Curriculum Web Service Runbook

Short Description:

This website is used for looking up the information which required for wht this course offer. The web service provides a gateway to access the information vis a REST-ful API

Required Software:

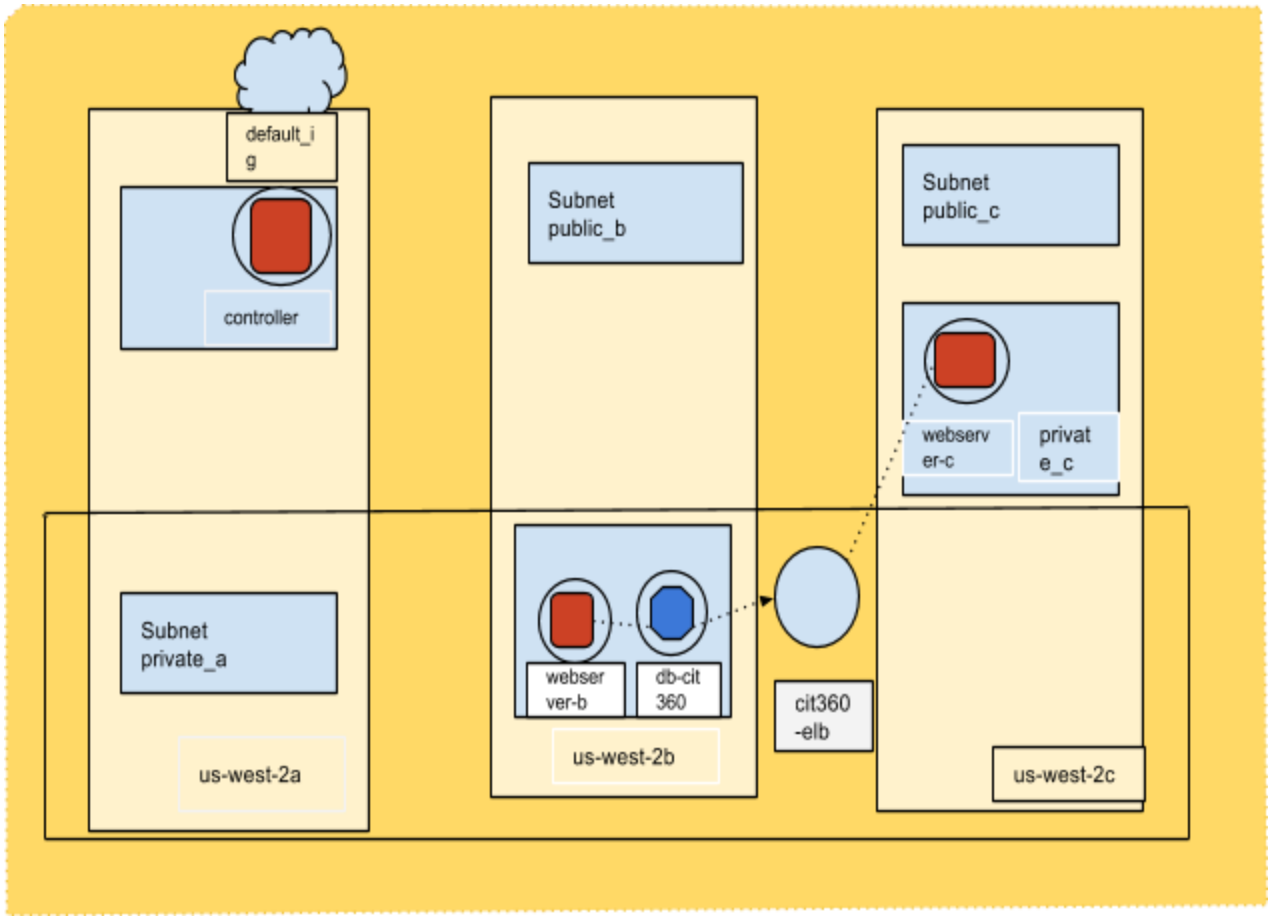
1) Amazon Web Services (AWS) account
- Accessible via provided user credentials
- Creating a new account

2) AWS Control Machine
- Ansible
- GIT
- MariaDB-client

3) Physical Machine
- AWS CLI
- GIT
- Terraform
- SSH

4) AWS Web Servers
- nginx
- php
- php-ldap
- mbstring
- phy-mycrypt
- php-mysql
- composer

Architecture Diagram



Deployment

To deploy the software, first we need to start with set up the environment on our physical machine, this action includes GIT, Terraform, AWS CLI, Deploy Terraform, Connect to the Bastion instance and deploy Ansible in purpose of configuring the web service.

Install GIT:
We need to go to https://git-scm.com/downloads, and select the installation which maches our computer version, and install it, on our terminal our Git config needs to be set as our user name.

Terraform:
First need to go to the http://www.terraform.io/downloads.html and select the appropriate package for our system, after extracting the zipped files, and choosing where we like to install it, the terraform should be add to the PATH of our system.

Amazon (AWS) Account:
For this step, first we need to already have an active Amazon account, and has the appropriate IAM group on it. After we make sure the above steps are done, we need to generate a key pair, we could do it via the web EC2 console, under the Network and Security, the key pair needs to be create with the name cit360.

AWS CLI:
For starting to set up the AWS CLI, we need to get the credentials from the AWS Website. First need to access the IAM users, choose users, choose our IAM user name, choose the security credentials tab and choose create access key. Then select show user security credential, and download the credentials.

Deploy Terraform:
First in the terminal we need to navigate to the terraform directory where the Git repo is.
$ cd path-to-file/cit-360/terraform
Then, run the $ terraform apply\
-var 'password=your -db-password' to create the infrastructure for the web service.

Connect to the Bastion instance and Deploy Ansible to Configure the web service:

This software will be deployed by using the Ansible on the control machine to install a webserver onto the EC2 instances.
First we use the key generated which we got in last step, SSH it into the bastion control instances. Second, install the Git, Third install Ansible, Forth Clone the Git redo, after that signing to AWS, going into EC2 console to get what the IP address is for each of the web servers and alter the host file.
We need to make sure that all the configuration files are in correct address.

Use the Web service:

Find the public DNS name of the ELB. this could be done either through the command line in our machine or from the AWS console from the EC2 dashboard.
Then copy and paste the address into a web browser, and we should see a working website.

Issues may be seen:
Failed to decrypt: if we run the ansible command without the correct password or without using the --ask-vault-pass this error will occur.
Fixing this error: we need to make sure that correct password used for decrypting the secrets.yml file

Connection Refused: this issue is when trying to access the website, and it displays a 500 error code saying connection refused.

Fixing this error: it could be not nginx service isn't properly running which cause this issue, at this point we need to restart with, $ sudo service ngix restart

Not reaching the service: when it is unreachable, and it gives the "Failed to connect to the host via ssh" error message.
Fixing this error: it may cause for ssh isn't properly configured between the control machine and the host. At this point we need to open the sshd_confing file on the host machine.
$ sudo vim /etc/ssh/sshd_config #use vi or nano if vim isn't installed
In the file: PublicAuthentication yes, and PasswordAuthentication ye
Next, Restart the sshd service with $ sudo service sshd restart, then on the control machine: $ ssh-copy-id *host IP address*
And at this point put the password to the host machine.