Wiki Link :

<http://oo-gitlab.nam.nsroot.net/aws_poc_team/gcb_doc-aws_discovery_poc_wiki/wikis/TechStack-Intro>

Required Cookbooks:

<http://oo-gitlab.nam.nsroot.net/aws_poc_team/gcb_chef_cookbook-aws_orchestrator>

<http://oo-gitlab.nam.nsroot.net/aws_poc_team/gcb_chef_cookbook-orchestrator_ec2_wrapper>

Steps to follow:

* Create the new repo **gcb\_environment-weedwacker** and modify the parameters as needed in the manifest.yml
  + VPC
  + Subnets
  + Key name
  + Image ID, if needed.
  + Other techstack specific paramters.
* Clone [**gcb\_chef\_cookbook-aws\_orchestrator** locally.](http://oo-gitlab.nam.nsroot.net/aws_poc_team/gcb_orchestrator-aws_orchestrator)
  + <http://oo-gitlab.nam.nsroot.net/aws_poc_team/gcb_chef_cookbook-aws_orchestrator.git>
  + <http://oo-gitlab.nam.nsroot.net/aws_poc_team/gcb_chef_cookbook-orchestrator_ec2_wrapper.git>
* Clone [**gcb\_chef\_cookbook-orchestrator\_ec2\_wrapper**](http://oo-gitlab.nam.nsroot.net/aws_poc_team/gcb_chef_cookbook-orchestrator_ec2_wrapper) locally.
  + Edit ssh\_key to use our own key in kitchen.yml
  + Set the below environment variables.
    - Export AWS\_SSH\_KEY\_ID=<JK\_AWS\_POC>
    - ~~export AWS\_DEFAULT\_PROFILE=WL3~~
    - Export ORCHESTRATOR\_NAME=<perftest>
    - Export ENV\_REPO\_NAME=<gcb\_environment-weedwacker> 🡪 name of the repo created earlier.
    - Export RUN\_MODE=<dev> 🡪 If this variable is not set, a kitchen converge will install and run Jenkins. Otherwise, the bootstrap needs to be executed manually.
  + Kitchen converge 🡪 This will spin up an EC2 instance, and install chef-DK and clone the required cookbooks from gitlab along with the other required bootstrap scripts.
  + kitchen login 🡪 login to the orchestrator EC2 instance.
    - sudo –i
    - bash-4.2# aws configure

AWS Access Key ID [None]:

AWS Secret Access Key [None]:

Default region name [None]: us-east-1

Default output format [None]:

bash-4.2#

* + - cd /orchestrator
    - chmod 700 bootstrap.sh
    - chmod 700 destroy\_stack.sh
    - export AWS\_REGION=us-east-1
    - export ORCHESTRATOR\_NAME=perftest
    - export ENV\_REPO\_NAME=gcb\_environment-weedwacker
    - edit bootstrap.sh to comment the “run\_stack.sh” 🡪 to make sure it doesn’t run any stack, until we modify the techstack manifest to comment out the unwanted stacks.
    - ./bootstrap.sh gcb\_tech\_stack-sawgrass clone 🡪 to clone the repo locally, to edit the techstack.
    - vi tech\_stacks/gcb\_tech\_stack-sawgrass/manifest.yml – to comment the unwanted.
    - Vi /orchestrator/environments/gcb\_environment-weedwacker/manifest.yml and change the key name.
    - edit bootstrap.sh to UNcomment the “run\_stack.sh”
    - set env variable “AWS\_REGION”; export AWS\_REGION=us-east-1
    - ./bootstrap.sh gcb\_tech\_stack-sawgrass
      * Issue : proxy had some invalid characters. Had to manually set the http\_proxy and https\_proxy env variables manually.

export http\_proxy=http://webproxy.wlb2.nam.nsroot.net:8080

export https\_proxy=http://webproxy.wlb2.nam.nsroot.net:8080

export HTTP\_PROXY=http://webproxy.wlb2.nam.nsroot.net:8080

export HTTPS\_PROXY=http://webproxy.wlb2.nam.nsroot.net:8080

* + - * Error : RDC was brought up in VPC different from the VPC where SGs were created. Weedwacker manifest was corrected.
      * Error : RDS on VPC2 couldn’t be connected from the orchestrator as the it was running on VPC1. Not a show stopper.
      * Error : Couldn’t download the packages to the EC2 in second VPC from the package servers hosted in the first VPC. Bootstrap failed. Show stopper.
    - ./destroy\_stack.sh perftest gcb\_environment-weedwacker gcb\_tech\_stack-sawgrass 🡪 to destroy a stack.