database VM with MySQL, Redis and Mongo.
4. Setup an automated CI/CD deployment using Jenkins.
5. Use Ansible to install and configure all systems.

Infrastructure and Assumptions

1. One VM for setup of Terraform, Ansible, OCI CLI.
2. Using Ansible, I want to perform tasks 2, 3 and 4.
3. A second VM (db-vm) used to setup MySQL, Redis and Mongo Databases.
4. A third VM (app-vm) used to setup two applications.
   1. one frontend app using Nuxt.js
   2. one backend app using Laravel
5. Frontend and backend can be exposed using Nginx or with ingress in Kubernetes (domains and SSL certs will be required). I have included an Nginx in my Ansible playbook.
6. Database subnet is private and only accessible to the backend application. Connecting to the DB Instance on the private subnet using Ansible as a Bastion. NAT gateway is setup for direct internet access.
7. A public subnet to allow external traffic to reach the frontend.
8. In a staging and prod environment, I will use Vault for any secure configs.
9. Names of Ansible tasks are explanatory and I have added comments for the terraform tasks.
10. Terraform cannot manage the actual connection of remote peering connections in OCI at the moment because of a limitation in the OCI Terraform provider. There is no resource or API exposed by OCI that allows Terraform to directly connect RPCs across different VCNs, especially when the connections involve DRGs.

Tools

1. Terraform
2. Ansible
3. Git
4. Jenkins
5. Nginx
6. Oracle Cloud VCN (Subnets, Route tables, Internet Gateway, Network Security Group, Security Lists, NAT Gateway)