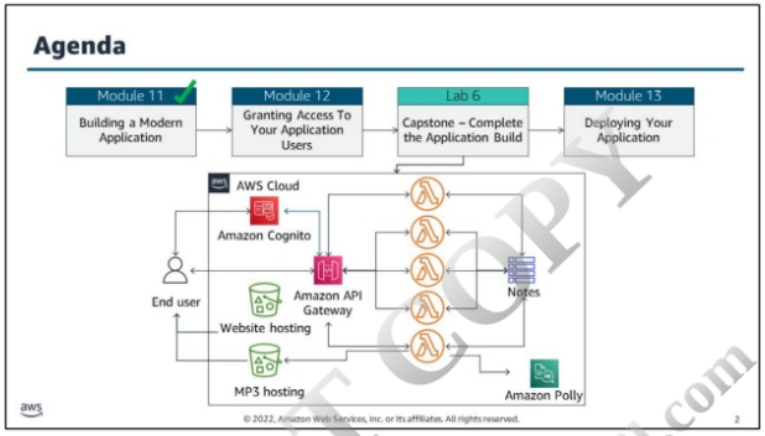
Printed by: sameersheik68@gmail.com. Printing is for personal, private use only. No part of this book may be reproduced or transmitted without publisher's prior permission. Violators will be prosecuted.

AWS Training and Certification

Module 12: Granting Access to Your Application Users



Module objectives

By the end of this module, you will be able to:

- Explore the authentication process using Amazon Cognito
- Manage user access and authorize serverless APIs
- Observe best practices for implementing Amazon Cognito
- Demonstrate the integration of Amazon Cognito and review the JWT token

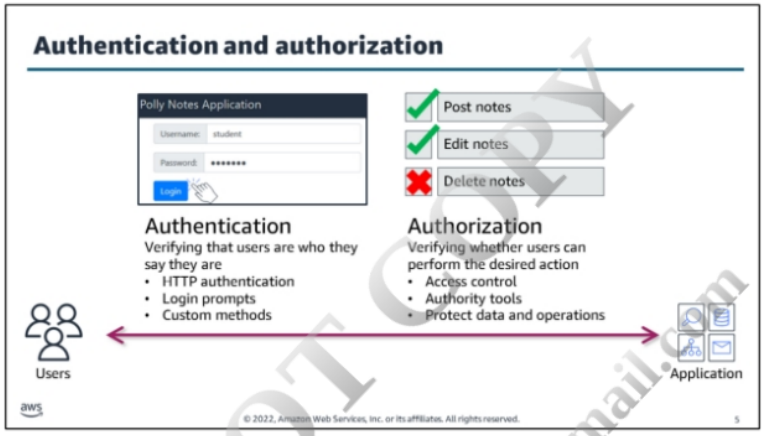
3

Printed by: sameersheik68@gmail.com. Printing is for personal, private use only. No part of this book may be reproduced or transmitted without publisher's prior permission. Violators will be prosecuted.

AWS Training and Certification
Module 12: Granting Access to Your Application Users

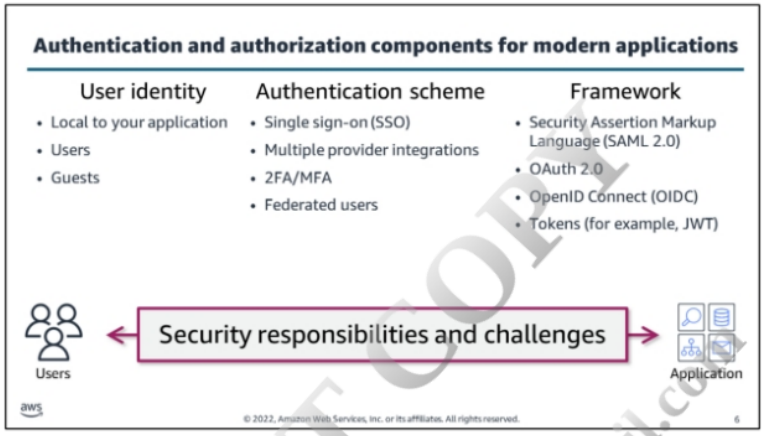
Complexities of authentication and authorization

Module 12: Granting Access to Your Application Users



Historically, application security from a user perspective required users to authenticate to the application to be granted authorizations. Users must verify to prove their identity before they are allowed to perform a desired action. User identity was woven into the application.

Printed by: sameersheik68@gmail.com. Printing is for personal, private use only. No part of this book may be reproduced or transmitted without publisher's prior permission. Violators will be prosecuted.

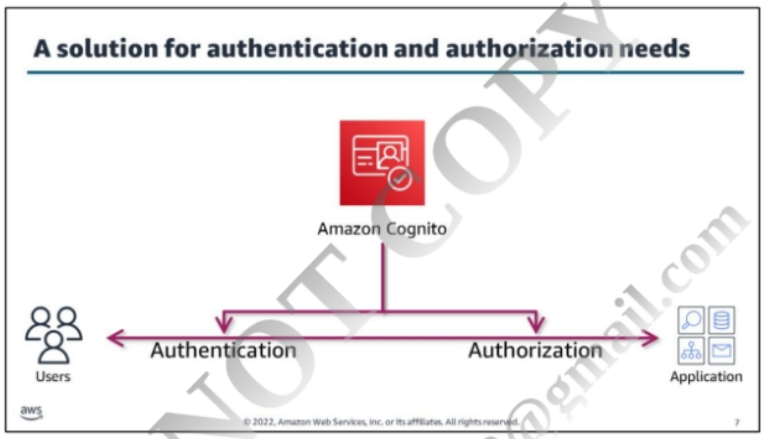


Modern applications are similar but include a greater set of challenges for a developer to track. Developers must manage users, implement a robust authentication scheme, and build a framework to enable security.

User identification and authentication are important to your company. Developers often spend time choosing and configuring the right services to support their applications.

User identity – Are users created locally in the application, or are they federated from another source? How will the application handle guest access?
Authentication scheme – Choosing and configuring the right services to support the applications.
Framework – What protocols and methods will the services require to implement the user security solution?

These are security responsibilities and challenges a developer must face when implementing a solution. Ask yourself whether you want to assume these responsibilities and challenges as a developer.

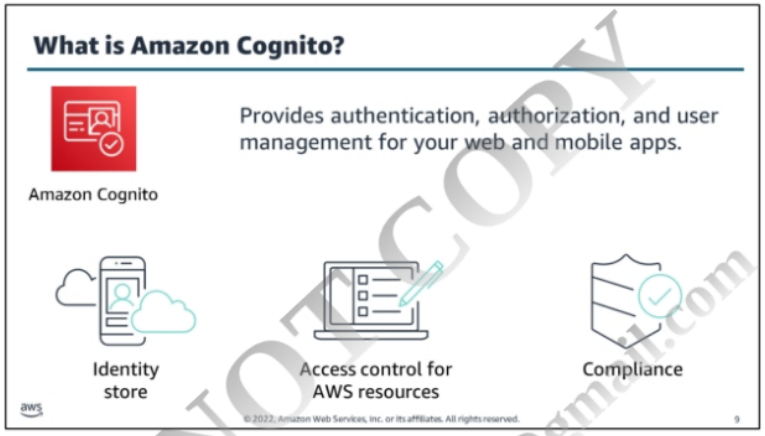


Amazon Cognito is the solution for the authentication and authorization needs of applications.

Printed by: sameersheik68@gmail.com. Printing is for personal, private use only. No part of this book may be reproduced or transmitted without publisher's prior permission. Violators will be prosecuted.

AWS Training and Certification
Module 12: Granting Access to Your Application Users

DO NOT COPY
sameersheik68@gmail.com



Using Amazon Cognito, your users can sign in directly to your applications with a user name and password. Alternatively, they can sign in through a third party such as Facebook, Login with Amazon, Google, or Sign in with Apple.

Amazon Cognito features include:

- **Identity store**
  An identity store for all your apps and users. With Amazon Cognito, your users can sign in through the following:
  - Social identity providers such as Google, Facebook, and Login with Amazon
  - Enterprise identity providers, such as Microsoft Active Directory, using SAML 2.0

- **Access control**
  With Amazon Cognito identity pools, you can create methods to grant user access to your AWS resources.

- **Support of multiple compliance programs**
  Advanced security features to protect your users.

Additional features
The following are additional features that you cam implement.

Printed by: sameersheik68@gmail.com. Printing is for personal, private use only. No part of this book may be reproduced or transmitted without publisher's prior permission. Violators will be prosecuted.

AWS Training and Certification Module 12: Granting Access to Your Application Users

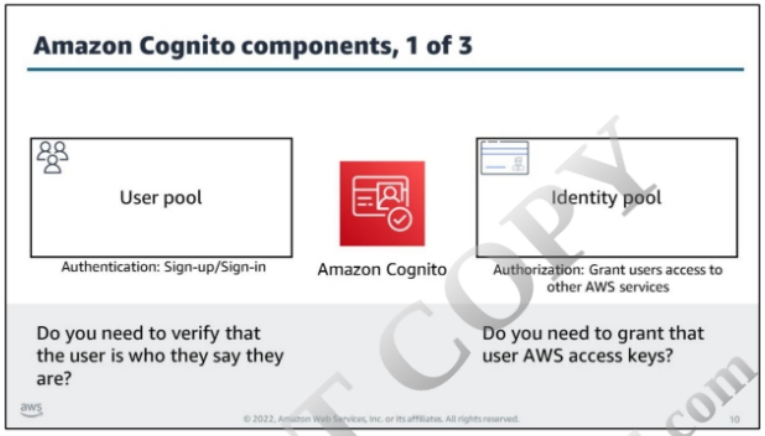
Integration

- Integration with AWS services
- Integration with external identify providers (IdPs) such as Google, Facebook, Sign in with Apple)
- Microsoft Active Directory integration

Security

- Data encrypted at rest and in transit; compliant with standards such as the following:
  - Payment card industry (PCI) data security standards (DSS)
  - System and Organization Controls (SOC)
  - International Organization for Standardization (ISO) 9001
- Standards-based authentication (OAuth 2.0, SAML 2.0, OpenID Connect)
- Multi-factor authentication (MFA)

© 2022 Amazon Web Services, Inc. or its affiliates. All rights reserved.



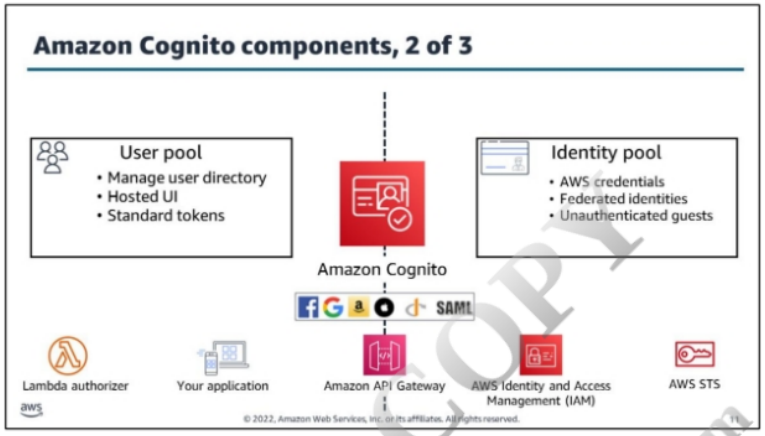
Amazon Cognito user pools
A user pool is a user directory in Amazon Cognito. With a user pool, your users can sign in to your web or mobile app through Amazon Cognito. Users can also sign in through social identity providers such as Google, Facebook, Login for Amazon, or Sign in with Apple, and through SAML 2.0 identity providers.

Amazon Cognito identity pools (federated identities)
Use Amazon Cognito identity pools (federated identities) to create unique identities for your users and federate them with identity providers. With an identity pool, you can obtain temporary, limited-privilege AWS credentials to access other AWS services.

Printed by: sameersheik68@gmail.com. Printing is for personal, private use only. No part of this book may be reproduced or transmitted without publisher's prior permission. Violators will be prosecuted.

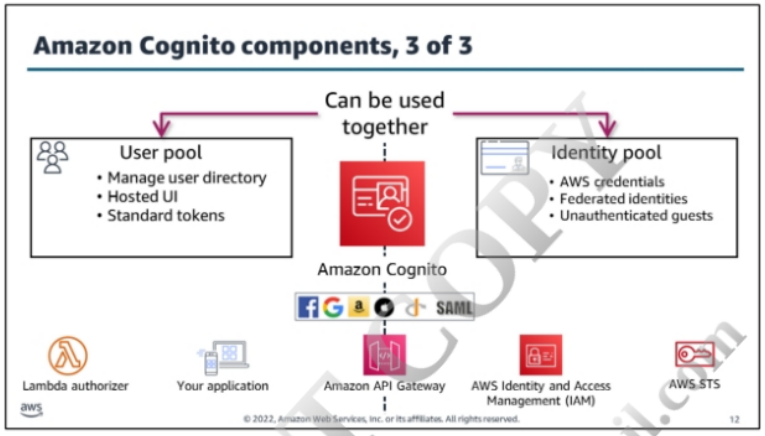
AWS Training and Certification

Module 12: Granting Access to Your Application Users



Amazon Cognito user pools
A user pool is a user directory in Amazon Cognito. With a user pool, your users can sign in to your web or mobile app through Amazon Cognito. Users can also sign in through social identity providers such as Google, Facebook, Login for Amazon, or Sign in with Apple, and through SAML 2.0 identity providers.

Amazon Cognito identity pools (federated identities)
Use Amazon Cognito identity pools (federated identities) to create unique identities for your users and federate them with identity providers. With an identity pool, you can obtain temporary, limited-privilege AWS credentials to access other AWS services.

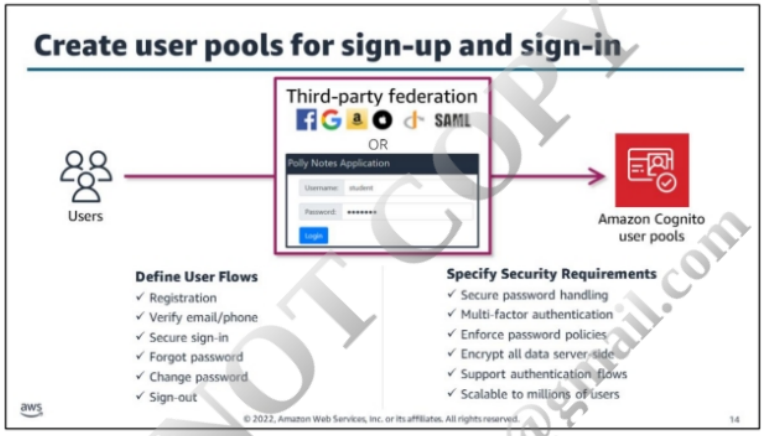


Amazon Cognito user pools and Amazon Cognito identity pools (federated identities) can be used together.

Printed by: sameersheik68@gmail.com. Printing is for personal, private use only. No part of this book may be reproduced or transmitted without publisher's prior permission. Violators will be prosecuted.

AWS Training and Certification Module 12: Granting Access to Your Application Users

DO NOT COPY
sameersheik68@gmail.com

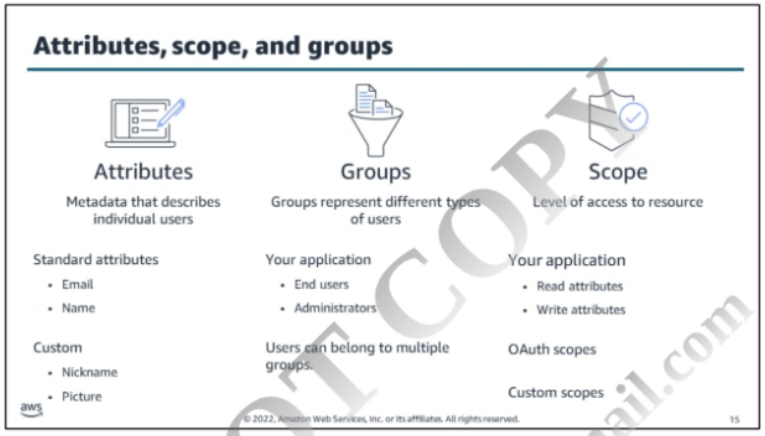


With Amazon Cognito User Pools, you can do the following:

- Choose how the user will sign in, such as options for verifying email, phone number, or user name.
- Require attributes for signing up and creating a user profile, such as email, birthdate, picture, custom attributes, and more.
- Specify policies (for example, password strength, require administrator for signup, and set expiration of temp passwords).
- Enable multi-factor authentication (MFA) and remembering user devices.
- Set options for recovering user accounts.
- Set options for sending short message service (SMS) and custom emails.
- Add tags so that you can categorize and manage user pools.
- Specify application clients with access to the user pool. App clients will receive a unique ID and an optional secret key to access the user pool.
- Set up customized workflows with triggers. You can select AWS Lambda functions to run with certain events (for example, pre-sign-up, post authentication).

Printed by: sameersheik68@gmail.com. Printing is for personal, private use only. No part of this book may be reproduced or transmitted without publisher's prior permission. Violators will be prosecuted.

AWS Training and Certification Module 12: Granting Access to Your Application Users



Attributes
Attributes are pieces of information that help you identify individual users, such as name, email, and phone number. You get a set of default attributes, called *standard attributes*, with all user pools. You can also add custom attributes to your user pool definition in the AWS Management Console.

Groups
With the support for groups in Amazon Cognito user pools, you can create and manage groups, add users to groups, and remove users from groups. Use groups to create collections of users to manage their permissions or to represent different types of users. You can assign an AWS Identity and Access Management (IAM) role to a group to define the permissions for members of a group.

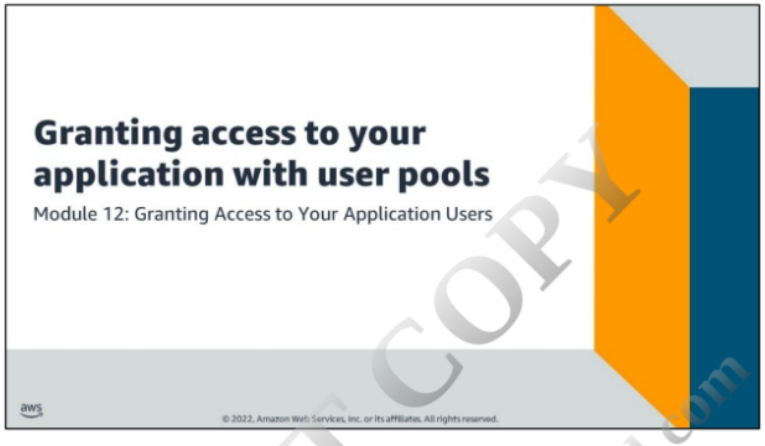
Because a user can belong to more than one group, each group can be assigned a precedence. This is a non-negative number that specifies the precedence of this group relative to the other groups that a user can belong to in the user pool. Zero is the top precedence value. Groups with lower precedence values take precedence over groups with higher or null precedence values. If a user belongs to two or more groups, the group with the lowest precedence value has the IAM role applied to the `cognito:preferred_role` claim in the user's ID token.

© 2022 Amazon Web Services, Inc. or its affiliates. All rights reserved.

Scope
A level of access that an application can request to a resource. Using Amazon Cognito, application developers can create their own OAuth 2.0 resource servers and define custom scopes in them. Custom scopes can then be associated with a client. The client can request them in OAuth2.0 authorization code grant flow, implicit flow, and client credentials flow. Custom scopes are added in the scope claim in the access token.

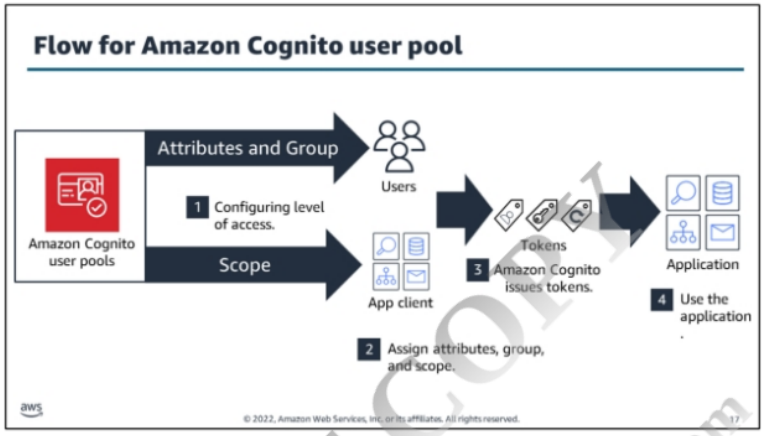
Printed by: sameersheik68@gmail.com. Printing is for personal, private use only. No part of this book may be reproduced or transmitted without publisher's prior permission. Violators will be prosecuted.

AWS Training and Certification Module 12: Granting Access to Your Application Users



DO NOT COPY
sameersheik68@gmail.com

© 2022 Amazon Web Services, Inc. or its affiliates. All rights reserved.



The Amazon Cognito user pool flow for granting access to the application is as follows:

1. **Configuring level of access.** Configure attributes to identify your application users, for example - username, email, etc. You can define which attributes are mandatory and needed to verify.
2. **Assign attributes, group, and scope.** Define which app clients will access this user pool. Create user groups as needed to manage collectively users' permissions or to represent different types of users. For each app client, you can define scope to control the level of access needed.
3. **Amazon Cognito issues tokens.** When a user signs into your app, Amazon Cognito verifies the login information. If the login is successful, Amazon Cognito creates a session and returns an ID, access, and refresh token for the authenticated user.
4. **Use the application.** You can use the tokens to grant your users access to your own server-side resources or to the Amazon API Gateway. Or you can exchange them for temporary AWS credentials to access other AWS services.

Printed by: sameersheik68@gmail.com. Printing is for personal, private use only. No part of this book may be reproduced or transmitted without publisher's prior permission. Violators will be prosecuted.

AWS Training and Certification Module 12: Granting Access to Your Application Users

Amazon Cognito tokens

Amazon Cognito tokens are of JSON Web Token (JWT) format. JWT tokens have three sections:

- **Header** – How to verify a token with encryption algorithm (alg) and the key ID (kid).
- **Payload** – Encoded information about the claim of the key. The ID token contains encoded user information, such as cognito:username, email, or phone number. The access token contains information about the authenticated user, including cognito:groups, username, and scope.
- **Signature** – Calculated based on the header and payload of the token.

ID token

Purpose: Contains details about your user, and confirms the identity.

Format: JWT, which defines the JSON object for sharing security information.

- **Header**– How to verify the token with encryption algorithm and key id
  { "kid" : "1234example=",
  "alg" : "RS256" }
- **Claim**– Payload with encoded user information, such as cognito:username, email, phone_number, and the following
  - **Sub** – Unique identifier for authenticated user
  - **Aud** – client_id that is used in the user authentication
  - **token_use**– Purpose of the token, such as ID
  - **auth_time**– When the authentication occurred

- **origin_jti**– JWT identifier that indicates where the authentication happened.
- **Jti** – Unique identifier of the JWT token

- **Signature**– Verify the key source and that it has not been altered.

You can set the ID token expiration to any value between 5 minutes and 1 day. This value can be set per application client.

Access Token
Format: JWT
Purpose: Authorize API operations of the user in the user pool, but does not send details about the user.

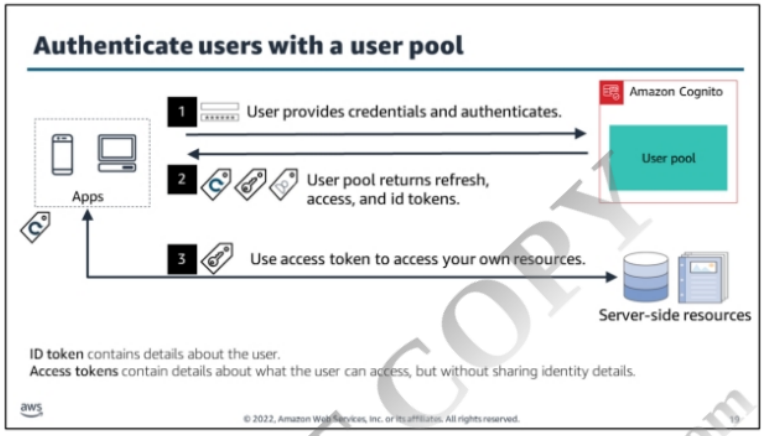
- **Header** – How to verify a token with encryption algorithm and key id).
- **Claim**– Information about the authenticated user, including cognito:groups, username, scope, and the following:
  - **sub**– Unique identifier (UUID) for the authenticated user.
  - **cognito:groups**– List of groups the user belongs to.
  - **token_use**– The intended purpose of the token, such as access.
  - **Scope** – List of OAuth 2.0 scopes that define what access the token provides.
  - **auth_time**– When the authentication occurred.
  - **iss**– Which user pool issued the claim.
  - **origin_jti**– The originating **JWT** identifier, from when the authentication occurred.
  - **jti**– Unique identifier of the JWT.
- Signature– Calculated based on the header and payload of the token.

Refresh Token
You can use the refresh token to retrieve new ID and access tokens. By default, the refresh token expires 30 days after your application user signs into your user pool. When you create an application for your user pool, you can set the application's refresh token expiration to any value between 60 minutes and 10 years.

You can set the access token expiration to any value between 5 minutes and 1 day. This value can be set per application client.

Printed by: sameersheik68@gmail.com. Printing is for personal, private use only. No part of this book may be reproduced or transmitted without publisher's prior permission. Violators will be prosecuted.

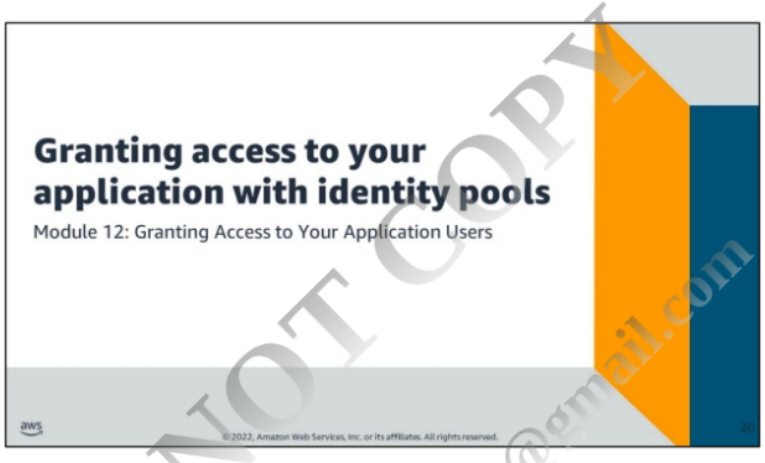
AWS Training and Certification Module 12: Granting Access to Your Application Users



You can enable your users to authenticate with a user pool. Your app users can sign in directly through a user pool or federate through a third-party identity provider (IdP). The user pool manages the overhead of handling the tokens that are returned from social sign-in through Facebook, Google, Login with Amazon, and Sign in with Apple. The user pool also manages the handling of tokens from OpenID Connect (OIDC) and SAML IdPs.

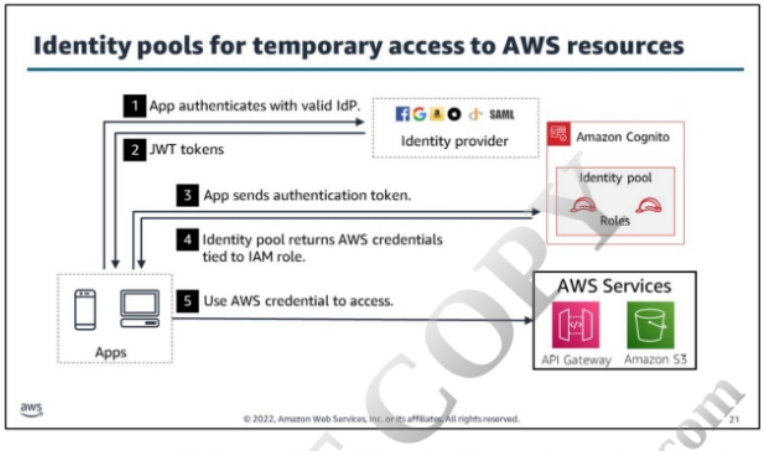
After a successful authentication, your web or mobile app will receive user pool tokens from Amazon Cognito. You can use those tokens to retrieve AWS credentials that allow your application to access other AWS services. Alternatively, you might choose to use them to control access to your server-side resources or to the Amazon API Gateway.

© 2022 Amazon Web Services, Inc. or its affiliates. All rights reserved.



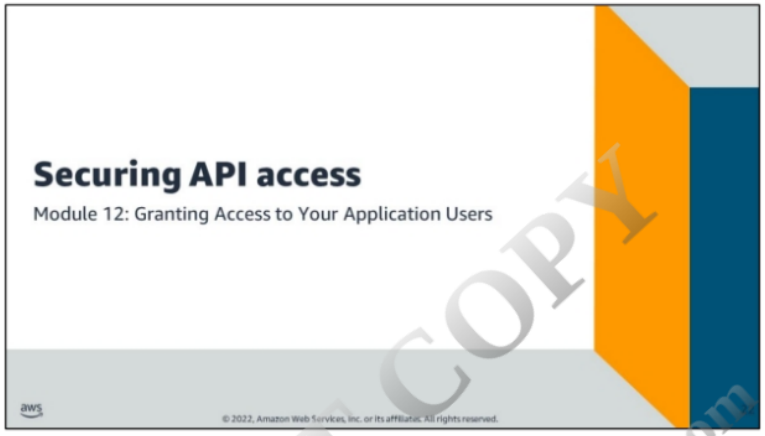
Printed by: sameersheik68@gmail.com. Printing is for personal, private use only. No part of this book may be reproduced or transmitted without publisher's prior permission. Violators will be prosecuted.

AWS Training and Certification Module 12: Granting Access to Your Application Users



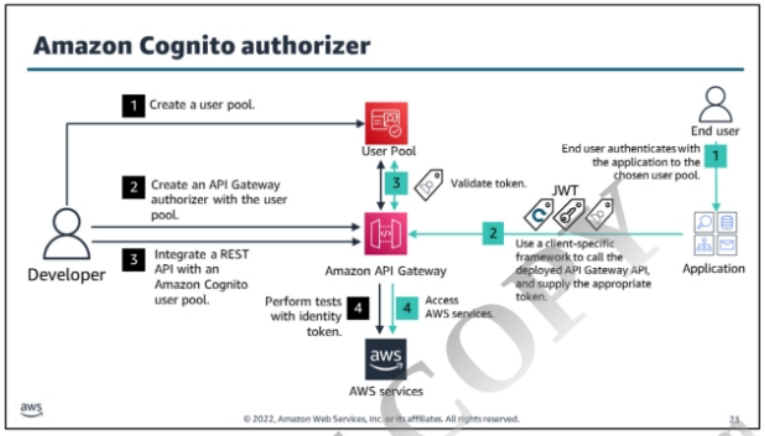
An identity pool is a store Amazon Cognito uses to keep your applications' federated identities organized. Identity pool associates federated identities from social identity providers with a unique user-specific identifier. Identity pools do not store any user profiles. With identity pools, you can authenticate users through an external identity provider. Identity pools provide temporary security credentials to access your application's backend resources in AWS or any service behind Amazon API Gateway.

An identity pool can be associated with one or many applications. If you use two different identity pools for two applications, the same end user will have a different unique identifier in each identity pool.



Printed by: sameersheik68@gmail.com. Printing is for personal, private use only. No part of this book may be reproduced or transmitted without publisher's prior permission. Violators will be prosecuted.

AWS Training and Certification Module 12: Granting Access to Your Application Users



Integrate Amazon Cognito APIs directly into your application by using the AWS SDK for your chosen programming language. Amazon Cognito APIs provide complete control over user experience and flows.

Creating and configure a user pool

To create and configure an Amazon Cognito user pool for your API, perform the following tasks:

1. Use the Amazon Cognito console, CLI/SDK, or API to create a user pool.
   You need user pool ID, client ID, and any client secret (if set any) for the rest of the configuration.
2. Use the API Gateway console, CLI/SDK, or API to create an API Gateway authorizer of *COGNITO_USER_POOLS* authorizer type with the chosen user pool.
3. Use the API Gateway console, CLI/SDK, or API to enable the authorizer on selected API methods. For Token source, use *Authorization* as the header name to pass the identity or access token that is returned by Amazon Cognito when a user signs in successfully.
4. Use the API Gateway Console or third-party tools to test the invocation by supplying an identity token that's provisioned from the user pool. For more information, see "Integrating Amazon Cognito With Web and Mobile Apps" in the *Amazon Cognito Developer Guide* (https://docs.aws.amazon.com/cognito/latest/developerguide/cognito-integrate-apps.html).

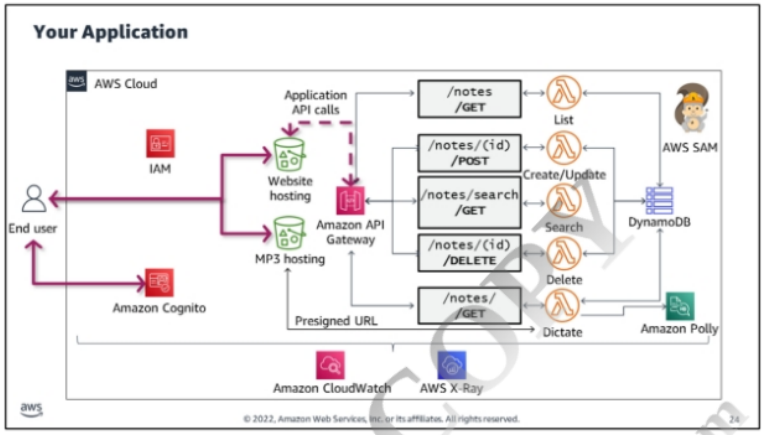
Calling API methods

To call any API methods with a user pool enabled, your API clients perform the following tasks:

1. Use the Amazon Cognito CLI/SDK or API to sign a user into the chosen user pool, and obtain an identity token or access token.
2. Use a client-specific framework to call the deployed API Gateway API and supply the appropriate token in the Authorization header.
3. API Gateway validates tokens with the Amazon Cognito user pool.
4. Invoke backend AWS services.

Printed by: sameersheik68@gmail.com. Printing is for personal, private use only. No part of this book may be reproduced or transmitted without publisher's prior permission. Violators will be prosecuted.

AWS Training and Certification Module 12: Granting Access to Your Application Users

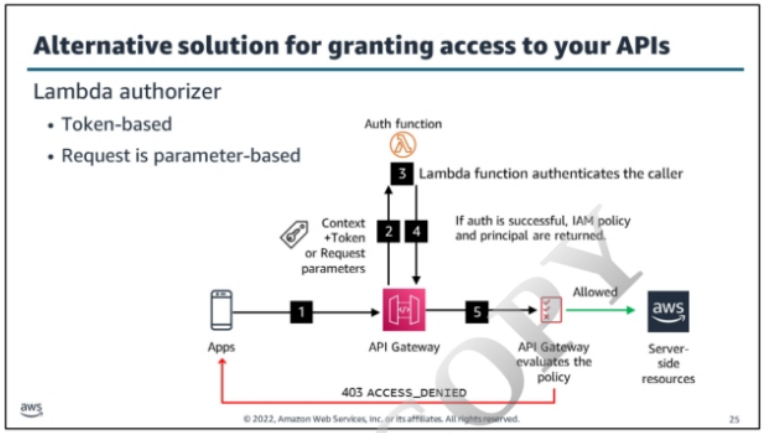


Amazon Cognito provides authentication, authorization, and user management for your web and mobile apps.

Users can sign in directly with a user name and password, or through a third party, such as Facebook, Login with Amazon, Google, or Sign in with Apple.

DO NOT COPY
sameersheik68@gmail.com

© 2022 Amazon Web Services, Inc. or its affiliates. All rights reserved. 516



Lambda authorizers
A Lambda authorizer is an API Gateway feature that uses a Lambda function to control access to your API.

The following are two types of Lambda authorizers:
- *Token-based Lambda authorizer* (also called a TOKEN authorizer) receives the caller's identity in a bearer token, such as a JWT or an OAuth token.
- *Request parameter-based Lambda authorizer* (also called a REQUEST authorizer) receives the caller's identity in a combination of headers, query string parameters, stageVariables, and $context variables.

API Gateway Lambda authorization workflow
1. The client calls a method on an API Gateway API method, passing a bearer token or request parameters.
2. API Gateway checks whether a Lambda authorizer is configured for the method. If it is, API Gateway calls the Lambda function.
3. The Lambda function authenticates the caller by means such as the following:
   - Calling out to an OAuth provider to get an OAuth access token.
   - Calling out to a SAML provider to get a SAML assertion.
   - Generating an IAM policy based on the request parameter values.
   - Retrieving credentials from a database.

Printed by: sameersheik68@gmail.com. Printing is for personal, private use only. No part of this book may be reproduced or transmitted without publisher's prior permission. Violators will be prosecuted.

AWS Training and Certification Module 12: Granting Access to Your Application Users

4. If the call succeeds, the Lambda function grants access by returning an output object containing at least an IAM policy and a principal identifier.
5. API Gateway evaluates the policy.
   - If access is denied, API Gateway returns a suitable HTTP status code, such as 403 ACCESS_DENIED.
   - If access is allowed, API Gateway runs the method. If caching is enabled in the authorizer settings, API Gateway also caches the policy so that the Lambda authorizer function doesn't need to be invoked again.

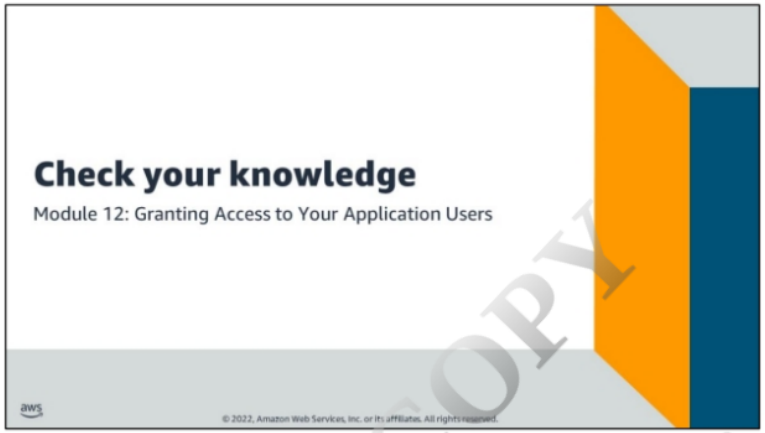
Authentication workflow for IAM identity-based policies
1. User federates into the application through a third-party party provider.
2. Amazon Cognito validates the user.
3. User issues a `GetOpenIdToken` and `AssumeRoleWithWebIdentity` requests.
4. User is validated with a token and issued AWS Security Token Service (AWS STS)
5. access to server-side resources.

For more information, see "Understanding Amazon Cognito Authentication Part 4: Enhanced Flow" in the Front-End Web & Mobile AWS blog (https://aws.amazon.com/blogs/mobile/understanding-amazon-cognito-authentication-part-4-enhanced-flow/).

© 2022 Amazon Web Services, Inc. or its affiliates. All rights reserved.

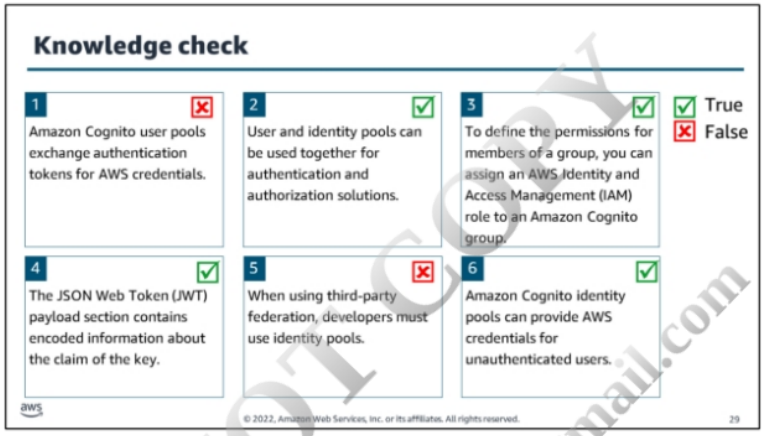
Printed by: sameersheik68@gmail.com. Printing is for personal, private use only. No part of this book may be reproduced or transmitted without publisher's prior permission. Violators will be prosecuted.

AWS Training and Certification
Module 12: Granting Access to Your Application Users



DO NOT COPY
sameersheik68@gmail.com

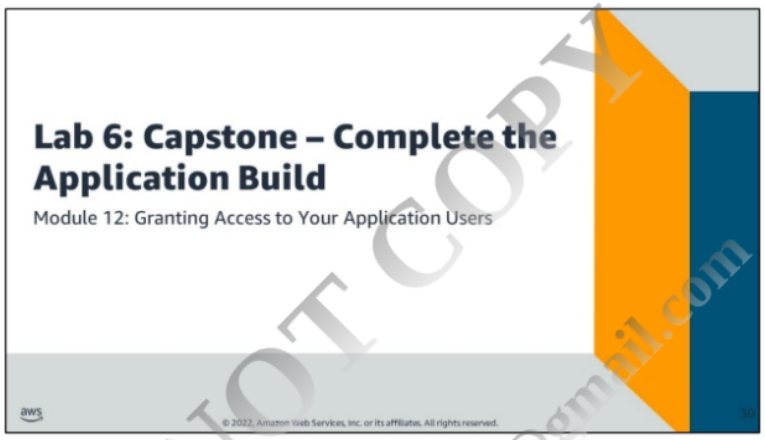
© 2022 Amazon Web Services, Inc. or its affiliates. All rights reserved.



1. (False) Amazon Cognito identity pools exchange authentication tokens for AWS credentials.
2. (True)
3. (True)
4. (True)
5. (False) Third-party federation is supported for both user pools and identity pools.
6. (True)

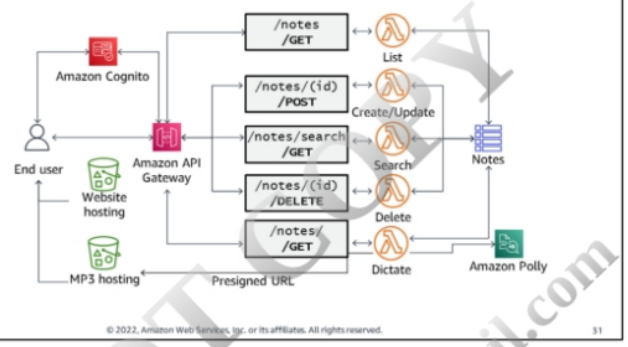
Printed by: sameersheik68@gmail.com. Printing is for personal, private use only. No part of this book may be reproduced or transmitted without publisher's prior permission. Violators will be prosecuted.

AWS Training and Certification Module 12: Granting Access to Your Application Users



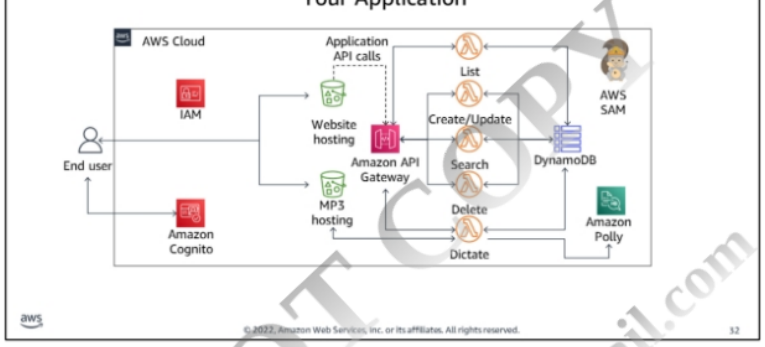
DO NOT COPY
sameersheik68@gmail.com

Lab 6 workflow: Authenticate users with Amazon Cognito



Printed by: sameersheik68@gmail.com. Printing is for personal, private use only. No part of this book may be reproduced or transmitted without publisher's prior permission. Violators will be prosecuted.

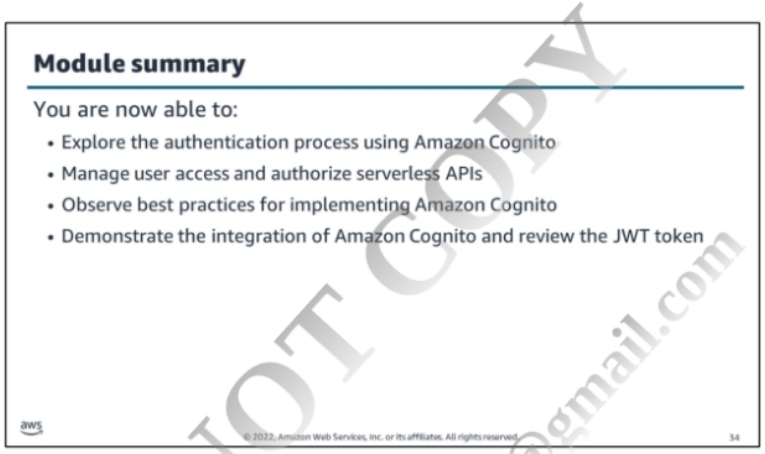
Lab 6 workflow: Complete the application build



1. IDE setup
2. Hosting your application
3. Database solutions
4. Compute with Lambda
5. Amazon API Gateway

Printed by: sameersheik68@gmail.com. Printing is for personal, private use only. No part of this book may be reproduced or transmitted without publisher's prior permission. Violators will be prosecuted.

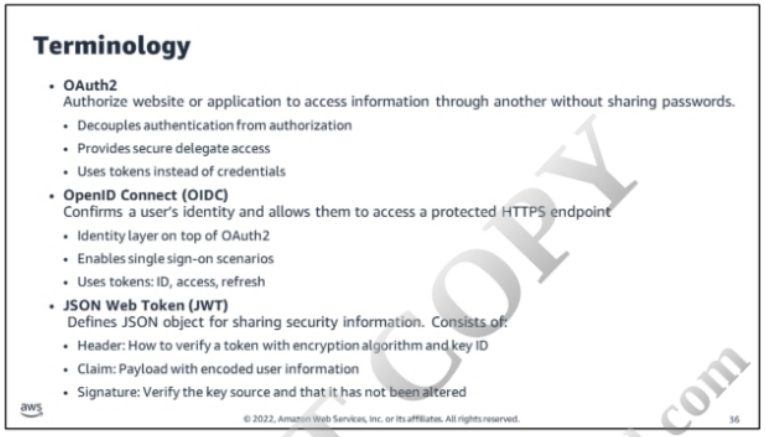
AWS Training and Certification
Module 12: Granting Access to Your Application Users



Printed by: sameersheik68@gmail.com. Printing is for personal, private use only. No part of this book may be reproduced or transmitted without publisher's prior permission. Violators will be prosecuted.

AWS Training and Certification

Module 12: Granting Access to Your Application Users



DO NOT COPY
sameersheik68@gmail.com