ProLUG Automation

Unit 7 Worksheet

Instructions

Fill out this sheet as you progress through the lab and discussions. Hold your worksheets until the end to turn them in as a final submission packet.

Discussion Questions:

Unit 7 Discussion Post 1: You are the team lead of a small Linux team maintaining 700 servers. Your management is always pushing for getting more from the systems and has been asking you to explore container environments, especially in the cloud. You read some blog posts about services and decide to write out your notes: https://aws.amazon.com/blogs/containers/amazon-ecs-vs-amazon-eks-making-sense-of-aws-container-services/

1. What are the major differences between container environments and Kubernetes orchestrated environments?
    a. Why might you just want a containerized environment?
    b. Why might you want an orchestrated environment?
    c. Can you compare and contrast them?

Unit 7 Discussion Post 2: Your team is having problems with a deployment. This is the code snippet they are using.

1. What is the provider they are using?
2. How many docker instance are they trying to run, and what are their names?
    a. What ports are they going to be running on?
3. Your team is having problems executing this and have brought it to you. What might you check, or do with terraform to try to resolve the issue?
    a. If it's telling you there are no providers?
    b. If it's saying there's a syntax problem (how can you find it)?
    c. If there are no resources created?

```
terraform {
  required_providers {
```

```
    docker = {
      source  = "kreuzwerker/docker"
      version = "~> 2.13.0"
    }
  }
}

provider "docker" {}

resource "docker_image" "nginx" {
  name         = "nginx:latest"
  keep_locally = false
}

resource "docker_container" "nginx8080" {
  image = docker_image.nginx.latest
  name  = "nginx8080"
  ports {
    internal = 80
    external = 8080
  }
}

resource "docker_container" "nginx8081" {
  image = docker_image.nginx.latest
  name  = "nginx8081"
  ports {
    internal = 80
    external = 8081
  }
}

resource "docker_container" "nginx8082" {
  image = docker_image.nginx.latest
  name  = "nginx8082"
  ports {
    internal = 80
    external = 8082
  }
}
```

Definitions/Terminology

Terms:

- Pipeline
- Inotify-tools

Notes During Lecture/Class:

Links:

- Packer labs: https://developer.hashicorp.com/packer
- Apptainer labs: https://ciq.com/products/apptainer/
- Docker build best practices: https://docs.docker.com/build/building/best-practices/

Useful tools:

Lab and Assignment

Unit7 Lab Container Environments- https://killercoda.com/het-tanis/course/Automation-Labs/Unit7_Container_Environments

Digging Deeper

1. What are some of the best practices around container deployments? https://docs.docker.com/build/building/best-practices/
    1. Why might we not want to ever run the “latest” tag in production?
    2. Why should an application be run as non-root?
    3. What is it to be an immutable container?
    4. What is it to be a sandboxed container?
        1. What does this mean from the kernel standpoint

Reflection Questions

1. What questions do you still have about this week?

2. How are you going to use what you've learned in your current role?