


# TASK 19

## 1. GitLab CI/CD :

GitLab CI/CD is an integrated software development tool within the **GitLab DevSecOps platform** that automates the entire software delivery process, from the initial code commit to deployment in production environments. It helps teams implement continuous methodologies including Continuous Integration (CI), Continuous Delivery (CD), and Continuous Deployment (CD).

- **.gitlab-ci.yml file:** The YAML file where pipeline logic, stages, and jobs are defined.
- **Pipelines:** The top-level components that orchestrate the sequence of building, testing, and deploying the code.
- **Jobs:** The fundamental tasks executed by a runner.
- **Stages:** Define the order in which jobs run.
- **Runners:** The execution agents that run the jobs (can be GitLab-hosted or self-managed).
- **CI/CD Variables:** Key-value pairs used to store and pass configuration or sensitive information (like API keys) securely to jobs.
- **Artifacts and Cache:** Mechanisms to store intermediate build results and speed up subsequent jobs by saving dependencies.

GitLab CI/CD persetup:

 **Ssh-keygen -t rsa** (create key pair in git bash - public key and private key)

```
Sarav@hp MINGW64 ~
$ ssh-keygen -t rsa
Generating public/private rsa key pair.
Enter file in which to save the key (/c/Users/Sarav/.ssh/id_rsa):
Enter passphrase for "/c/Users/Sarav/.ssh/id_rsa" (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /c/Users/Sarav/.ssh/id_rsa
Your public key has been saved in /c/Users/Sarav/.ssh/id_rsa.pub
The key fingerprint is:
SHA256:vqYga7V1LKkR1kwwqx33ymcJj2jjXScnT0IbtlfzYZk Sarav@hp
The key's randomart image is:
+---[RSA 3072]-----+
|  o.                  |
|  o.                  |
| o+.                 |
| 0000.              o |
| . . . .oS   o E    |
|  oo+Oo= . + .      |
| ..==+O& = .        |
| . =O+  +. @         |
| .. . oo. .          |
+---[SHA256]-----+
```

## Private key

```
Sarav@hp MINGW64 ~  
$ cat ~/.ssh/id_rsa  
-----BEGIN OPENSSH PRIVATE KEY-----  
b3B1bnNzaC1rZXktdjEAAAABG5vbmUAAAABbm9uZQAAAAAAAAABAAABlWAAAdzc2gtcn  
NhAAAAAwEAAQAAAYEazBOQUH/kk1K+1VwdtFM6+x5eBmfefX0O1/BqOPE45U7KiK+UHRM  
YP35DrWhH9fQ2GMbVoAcncGnu7oRXBe8pBwcnqzset6o0/V7xmfc1PguwKuId12FS92/  
7wkjKDBVSGZMSQg0NcEK4LV07SyzPrabeMGiasJQ2ks/NcO4HYqHsJv/Vv1UNno1uPNR05  
2pWCDSSqChUvCV2U20v/yELi00zBGa1JyGIB1YAwfrLyoZKwps7j0tGP+TMSwuo1D+v5Na  
ezdhs9ZBpDiiE3t2U9DqZ3nY1iBqWE1d++KnAqUs/DFwKha83Fj0t0YCsRT3WziFDB8UHM  
u3Ecj42ABDvJQishZOWRBtcm1pbXB9G76kk25RS/1ioEni1FA47HANIVbyP0LXIAuoSG5G  
32bmHs1WiumMcIAqyWH70zqcid7uwOr4ELYPH5PO5HIbpv/w7fv8LtJvmtkS0EK+shnTYG  
25t1VTyOuZUssu2u0TRW49Ex7ITy586YXAA3S9FjAAAFgFhEnuNYRj7jAAAAB3NzaC1yc2  
EAAAGBAMwTqrh/5JJSvtVvnbRTovseXgZn3n19Dtfwajjx00Voyoiv1B60TGD9+Q61oR/X  
0NhjG1aAHDXBp106EVwXvKQCHj6p87HreqNP1e8ZnwpT4LsCriInZdhUvdv+8JCZAwwUhs  
zEkINDXBcuC1a00ssz62m3jBomrCUNpLPzXDuB2Kh7Cb/1b5VDZ6jbjazudqVgg0rkgoV  
Lw1d1Njr/8hC4tdSwRmtSchAdwAFn6y8qGS1quu49LRj/kzEsLqNQ/1eTwns3YUvWQAQ4  
ohn7d1PQ6md52NYgalhNXfvipwK1LPwxcCOWvNxyZrdGAReU91mYhQwFfITLtxHI+NgAQ7  
yUIrB2TSEqbXJpaw1wfrU+pJNuUuv9YqBJ4pRQ00xwJyL28j9C1yALqEhUrT9mzIUpviLp  
jHIGEMlh+9M6nIne7sDq+BC2KR+TzuRyG6Vf8031fc7Sb5pLutBCvrIZ02Bk+bdVU8jrmV  
LLLtrjK0VuPRMeyE8ufOmFwAN0VRYWAAAAMBAAEAAAGAF8LiDID6TXShoQdK9B1iF1hoFo  
PIgAv47Soit+b7n0zDoD1Gocc/GBFPgmrwy72xJMDFWjh92onQaxXrF4dGq5pVABRXd4w/  
cexwba/UrhB+G3LVxkbtZiGcdFmxYOSv+VrnfHLBRhyoXON80859X144Ns2W6SSU6UDhay  
JSHBXOQa+xvuWGMKGchhXAXI996v16yasQ8K5EFRm++ag2K/cTDwhso19jNor9c60U8w5H  
LzuGLCLS2/pn17geoxs/GldgGu71817sYCUQo9Pb4LcWLGgA9cP8VpVmh28VjuGuwnM8YZ  
43FEh+Xmmn9hXkOCuI+FpViw4BND2BBka7myZzzSkwfJATORpmbq2YYVaY1JZYS2FZ41KB  
R1YnFGaGFswW8uB7LYOB+yJ8E9JR02JeohvEjv8N6wdnD06LPJ0Dhw3k20ardM4pswIEHg  
9mDT3Q20mnXu2i8xkQAigrXADLa0s00gIQEQowi8kiafwqHSCd2W6sZUOYiGQG960hAAAA
```

## Public key

```
Sarav@hp MINGW64 ~  
$ cat ~/.ssh/id_rsa.pub  
ssh-rsa AAAAB3NzaC1yc2EAAAADAQABAAQGDME6q4f+SSur7VVZ20Uzr7H14GZ959fQ7X8Go48Tj1TsqIr5QetExg/f  
k0taEf19DYxtWgBw1wadTuhFcF7ykHBYeqf0x63qjT9XvgZ8KU+c7Aq4iJ2XYVL3b/vCQmQMFVIBmxJCDQ1wQrgtWjTL  
M+tpt4waJqwlDaS281w7gdioewm/9w+VQ2eiw482s7na1YINKyokFS8JXZTY6//IQUlQ7MEZrUnIYGHVgBZ+svKhkpa1Lu  
PS0Y/5MXLC6jUP5Xk1p7N2FL1kGkOKITE3ZT0opnedjWIGpYTY374qcCpSz8MXAQFrzcWm63RgKXFPdzmIUMHxSEY7cRyP  
jYAE081CKwdk7BEG1yaW1tch0bvqSTb1FL/WKgSeKUUDjScCci9vI/QtcgC6hIbkbFzsyFKVYi6YxyIBDJYfvTOPYJ3u7A  
6vgQtikfk87kchulX/Dt9Xwu0m+as1LQQR6yGdNgZPm3VVPI651Syy7a45NFbj0THshPLnzphcADdLOWM= Sarav@hp
```

## Update the ssh key into your gitlab account (past your public key)

The screenshot shows the GitLab User Settings page for the user 'Sarav@hp'. The 'SSH Keys' section is active, displaying the details of an SSH key. The key is named 'SSH Key: Sarav@hp' and is of type 'Authentication & Signing'. It was created on 'Dec 10, 2025 9:08am' and has never been used. The key expires on 'Dec 10, 2026 12:00am'. The public key is displayed as a long string of characters, and the fingerprints for MD5 and SHA256 are also shown.

| Usage type               | Created             | Last used | Expires              |
|--------------------------|---------------------|-----------|----------------------|
| Authentication & Signing | Dec 10, 2025 9:08am | Never     | Dec 10, 2026 12:00am |

**SSH Key**

```
ssh-rsa AAAAB3NzaC1yc2EAAAADAQABAAQGDME6q4f+SSur7VVZ20Uzr7H14GZ959fQ7X8Go48Tj1TsqIr5QetExg/fk0taEf19DYxtWgBw1wadTuhFcF7ykHBYeqf0x63qjT9XvgZ8KU+c7Aq4iJ2XYVL3b/vCQmQMFVIBmxJCDQ1wQrgtWjTLLM+tpt4waJqwlDaS281w7gdioewm/9w+VQ2eiw482s7na1YINKyokFS8JXZTY6//IQUlQ7MEZrUnIYGHVgBZ+svKhkpa1LuPS0Y/5MXLC6jUP5Xk1p7N2FL1kGkOKITE3ZT0opnedjWIGpYTY374qcCpSz8MXAQFrzcWm63RgKXFPdzmIUMHxSEY7cRyPjYAE081CKwdk7BEG1yaW1tch0bvqSTb1FL/WKgSeKUUDjScCci9vI/QtcgC6hIbkbFzsyFKVYi6YxyIBDJYfvTOPYJ3u7A6vgQtikfk87kchulX/Dt9Xwu0m+as1LQQR6yGdNgZPm3VVPI651Syy7a45NFbj0THshPLnzphcADdLOWM= Sarav@hp
```

**Fingerprints**

| MD5   | SHA256                                      |
|---|---|
| dc:f3:7a:cf:9d:a6:21:09:0d:9e:c3:3f:ce:8a:c7:47 | vqYga7V1LKKR1kwwqx33ymcJj2jjXScnT0IbtlfzYZk |




## Result :

User Settings / SSH Keys

Search settings

### SSH Keys

SSH keys allow you to establish a secure connection between your computer and GitLab. SSH fingerprints verify that the client is connecting to the correct host. Check the [current instance configuration](#).

| Your SSH keys  1 |   |                          |               |           |            | <a href="#">Add new key</a>  |
|---|---|--------------------------|---------------|-----------|------------|--|
| Title   | Key   | Usage type               | Created       | Last used | Expires    | Actions  |
| Sarav@hp  |  dc:f3:7a:cf:9d:a6:21:09:0d:9e:c3:3f:ce:8a:c7:47 | Authentication & Signing | in 10 seconds | Never     | 2026-12-10 | <a href="#">Revoke</a>  |

## Create IAM user in your aws account

### Users (1) [Info](#)

An IAM user is an identity with long-term credentials that is used to interact with AWS in an account.



Delete

Create user

Q Search

< 1 > 

| <input type="checkbox"/> | User name             | Path | Group | Last activity | MFA | Password age | Console last sign-in |
|--------------------------|-----------------------|------|-------|---------------|-----|--------------|----------------------|
| <input type="checkbox"/> | <a href="#">saran</a> | /    | 0     | -             | -   | -            | -                    |

## Generate the access key and secret access key in your IAM user


### Retrieve access keys [Info](#)

#### Access key

If you lose or forget your secret access key, you cannot retrieve it. Instead, create a new access key and make the old key inactive.

Access key

Secret access key

 AKIA4O3W5DSTTKKGV7O

 \*\*\*\*\* [Show](#)

#### Access key best practices











- Never store your access key in plain text, in a code repository, or in code.
- Disable or delete access key when no longer needed.
- Enable least-privilege permissions.
- Rotate access keys regularly.

For more details about managing access keys, see the [best practices for managing AWS access keys](#).

## Update the gitlab project variables into IAM access key and secret access key

### Project variables

Variables can be accidentally exposed in a job log, or maliciously sent to a third party server. The masked variable feature can help reduce the risk of accidentally exposing variable values, but is not a guaranteed method to prevent malicious users from accessing variables. [How can I make my variables more secure?](#)

| CI/CD Variables </> 2  |   | Reveal values   | Add variable  |
|--|---|---|---|
| Key ↑  | Value   | Environments  | Actions   |
| AWS_ACCESS_KEY_ID <br>AWS_ACCESS_KEY_ID<br>Masked         | .....  | All (default)  |   |
| AWS_SECRET_ACCESS_KEY <br>AWS_SECRET_ACCESS_KEY<br>Masked | .....  | All (default)  |   |