* Azure CLI basics
* Azure Portal Login
* MFA setup
* Resource Group Creation
* VM Creation
* Subscription
* Keyvault
* Storage Account
* Shared Images
* GitLab Configuration
* Code checkin/checkout
* Branch Creation
* Create merge request
* Gitlab Pipeline
* foodcritic
* chef rspec
* chef kitchen test
* packer image creation
* upload image into storage account
* Cookbook basics
* checkout git lab cookbook project into local,
* how to make cookbook run in local,
* run kitchen test in workstation,
* dependency cookbooks,
* debug cookbook errors
* ARM Template
* CLI deployment
* deployment in local
* deployment validation
* debug ARM deployment errors and logs

**Azure CLI basics**

* Workstation-logging Bastion server using Host name
* ssh IP gives access to Azure portal CLI(workstation)
* Region group is created through console azure VM is launched from azure CLI.
* Azure console is used comparatively less to CLI.

**Azure Portal Login**

* Portal Login ID with domain digitalaviationservices.com will be provided
* Initially default password is used to login the portal that needs to changed accordingly.

**MFA Setup-Multi Factor Authentication**

Once password is set authentication call should be answered followed by the PIN entry.

Authentication method can be set up to our convenience.

**Resource Group Creation**

In Microsoft azure console resource group creation is done manually through console later called through CLI commands(refers resource group,2json files) to launch A virtual machine.

Region available for us is always westUS

**VM Creation**

*az group deployment validate  --resource-group <ifs-pac-westus-rg> --template-file azuredeploy.json --parameters parameters/dev/westus.parameters.json*

GIT lab files useful for a VM are cloned in our workstation

CLI commands use Resource group and 2 json files for a VM creation

**Subscription**

Subscription is nothing but the region/environment where we can switch in our workstation and make code changes(for eg:Fedproxy dev, Fedproxy test)

**Keyvault**

To save vm passwords ,license key secrets,ash token,console password

For rg: Ifs-global-dev-kv

**Storage Account**

To upload image files and gpg

Gpg-file for importing to workstation converting to zip file fr installation of software.

**Shared Images**

Once the pipeline is triggered the application is built as an image is build shall be uploaded manully

in shared images.

images for inter-region uses

**Code checkin/checkout**

* **Git clone**-replicates the master copy in our workstation
* **Git checkout**-creates a working copy of the master branch to make code changes.
* **Git commit**-changes are made in our working copy but pipeline is not triggered.
* **Git push**-uploads the code changes in to GIT lab.A pipeline is triggered for each and every push.

**Branch Creation**

**Git branch ‘name’**-that creates a separate branch of master copy

**Create merge request**

So once code changes are committed and pushed pipeline is triggered if the test is successful then request to merge the code to master is raised to HCL team,narendra.

**Git Lab configuration-**

Own user ID and password

**3 stages of testing**

Once the pipeline is triggered three stages of testing is done.

* foodcritic
* chef rspec
* chef kitchen test

**packer image creation**

This readily available image for deploying a virtual machine.

**Upload image to storage account**

So once a commit is done and changes are pushed to GITlab a pipeline is triggered that

Generates a image of that software’s code and deploys it in to a container.The respective image

Shall be uploaded manually in the storage account.

* While launching a command from CLI refers eastus.json file and the parameters file that needs to be passes.

Command:Az account show:shows the region(subscription) that we are present now

We deploy vm inside subscription

Services for deployment

Cookbooks affects the service.

**Cookbook basics**

Cookbook is directory with variety of files like Ruby,yml,spec etc.

**PFC,PAC,PF base line,PAE,PFE** are the types of cookbooks

Every dependency in recipe file should be changed accordingly in spec file

**checkout git lab cookbook project into local,**

command: checkout “url” “dest\_path”

**how to make cookbook run in local,,run kitchen test in workstation,**

running locally(kitchen test “filename”)

**dependency cookbooks**

PFbaseline(directory) is PFE,PFC

PAC PAE are inter-dependent to each other

**debug cookbook errors**

kitchen test ”cook book name” for eg:syntax,dependency

**ARM Templates**

**4 different arm templates**

* Ping Federate Console
* Ping Access Console
* Ping Access Engine
* Ping federate Engine

**CLI deployment**

Deployment done from CLI rather than console

**Deployment in local**

Run kitchen test in workstation:

Kitchen test “file name”-that tests the file locally instead of going in to a pipeline.

**Deployment Validation**

Debugging error logs from Kitchen test output

**debug ARM deployment errors and logs(ARM-arm template)**

* Error logs can be debugged in Azure Console Issue during or in **/temp/bootstrap.log** file inside the virtual machine.
* For instance, license issue in resource group will be checked in Keyvault/’rg name’/pf license value by creating a license with the key from key-vault of the virtual machine
* If an issue is fixed, then virtual machine should be redeployed from the virtual machine not from the workstation
* Launching virtual machine creates virtual machine, network Interface, disk and key vaults.

cli command to launch a vm

vm redeoply command.