**README**

**Paul Emhoff**

**IaC Project – Dell Technologies; For Aaron Baldie**

**Job Req: R81458**

# Background

I am applying for a Systems Engineer role within Dell technologies. Aaron Baldie, the hiring manager, has asked applicants to submit an IaC Project to demonstrate abilities around deploying Infrastructure as Code via open source tools. This document, along with the attached code samples represent my submission. These documents will be put into my personal GitHub Repository prior to any interview panel discussion.

# Assumptions on what’s needed to complete the project:

* Understand the Objective
* Define the scope
* Create checklist of all ‘asks’
* Determine and line up resources/tools
* Create a document that visually demonstrates connectivity between a series of VM’s
* Create configuration files
* Demonstrate the work product

# Understand the objective:

* While I’m doing this project as a job interview, I’ll view it as a request from a customer – and pose my solution back to the customer in that vein.
* The “Customer” has asked me to use configuration files and opensource frameworks to deploy a *series* of VM’s or containers to a public/private cloud and to create these VM’s or containers at *zero* cost.
* I’m assuming the customer wants IaC because it will save them OPEX and CAPEX costs, decrease their time to value, and that IaC will provide a platform for further digital transformation and growth.
* Demonstrate the process by which the customer can manage IaC configuration files – in a logical and easily-understood way.
* The objective appears to be both process-focused and outcome or application-focused. In other words, the customers wants to see *how* IaC is done, and would like to see the code to make it work.

# Define the scope:

* Although specific tools are not being requested, I will limit the scope to certain tools and systems – mostly because those that the tools and systems that I’ve learned.
* Focus the process on a Site Reliability Engineering model

# “Asks” Checklist:

|  |  |  |
| --- | --- | --- |
| **Ask** | **Status** | **Notes** |
| Apply for position and submit resume | Done. | Awaiting Eric Boardman’s approval |
| Document assumptions of what will be needed to complete the project; Enter into README. | Done. | See above |
| Use a public source control system to store configuration files and related code | Done. | Will use my personal GitHub account. Awaiting names of panel members so I can add them as Collaborators in GitHub. |
| Cost NO money to create configurations | Done. | Using my personal AWS Free-Tier account and Terraform. Note: While the *creation* of these VM’s is free, there may be a cost to *running* them. |
| Be as creative as possible within the 7 day allotted time frame | Done. |  |
| Create a “series” of VM’s or Containers | Done. | Web, App, and DB VM’s have been spun up on AWS – although they are not running any real services. |
| “Show your work around how you leverage the configuration files to create a known good configuration and the application [process] of building up and tearing down the virtualized environment” | Done. | .TF files that begin with DEV are in Development. .TF files that begin with PROD are ready for Production (submission, in this case…). |
| Due by 04-June-2019 @ 2:00PM PST | Done. | Sent via email, June 3, 2019, 10:20PM |

# Resources and Tools

* Hashicorp Terraform (latest version)
* Linux, running in a Windows Subsystem for Linux environment
* GitHub
* Access Key’s and Secret Key’s from my personal AWS account; not cloned to GitHub
* Documentation: visual maps, README’s, annotated configuration files, and a logical naming convention to keep track of Known Good Configurations (PROD vs DEV).

# VM Map and Network Connectivity

<See below diagram>



# Known Issues

* Cannot currently connect to the Web instance via SSH. SSH key is not being accepted by the Web Instance at AWS.
* No real services are running on the Web, App, or DB servers. At this time, these instances are placeholders. Next step is to add appropriate services to them.
* Syntax of the .TF files needs to be cleaned up – to ensure consistency in the code and formatting.
* The actual Outside Interface (I/F) of the Web server may change if an instance has been Terminated via the AWS Console. The above diagram’s Web Server Outside I/F is not specified due to this.
* GitHub’s capabilities around branching, check-in/check-out need to be fully utilized. The “Archive” folder doesn’t utilize GitHub’s goal.