Windows Jenkins-IICS implementation

# Pre-requisites:

1. Java JDK installed
2. Git and GithubDesktop installed on machine
3. Jenkins installed on localhost:8080 <https://www.jenkins.io/download/thank-you-downloading-windows-installer-stable/> (Install all recommended plugins)

Note: **Install this outside C drive, eg. Documents**

1. Admin rights on machine
2. Github Account public/ Deloitte Github account
3. IICS Source and Target Orgs (e.g. Dev and QA)

# Step I : How to change Jenkins running user (Admin rights on machine required)

**Change Running user on Jenkins:**

Windows menu à Search for Services application à right click and run as admin à go to jenkins service à right click on Jenkins à Stop service à right click on Jenkins à click on log on tab à Select ‘This account’ à Enter your machine account details (Complete Deloitte username [yourusername@deloitte.com](mailto:yourusername@deloitte.com) and Deloitte laptop password) à apply à stop service and start service

# Step II: Download IICS CLI (This will be your utilHome)

1. Go to Github repo: <https://github.com/InformaticaCloudApplicationIntegration/Tools>
2. Click on ‘Code’
3. Select ‘Download Zip’
4. Unzip above downloaded folder inside Documents e.g. within Documents\IICS
5. This folder has v1 and v2 of IICS CLI for Linux, Mac and Windows

# Step III: Check IICS CLI on local machine

1. Go to above downloaded folder for CLI
2. Open àTools-masteràIICS Asset Management CLIàv2
3. Open your machine OS folder eg. Win\_x86
4. Run cmd from this folder
5. Run commandà iics.exe –help
6. Check if it is returning all help arguments
7. If yes then that means IICS CLI is working on local machine

# Step IV: Setup Integration between Jenkins and Git

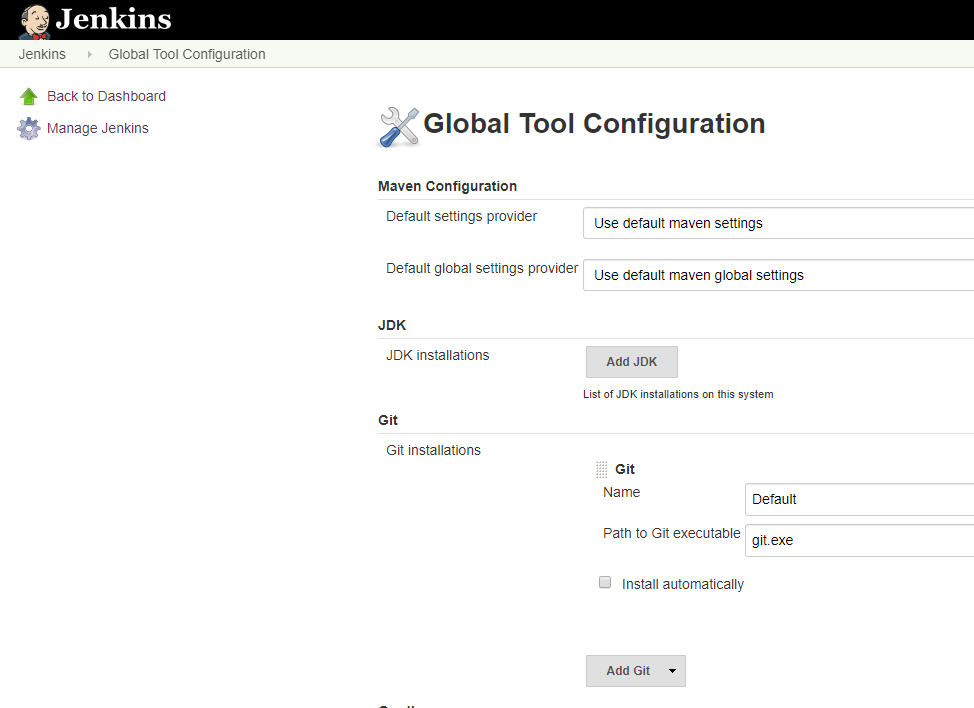
You will need to perform below steps (in the same order) to ensure that Jenkins is able to access Git repository:

Download Git command line and set it up on the machine where Jenkins is running.

Put the git.exe in the SYSTEM PATH e.g if git.exe is in C:\Program Files\Git\cmd, you need to put C:\Program Files\Git\cmd in the PATH

If Git is already installed, then perform below step:

Go to Jenkins -> Manage Jenkins -> Global Tool Configuration and enter Path to Git executable to git.exe as shown below.



Restart Jenkins Service

Use SSH Git repository access. You will need to use this URL while trying to access Git repository from Jenkins

Below snapshot is for Git repo hosted on Deloitte GitHub. Go to Code and choose Use SSH

A screenshot of a social media post

Description automatically generated

Generate SSH key where Jenkins Server is running. Follow below steps:

Generate key using this command: ssh-keygen -t rsa -b 4096 -C [your\_email@example.com](mailto:your_email@example.com).

In case you do not have SSH, download SSH. Please refer How to use ssh-keygen to generate a new SSH key

Please refer [How to use ssh-keygen to generate a new SSH key | SSH.COM](https://network.informatica.com/external-link.jspa?url=https%3A%2F%2Fwww.ssh.com%2Fssh%2Fkeygen%2F)

**Note**: Please make sure your private SSH key does not have a paraphrase, otherwise you get permission denied error in Jenkins.

For example:

**C:\Program Files (x86)\Jenkins\workspace\Backup assets tagged by release to Git>git fetch origin**

**git@github.com: Permission denied (publickey).**

**fatal: Could not read from remote repository.**

Add new SSH key to Version Control System.

For GitHub, Navigate to Profile Settings -> SSH and GPG keys and add New SSH key. Create an appropriate title and copy the contents of public key into Key section.

A screenshot of a cell phone

Description automatically generated

If using Deloitte Github Repo, Click on ‘Enable SSO’ and Click on dropdown to Authorize with Deloitte. This step is important for connecting to repository from Jenkins.

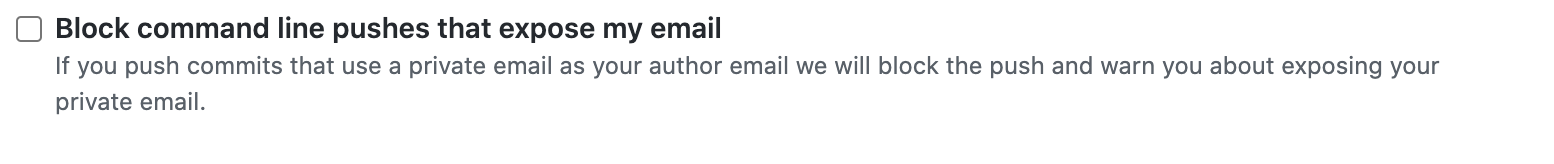
A screenshot of a cell phone

Description automatically generated

**Note: Allow command line pushes from Setting**

**Go to Github account--> Setting --> Email --> uncheck below checkbox:**

**Block command line pushes that expose my email**



Step V: Setup credentials in Jenkins

You will need to create Credentials in Jenkins through Jenkins -> Credentials -> System -> Global credentials (unrestricted) -> Add Credentials to ensure that the authentication details are secure and not exposed to the user looking at logs. We need to define three Credentials (used in the sample script)

A screenshot of a cell phone

Description automatically generated

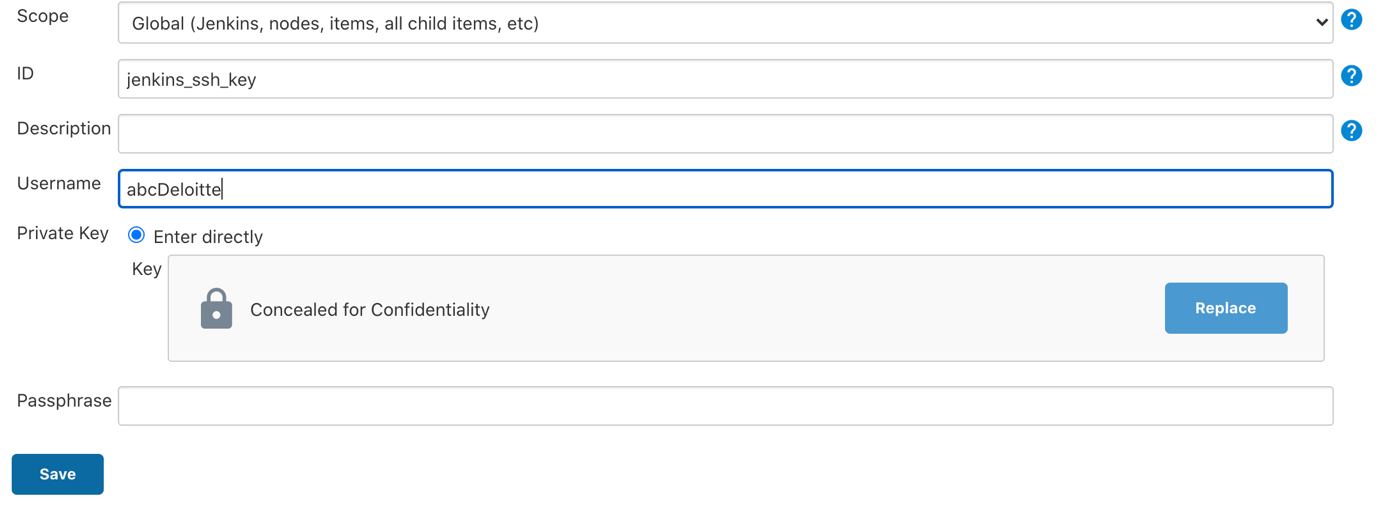
**jenkins\_ssh\_key:** This is needed to ensure that Git is able to access the Jenkins server running on a local machine using private key.

Create a credential with Kind as **SSH Username with private key**

**ID** should be **jenkins\_ssh\_key**

**Username should be the GitHub username**

Copy the contents of private key generated with the ssh-keygen command earlier (id\_rsa) into the Private Key section as given below:



**dev\_login:** This is needed to login to Development environment and export / import assets.

Create a credential with Kind as **Username with password**

Username should be the IICS Development environment username that you would want to access

Password should be the IICS Development environment password that you would want to access

**ID** should be **dev\_login**

A screenshot of a cell phone

Description automatically generated

**qa\_login:** This is needed to login to QA environment and export / import assets.

Create a credential with Kind as **Username with password**

Username should be the IICS QA environment username that you would want to access

Password should be the IICS QA environment password that you would want to access

**ID** should be **qa\_login**

A screenshot of a cell phone

Description automatically generated

Step VI: Download the Pipeline script

**Sample Jenkins Pipeline script is attached. Please copy this under the Pipeline section on Pipeline Job.**

****

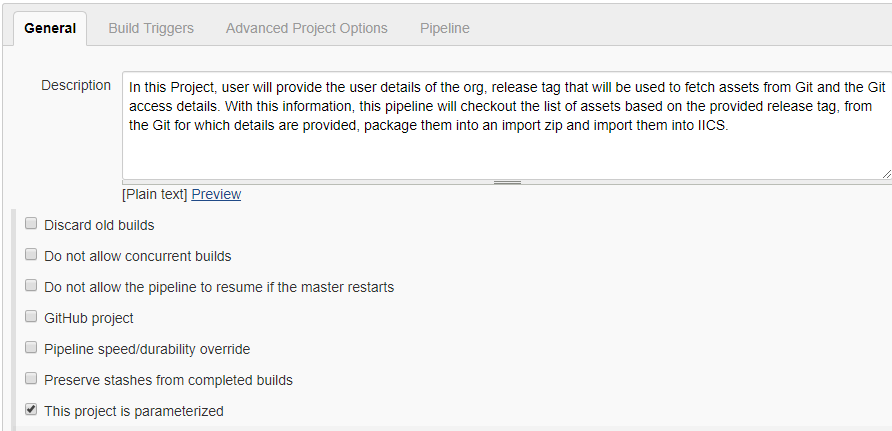
Step VII: Setup Jenkins Pipeline

**Prerequisite**: You need to have access to Jenkins server. Create a new Jenkins Pipeline and give it an appropriate name.

**Note**: You may want to reuse the Title : "Restore assets from Git based on Release Version" and Description : "In this Project, user will provide the user details of the org, release tag that will be used to fetch assets from Git and the Git access details. With this information, this pipeline will checkout the list of assets based on the provided release tag, from the Git for which details are provided, package them into an import zip and import them into IICS." for the attached script as Title and description for your script as well.

Step VIII: Setup Input Parameters

As you would have noticed, this script need user inputs. Please edit the Pipeline and mark this Project as Parameterized.



**Please setup below inputs:**

* Choice Parameter: **SourceDetails** - Which environment do you need to extract the assets from?

A screenshot of a social media post

Description automatically generated

* Choice Parameter: **TargetDetails** - Which environment do you need to deploy the assets to?

A screenshot of a cell phone

Description automatically generated

* Choice Parameter: **GitStore** - Which Git repository you want to use for Version Control? Use SSH access URL you fetched for Version Control System.

A screenshot of a cell phone

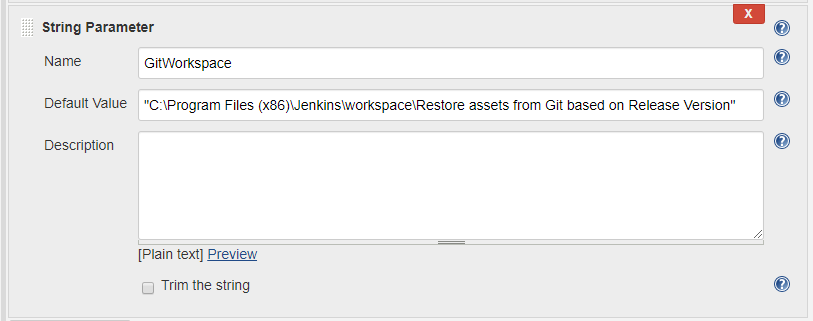
Description automatically generated

* String Parameter: **ReleaseVersion** - What is your (release) filter criteria for the assets that need to be version Controlled?

A screenshot of a social media post

Description automatically generated

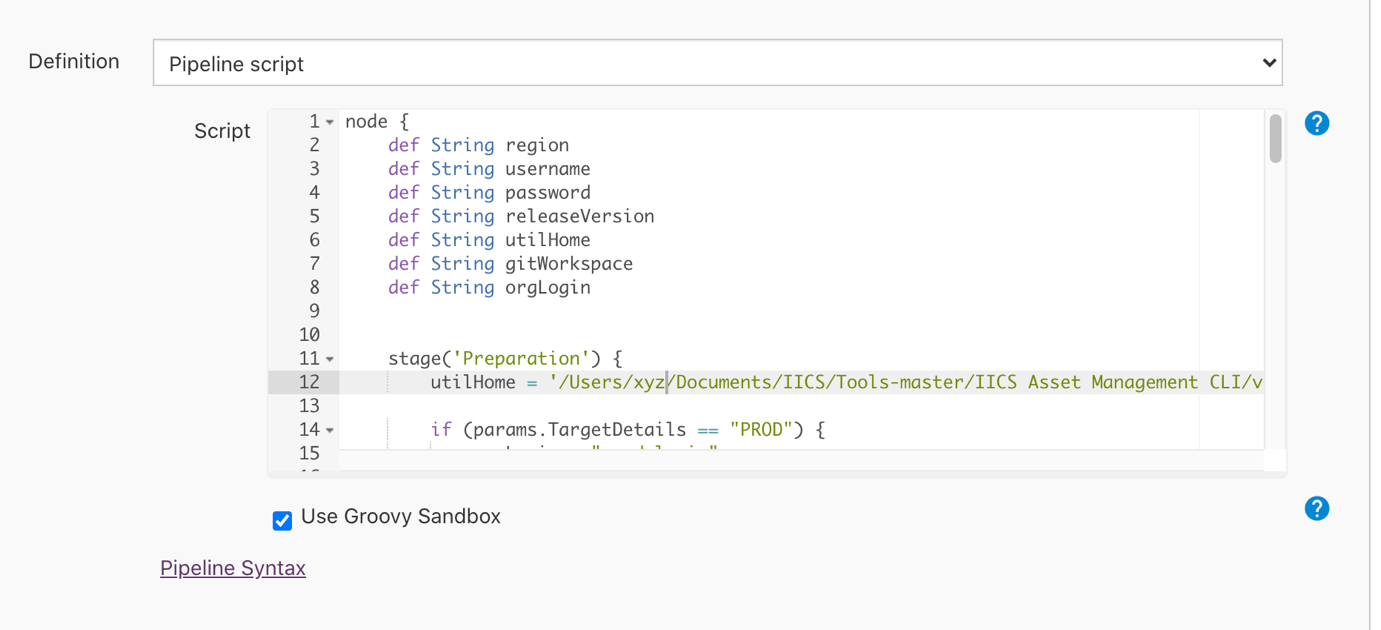
* String Parameter: **GitWorkspace** - What is your local Git Workspace where the assets be persisted locally?



Note: In addition, you will also need to modify the script:

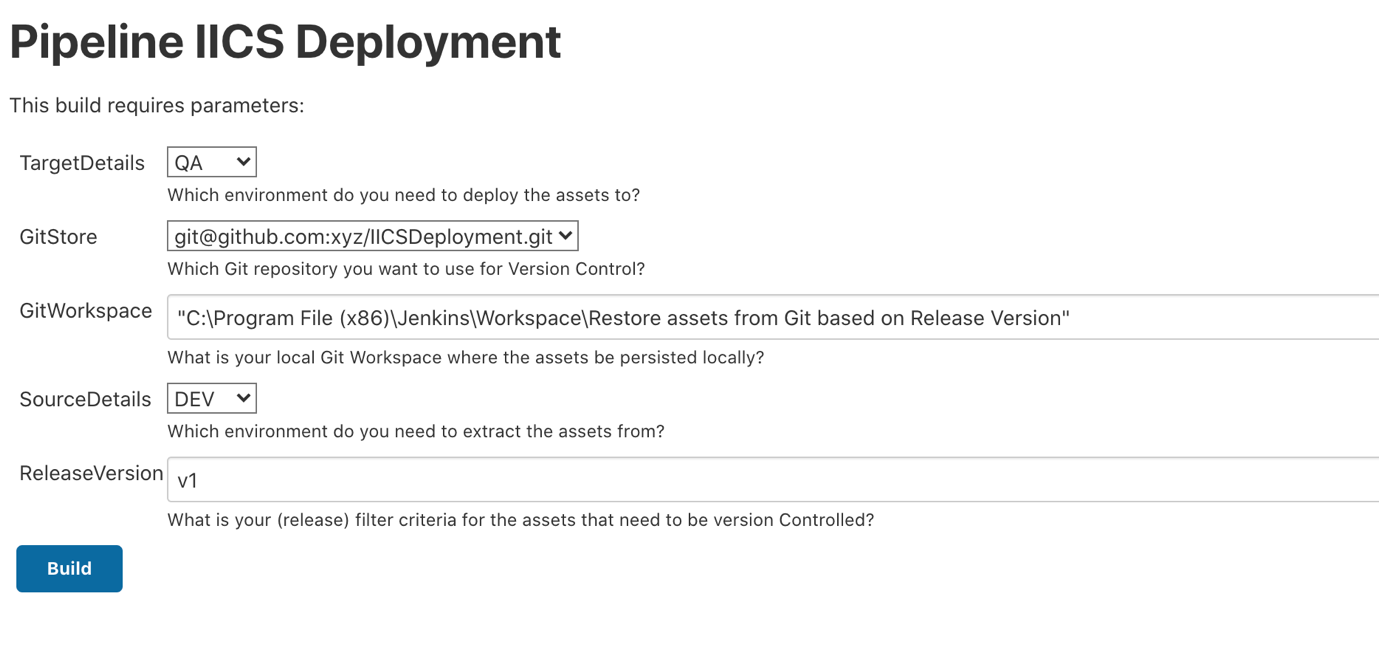
**utilHome**: This should be set to the directory where the asset management CLI v2 iics.exe is downloaded

Copy script to Pipeline section:



Step IX: Execute the pipeline

To execute the pipeline, just execute the Project with "Build with Parameters" and provide user inputs.



You should be able to see the status of execution in various stages as given below: