



JJ Tech

## AWS - EC2 Image builder POC

⌚ Status	READY TO PULL	<input checked="" type="checkbox"/>	👤 Assignees	DE
📅 Dates	Empty		📌 Priority	Normal
🕒 Time Estimate	Empty		⌚ Sprint Points	Empty
🏷️ Tags	aws		🔗 Relationships	Empty

Add description

Details Subtasks Action Items

1. Review of Amazon EC2 Image Builder
2. Console hands on EC2 Image Builder
3. Create an windows server from the new image
4. RDP in windows server
5. Session into windows server
6. Integrate with Patch Manager. We will create a patch baseline and use run command
7. Introduction to Gitlab
8. Hands-on creating a gitlab account
9. Deploying image builder infrastructure build through terraform using gitlab CICD

## Build and Distribute Golden Images using Amazon EC2 Image Builder

1. First, we will give you a brief overview of the Image Builder concepts.

+1 (410) 8887049  
contact@jjtechinc.co  
www.jjtechinc.co  
14103 Hammermil Field Dr Bowie MD 20720

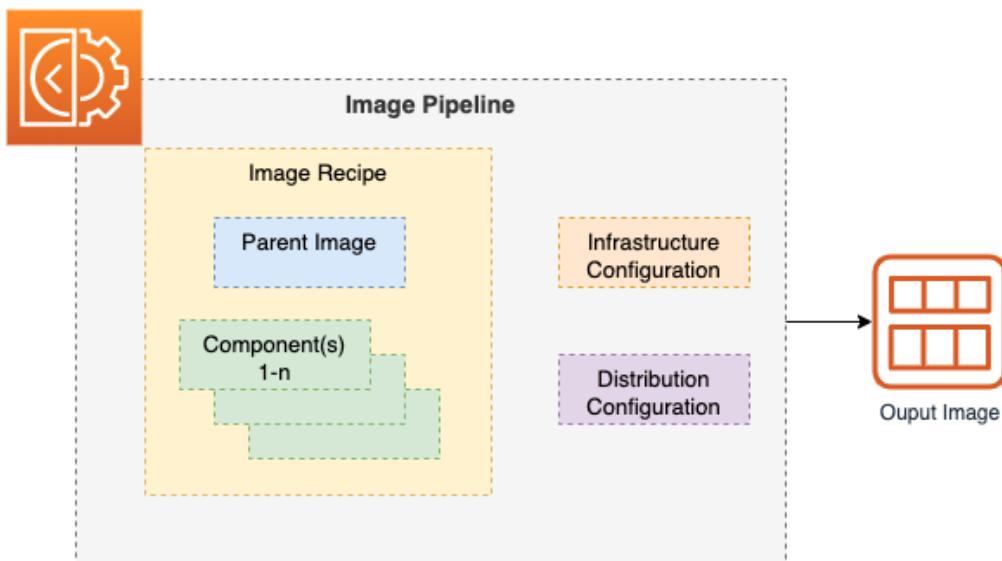


JJ Tech

2. Second, you will **create your first EC2 Image Builder Pipeline** using the [AWS Console](#). This will get you hands-on experience with the core concepts of EC2 Image Builder.
3. Third, you will **create your first EC2 Image Builder Pipeline** using the [gitlab CI/CD](#).
4. Finally, you'll learn how to develop your own components.

## 1.1 Concepts

- Let review a few concepts



### 1.1.1 Image Pipeline

The **main component** is the **Image Pipeline** itself.

A pipeline needs at least an **Image Recipe** and an **Infrastructure Configuration**.

### 1.1.2 Image Recipe



JJ Tech

The **Image Recipe** is a versioned resource that **contains a Parent Image AND one or more Components** to build your image.

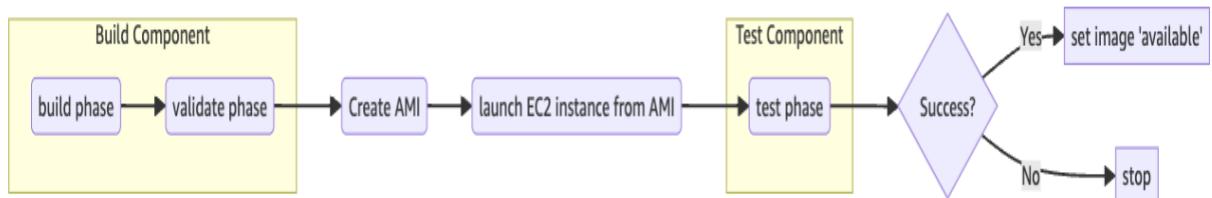
## Parent Image

The EC2 Image Builder "parent" image of the image recipe. It can either be:

1. The Amazon Resource Name (ARN) of an EC2 Image Builder image.
  - o Format: `arn:aws:imagebuilder:${AWS::Region}:aws:image/${ParentImageName}/${ImageVersion}`
  - o Example: `arn:aws:imagebuilder:eu-west-1:aws:image/windows-server-2019-english-full-base-x86/2020.x.x`
2. Or an Amazon Machine Image (AMI) ID.
  - o Format: `ami-0123456789abcdefg`

## Components

Components are the **building blocks** that are consumed by an image recipe, for example, packages **for installation, security hardening steps, and tests**. You **define one or more component(s)**, that describe how to build, validate, and test your image.



### 1.1.3 Infrastructure Configuration

Image Builder will use an EC2 instance to build your images. An infrastructure configuration defines the environment in which your image will be built and tested.

#### Info

The only required fields are:



JJ Tech

- Name
- Instance Profile Name

Other configuration such as `securityGroupIds` and `subnetId` are optional. If you do not provide these, EC2 Image Builder will use the default VPC.

## 1.1.4 Distribution Configuration

A distribution configuration can be used to share an image. You distribute your image to selected AWS Regions, accounts or organization and give `LaunchPermission` to AWS Accounts after it passes tests in the image pipeline.

A Distribution Configuration is **optional!** It's mutable, and **NOT** versioned.

### Hands-On

## 1.2 Create EC2 Image Builder Pipeline

Now the hands-on exercise workshop will start. This module consists of **two main parts**,

1. **First**, we will guide you through **creating** your first **EC2 Image Builder Pipeline** using the [AWS Console](#).

This will make you more familiar with the core concepts of EC2 Image Builder.

2. **Second**, we will guide you in **building your own EC2 Image Builder Components**.

### 1.2.1 Creating an Image Pipeline



JJ Tech

Navigate to the [EC2 Image Builder Console](#)

1. In the top search bar, search and select EC2 Image Builder

The screenshot shows the AWS search interface with the search term 'EC2 Image Builder' in the bar. Below the search bar, there are three main categories: Services (38), Features (71), and Resources (New). On the right side of the search results, there is a pinned service card for 'EC2 Image Builder' with a star icon.

**Tip!**

You can click the to favorite the service. This will pin it in the top navigation bar.

As we will be coming to this page a lot, I recommend you favorite it.

This screenshot shows the AWS services page. The 'EC2 Image Builder' service is visible in the list and has a star icon next to it, indicating it is favorited.

## Specify pipeline details

### General

#### Pipeline name

JJTech-Image-Pipeline

Enter a name that has not been used. Maximum 70 characters. Letters, numbers, spaces, -, and \_ are allowed

#### Description

Enter description

Maximum 1024 characters.

#### Enhanced metadata collection (for AMI only)

EC2 Image Builder uses AWS Systems Manager Inventory to collect additional information about the images you create. This information is verified before the creation of an image to ensure compatibility between components and images.

**Enable enhanced metadata collection**

If you deselect this box, the collection of additional information from images created for this pipeline will be disabled.

+1 (410) 8887049

contact@jjtechinc.co

www.jjtechinc.co

14103 Hammermil Field Dr Bowie MD 20720



JJ Tech

**Build schedule** [Info](#)

Schedule options

**Schedule builder**  
Automatically run the pipeline using a job schedule. The default schedule is every Monday at 9:00 UTC.

**CRON expression**  
Automatically run the pipeline using a syntax that specifies the time and intervals to run it.

**Manual**  
The pipeline will run when you initiate it.

Run pipeline every **Week** on **Monday** at **09:00** Time zone **UTC**

Dependency update settings  
Choose how you would like to apply the build schedule when a dependency, such as an image or component, is updated.

**Run pipeline based on schedule**  
The pipeline will run according to the regular schedule.

**Run pipeline at the scheduled time if there are dependency updates**  
If you choose this option, you must use semantic versioning (x.x.x) for your components, and always use latest for the base image.

This applies to the following image recipe dependencies:

## 1.2.2 Choose recipe

An image recipe is a document that defines the components to be applied to the base images to create the desired configuration for the output image. After a recipe has been created, it cannot be modified. A new version must be created in order to change components.

**Choose recipe**

An image recipe is a document that defines the components to be applied to the base images to create the desired configuration for the output image. After a recipe has been created, it cannot be modified. A new version must be created in order to change components.

Recipe

Configuration options

Use existing recipe  Create new recipe

Image type  
Choose the image type

Output type

Amazon Machine Image (AMI)   
 Docker image 

+1 (410) 8887049  
contact@jjtechinc.co  
www.jjtechinc.co  
14103 Hammermill Field Dr Bowie MD 20720



JJ Tech

**General**

Enter a name and description for this configuration.

Name:  Maximum of 128 characters. Letters, numbers, spaces, -, and \_ are allowed

Version:  Use the format: major.minor.patch

Description - optional:  
 Maximum of 1024 characters

**Base image** [Info](#)

Select image  
You can select the base image from a list of Image Builder managed images or Amazon Machine Images (AMIs) that your account has access to or import a virtual image.

Select managed images  
Image Builder managed images created by you, shared with you, or provided by AWS.

AWS Marketplace images  
Choose from images you subscribed to in AWS Marketplace.

Enter custom AMI ID  
The AWS Systems Manager Agent (SSM Agent) must be pre-installed in the selected AMI.

Import base image  
Import from your VM into Image Builder and use the converted image as the base image in your recipe.

**Image Operating System (OS)**  
Image Builder supports Amazon Linux, Windows, Ubuntu, CentOS, RHEL, and SLES.

Amazon Linux  
Amazon Linux 2 and 2023

Windows  
Windows Server 2016, 2019, 2004, 20H2, and 2022

Ubuntu  
Ubuntu 18.04 LTS, 20.04 LTS, 22.04 LTS, and 24.04 LTS

CentOS  
CentOS 7 and 8

Red Hat Enterprise Linux (RHEL)  
RHEL 7, 8, and 9

SUSE Linux Enterprise Server (SLES)  
SLES 12 and 15

**Image origin**  
Choose the image to configure from a list of previously created pipeline images, images shared with you or a quick start list to help you get started. You could also enter a custom AMI ID to define the base image.

Quick start (Amazon-managed)  
Amazon curated images to help you get started

Images owned by me  
Images you created with Image Builder

Images shared with me  
Images shared with this account

**Image name**  
Choose an image based on the above selections.  
  
Owner: Amazon OS: Windows

**Auto-versioning options**  
Choose the OS version that the pipeline can automate for future builds.

Use latest available OS version

Use selected OS version

Specify OS version

+1 (410) 8887049  
contact@jjtechinc.co  
www.jjtechinc.co  
14103 Hammermill Field Dr Bowie MD 20720



JJ Tech

### 1.2.3 Adding Components

We will now add components to our AMI. In this example we'll use components provided by AWS

**Components** Info

Components are software scripts that define the custom configuration for an image. Components cannot be modified or replaced after a recipe is created. Automatic version choices are provided for each component. A maximum of 20 components (including build and test) can be applied to a recipe.

**Step 1: Choose build components to produce the desired output AMI**

Build components are software scripts that define a sequence of steps for downloading, installing, and configuring software packages. They also define validation steps.

**Build components - Windows (16)**

Filter owner: Amazon-managed

Find components by name:  X

Page: 1 / 2 / ... / 3 / 4 / 5

Name	Description
amazon-cloudwatch-agent-windows	Installs the latest version of the Amazon CloudWatch agent. This component installs only the agent. You must take additional steps to configure and use the Amazon CloudWatch agent. For more information, see the documentation at <a href="https://docs.aws.amazon.com/AmazonCloudWatch/latest/monitoring/install-CloudWatch-Agent-on-EC2-Instance.html">https://docs.aws.amazon.com/AmazonCloudWatch/latest/monitoring/install-CloudWatch-Agent-on-EC2-Instance.html</a> . Owner: Amazon ARN: arn:aws:imagebuilder:us-east-1:aws:component/amazon-cloudwatch-agent-windows/x.x.x

Installs Amazon Corretto 11 for Windows in accordance with the Amazon Corretto 11 User Guide at [Amazon Corretto 11 User Guide](#).



**Selected components (1)**

Expand the component to view versioning options and input parameters. To sort the build sequence, drag the components up and down.

Sequence	Component (drag the component up or down to change the sequence)	Owner	Expand all
1	amazon-cloudwatch-agent-windows	Owner: Amazon	<input checked="" type="checkbox"/> Expand all

Component (drag the component up or down to change the sequence): amazon-cloudwatch-agent-windows

Owner: Amazon

Versioning options

**Step 2: Optional Select tests to verify the output AMI (post-build)**

Test components are a sequence of steps used to verify that the output image built by your image pipeline is functioning as expected.

**Test components - Windows (8)**

Filter owner: Amazon-managed

Find components by name:  X 1 match

Page: 1 / 2 / ... / 3 / 4 / 5

Name	Description
simple-boot-test-windows	Executes a simple boot test. Owner: Amazon ARN: arn:aws:imagebuilder:us-east-1:aws:component/simple-boot-test-windows/x.x.x



**What will this do?**

Image builder will run through a build, validate and test phase. The build and validate phase run on an EC2 Instance created from the recipe's parent image. When these complete successfully Image Builder creates an AMI. From that AMI a new EC2 instance is launched to run the tests. If the tests pass, the Image is distributed via the distribution configuration.



+1 (410) 8887049  
contact@jjtechinc.co  
www.jjtechinc.co  
14103 Hammermill Field Dr Bowie MD 20720



JJ Tech

### Define image creation process

An image recipe is a document that defines the components to be applied to the base images to create the desired configuration for the output image. After a recipe has been created, it cannot be modified. A new version must be created in order to change components.

Type

Default workflows  
Use service default build and test workflows combination.

Custom workflows  
Combine customized build and test workflows of your choice.

Create workflow

Cancel Previous Next

### 1.2.4 Infrastructure Configuration

Define infrastructure configuration - *optional*

Image Builder launches EC2 instances in your account to customize images and run validation tests. The following settings specify infrastructure details for the instances that will run in your AWS account.

Infrastructure configuration

Configuration options

Create infrastructure configuration using service defaults

Use existing infrastructure configuration

Create a new infrastructure configuration

General

Enter a name and description for this configuration.

Name

JJTech-Infra-Config

Maximum of 128 characters. Letters, numbers, spaces, -, and \_ are allowed

Description - *optional*

Maximum of 1024 characters

IAM role **AdminFullAccess**

Select a role to associate with the instance profile. This role defines what permissions the instances launched by EC2 Image Builder will have in your account. These permissions are used to download and execute your components, upload logs to CloudWatch, and perform any additional actions specified in your selected components.

Create new role

Let's configure the Instance type, VPC, Subnet and Security Group as follows:

- Select Instance type
- Select DefaultVpc
- Select applicationSubnet1
- Select workshop Security Group

+1 (410) 8887049  
contact@jjtechinc.co  
www.jjtechinc.co  
14103 Hammermil Field Dr Bowie MD 20720



JJ Tech

**AWS infrastructure**  
Service-specific defaults will be applied if you do not select values.

**Instance type** [Info](#)  
Select one or more instance types to customize your image.

**m6i.large**

**SNS topic** [Info](#)  
Select an SNS topic to receive notifications and alerts from EC2 Image Builder

**VPC, subnet and security groups**  
Specify advanced settings to launch the instance that will customize your image.

**Virtual Private Cloud (VPC)** [Info](#)  
Select a VPC to launch your instance.

**Subnet ID**  
The subnet associated with the selected VPC

**Security groups**  
The security group associated with the selected VPC.  
   
**sg-05ace818a7b1b85a5**

**Define distribution settings - optional**  
Distribution settings include specific Region settings for encryption, launch permissions, accounts that can launch the output AMI, the output AMI name, and license configurations.

**Distribution settings**

**Configuration options**

Create distribution settings using service defaults

Use existing distribution settings

Create new distribution settings

**Region settings**  
The AMI will be distributed to the current Region by default.

Region	Output AMI name	Encryption (KMS key)	Target accounts for distribution	Principals with shared permission	Target accounts for faster launch configuration	Associated license configuration	Launch template configuration	Set launch template default version
us-east-1	-	Configured in storage options	-	-	-	-	-	-

**Cancel** **Previous** **Next**

+1 (410) 8887049  
contact@jjtechinc.co  
www.jjtechinc.co  
14103 Hammermil Field Dr Bowie MD 20720



JJ Tech

Step 4: Infrastructure configuration

Edit

Infrastructure details

Configuration name	Configuration description	IAM role	ARN
-	-	VPC ID	-
Instance type	SNS	Subnet	-
-	-	Terminate instance on failure	-
Security group	Key pair	Enabled	Logs
-	-	Resource tags	-
Metadata version	Metadata token response hop limit	-	Tags
-	-	-	-

Step 5: Distribution settings

Edit

Distribution details

Cancel

Previous

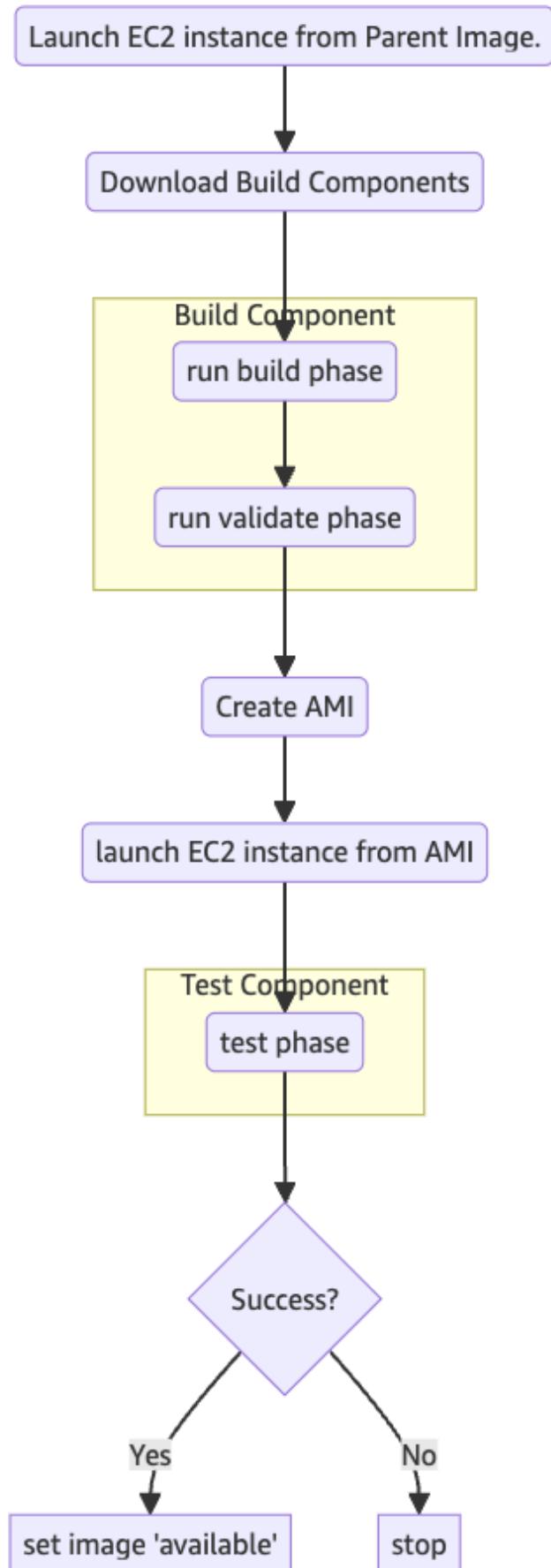
Create pipeline



+1 (410) 8887049  
contact@jjtechinc.co  
www.jjtechinc.co  
14103 Hammermil Field Dr Bowie MD 20720



JJ Tech





JJ Tech

## 1.2.6 Run the pipeline

EC2 Image Builder > Image pipelines > JJTech-Image-Pipeline

### JJTech-Image-Pipeline AMI

Summary			
Description -	Creation time September 10, 2024, 10:04 (UTC-06:00)	Time of last run -	Time of next run September 16, 2024, 03:00 (UTC-06:00)
IAM role AdminFullAccess	Build schedule (UTC) cron(0 9 ? * mon)	Pipeline status <span style="color: green;">Enabled</span>	ARN <code>arn:aws:imagebuilder:us-east-1:590183825830:image-pipeline/jjtech-image-pipeline</code>
Enhanced metadata collection Enabled			

Summary

Description -	Creation time September 10, 2024, 10:04 (UTC-06:00)	Time of last run -	Time of next run September 16, 2024, 03:00 (UTC-06:00)
IAM role AdminFullAccess	Build schedule (UTC) cron(0 9 ? * mon)	Pipeline status <span style="color: green;">Enabled</span>	ARN <code>arn:aws:imagebuilder:us-east-1:590183825830:image-pipeline/jjtech-image-pipeline</code>
Enhanced metadata collection Enabled			

**Output images**

Output images produced by the pipeline

Find output images	Version	Type	Creation time	Image status	Reason for failure	ARN	Log stream
<input type="checkbox"/>	<a href="#">1.0.0/1</a>	AMI	September 10, 2024, 10:07 (UTC-06:00)	Creating	-	<code>arn:aws:imagebuilder:us-east-1:590183825830:image-jjtech-window-recipe/1.0.0/1</code>	<a href="#">/aws/imagebuilder/JJTech-Window-Recipe/1.0.0/1</a>

[Navigate to the EC2 Console](#)

Here you will soon find an ec2 instance that is being used for building the image.

Instances (1) Info

Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone	Public IPv4 DNS	Public IPv4
Build instance ...	i-01e653ccb5af08988	Running	m6i.large	Initializing	View alarms +	us-east-1a	ec2-54-160-242-126.co...	54.160.242...

+1 (410) 8887049  
contact@jjtechinc.co  
www.jjtechinc.co  
14103 Hammermill Field Dr Bowie MD 20720



JJ Tech

Once the instance has finished building, it will be terminated and your new ami will be available.

This is your new image

Amazon Machine Images (AMIs) (1/2) Info

Owned by me Find AMI by attribute or tag

Actions Launch instance from AMI

Name	AMI name	AMI ID	Source	Owner	Visibility	Status
<input checked="" type="checkbox"/> JJTech-Window-Recipe 2024-0...	ami-0f2cd7d6c4532c4e7	590183825830/JJTech-Window-Recipe ...	590183825830	Private	Available	
<input type="checkbox"/> amazon-linux-2	amazon-linux-2-2024-08-30T1...	ami-0c0b6d99ebb1184be	590183825830/amazon-linux-2-2024-...	590183825830	Private	Available

AMI ID: ami-0f2cd7d6c4532c4e7

Details Permissions Storage Tags

Enable fast launch for your Windows AMI instances. - Recommended Improve launch times for Windows Server AMIs by up to 65%. You only pay for the storage of the snapshots created to speed up the instance launch times. Learn more

Enable fast launch X

AMI ID: ami-0f2cd7d6c4532c4e7

Image type: machine

Platform details: Windows

Root device type: EBS

AMI name: JJTech-Window-Recipe 2024-09-10T16-14-45.574994Z

Owner account ID: 590183825830

Architecture: x86\_64

Usage operation: RunInstances:0002

Root device name: /dev/sda1

Status: Available

Source: 590183825830/JJTech-Window-Recipe 2024-09-10T16-14-45.574994Z

Virtualization type: hvm

Boot mode: name

State reason: -

Creation date: Tue Sep 10 2024 10:25:55 GMT-0600 (Mountain)

Kernel ID: -

This is your new image name

### 1.2.7 – Create a new Instance from golden Image and connect to it.

Select the Image

Amazon Machine Images (AMIs) (1/1) Info

Owned by me Find AMI by attribute or tag

Actions Launch instance from AMI

Name	AMI name	AMI ID	Source	Owner	Visibility	Status
<input checked="" type="checkbox"/> JJTech-Window-Recipe 2024-0...	ami-0f2cd7d6c4532c4e7	590183825830/JJTech-Window-Recipe ...	590183825830	Private	Available	

+1 (410) 8887049  
contact@jjtechinc.co  
www.jjtechinc.co  
14103 Hammermill Field Dr Bowie MD 20720



JJ Tech

## Launch an instance [Info](#)

Amazon EC2 allows you to create virtual machines, or instances, that run on the AWS Cloud. Quickly get started by following the simple steps below.

### Name and tags [Info](#)

Name

Windows server

Add additional tags

### Application and OS Images (Amazon Machine Image) [Info](#)

An AMI is a template that contains the software configuration (operating system, application server, and applications) required to launch your instance. Search or Browse for AMIs if you don't see what you are looking for below.

Search our full catalog including 1000s of application and OS images

AMI from catalog

Recents

My AMIs

Quick Start

Name

JJTech-Window-Recipe 2024-09-10T16-14-45.574994Z

Description

Image ID

ami-0f2cd7d6c4532c4e7

Published  
2024-09-10T16:25:55.00  
0Z

Architecture  
x86\_64

Virtualization  
hvm

Root device  
type  
ebs

ENI Enabled  
Yes



Browse more AMIs  
Including AMIs from AWS, Marketplace and the Community

### Summary

Number of instances [Info](#)

1

Software Image (AMI)

JJTech-Window-Recipe 2024-09-1...read more  
ami-0f2cd7d6c4532c4e7

Virtual server type (instance type)

t2.micro

Firewall (security group)

New security group

Storage (volumes)

1 volume(s) - 30 GiB

Free tier: In your first year includes 750 hours of t2.micro (or t3.micro in the Regions in which t2.micro is unavailable) instance usage on free tier AMIs per month, 750 hours of public IPv4 address usage per month, 30 GiB of EBS storage, 2 million IOs, 1 GB of snapshots, and 100 GiB of bandwidth to the internet.

Cancel

Launch Instance

Review commands

### Instance type [Info](#) | Get advice

Instance type

t2.micro

Family: t2 1 vCPU 1 GiB Memory Current generation: true  
On-Demand Windows base pricing: 0.0162 USD per Hour  
On-Demand SUSE base pricing: 0.0116 USD per Hour  
On-Demand RHEL base pricing: 0.026 USD per Hour  
On-Demand Linux base pricing: 0.0116 USD per Hour

Free tier eligible

All generations

Compare instance types

Additional costs apply for AMIs with pre-installed software

### Key pair (login) [Info](#)

You can use a key pair to securely connect to your instance. Ensure that you have access to the selected key pair before you launch the instance.

Key pair name - required

Select

For Windows instances, you use a key pair to decrypt the administrator password. You then use the decrypted password to connect to your instance.

Create new key pair

We need this keypair to decrypt our with instance password

+1 (410) 8887049

contact@jjtechinc.co

www.jjtechinc.co

14103 Hammermill Field Dr Bowie MD 20720



JJ Tech

### Create key pair

**Key pair name**  
Key pairs allow you to connect to your instance securely.  
 (highlighted)

The name can include up to 255 ASCII characters. It can't include leading or trailing spaces.

**Key pair type**

RSA  
A encrypted private and public key pair

ED25519  
ED25519 encrypted private and public key pair (Not supported for Windows instances)

**Private key file format**

.pem  
For use with OpenSSH

.ppk  
For use with PUTTY

**Warning:** When prompted, store the private key in a secure and accessible location on your computer. You will need it later to connect to your instance. [Learn more](#) (highlighted)

Once the keypair is created, make sure you know where the downloaded private key is stored in your computer

(arrow pointing from the warning text to the red box)

(arrow pointing from the 'Create key pair' button to the red box)

Cancel Create key pair

**Additional charges apply** when outside of **free tier allowance**

**Firewall (security groups) | Info**  
A security group is a set of firewall rules that control the traffic for your instance. Add rules to allow specific traffic to reach your instance.

Create security group (highlighted)

Select existing security group

We'll create a new security group called 'launch-wizard-3' with the following rules:

Allow SSH traffic from Anywhere  
Helps you connect to your instance  
0.0.0.0/0

Allow HTTPS traffic from the Internet  
To set up an endpoint, for example when creating a web server

Allow HTTP traffic from the internet  
To set up an endpoint, for example when creating a web server

**Warning:** Rules with source of 0.0.0.0/0 allow all IP addresses to access your instance. We recommend setting security group rules to allow access from known IP addresses only. (highlighted)

This instance needs to connect to ssm so we need internet access

(arrow pointing from the 'Create security group' button to the red box)

(arrow pointing from the 'Anywhere' dropdown to the red box)

(arrow pointing from the 'Allow SSH traffic from' checkbox to the red box)

(arrow pointing from the 'Allow HTTPS traffic from the Internet' checkbox to the red box)

(arrow pointing from the 'Allow HTTP traffic from the internet' checkbox to the red box)

(arrow pointing from the 'Warning' text to the red box)

**Advanced details** Info

**Domain join directory** Info  
 (highlighted) Create new directory

**IAM instance profile** Info  
 (highlighted) Create new IAM profile

Click on Launch Instance

+1 (410) 8887049  
contact@jjtechinc.co  
www.jjtechinc.co  
14103 Hammermill Field Dr Bowie MD 20720



JJ Tech

Lets connect to the instance with both RDP and Session manager

## 2.1 - Connect using Session manager

- Select the instance and click on Connect

Instances (1/1) [Info](#)

Last updated less than a minute ago

Find Instance by attribute or tag (case-sensitive)

All states

Name: Windows server | Instance ID: i-0d4034ce0155a8b00 | Instance state: Running | Instance type: t2.micro | Status check: 2/2 checks passed | Alarm status: | Availability Zone: us-east-1d | Public IPv4 DNS: ec2-3-95-183-148.com... | Public IPv4: 3.95.183.14

Actions | Launch instances

### Connect to instance [Info](#)

Connect to your instance i-0d4034ce0155a8b00 (Windows server) using any of these options

Session Manager | RDP client | EC2 serial console

Session Manager usage:

- Connect to your instance without SSH keys, a bastion host, or opening any inbound ports.
- Sessions are secured using an AWS Key Management Service key.
- You can log session commands and details in an Amazon S3 bucket or CloudWatch Logs log group.
- Configure sessions on the Session Manager [Preferences](#) page.

Cancel | **Connect**

Session ID: root-gwfancnw04nlbxisxfrmlcfvwae | Instance ID: i-0d4034ce0155a8b00 | [Terminate](#)

```
PS C:\Windows\system32> PS C:\Windows\system32>
```

This is your new instance terminal

## 2.2 – Connect using RDP

+1 (410) 8887049  
contact@jjtechinc.co  
www.jjtechinc.co  
14103 Hammermil Field Dr Bowie MD 20720



JJ Tech

- Select the instance and click on Connect

### Connect to instance Info

Connect to your instance i-0d4034ce0155a8b00 (Windows server) using any of these options

Session Manager    **RDP client**    EC2 serial console

Instance ID  
 **i-0d4034ce0155a8b00 (Windows server)**

Connection Type  
 **Connect using RDP client**  
Download a file to use with your RDP client and retrieve your password.

**Connect using Fleet Manager**  
Connect to your instance using Fleet Manager Remote Desktop.

You can connect to your Windows instance using a remote desktop client of your choice, and by downloading and running the RDP shortcut file below:

**Download remote desktop file**

When prompted, connect to your instance using the following username and password:

Public DNS  
 ec2-3-95-183-148.compute-1.amazonaws.com

Username Info  
 Administrator

Password    **Get password**

If you've joined your instance to a directory, you can use your directory credentials to connect to your instance.

**Go copy the value of the keypair we downloaded earlier and post it below**

**Cancel**

### Get Windows password Info

Use your private key to retrieve and decrypt the initial Windows administrator password for this instance.

Instance ID

**i-0d4034ce0155a8b00 (Windows server)**

Key pair associated with this instance

Image-builder-keypair

Private key

Either upload your private key file or copy and paste its contents into the field below.

**Upload private key file**

Private key contents - optional

```
-----BEGIN RSA PRIVATE KEY-----  
MIGfMA0GCSqGSIb3DQEBAQEAoGAd8lZxwvPzxt710MMatJbq93wJ83EyLfPpf+XdlfjVRtUQtqeQ6u6UJS  
cusDlduhj51ve6PwCfsVM37eoDZEvwJYUIUN/YWAXPeu5Pxcdx/xxC1NBu7RgeL  
fZ2AHpVeneWO11wouFvGOo8XS+eZrPqgAAe5KQZJ/KSFcW2xQpA=  
-----END RSA PRIVATE KEY-----
```

**This is used to decrypt the password**

**Cancel**

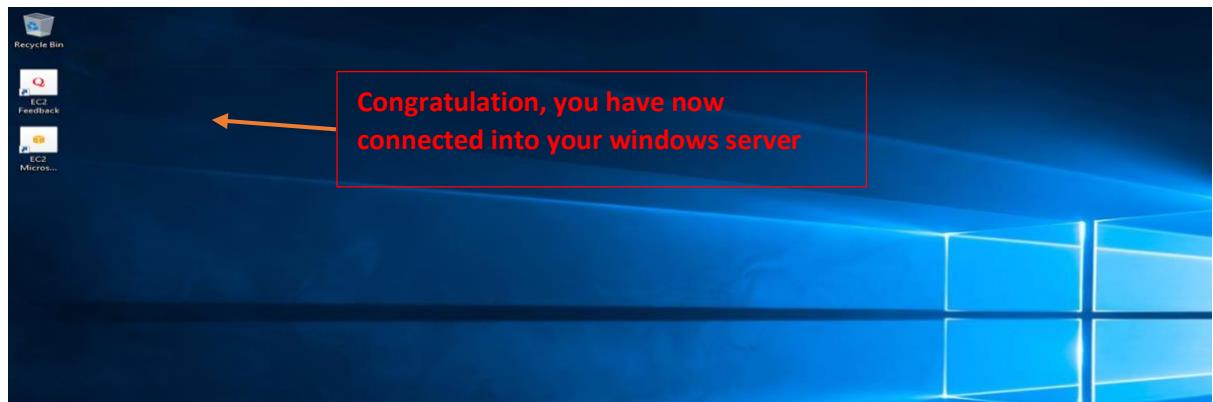
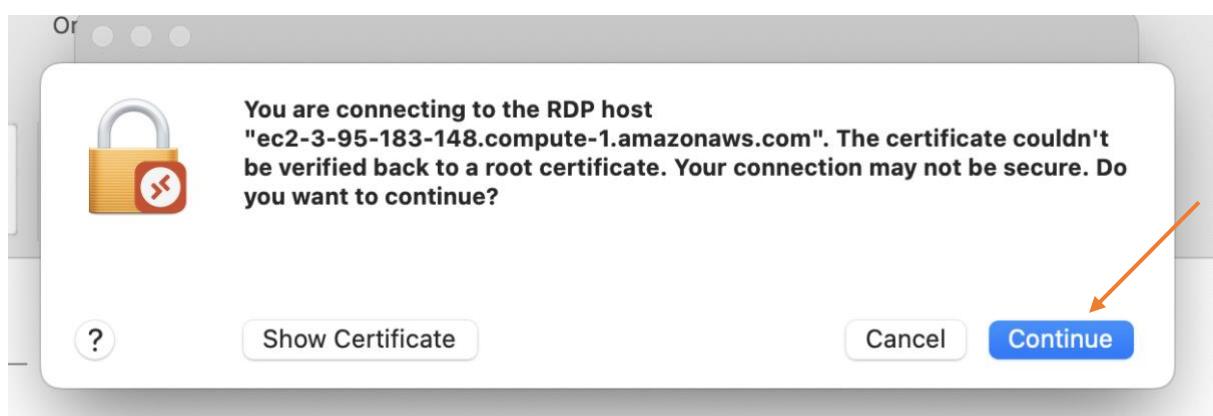
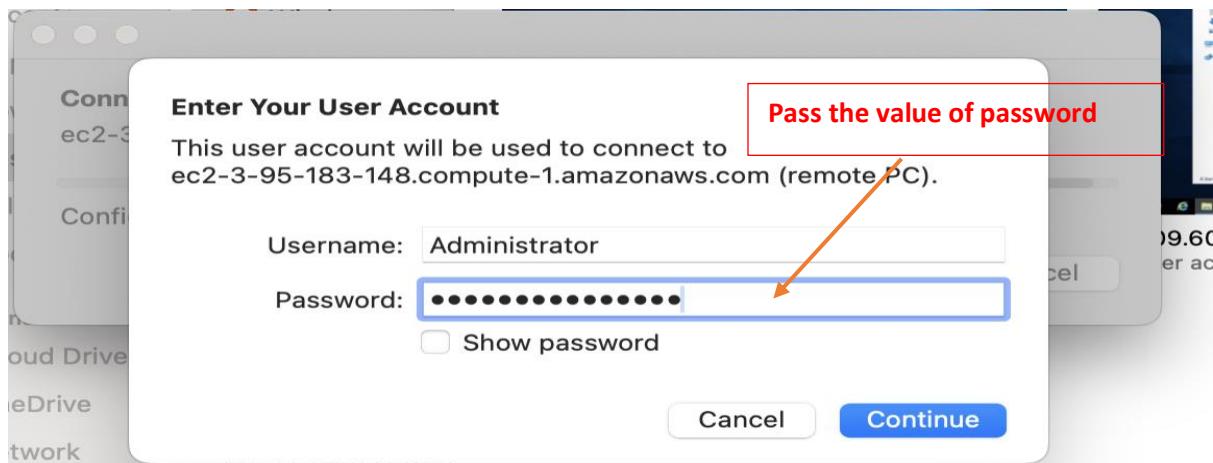
**Decrypt password**

+1 (410) 8887049  
contact@jjtechinc.co  
www.jjtechinc.co  
14103 Hammermill Field Dr Bowie MD 20720



JJ Tech

Get the password and pass it below.



+1 (410) 8887049  
contact@jjtechinc.co  
www.jjtechinc.co  
14103 Hammermil Field Dr Bowie MD 20720



JJ Tech

## 3.0 – Let's Manage Patching through AWS Systems Manager

### 3.1 – Using Patch Manager

Patch Manager provides options to scan your instances and report compliance on a schedule, install available patches on a schedule, and patch or scan instances on demand whenever you need to.

In this exercise, we will create a patch baseline and scan for missing patches.

Navigate to systems manager dashboard click on this link

[Systems-manager](#)

AWS Systems Manager Patch Manager

Manage patch compliance across the organization

Using Patch Manager, you can deploy patches simultaneously to applications and nodes across your organization. You can monitor patch compliance account by account.

How it works

- Create patch policy
- View dashboard
- View compliance reports

Patch your instances

Expedite patching by creating a patch policy to apply operating system patches across the organization, and track compliance account by account.

Create patch policy

Start with an overview

Use cases and blog posts

More resources

Take note of the instance we created

We can see this instance here because it meets the criterial for ssm managed instance

Dashboard | **Compliance reporting** | Patch baselines | Patches | Settings

Node patching details (1)

Name	Node ID	Patch configuration name	Patch configuration type	Compliant
Windows server	i-0d4034ce0155a8b00	-	Patch group	Never

+1 (410) 8887049  
contact@jjtechinc.co  
www.jjtechinc.co  
14103 Hammermill Field Dr Bowie MD 20720



JJ Tech

1. Choose "Patch Baselines"
2. Select "Create patch baseline".
3. Give it a name like: JJTech-Window-PatchBaseline, and set the **Operating system** to Windows.

The screenshot shows the AWS Systems Manager Patch Manager interface. At the top, there are navigation links: AWS Systems Manager > Patch Manager > Patch baselines. Below this is the Patch Manager header with tabs: Dashboard, Compliance reporting, Patch baselines (which is selected), Patches, and Settings. A yellow arrow points to the 'Patch baselines' tab. To the right of the tabs are buttons for 'Patch now' and 'Create patch policy'. Below the tabs, there's a search bar labeled 'Filter patch baselines' and a table with 17 entries. A second yellow arrow points to the 'Create patch baseline' button at the top right of the table area. The table includes columns for 'View details', 'Edit', 'Delete', and another 'Create patch baseline' button. At the bottom of the table area, there are navigation arrows for page 1 of 2. The main content area below the table is titled 'Approval rules for operating systems' and contains instructions for creating auto-approval rules. It includes sections for 'Operating system rule 1', 'Products', 'Classification', 'Severity', 'Auto-approval' (with radio buttons for 'Approve patches after a specified number of days' (selected) and 'Approve patches released up to a specific date'), and 'Compliance reporting - optional' (with a dropdown set to 'Unspecified'). There are also 'Add rule' and '9 remaining' buttons at the bottom of this section. Orange arrows from the left side point to the 'Select products', 'Select classifications', and 'Select severities' dropdown menus in the 'Operating system rule 1' section.

Let's Run a patch with on-demand scan

The screenshot shows the AWS Systems Manager Patch Manager interface again. At the top, there are navigation links: AWS Systems Manager > Patch Manager > Patch baselines. Below this is the Patch Manager header with tabs: Dashboard, Compliance reporting, Patch baselines (selected), Patches, and Settings. To the right of the tabs are buttons for 'Patch now' and 'Create patch policy'. Below the tabs, there's a search bar labeled 'Filter patch baselines' and a table with 17 entries. The table includes columns for 'View details', 'Edit', 'Delete', and another 'Create patch baseline' button. At the bottom of the table area, there are navigation arrows for page 1 of 2. The main content area below the table is titled 'Overview of patching operations - new'.

+1 (410) 8887049  
contact@jjtechinc.co  
www.jjtechinc.co  
14103 Hammermil Field Dr Bowie MD 20720



JJ Tech

## Patch instances now Info

### Basic configuration

Scan for missing patches or install patches, with or without rebooting. For more patching options, use the [Configure patching](#) page.

#### Patching operation

- Scan
- Scan and install

#### Instances to patch

Choose whether to patch all instances or only the instances you specify.

- Patch all instances
- Patch only the target instances I specify

#### Target selection

Choose a method for selecting targets.

- Specify instance tags  
Specify one or more tag key-value pairs to select instances that share those tags.

- Choose instances manually  
Manually select the instances you want to register as targets.

- Choose a resource group  
Choose a resource group that includes the resources you want to target.

i-0d4034ce0155a8b00 X

#### Managed instances (1/1)

Select one or more instances you want to patch.

Filter instances					
	Name	Instance ID	Platform type	Operating system	Instance state
<input checked="" type="checkbox"/>	Windows server	i-0d4034ce0155a8b00	Windows	Microsoft Windows Server 2019 Datacenter	<span>running</span>

#### Patching log storage

Select or create an S3 bucket for storing patching operation logs. Select Do not store logs if you don't require log information.

- Do not store logs

< 1 >

Patch now

## Navigate to Run Command

AWS Systems Manager > Run Command

Commands Command history

### Commands

Search executing commands

Congratulation your server is now patched

Requested date Document name Comment

Command ID Status Requested date Document name Comment

d86c966a-a60d-4736-bdda-e5ab32dc710e In Progress Thu, 12 Sep 2024 16:54:00 GMT AWS-RunPatchBaseline 83e18811-7649-442c-9496-58c1468b4bc3:dcade4d1-d8b

AWS Systems Manager > Run Command > Command ID: d86c966a-a60d-4736-bdda-e5ab32dc710e

Command ID: d86c966a-a60d-4736-bdda-e5ab32dc710e

### Command status

Overall status Success Detailed status Success # targets 1 # completed 1 # error 0 # delivery timed out 0

Watch the status

### Targets and outputs

Search command invocations

Instance ID Instance name Status Detailed Status Start time Finish time

i-0d4034ce0155a8b00 EC2AMAZ-DA94D16.WORKGROUP Success Success Thu, 12 Sep 2024 16:54:00 GMT Thu, 12 Sep 2024 16:54:29 GMT

+1 (410) 8887049

contact@jjtechinc.co

www.jjtechinc.co

14103 Hammermill Field Dr Bowie MD 20720



JJ Tech

## 4.0 - GitLab/Terraform.

Let's convert our ec2 image builder process into a code and use Gitlab CICD to deploy it to AWS. We will be testing now using amazon linux ami

### 4.1 - Gitlab

#### What is Gitlab?

Before we dive into definition for Gitlab, first we need to understand few terminologies. We often come across these terms like Git, Gitlab, GitHub, and Bitbucket. Let's define them:

**Git** - It is a source code versioning system that lets you locally track changes and push or pull changes from remote resources.

**GitLab, GitHub, and Bitbucket** - Are services that provides remote access to Git repositories. In addition to hosting your code, the services provide additional features designed to help manage the software development lifecycle. These additional features include managing the sharing of code between different people, bug tracking, Documentation(wiki) and other tools for 'social coding'.

- **GitHub** is a publicly available, free service which requires all code (unless you have a paid account) be made open. Anyone can see code you push to GitHub and offer suggestions for improvement. GitHub currently hosts the source code for tens of thousands of open-source projects.
- **GitLab** is a github like service that organizations can use to provide internal management of git repositories. It is a self-hosted Git-repository management system that keeps the users code private. You can use it for code management including code deployment.

##### 4.1.1 – Create a gitlab account

<https://about.gitlab.com/>



JJ Tech

Free 30-day trial

## GitLab Ultimate

Sign up for your free trial with:



Or create your own GitLab account

Fill out below with your own information

First name	Last name
JJTech	Inc
Username	
jjtech-inc	
Username is available.	
Email	
danny@jjtechinc.com	
Password	
.....	
<b>Continue</b>	

By clicking Continue or registering through a third party you accept the GitLab

- Check your email and provide the verification code
- Then click Next

+1 (410) 8887049  
contact@jjtechinc.co  
www.jjtechinc.co  
14103 Hammermil Field Dr Bowie MD 20720



JJ Tech

## Welcome to GitLab, JJTech!

To personalize your GitLab experience, we'd like to know a bit more about you. We won't share this information with anyone.

Role

I'm signing up for GitLab because:

Why are you signing up? (optional)

Who will be using this GitLab trial?  
 Just me  
 My company or team

Email updates (optional)  
 I'd like to receive updates about GitLab via email

**Continue**

## Create or import your first project

Projects help you organize your work. They contain your file repository, issues, merge requests, and so much more.

**Create**      Import

Group name

Project name

Select a template (optional)  
Get started with one of our popular project templates. [?](#)

You can choose any option here

Your project will be created at:  
<https://gitlab.com/jjtech7251637/ec2-image-builder>

You can always change your URL later

**Create project**

Your Profile

You gitlab account is set up

+1 (410) 8887049  
contact@jjtechinc.co  
www.jjtechinc.co  
14103 Hammermill Field Dr Bowie MD 20720



JJ Tech

You can't push or pull repositories using SSH until you add an SSH key to your profile.

Add SSH key Don't show again

Get the most out of your trial with space for more members

During your trial, invite as many members as you like to ec2-image-builder to collaborate with you. When your trial ends, you'll have a maximum of 1 member in the Free tier, or you can get more by upgrading to a paid tier.

Invite more members Explore paid plans

Name	Last commit	Last update
public	Initialized from 'Pages/Plain HTML' proj...	10 months ago
.gitlab-ci.yml	Initialized from 'Pages/Plain HTML' proj...	10 months ago
README.md	Initialized from 'Pages/Plain HTML' proj...	10 months ago

You can interact with gitlab from your local terminal using HTTP or SSH  
To use ssh, you will need to generate an ssh keypair and share it with gitlab

#### 4.1.2 - Let's add an ssh key [generate ssh key](#)

Git to your terminal

```
bash-3.2$ ssh-keygen -t rsa -b 2048
```

Generating public/private rsa key pair.

Enter file in which to save the key (/Users/dedie/.ssh/id\_rsa): This is optional  
/Users/dedie/.ssh/gitlab-key



+1 (410) 8887049  
contact@jjtechinc.co  
www.jjtechinc.co  
14103 Hammermill Field Dr Bowie MD 20720



JJ Tech

```
bash-3.2$ ssh-keygen -t rsa -b 2048
Generating public/private rsa key pair.
Enter file in which to save the key (/Users/dedie/.ssh/id_rsa): /Users/dedie/.ssh/gitlab-key
Enter passphrase (empty for no passphrase): Hit enter all
Enter same passphrase again: the way
Your identification has been saved in /Users/dedie/.ssh/gitlab-key
Your public key has been saved in /Users/dedie/.ssh/gitlab-key.pub
The key fingerprint is:
SHA256:eJGkQrBZqVRFFJgnWnbc3f4rtjUlgFVr+ekpHgKp+HQ dedie@GIM-X9DW19QCDV
The key's randomart image is:
+---[RSA 2048]---+
.o+o=... ...
.=B +o...o.
.o=+. o ... +
o . . . .o ...
. So ...o
... . oo.
. o E . oo+
o . =.+
. ..+
+---[SHA256]---+
```

This optional, if you don't provide the path, the key will be stored in .ssh/id\_rsa

Let copy the content of the public we generated

```
# cat ~/.ssh/gitlab-key.pub
```

```
bash-3.2$ cat ~/.ssh/gitlab-key.pub
ssh-rsa AAAAB3NzaC1yc2EAAAQABAAQcvkPRzUBWtuNOZ8likSUKqag34isM+5c7WZqDYuvGr72jUMMGcjq+LamSDUl5fPkJ9fYi6Da
qwU7V421Xt+Ie6uANFWw9F1F1ZAE/3R1gkox3MsYADV8cfwnN7Ssj5pXyh6bTUmpr2s8JBwgF5wZB1gCWN8GXuQCl/NWOYEB/HVOpireZkB0x
guzdXqrB4gUYJZwC8tntSqtF6XUd+Cmu42GXgM68MoXL17CMubJLj3YmvmDkogRgkgEUK7rl+jHd8EqB8tsQhoYzTdQYlPy2iR/ynZbrYgyqxn8
+0oBb4RRgPjLrdiu7FC6wBZS4HLPWqumMjvyNytJVOWGjQtv3X dedie@GIM-X9DW19QCDV
bash-3.2$
```

Let add the key to gitlab

1. Sign in to your GitLab account.
2. On the left sidebar, select your avatar.
3. Select **Edit profile**.
4. On the left sidebar, select **SSH Keys**.
5. Select **Add new key**.

+1 (410) 8887049

contact@jjtechinc.co

www.jjtechinc.co

14103 Hammermill Field Dr Bowie MD 20720



JJ Tech

The screenshot shows the JJ Tech GitLab interface. On the left, a sidebar menu is open for the user 'JJTech Inc' (@jjtech-inc). The menu includes options like 'Set status', 'Edit profile' (which has an orange arrow pointing to it), 'Preferences', 'Switch to GitLab Next', 'Sign out', and 'Merge requests'. Below this is a 'To-Do List' section. At the top of the page, there's a navigation bar with icons for user profile, search, and project creation, followed by the text 'Your work / Projects'. The main content area is titled 'Projects' and shows a list with one item: 'JJTech / ec2-image-builder' (Owner). There are tabs for 'Yours' (1), 'Starred' (0), 'Personal', and 'Inactive'. A search bar at the bottom of the projects list allows for 'Search or filter results...'.

The screenshot shows the 'User Settings / SSH Keys' page. On the left, a sidebar lists various settings: Profile, Account, Billing, Applications, Chat, Access tokens, Emails, Password, Notifications, SSH Keys (which has an orange arrow pointing to it), GPG Keys, Preferences, Comment Templates, Active Sessions, Authentication Log, and Usage Quotas. The main content area is titled 'SSH Keys' and contains a message: '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](#).'. Below this is a section titled 'Your SSH keys' with a count of '0'. A purple padlock icon is shown. To the right, there's a button labeled 'Add new key' with an orange arrow pointing to it. At the bottom right of the page, there's a reminder: '1 Reminder'.

+1 (410) 8887049  
contact@jjtechinc.co  
www.jjtechinc.co  
14103 Hammermil Field Dr Bowie MD 20720



JJ Tech

### SSH Keys

SSH keys allow you to establish a secure connection between your computer and GitLab. SSH fingerprints verify that the client is connected correctly.

Your SSH keys

**Add an SSH key**

Add an SSH key for secure access to GitLab. [Learn more](#).

**Key**

```
ssh-rsa AAAAB3NzaC1yc2EAAAQABAAQCVkPRzUBWtuNOZ8IkSUKqag34isM+5c7WZqDYuvGr72jUUMMOCjQ+LamSDUL5fPkJ9fYi3MsYADV8CfwNn7SsJ5pXyh6bTUmpr2s8JBwGF5wZB1gCWN8GXuQCI/NWOYEB/HVQprieZkB0xguzdXqrB4gUYJZwC8tntSqF6XgRgkqEUK7r!+jHd8EqB8tsQhoYzTdQYIPy2IR/ynZbrYgyqxn8+OoBb4RRgPJLrdlu7FC6wBZS4HLPWgumMjvyNytJVOWGjQtv3Xded
```

Begins with 'ssh-rsa', 'ecdsa-sha2-nistp256', 'ecdsa-sha2-nistp384', 'ecdsa-sha2-nistp511', 'ssh-ed25519', 'sk-ecdsa-sha2-nistp256', 'ed25519@openssh.com'.

**Title** **Give it a name**

dan-laptop  
Key titles are publicly visible.

**Usage type** Authentication & Signing

**Expiration date** 2025-09-12   
Optional but recommended. If set, key becomes invalid on the specified date.

**Add key** **Cancel**

Now that your key is added to gtolab, let go back to gitlab main page

The screenshot shows the GitLab main interface. On the left, there's a sidebar with 'User settings' and 'Profile' sections. The main area has a search bar at the top. Below it, there's a 'Projects' section with a list of projects. One project, 'JJTech / ec2-image-builder', is highlighted with a red box and labeled 'Owner'. An orange arrow points from the 'Profile' section in the sidebar to the 'Owner' label in the projects list.

Lets clone this project to our local laptop.

+1 (410) 8887049  
contact@jjtechinc.co  
www.jjtechinc.co  
14103 Hammermill Field Dr Bowie MD 20720



JJ Tech

ec2-image-builder

Initialized from 'Pages/Plain HTML' project template

Name Last commit Last update

public	Initialized from 'Pages/Plain HTML' proj...	10 months ago
.gitlab-ci.yml	Initialized from 'Pages/Plain HTML' proj...	10 months ago
README.md	Initialized from 'Pages/Plain HTML' proj...	10 months ago

Project information

- 1 Commit
- 1 Branch
- 0 Tags
- 2 KIB Project Storage

README  
CI/CD configuration  
+ Add LICENSE  
+ Add CHANGELOG

ec2-image-builder

Initialized from 'Pages/Plain HTML' project template

Clone with SSH  
git@gitlab.com:jjtech7251637/ec2-image-builder.git

Clone with HTTPS  
https://gitlab.com/jjtech7251637/ec2-image-builder

You can use either ssh or http

## SSH Option

```
bash-3.2$ git clone git@gitlab.com:jjtech7251637/ec2-image-builder.git
```

## HTTP option

```
bash-3.2$ git clone https://gitlab.com/jjtech7251637/ec2-image-builder.git
Cloning into 'ec2-image-builder'...
remote: Enumerating objects: 7, done.
remote: Total 7 (delta 0), reused 0 (delta 0), pack-reused 7 (from 1)
Receiving objects: 100% (7/7), done.
bash-3.2$
```

Now cd into your cloned project

```
# cd ec2-image-builder
```

```
bash-3.2$ cd ec2-image-builder/
```

To see the full content of your folder user ls -al

+1 (410) 8887049  
contact@jjtechinc.co  
www.jjtechinc.co  
14103 Hammermill Field Dr Bowie MD 20720



JJ Tech

```
[bash-3.2$ ls -al
total 16
drwxr-xr-x  6 dedie  staff   192 13 Sep 11:19 .
drwxr-xr-x  3 dedie  staff    96 13 Sep 11:19 ..
drwxr-xr-x 12 dedie  staff   384 13 Sep 11:19 .git
-rw-r--r--  1 dedie  staff   339 13 Sep 11:19 .gitlab-ci.yml
-rw-r--r--  1 dedie  staff  2187 13 Sep 11:19 README.md
drwxr-xr-x  4 dedie  staff   128 13 Sep 11:19 public
bash-3.2$ ]
```

We can now write our codes and push to gitlab.

If we need to deploy to any environment, like AWS, we need to provide gitlab access to our aws account. There are multiple ways you can do this.

One option is to use environment variables in gitlab. This is the option we will implement. You can also pass your access keys or iam role profile directly into your .gitlab-ci.yml. ( You can try this on your own)

## Let provide Gitlab with access keys of a user with full aws access.

If you don't already have access keys to use, login to aws and create a user called **gitlab**

The screenshot shows the 'Create user' wizard in the AWS IAM console. It's on Step 1: Specify user details. The 'User name' field contains 'gitlab'. A note below says: 'The user name can have up to 64 characters. Valid characters: A-Z, a-z, 0-9, and + = , @ \_ - (hyphen)'. There's an optional checkbox for 'Provide user access to the AWS Management Console'. A note below it says: 'If you're providing console access to a person, it's a best practice [link] to manage their access in IAM Identity Center.' At the bottom right, there are 'Cancel' and 'Next Step' buttons, with 'Next Step' being highlighted by an orange box and an arrow.

+1 (410) 8887049  
contact@jjtechinc.co  
www.jjtechinc.co  
14103 Hammermill Field Dr Bowie MD 20720



JJ Tech

Step 2  
Set permissions

Step 3  
Review and create

Add user to an existing group or create a new one. Using groups is a best-practice way to manage user's permissions by job functions. [Learn more](#)

**Permissions options**

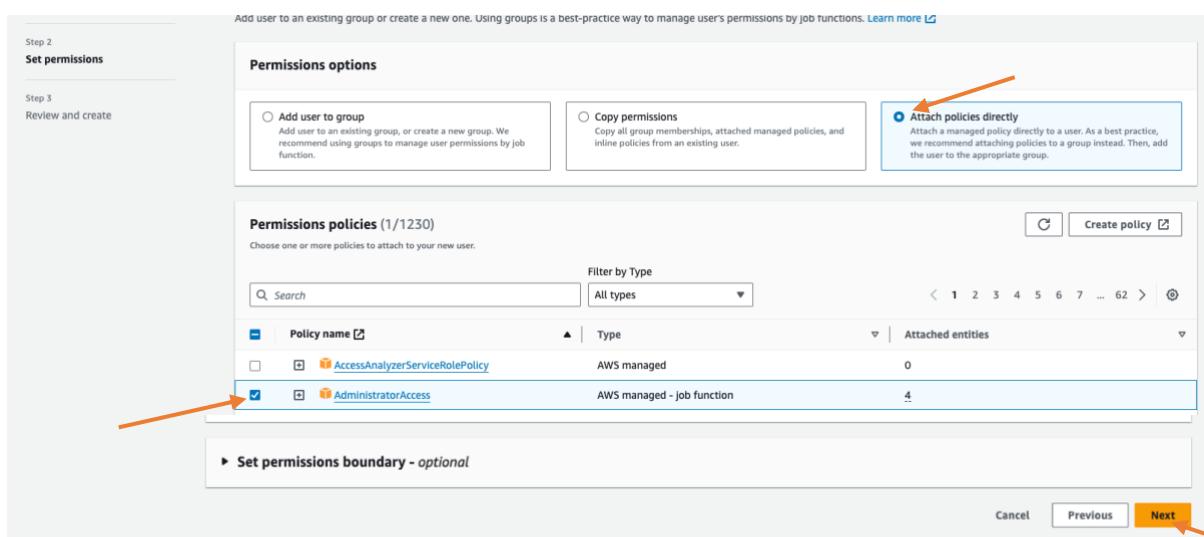
- Add user to group  
Add user to an existing group, or create a new group. We recommend using groups to manage user permissions by job function.
- Copy permissions  
Copy all group memberships, attached managed policies, and inline policies from an existing user.
- Attach policies directly  
Attach a managed policy directly to a user. As a best practice, we recommend attaching policies to a group instead. Then, add the user to the appropriate group.

**Permissions policies (1/1230)**  
Choose one or more policies to attach to your new user.

Policy name	Type	Attached entities
<a href="#">AccessAnalyzerServiceRolePolicy</a>	AWS managed	0
<a href="#">AdministratorAccess</a>	AWS managed - job function	4

**Set permissions boundary - optional**

Cancel Previous **Next**



**Permissions summary**

Name	Type	Used as
<a href="#">AdministratorAccess</a>	AWS managed - job function	Permissions policy

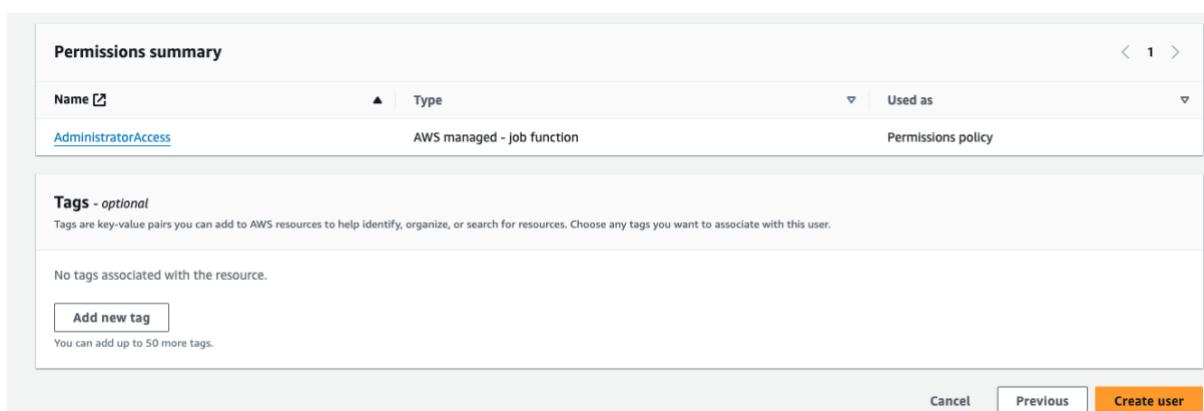
**Tags - optional**  
Tags are key-value pairs you can add to AWS resources to help identify, organize, or search for resources. Choose any tags you want to associate with this user.

No tags associated with the resource.

Add new tag

You can add up to 50 more tags.

Cancel Previous **Create user**



## Select the user you created and go to security credential

Permissions | Groups | Tags | **Security credentials** | Access Advisor

**Console sign-in**

Console sign-in link  
<https://590183825830.signin.aws.amazon.com/console>

Console password  
Not enabled

**Enable console access**

**Multi-factor authentication (MFA) (0)**

Use MFA to increase the security of your AWS environment. Signing in with MFA requires an authentication code from an MFA device. Each user can have a maximum of 8 MFA devices assigned. [Learn more](#)

Type	Identifier	Certifications	Created on
No MFA devices. Assign an MFA device to improve the security of your AWS environment			

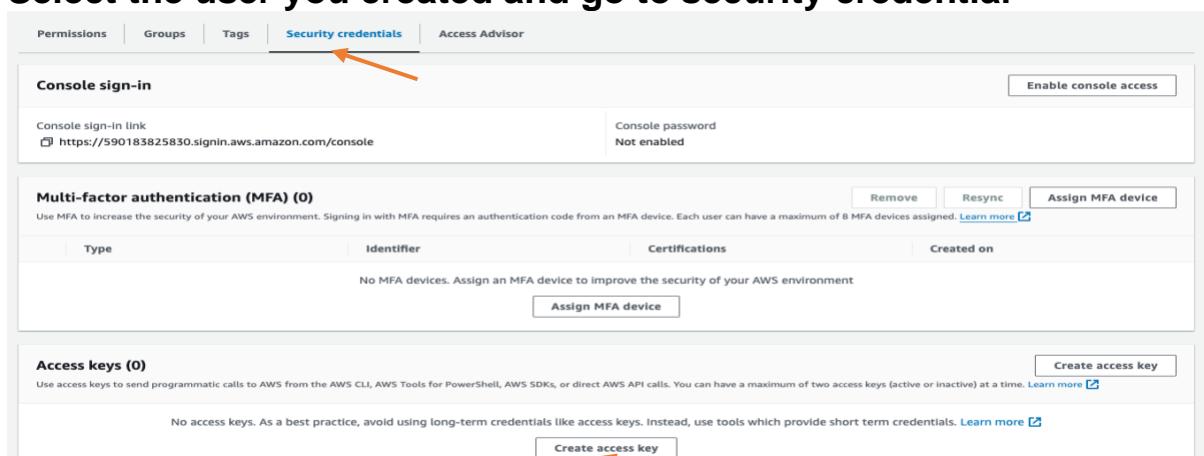
**Assign MFA device**

**Access keys (0)**

Use access keys to send programmatic calls to AWS from the AWS CLI, AWS Tools for PowerShell, AWS SDKs, or direct AWS API calls. You can have a maximum of two access keys (active or inactive) at a time. [Learn more](#)

No access keys. As a best practice, avoid using long-term credentials like access keys. Instead, use tools which provide short term credentials. [Learn more](#)

**Create access key**



## For use care – select Command Line interface

+1 (410) 8887049  
contact@jjtechinc.co  
www.jjtechinc.co  
14103 Hammermill Field Dr Bowie MD 20720



JJ Tech

IAM > Users > gitlab > Create access key

Step 1  
Access key best practices & alternatives

Step 2 - optional  
Set description tag

Step 3  
Retrieve access keys

**Access key best practices & alternatives** Info

Avoid using long-term credentials like access keys to improve your security. Consider the following use cases and alternatives.

**Use case**

Command Line Interface (CLI)  
You plan to use this access key to enable the AWS CLI to access your AWS account.

Local code  
You plan to use this access key to enable application code in a local development environment to access your AWS account.

**Alternatives recommended**

- Use [AWS CloudShell](#), a browser-based CLI, to run commands. [Learn more](#)
- Use the [AWS CLI V2](#) and enable authentication through a user in IAM Identity Center. [Learn more](#)

**Confirmation**

I understand the above recommendation and want to proceed to create an access key.

Cancel **Next**

IAM > Users > gitlab > Create access key

Step 1  
Access key best practices & alternatives

Step 2 - optional  
Set description tag

Step 3  
Retrieve access keys

**Set description tag - optional** Info

The description for this access key will be attached to this user as a tag and shown alongside the access key.

Description tag value

Describe the purpose of this access key and where it will be used. A good description will help you rotate this access key confidently later.

Maximum 256 characters. Allowed characters are letters, numbers, spaces representable in UTF-8, and: \_ : / = + - @

Cancel Previous **Create access key**

**Access key created**

This is the only time that the secret access key can be viewed or downloaded. You cannot recover it later. However, you can create a new access key any time.

IAM > Users > gitlab > Create access key

Step 1  
Access key best practices & alternatives

Step 2 - optional  
Set description tag

Step 3  
Retrieve access keys

**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
<input type="text"/> AKIAIOS2NSQGTOZBQAM2H	<input type="text"/> ***** <a href="#">Show</a>

**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](#).

Download .csv file **Done**

We now need to use these keys in gitlab

Go back to gitlab select you ec2-image-builder project.

+1 (410) 8887049  
contact@jjtechinc.co  
www.jjtechinc.co  
14103 Hammermill Field Dr Bowie MD 20720



JJ Tech

## Select settings > CICD > Variables

The screenshot shows the GitLab CI/CD Variables settings page. The navigation bar on the left has arrows pointing to the 'Variables' section and the 'Add variable' button. A red box highlights the 'values' field in the 'Pipeline trigger tokens' section.

**Variables**  
Variables store information that you can use in job scripts. Each project can define a maximum of 8000 variables. [Learn more](#).

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?](#)

Variables can have several attributes. [Learn more](#).

- Visibility: Set the visibility level for the value. Can be visible, masked, or masked and hidden.
- Flags
  - Protected: Only exposed to protected branches or protected tags.
  - Expanded: Variables with \$ will be treated as the start of a reference to another variable.

**CI/CD Variables** </> 0

Key ↑	Value	Environments	Actions
There are no variables yet.			

**Group variables (inherited)**  
These variables are inherited from the parent group.

**CI/CD Variables** </> 0

Key	Environments	Group
There are no variables yet.		

**CI/CD Settings**  
Runners are processes that pick up and execute CI/CD jobs for GitLab. [What is GitLab Runner?](#)

**Artifacts**  
A job artifact is an archive of files and directories saved by a job when it finishes.

**Variables**  
Variables store information that you can use in job scripts. Each project can define a maximum of 8000 variables. [Learn more](#).

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?](#)

Variables can have several attributes. [Learn more](#).

- Visibility: Set the visibility level for the value. Can be visible, masked, or masked and hidden.
- Flags
  - Protected: Only exposed to protected branches or protected tags.
  - Expanded: Variables with \$ will be treated as the start of a reference to another variable.

**CI/CD Variables** </> 2

Key ↑	Value	Environments
AWS_ACCESS_KEY_ID	*****	All (default)
AWS_SECRET_ACCESS_KEY	*****	All (default)

**Group variables (inherited)**  
These variables are inherited from the parent group.

**CI/CD Variables** </> 0

Key	Environments
There are no variables yet.	

**Pipeline trigger tokens**  
Trigger a pipeline for a branch or tag by generating a trigger token and using `curl -X POST` call. The token impersonates a user's project access and permission.

**Deploy freezes**

**Add variable**

Type: Variable (default)

Environments: All (default)

Visibility: Visible (selected)

Flags: Protect variable (selected)

Description (optional): The description of the variable's value or usage.

Key: AWS\_ACCESS\_KEY\_ID

Value: AKIAY52NSQGTNSTK7VK

Notes: You can use CI/CD variables with the same name in different places, but the variables might overwrite each other. What is the order of precedence for variables?

Variable value will be evaluated as raw string.

**Reveal values**  Add

**CI/CD Variables** </> 2

Key ↑	Value	Environments
AWS_ACCESS_KEY_ID	*****	All (default)
AWS_SECRET_ACCESS_KEY	*****	All (default)

+1 (410) 8887049

contact@jjtechinc.co

www.jjtechinc.co

14103 Hammermill Field Dr Bowie MD 20720



JJ Tech

## Let's go write some terraform code for our ec2 image builder infrastructure

- Don't worry I have a code ready for you. Here is the link [ec2-ami-builder](#)

Once we have finished modifying our code. We will push it to gitlab by running the following commands

- 1 – git add .
- 2 - git commit -m “ec2-image-builder”
- 3 – git push

Go verify that your code is now in github

Let's run a build

The screenshot shows the GitLab interface for the project 'ec2-image-builder'. The sidebar on the left has a 'Build' section highlighted with an orange arrow. The main content area shows a commit from 'JJTech Inc' made 3 minutes ago, updating two files. Below the commit, a table lists several files with their last commit times.

Name	Last commit	La
scripts	Update 6 files	10 mi
.gitlab-ci.yml	Update 6 files	10 mi
IE.md	Initialized from 'Pages/Plain HTML' proj...	10 m
n.tfvars	Update 6 files	10 mi
es.tf	Update 6 files	10 mi
README.md		



JJ Tech

Ultimate Trial Day 1/30  
Learn about features

Project ec2-image-builder

- Learn GitLab 33%
- Pinned
- Issues 0
- Merge requests 0
- Manage >
- Plan >
- Code >
- Build >
- Pipelines
- Jobs

All 0 Finished Branches Tags

Filter pipelines Q Show Pipeline ID

Clear runner caches Run pipeline

There are currently no pipelines.

Run pipeline

Run for branch name or tag master

Variables

Variable Input variable key Input variable value

Specify variable values to be used in this run. The variables specified in the configuration file as well as CI/CD settings are used by default. Variables specified here are expanded and not masked.

Run pipeline Cancel

redad / ec2-ami-builder / Pipelines / #1452910503

moved files

Running Daniel Eddie created pipeline for commit 3d3f7726 just now

For main latest 40 jobs In progress, queued for 0 seconds

This your pipeline

```
graph LR; validate[validate] --> plan[plan]; plan --> apply[apply]; apply --> destroy[destroy]
```

+1 (410) 8887049  
contact@jjtechinc.co  
www.jjtechinc.co  
14103 Hammermil Field Dr Bowie MD 20720



JJ Tech

The screenshot shows a GitLab project interface for a 'main' branch. The pipeline section displays a sequence of four stages: validate, plan, apply, and destroy. Each stage has a green checkmark icon indicating it has been completed. The 'plan' stage also features a circular refresh icon. Below the pipeline diagram, there is a terminal window showing Terraform command-line output. An orange arrow points from the text 'Plan: 10 to add, 0 to change, 0 to destroy.' to the pipeline diagram.

Expand the plan you should see 10 resources been deployed

```
589      + object_lock_enabled      = (known after apply)
590      + policy                  = (known after apply)
591      + region                  = (known after apply)
592      + request_payer           = (known after apply)
593      + tags_all                = {
594          + "environment" = "sandbox"
595      }
596      + website_domain          = (known after apply)
597      + website_endpoint        = (known after apply)
598  }
599 Plan: 10 to add, 0 to change, 0 to destroy.
600
601 Saved the plan to: tfplan
602 To perform exactly these actions, run the following command to apply:
603   terraform apply "tfplan"
```

If you are satisfied with the plan, run the apply job.

The screenshot shows a GitLab project interface for a 'main' branch. The pipeline section displays a sequence of four stages: validate, plan, apply, and destroy. The 'validate' and 'plan' stages have green checkmark icons. The 'apply' stage has a circular refresh icon, indicating it is currently running. The 'destroy' stage has a grey gear icon. The pipeline diagram is identical to the one shown in the previous screenshot, but the 'apply' stage is now active.

+1 (410) 8887049  
contact@jjtechinc.co  
www.jjtechinc.co  
14103 Hammermill Field Dr Bowie MD 20720



JJ Tech

The screenshot shows a GitLab pipeline interface. On the left, there's a sidebar with navigation links: 'Search or go to...', 'Issues', 'Merge requests', and 'Manage'. The main area displays a pipeline job named 'apply'. The status bar at the top indicates it is 'Running' and was 'Started just now by Daniel Edie'. Below the status, there's a large black progress bar with three white dots in the center. To the right of the progress bar, there's some small, illegible text.

Monitor your job, it should take about 12 -15 minutes

```
404 aws_imagebuilder_image...: Still creating... [12m30s elapsed]
405 aws_imagebuilder_image...: Creation complete after 12m33s [id=arn:aws:imagebuilder:us-east-1:590183825830:image/amazon-linux-recipe/1.1.0/5]
406 Apply complete! Resources: 10 added, 0 changed, 0 destroyed.
v 407 Cleaning up project directory and file based variables
408 Job succeeded
```

Go to AWS and check you ami. Filter by Owned by me. You should see an ami created by image builder

The screenshot shows the AWS EC2 Dashboard. In the top left, there are navigation links: 'EC2 Global View', 'Events', 'Console-to-Code Preview', and 'Instances'. The main content area is titled 'Amazon Machine Images (AMIs) (1) Info'. It shows a single AMI entry: 'Name': 'amazon-linux-2', 'AMI name': 'amazon-linux-2-2024-09-13T18-23-41.215Z', 'AMI ID': 'ami-0b3f61e55ec33d9f2', 'Source': '590183825830/amazon-linux-2-2024...', 'Owner': '590183825830', and 'Visibility': 'Private'. There are buttons for 'Actions', 'Launch Instance from AMI', and a 'Recycle Bin' button. A red arrow points to the 'Owned by me' dropdown menu in the top left of the table header.

Go ahead a run the destroy job. This will destroy.

```
676 aws_imagebuilder_component.components['disable_ipv6']: Destruction complete after 1s
677 aws_imagebuilder_component.components['install_docker']: Destruction complete after 1s
678 aws_imagebuilder_component.cw_agent: Destruction complete after 1s
v 679 Destroy complete! Resources: 10 destroyed.
680 Cleaning up project directory and file based variables
681 Job succeeded
```

!!!! Congratulation, you have successfully built a custom **aws ami** using **terraform** and **gitlab**.

+1 (410) 8887049  
contact@jjtechinc.co  
www.jjtechinc.co  
14103 Hammermill Field Dr Bowie MD 20720