**Pipeline and Gitlab**

**Gitlab Project Check-in**

**Pre- requisites:**

1. Project is created by Git-master under Group
2. Project has main and test branches created
3. Project has 1 test file checked in by Git-master to unprotect the branches for developer to check-in the code.
4. SSH public key should be added to Git account.

**SSH KeyGen**

Below steps are required for SSH key generation for Gitlab and deployment

Below steps are required to generate private and public key:

ssh-keygen -t rsa -C "<your email>" -b 4096

view PrivateKey:

cat ~/.ssh/id\_rsa

view PublicKey:

cat ~/.ssh/id\_rsa.pub

**Git Check-in Details:**

Developer to check-in the code into main branch only, master should not hold any project files.

Developer to add files using Git bash or Git SSH.

Developer to clone the project folder to local.

Git Commands:

1. Add files
2. Commit changes
3. Push changes to main branch

**Pipeline Creation:**

**Pre- requisites:**

1. Project checked into Gitlab under project group in main branch
2. Concourse deployment folder and its files added to Gilab main branch

**Pipeline creation steps:**

1. Set fly target – login into pcf

fly --target manulife\_ci login --concourse-url http://concourse.sys.dev.gsdcf.manulife.com/

*Username: admin*

*Password: G$dc0ncourse*

1. Set pipeline

fly -t manulife\_ci set-pipeline --pipeline <pipeline\_name> --config pipeline.yml --load-vars-from

fly -t manulife\_ci set-pipeline -c pipeline.yml -p EFSDocumentManagement -n -l ../concourse-credentials.yml

/\* This is what you will see

pipeline created!

you can view your pipeline here: http://concourse.sys.dev.gsdcf.manulife.com/teams/main/pipelines/myPublisherPipeline

\*/

1. Unpause pipeline – un-pauses the pipeline to run

fly -t manulife\_ci unpause-pipeline -p <pipeline\_name>

C:\Concorse\fly -t manulife\_ci unpause-pipeline -p myPublisherPipeline

1. Trigger job – used to trigger individual jobs

fly -t manulife\_ci trigger-job –j myPublisherPipeline/prepare-cf-dev

1. Destroy Pipeline

fly -t manulife\_ci destroy-pipeline --pipeline <<Pipeline Name>>

fly -t manulife\_ci destroy-pipeline --pipeline myPublisherPipeline

**Issues:**

Steps to fix permission issue in folder (from 100644 → 100755):

1. cd concourse/shared/scripts -- your path to sh scripts

2. add a blank line at the end

3. git add

4. find . -name '\*.sh' | xargs git update-index --chmod=+x

5. git commit -m "chmod 755"

6. git push

**Frequent issues with Gitlab:**

1. Git Branch **issue**

$ git checkout main

error: pathspec 'main' did not match any file(s) known to git.

**Resolution1**: check if project on gitlab UI has ‘main’ branch created

**Resolution2**: create new branch using command ‘git branch main’ and then move to branch using ‘git checkout <branch name>’

**Resolution3**: if this is the first time you are moving to branch then remove the project folder from your local git repo and clone it again.

rm -rf <Project>

git clone <git url>

1. If you get error your branch is ahead of you then you need to do ‘git pull’
2. SSH key issues

**Resolution1**: Check your SSH keys are created, use below command for public and private key.

**PrivateKey**: cat ~/.ssh/id\_rsa

**PublicKey**: cat ~/.ssh/id\_rsa.pub

**Resolution2**: Check your public SSH key is added to your gitlab UI account.

**Resolution3**: If it still doesn’t work then recreate your keys using below command and add public SSH key to your Gitlab UI.

1. Generic add/commit issues:

**Resolution1**: Check if you are connected to correct git repo and branch

**Resolution2**: delete .git and .gitignore and re initiate the repo as git ‘git init’.





