



GITLAB Salesforce (SFDC) CICD setup DevOps

RASHID AHMAD

Setting Up Your Project

- 1.Utilize any type of org, such as Sandbox, Production, or a free [Developer Edition](#) org.
2. Create [GitLab](#) account or use Existing one.
- 3.[Create](#) a new GitLab project or [import](#) our quick start .
- 5.[Configure](#) CI/CD environment variables in your GitLab project.
- 6.Software needed [GIT](#), [Salesforce CLI](#), OpenSSL

Script on Git

rashid@Rashid MINGW64 /d/dev/2024/Salesforce

```
$ git clone https://gitlab.com/sfdc7562228/CICD-Salesforce.git
```

Cloning into 'CICD-Salesforce'...

remote: Enumerating objects: 3, done.

remote: Counting objects: 100% (3/3), done.

remote: Compressing objects: 100% (2/2), done.

remote: Total 3 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)

Receiving objects: 100% (3/3), done.

rashid@Rashid MINGW64 /d/dev/2024/Salesforce

```
$ cd CICD-Salesforce/
```



```
MINGW64:/d/dev/2024/Salesforce/CICD-Sal
.git/      README.md  keys/
rashid@Rashid MINGW64 /d/dev/2024/S
$ cd keys
rashid@Rashid MINGW64 /d/dev/2024/S
$ openssl genrsa -aes256 -passout p
rashid@Rashid MINGW64 /d/dev/2024/S
$ openssl rsa -passin pass:gsahdg -
writing RSA key
rashid@Rashid MINGW64 /d/dev/2024/S
$ ^[[200~rm server.pass.key~
bash: '$'\E[200~rm': command not fou
rashid@Rashid MINGW64 /d/dev/2024/S
$ rm server.pass.key
rashid@Rashid MINGW64 /d/dev/2024/S
$ openssl req -new -key server.key
You are about to be asked to enter
```

Validate connection

```
sfdx force:auth:jwt:grant --client-id  
3MVG9n_HvETGhr3BkBFX08q5GHB7U8RISY9XmPg4OtX6_qnEEtMNq6EPuy3iBkaodNu7RqZToH  
mq_bUe5964v -f server.key --username rashidahmad03@abc.com -a cicdorg --instance-url  
https://rashidabc-dev-ed.my.salesforce.com/
```

Environment Variables

Env Var	Description
CONSUMER_KEY	From your JWT-based connected app on Salesforce, retrieve the generated Consumer Key from your Dev Hub org.
USER_NAME	This username is the username that you use to access your Dev Hub.
ENDPOINT	the login URL of the instance the org lives on.
DECRYPTION_KEY	server.key encryption key.
DECRYPTION_IV	server.key encryption initialization Vector.

Here are some helpful links that can assist you in understanding

<https://devcenter.heroku.com/articles/ssl-certificate-self>

https://developer.salesforce.com/docs/atlas.en-us.sfdx_dev.meta/sfdx_dev/sfdx_dev_auth_key_and_cert.htm

https://developer.salesforce.com/docs/atlas.en-us.sfdx_dev.meta/sfdx_dev/sfdx_dev_auth_connected_app.htm

https://developer.salesforce.com/docs/atlas.en-us.sfdx_dev.meta/sfdx_dev/sfdx_dev_intro.htm
