

# Amazon Cognito

Amazon Cognito is a service offered by AWS (Amazon Web Services) that helps you manage user authentication, authorization, and access control for your web and mobile applications. It provides secure user sign-up, sign-in, and access to your apps by enabling you to easily authenticate users through social identity providers like Facebook, Google, and Apple, as well as through your own custom identity provider.

## Key Features of Amazon Cognito:

Amazon Cognito is a comprehensive identity management solution within the AWS ecosystem, designed to simplify the process of adding authentication and authorization to web and mobile applications. Here's a more detailed overview:

### 1. User Pools

- User Sign-Up and Sign-In: Amazon Cognito User Pools provide user directories that allow users to sign up and sign in to your apps. The sign-up process can include user verification through email or phone number. You can customize the sign-up process with required attributes, such as name or address, and configure it to support multi-factor authentication (MFA).
- OAuth 2.0 and SAML 2.0 Support: User Pools support OAuth 2.0 authorization flows, enabling easy integration with various identity providers. You can also integrate with SAML-based identity providers for enterprise use cases.
- Customizable UI: You can customize the UI for the hosted sign-up and sign-in pages, or you can create your own UI while using the Cognito APIs in the backend.

### 2. Identity Pools

- Authentication and Authorization: With Identity Pools, you can create unique identities for users and authenticate them through Cognito User Pools or other identity providers. Identity Pools enable users to get temporary AWS credentials to directly access AWS services like Amazon S3, DynamoDB, and others.
- Unauthenticated Users: Identity Pools allow you to define permissions for unauthenticated users (guest users) who have not signed in. This is useful for allowing limited access to resources before users sign up or sign in.
- Role-Based Access Control (RBAC): You can define IAM roles with specific permissions and map users to these roles based on their identity pool or attributes, allowing for fine-grained access control.

### 3. Federated Identities

- Social Identity Providers: Amazon Cognito supports authentication through popular social identity providers like Facebook, Google, Amazon, and Apple. This allows users to sign in using their existing social media credentials.

- Enterprise Identity Providers: You can also integrate with enterprise identity providers using SAML 2.0 or OpenID Connect (OIDC), enabling single sign-on (SSO) for enterprise applications.
- Custom Identity Providers: If your application uses a custom authentication mechanism, you can integrate it with Cognito by defining a custom identity provider.

## 4. Security and Compliance

- Multi-Factor Authentication (MFA): Cognito supports both SMS-based and TOTP (Time-based One-Time Password) MFA, enhancing the security of user accounts.
- Encryption: All data managed by Amazon Cognito is encrypted at rest and in transit, ensuring the security and privacy of user data.
- Compliance: Amazon Cognito is compliant with several industry standards, including GDPR, HIPAA, and SOC 2, making it suitable for applications with strict regulatory requirements.

## 5. Integration with Other AWS Services

- AWS Lambda: You can use Lambda triggers to customize authentication workflows. For example, you can use a pre-sign-up trigger to validate user attributes or a post-confirmation trigger to send a welcome email.
- Amazon API Gateway: Cognito can be integrated with API Gateway to provide authentication and authorization for your APIs, leveraging Cognito's user pools for token-based authentication.
- AWS Amplify: If you're building mobile or web applications using AWS Amplify, Amazon Cognito is the default solution for handling authentication, providing a seamless integration experience.

## 6. Analytics and Monitoring

- Amazon CloudWatch: You can monitor Amazon Cognito metrics using Amazon CloudWatch to track the performance and usage of your user pools and identity pools.
- AWS CloudTrail: Track all API calls made to Cognito using AWS CloudTrail, providing you with an audit trail of all activities within your Cognito environment.

## 7. Customization and Extensibility

- Custom Attributes: You can define custom attributes in your user pools to store additional information specific to your application.
- Custom Authentication Flows: Cognito allows you to implement custom authentication flows using AWS Lambda triggers. This gives you control over how authentication is handled, enabling scenarios like custom passwordless login mechanisms.

## 8. Pricing

- User Pools: Pricing is based on the number of Monthly Active Users (MAUs). You are charged for each MAU that authenticates during the month.

- Identity Pools: Charges are based on the number of operations, such as token exchanges or user synchronizations. Additional charges apply if your users access other AWS services using temporary credentials provided by Cognito.
- Free Tier: Amazon Cognito offers a free tier, which includes a limited number of MAUs and operations, making it a cost-effective option for small applications.

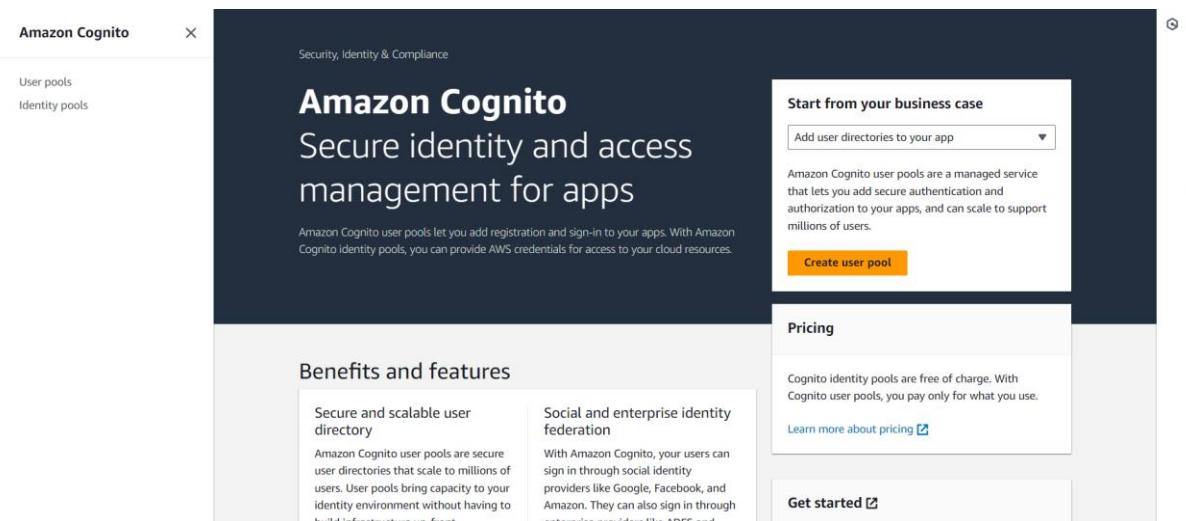
## 9. Use Cases

- Mobile and Web Apps: Cognito is widely used in mobile and web apps that require secure user management, particularly in e-commerce, social networking, and content delivery platforms.
- Enterprise Applications: Its ability to integrate with SAML and OIDC makes it a strong choice for enterprise applications that need to support single sign-on (SSO) with existing corporate identities.
- IoT Applications: Cognito's support for unauthenticated access and fine-grained access control makes it suitable for IoT applications where devices or sensors need access to AWS services.

Amazon Cognito provides a powerful and flexible way to manage user identities and permissions, making it a popular choice for developers building scalable and secure applications on AWS.

### 😊 To begin with the Lab:

1. In your AWS Console, search for Cognito and go toward its dashboard, then from here you have to click on Create user pools.



2. Now on step 1 you have to choose the same specifications as you can see below in the snapshot and click on next.

## Authentication providers

Configure the providers that are available to users when they sign in.

### Provider types

Choose whether users will sign in to your Cognito user pool, a federated identity provider, or both. Amazon Cognito has different pricing for federated users and user pool users. [Learn more about pricing](#)

#### Cognito user pool

Users can sign in using their email address, phone number, or user name. User attributes, group memberships, and security settings will be stored and configured in your user pool.

#### Federated identity providers

Users can sign in using credentials from social identity providers like Facebook, Google, Amazon, and Apple; or using credentials from external directories through SAML or Open ID Connect. You can manage user attribute mappings and security for federated users in your user pool.

### Cognito user pool sign-in options | [Info](#)

Choose the attributes in your user pool that are used to sign in. If you select only one attribute, or you select a user name and at least one other attribute, your user can sign in with all of the selected options. If you select only phone number and email, your user will be prompted to select one of the two sign-in options when they sign up.

#### User name

#### Email

#### Phone number

### User name requirements

#### Allow users to sign in with a preferred user name

#### Make user name case sensitive

**⚠️** Cognito user pool sign-in options can't be changed after the user pool has been created.

3. Then in step 2 you have to choose custom for password policy mode and set the password minimum length to 6 characters. After that disable all password requirements and scroll down.

## Password policy [Info](#)

Create a password policy to define the length and complexity of the passwords your users can set.

Password policy mode | [Info](#)

Cognito defaults

Use default password requirements.

Custom

Use password requirements that you define.

Password minimum length

6

character(s)

Must be a number between 6 and 99. We strongly recommend that you require passwords to be at least 8 characters in length.

Password requirements

- Contains at least 1 number
- Contains at least 1 special character
- Contains at least 1 uppercase letter
- Contains at least 1 lowercase letter

Temporary passwords set by administrators expire in

7

day(s)

Must be a number between 0 and 365.

4. Now say no to multi-factor authentication but enable user account recovery as shown below and click on next.

## Multi-factor authentication

Configure secure access to your app by enforcing multi-factor authentication (MFA) during the user sign-in process. MFA settings are applied to all app clients.

MFA enforcement | [Info](#)

Require MFA -

Recommended

Users must provide an additional authentication factor when signing in.

Optional MFA

Users can sign in with a single authentication factor, and can choose to add additional authentication factors.

No MFA

Users can only sign in with a single authentication factor. This is the least secure option.

## User account recovery

Configure how users will recover their account when they forget their password. Recipient message and data rates apply.

### Self-service account recovery | [Info](#)

#### Enable self-service account recovery - Recommended

Allow forgot-password operations in your user pool. In the hosted UI sign-in page, a "Forgot your password?" link is displayed. When this feature is not enabled, administrators reset passwords with the Cognito API.

### Delivery method for user account recovery messages | [Info](#)

Select how your user pool will deliver messages when users request an account recovery code. SMS messages are charged separately by Amazon SNS. Email messages are charged separately by Amazon SES. [Learn more about pricing](#)

- Email only
- SMS only
- Email if available, otherwise SMS
- SMS if available, otherwise email
- SMS if available, otherwise email, and allow a user to reset their password via SMS if they are also using it for MFA

[Cancel](#)

[Previous](#)

[Next](#)

5. So, for step 3 you have to keep this step to default settings and move to the next step.

[Amazon Cognito](#) > [User pools](#) > [Create user pool](#)

Step 1

[Configure sign-in experience](#)

Step 2

[Configure security requirements](#)

Step 3

[Configure sign-up experience](#)

Step 4

[Configure message delivery](#)

Step 5

[Integrate your app](#)

Step 6

[Review and create](#)

## Configure sign-up experience [Info](#)

Determine how new users will verify their identities when signing up and which attributes should be required or optional during the user sign-up flow.

### Self-service sign-up [Info](#)

Choose whether new users of your app can register for an account themselves.

#### Enable self-registration

Display a "Sign up" link on the sign-in page in the hosted UI, and allow the use of public APIs to create new user accounts. When this feature is not enabled, federation and administrative API operations create user profiles.

If you activate user sign-up in your user pool, anyone on the internet can sign up for an account and sign in to your apps. Don't enable self-registration in your user pool until you want to open your app to public sign-up.

[Learn more](#)

### Attribute verification and user account confirmation

Choose between Cognito-assisted and self-managed user attribute verification and account confirmation. Only verified attributes can be used for sign-in, account recovery, and MFA. A user account must be confirmed either by attribute verification, or user pool administrator confirmation, before a user is allowed to sign in.

6. Now for step 4 choose Send email with Cognito and click on next.

## Email

Configure how your user pool sends email messages to users.

Email provider | [Info](#)

Send email with Amazon SES - Recommended

Send emails using an Amazon SES verified identity in your account. We recommend this option for higher email volume and production workloads.

Send email with Cognito

Use Cognito's default email address as a temporary start for development. You can use it to send up to 50 emails a day.

You must have configured a verified sender with [Amazon SES](#) to use the SES feature. [Learn more](#)

SES Region [Info](#)

Asia Pacific (Mumbai)

FROM email address | [Info](#)

By default "no-reply@verificationemail.com" will be used. You can also choose a different email address that you have previously verified with Amazon SES.



REPLY-TO email address - *optional* | [Info](#)

If you set an invalid reply-to address, sending restrictions may be imposed on your account.

[Cancel](#)

[Previous](#)

[Next](#)

7. For step 5, first give it a user pool name and enable use the Cognito hosted UI.

## Integrate your app [Info](#)

Set up app integration for your user pool with Cognito's built-in authentication and authorization flows.

### User pool name

Create a friendly name for your user pool.

User pool name

User pool names are limited to 128 characters or less. Names may only contain alphanumeric characters, spaces, and the following special characters: + = , . @ -

Your user pool name can't be changed once this user pool is created.

### Hosted authentication pages

Choose whether to use Cognito's Hosted UI and OAuth 2.0 server for user sign-up and sign-in flows.

Use the Cognito Hosted UI

Build hosted sign-up, sign-in, and OAuth 2.0 service endpoints in Amazon Cognito. When this feature is not enabled, use Cognito API operations to perform sign-up and sign-in.

8. After that in domain choose use a Cognito domain and give a unique domain name.

**Domain Info**  
Configure a domain for your Hosted UI and OAuth 2.0 endpoints. To use the Hosted UI, you must choose a domain where authentication endpoints will be created.

**Domain type**

**Use a Cognito domain**  
Enter an identifying prefix to use in an Amazon-owned domain. For production apps, we recommend using a custom domain instead.

**Use a custom domain**  
Enter a domain that you own for Cognito-hosted sign-up and sign-in pages. You must provide a DNS record and an AWS Certificate Manager (ACM) certificate to use a custom domain. We recommend using a custom domain for production workloads.

**Cognito domain**  
Enter a domain prefix.  
 .auth.ap-south-1.amazoncognito.com

Domain prefixes may only include lowercase, alphanumeric characters, and hyphens. You can't use the text aws, amazon, or cognito in the domain prefix. Your domain prefix must be unique within the current Region.

Available

9. Then in the initial app client you should choose a public client and give an App client name. For client secret choose Don't generate.

**Initial app client**  
Configure an app client. App clients are single-app platforms in your user pool that have permissions to call unauthenticated API operations. A user pool can have multiple app clients.

**App type | Info**  
Select an app type and we will automatically populate common default settings. You can add additional app clients after the user pool is created.

**Public client**  
A native, browser or mobile-device app. Cognito API requests are made from user systems that are not trusted with a client secret.

**Confidential client**  
A server-side application that can securely store a client secret. Cognito API requests are made from a central server.

**Other**  
A custom app. Choose your own grant, auth flow, and client-secret settings.

**App client name | Info**  
Enter a friendly name for your app client.

App client names are limited to 128 characters or less. Names may only contain alphanumeric characters, spaces, and the following special characters: + = , . @ -

**Client secret | Info**  
Choose whether your app client will have a client secret. Client secrets are used by the server-side component of an app to authorize API requests. Using a client secret can prevent a third party from impersonating your client.

Generate a client secret

Don't generate a client secret

10. Now for the allowed callback URL give any localhost and expand Advanced app client settings. Here you just need to check the application flows. These options should be selected which you can see below.

11. Then leave everything as it is and create your user pool.

## Allowed callback URLs | [Info](#)

Enter at least one callback URL to redirect the user back to after authentication. This is typically the URL for the app receiving the authorization code issued by Cognito. You may use HTTPS URLs, as well as custom URL schemes.

### URL

<https://localhost:8000>

[Remove](#)

Length of callback URL must be between 1 and 1024 characters. Valid characters are letters, marks, numbers, symbols, and punctuations. Amazon Cognito requires HTTPS over HTTP except for http://localhost for testing purposes only. App callback URLs such as myapp://example are also supported. Must not contain a fragment.

[Add another URL](#)

You can add 99 more URLs

## ▼ Advanced app client settings

We have populated suggested authentication flows, OAuth 2.0 grant types, and OIDC scopes based on the selections you made earlier.

## Authentication flows | [Info](#)

Choose authentication flows that your app will support. Refresh token authentication is always enabled. We have populated options based on your app type.

[Select authentication flows](#)

ALLOW\_REFRESH\_TOKEN\_AUTH [X](#)  
Refresh token based authentication

ALLOW\_USER\_SR\_P\_AUTH [X](#)  
SRP (secure remote password) protocol based authentication

ALLOW\_CUSTOM\_AUTH [X](#)  
Lambda trigger based custom authentication

12. Below you can see that your user pool has been created successfully. You can go inside it and check the configurations.

⌚ User pool 'demo-ecommerce-pool' has been created successfully. [View details](#) [X](#)

[Amazon Cognito](#) > User pools

**New from Amazon Verified Permissions!** Cognito user group authorization for API Gateway  
You can now create group-aware authorization policies for your APIs with Amazon Verified Permissions, a fine-grained authorization service for applications. [Learn more](#)

[Go to Amazon Verified Permissions](#)

**User pools (1) [Info](#)**

View and configure your user pools. User pools are directories of federated and local user profiles. They provide authentication options for your users.

User pool name	User pool ID	Created time	Last updated time
<a href="#">demo-ecommerce-pool</a>	ap-south-1_X7Tklyh83	1 second ago	1 second ago