AWS re:Invent

SEC403-R

AWS Identity: Using Amazon Cognito for serverless consumer apps

Jesse Fuchs
Sr. Security Solutions Architect
Amazon Web Services

Greg McConnel
Sr. SA Manager, Security & Compliance
Amazon Web Services

Agenda

Overview

User sign-up and sign-in

Backend authorization with Amazon API Gateway

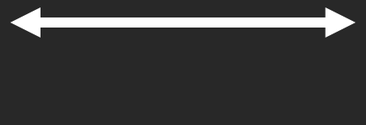
Retrieving and using temporary AWS credentials

Wrap-up

Amazon Cognito overview

Web and Mobile Apps

Developers focus on what is special about their app



Amazon Cognito

Amazon Cognito handles authentication and identity

Managed User Directory

Hosted UI

Standard Tokens

Federation

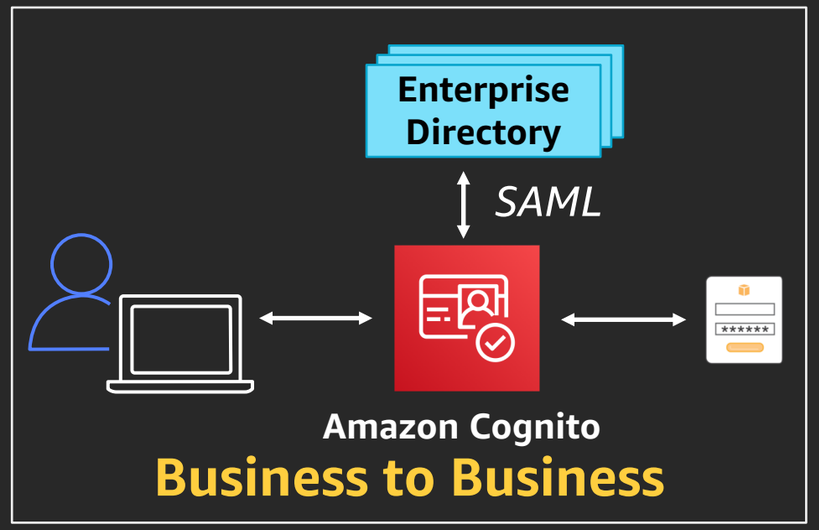
SAML

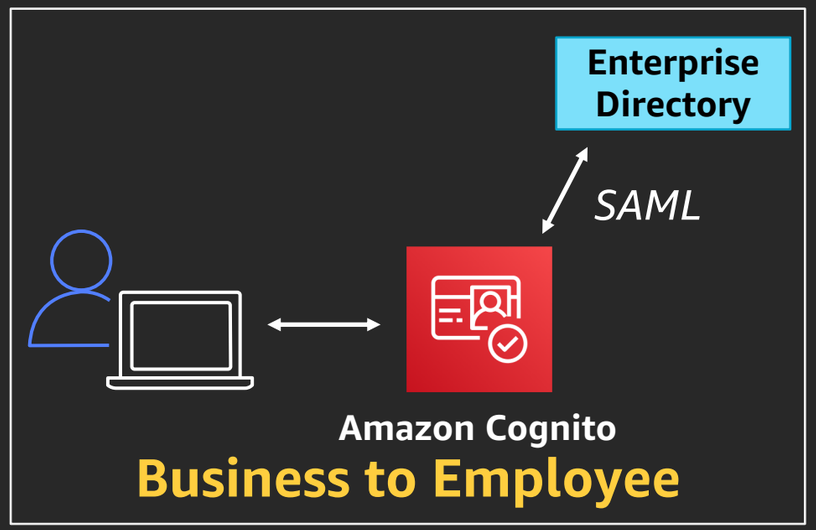
AWS Credentials

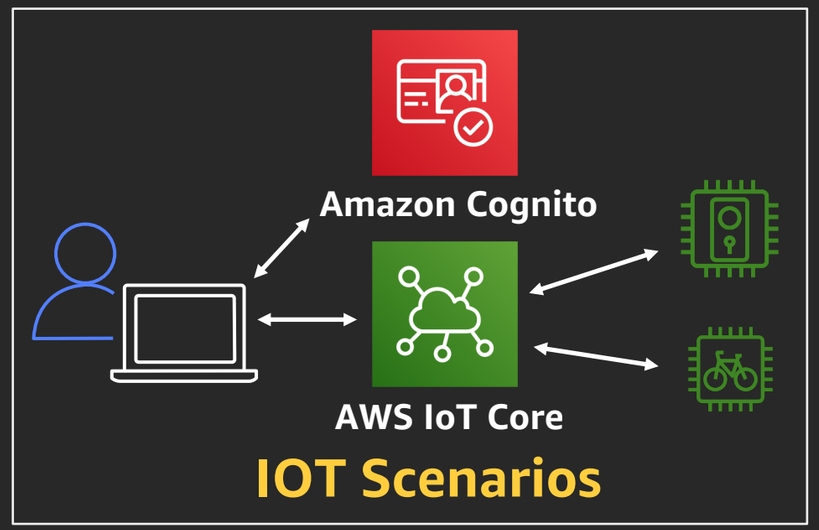
User Pools

Identity Pools

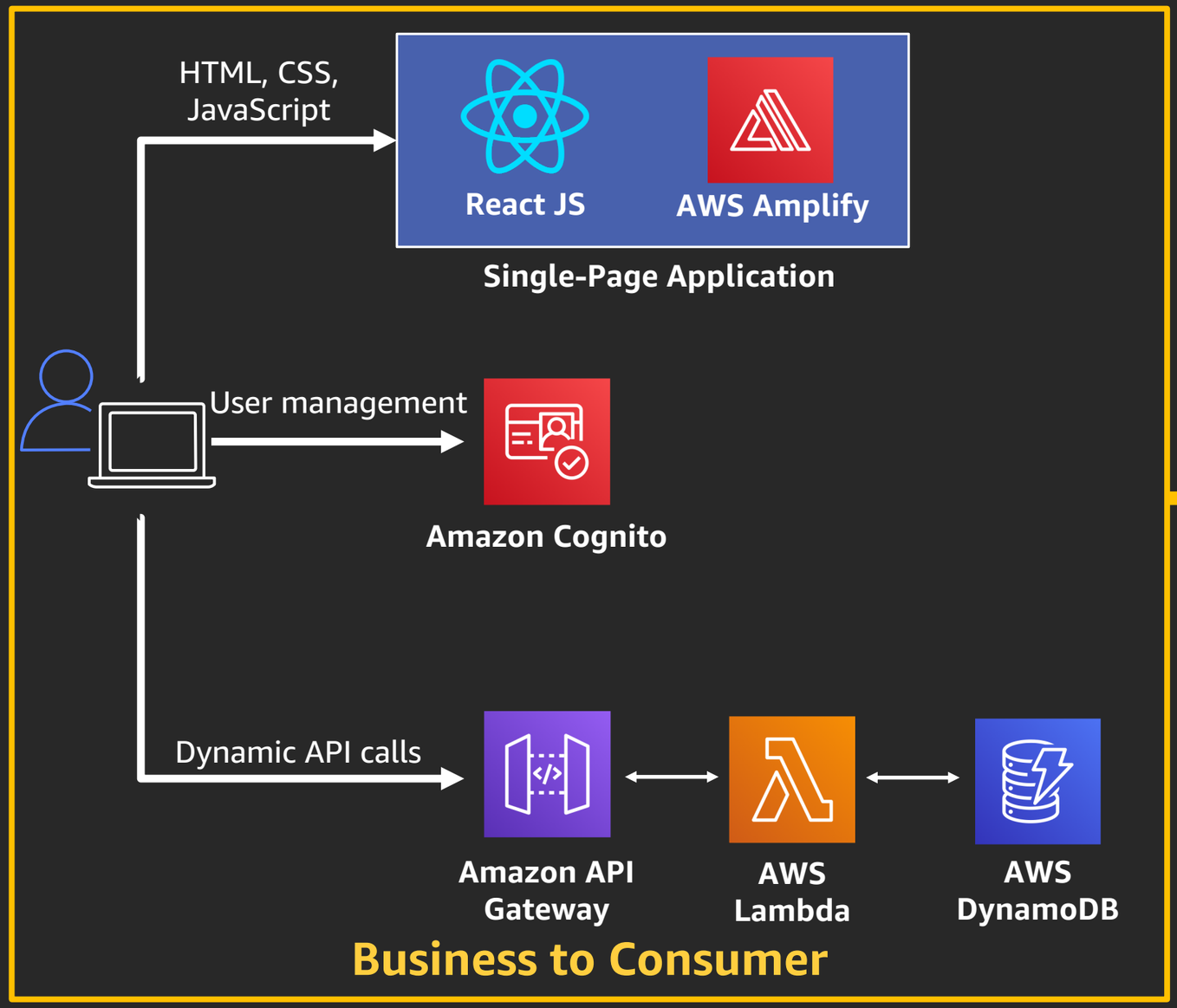
Amazon Cognito: Identity management scenarios

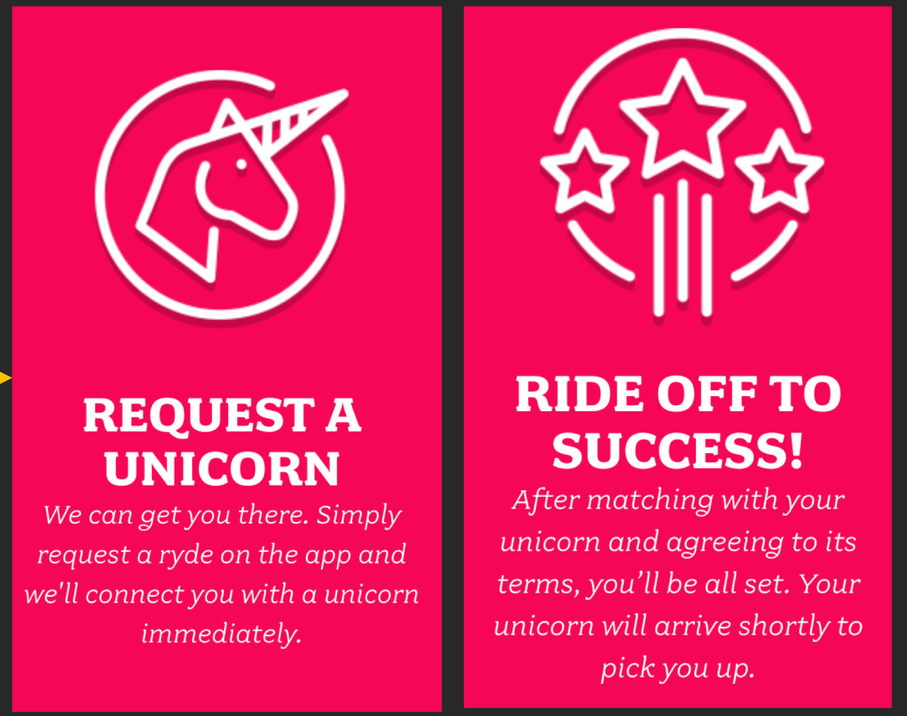






Today's workshop – Wild Rydes





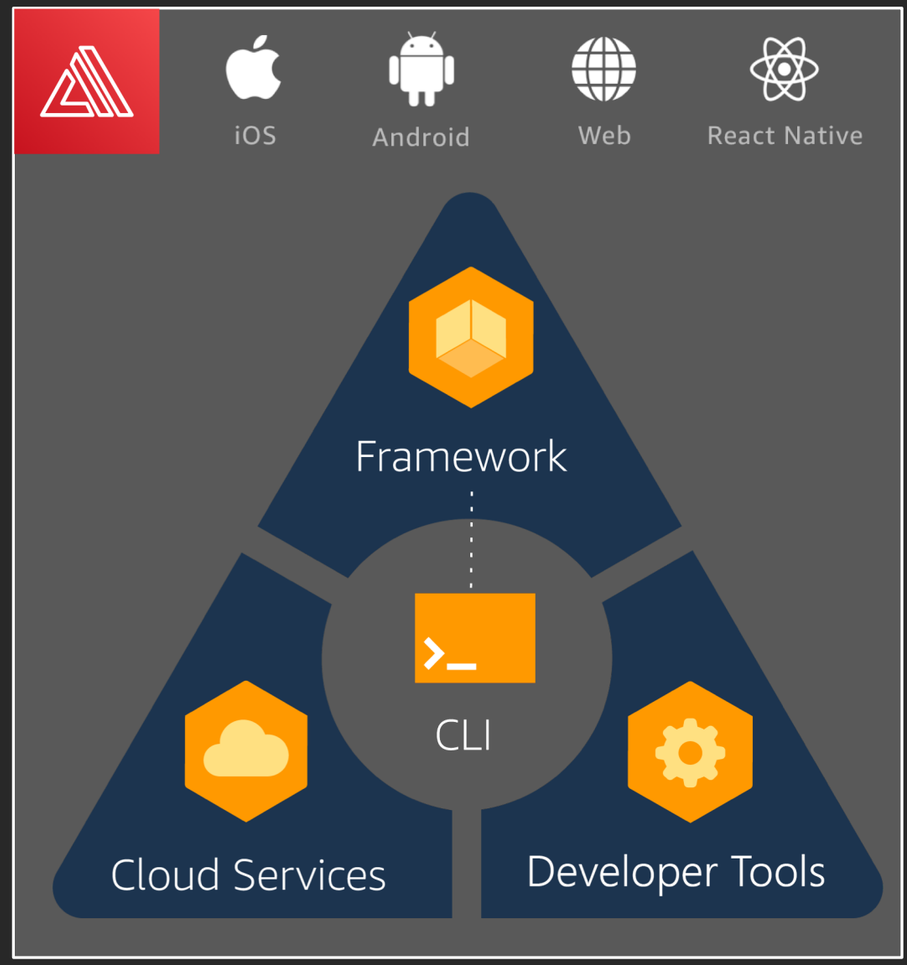
Development with AWS Amplify

The fastest way to develop cloud-powered apps

The **Amplify Framework**, an open-source client framework, includes libraries, a CLI toolchain, and UI components

The **CLI toolchain** enables easy integration with cloud services such as Amazon Cognito, AWS AppSync, and Amazon Pinpoint

Developer tools for building, testing, deploying, and hosting the entire app – frontend and backend



Best for: Native mobile apps and JavaScript-based web apps.

User sign-up and sign-in

Amazon Cognito user pools – Comprehensive user flows

User Sign-Up and Sign-In

User Profile Data

Reset Password

Token-based Authentication

Email or Phone Number Verification

Multi-factor Authentication

Customize these User Flows Using AWS Lambda

Custom user flows using Lambda triggers

Category	Lambda trigger	Example scenarios
Custom authentication flow	**Define auth challenge**	Determines the next challenge in a custom auth flow
	Create auth challenge	Creates a challenge in a custom auth flow
	Verify auth challenge response	Determines whether a response is correct in a custom auth flow
Authentication events	**Pre-authentication**	Custom validation to accept or deny the sign-in request
	Post-authentication	Event logging for custom analytics
	Pre-token generation	Customize claims in the ID token
Sign-up	**Pre-sign-up** ← Module 1	Custom validation to accept or deny the sign-up request
	Post-confirmation	Custom welcome messages or event logging for custom analytics
	Migration	Migrate users and retain existing passwords
Messages	**Custom message** ← Module 1	Advanced customization and localization of messages

Amazon Cognito user pool tokens overview

Identity Token

- JSON Web Token (JWT)
- Can be used for authentication
- Includes user profile information
  - Attributes
  - Amazon Cognito groups
- Expires in 1 hour

Access Token

- JSON Web Token (JWT)
- Used to authorize requests including APIs
- Includes
  - OAuth scopes
  - Amazon Cognito groups
- Expires in 1 hour

Refresh Token

- Opaque blob
- Used to get new Id and Access tokens without re-authenticating
- Expiration configurable from 1 day to 10 years

Dissecting a JSON Web Token (JWT)


```
eyJraWQiOiI5ZXJydERLbHRxOFl3YUp5MkdadE9ieWtSREVB
OVNCNGlEVDZ2V21UZVFFPSIsImFsZyI6IlJTMjU2In0.eyJz
dWIiOiI2ZjU1NzM2OC1hODg0LTQ4NGUtYjY2Mi05ZmM2OWYz
YzM4MDIiLCJhdWQiOiI2bGtmczcwcm92a3ViaXJoMXF0bnR2
ajAxMiIsImVtYWlsX3ZlcmlmaWVkIjp0cnVlLCJ0b2tlbl91
c2UiOiJpZCIsImF1dGhfdGltZSI6MTQ3ODQ0OTA2MCwiaXNz
IjoiaHR0cHM6XC9cL2NvZ25pdG8taWRwLnVzLWVhc3QtMS5h
bWF6b25hd3MuY29tXC91cy1lYXN0LTFfWE1sVVc5c1V5Iiwi
Y29nbml0bzp1c2VybmFtZSI6InRlc3QxMjMiLCJleHAiOjE0
Nzg0NTI2NjAsImdpdmVuX25hbWUiOiJUZXN0IiwiaWF0Ijox
NDc4NDQ5MDYwLCJmYW1pbHlfbmFtZSI6IlRlc3QiLCJlbWFp
bCI6InRyYW5qaW1AYW1hem9uLmNvbSJ9.atQO0SJg9V97d6t
YonHNx0q7Zuof8-d-q0u69zNnuSJtmzGvOAW97tP2e3GydY9
K8q_2kG2IzkpEMUEdaeWjz2qG5dS328Scm6pRDPpC5pOkU8y
mjH7DBPfVXhtgS3iOhyleFhtmaTaYb_lYLpaaV10m8sVFOMH
tjdfrAm26Fq7zyjWYTSfzhqud29Ti4zn9PhcE7aL3s7BB8CJ
18_yFXSoG5CYCpLszvHazx1cbmPoXFrlFlPvZ07Oy8EbOaGs
4CukmoYiV-5RnZsA9JXj405Kp50k-v8HCL6ZACDw3OYMV87P
e6PuEqbzQLlc8BufKThm0xBiO6NJtvI7iC2sEIQ
```


- Open standard (RFC 7519)
- Compact, URL-safe means of representing claims
- Used for securely transmitting information between parties
- Digitally signed
- Optionally encrypted

Dissecting a JWT

eyJraWQiOiI5ZXJydERLbHRxOFl3YUp5MkdadE9ieWtSREVB
OVNCNGlEVDZ2V21UZVFFPSIsImFsZyI6IlJTMjU2In0.eyJz
dWIiOiI2ZjU1NzM2OC1hODg0LTQ4NGUtYjY2Mi05ZmM2OWYz
YzM4MDIiLCJhdWQiOiI2bGtmczcwcm92a3ViaXJoMXF0bnR2
ajAxMiIsImVtYWlsX3ZlcmlmaWVkIjp0cnVlLCJ0b2tlbl91
c2UiOiJpZCIsImF1dGhfdGltZSI6MTQ3ODQ0OTA2MCwiaXNz
IjoiaHR0cHM6XC9cL2NvZ25pdG8taWRwLnVzLWVhc3QtMS5h
bWF6b25hd3MuY29tXC91cy1lYXN0LTFfWE1sVVc5c1V5Iiwi
Y29nbml0bzp1c2VybmFtZSI6InRlc3QxMjMiLCJleHAiOjE0
Nzg0NTI2NjAsImdpdmVuX25hbWUiOiJUZXN0IiwiaWF0Ijox
NDc4NDQ5MDYwLCJmYW1pbHlfbmFtZSI6IlRlc3QiLCJlbWFp
bCI6InRyYW5qaW1AYW1hem9uLmNvbSJ9.atQO0SJg9V97d6t
YonHNx0q7Zuof8-d-q0u69zNnuSJtmzGvOAW97tP2e3GydY9
K8q_2kG2IzkpEMUEdaeWjz2qG5dS328Scm6pRDPpC5pOkU8y
mjH7DBPfVXhtgS3iOhyleFhtmaTaYb_lYLpaaV10m8sVFOMH
tjdfrAm26Fq7zyjWYTSfzhqud29Ti4zn9PhcE7aL3s7BB8CJ
18_yFXSoG5CYCpLszvHazx1cbmPoXFrlFlPvZ07Oy8EbOaGs
4CukmoYiV-5RnZsA9JXj405Kp50k-v8HCL6ZACDw3OYMV87P
e6PuEqbzQLlc8BufKThm0xBiO6NJtvI7iC2sEIQ

Header

```
{
  "kid":"9errtDKltq8YwaJy2GZtObykRDEA9SB4iDT6vWmTeQE=",
  "alg":"RS256”
}
```

Public Key: **https://cognito-idp.{region}.amazonaws.com/{userPoolId}/.well-known/jwks.json**

Dissecting a JWT

eyJraWQiOiI5ZXJydERLbHRxOFl3YUp5MkdadE9ieWtSREVB
OVNCNGlEVDZ2V21UZVFFPSIsImFsZyI6IlJTMjU2In0.eyJz
dWIiOiI2ZjU1NzM2OC1hODg0LTQ4NGUtYjY2Mi05ZmM2OWYz
YzM4MDIiLCJhdWQiOiI2bGtmczcwcm92a3ViaXJoMXF0bnR2
ajAxMiIsImVtYWlsX3ZlcmlmaWVkIjp0cnVlLCJ0b2tlbl91
c2UiOiJpZCIsImF1dGhfdGltZSI6MTQ3ODQ0OTA2MCwiaXNz
IjoiaHR0cHM6XC9cL2NvZ25pdG8taWRwLnVzLWVhc3QtMS5h
bWF6b25hd3MuY29tXC91cy1lYXN0LTFfWE1sVVc5c1V5Iiwi
Y29nbml0bzp1c2VybmFtZSI6InRlc3QxMjMiLCJleHAiOjE0
Nzg0NTI2NjAsImdpdmVuX25hbWUiOiJUZXN0IiwiaWF0Ijox
NDc4NDQ5MDYwLCJmYW1pbHlfbmFtZSI6IlRlc3QiLCJlbWFp
bCI6InRyYW5qaW1AYW1hem9uLmNvbSJ9.atQO0SJg9V97d6t
YonHNx0q7Zuof8-d-q0u69zNnuSJtmzGvOAW97tP2e3GydY9
K8q_2kG2IzkpEMUEdaeWjz2qG5dS328Scm6pRDPpC5pOkU8y
mjH7DBPfVXhtgS3iOhyleFhtmaTaYb_lYLpaaV10m8sVFOMH
tjdfrAm26Fq7zyjWYTSfzhqud29Ti4zn9PhcE7aL3s7BB8CJ
18_yFXSoG5CYCpLszvHazx1cbmPoXFrlFlPvZ07Oy8EbOaGs
4CukmoYiV-5RnZsA9JXj405Kp50k-v8HCL6ZACDw3OYMV87P
e6PuEqbzQLlc8BufKThm0xBiO6NJtvI7iC2sEIQ

Header

```
{
  "kid":"9errtDKltq8YwaJy2GZtObykRDEA9SB4iDT6vWmTeQE=",
  "alg":"RS256”
}
```

Payload

```
{
  "sub":"6f557368-a884-484e-b662-9fc69f3c3802",
  "aud":"6lkfs70rovkubirh1qtntvj012",
  "email_verified":true,
  "token_use":"id",
  "auth_time":1478449060,
  "iss":"https://cognito..aws.com/us-east-1_XMlUW9sUy",
  "cognito:username":”jane|example|com",
  "custom:genre":"jazz",
  "exp":1478452660,
  "given_name":"Test",
  "iat":1478449060,
  "phone_number":"+12345550100"
}
```

Dissecting a JWT

eyJraWQiOiI5ZXJydERLbHRxOFl3YUp5MkdadE9ieWtSREVB
OVNCNGlEVDZ2V21UZVFFPSIsImFsZyI6IlJTMjU2In0.eyJz
dWIiOiI2ZjU1NzM2OC1hODg0LTQ4NGUtYjY2Mi05ZmM2OWYz
YzM4MDIiLCJhdWQiOiI2bGtmczcwcm92a3ViaXJoMXF0bnR2
ajAxMiIsImVtYWlsX3ZlcmlmaWVkIjp0cnVlLCJ0b2tlbl91
c2UiOiJpZCIsImF1dGhfdGltZSI6MTQ3ODQ0OTA2MCwiaXNz
IjoiaHR0cHM6XC9cL2NvZ25pdG8taWRwLnVzLWVhc3QtMS5h
bWF6b25hd3MuY29tXC91cy1lYXN0LTFfWE1sVVc5c1V5Iiwi
Y29nbml0bzp1c2VybmFtZSI6InRlc3QxMjMiLCJleHAiOjE0
Nzg0NTI2NjAsImdpdmVuX25hbWUiOiJUZXN0IiwiaWF0Ijox
NDc4NDQ5MDYwLCJmYW1pbHlfbmFtZSI6IlRlc3QiLCJlbWFp
bCI6InRyYW5qaW1AYW1hem9uLmNvbSJ9.atQO0SJg9V97d6t
YonHNx0q7Zuof8-d-q0u69zNnuSJtmzGvOAW97tP2e3GydY9
K8q_2kG2IzkpEMUEdaeWjz2qG5dS328Scm6pRDPpC5pOkU8y
mjH7DBPfVXhtgS3iOhyleFhtmaTaYb_lYLpaaV10m8sVFOMH
tjdfrAm26Fq7zyjWYTSfzhqud29Ti4zn9PhcE7aL3s7BB8CJ
18_yFXSoG5CYCpLszvHazx1cbmPoXFrlFlPvZ07Oy8EbOaGs
4CukmoYiV-5RnZsA9JXj405Kp50k-v8HCL6ZACDw3OYMV87P
e6PuEqbzQLlc8BufKThm0xBiO6NJtvI7iC2sEIQ

Header

```
{
  "kid":"9errtDKltq8YwaJy2GZtObykRDEA9SB4iDT6vWmTeQE=",
  "alg":"RS256”
}
```

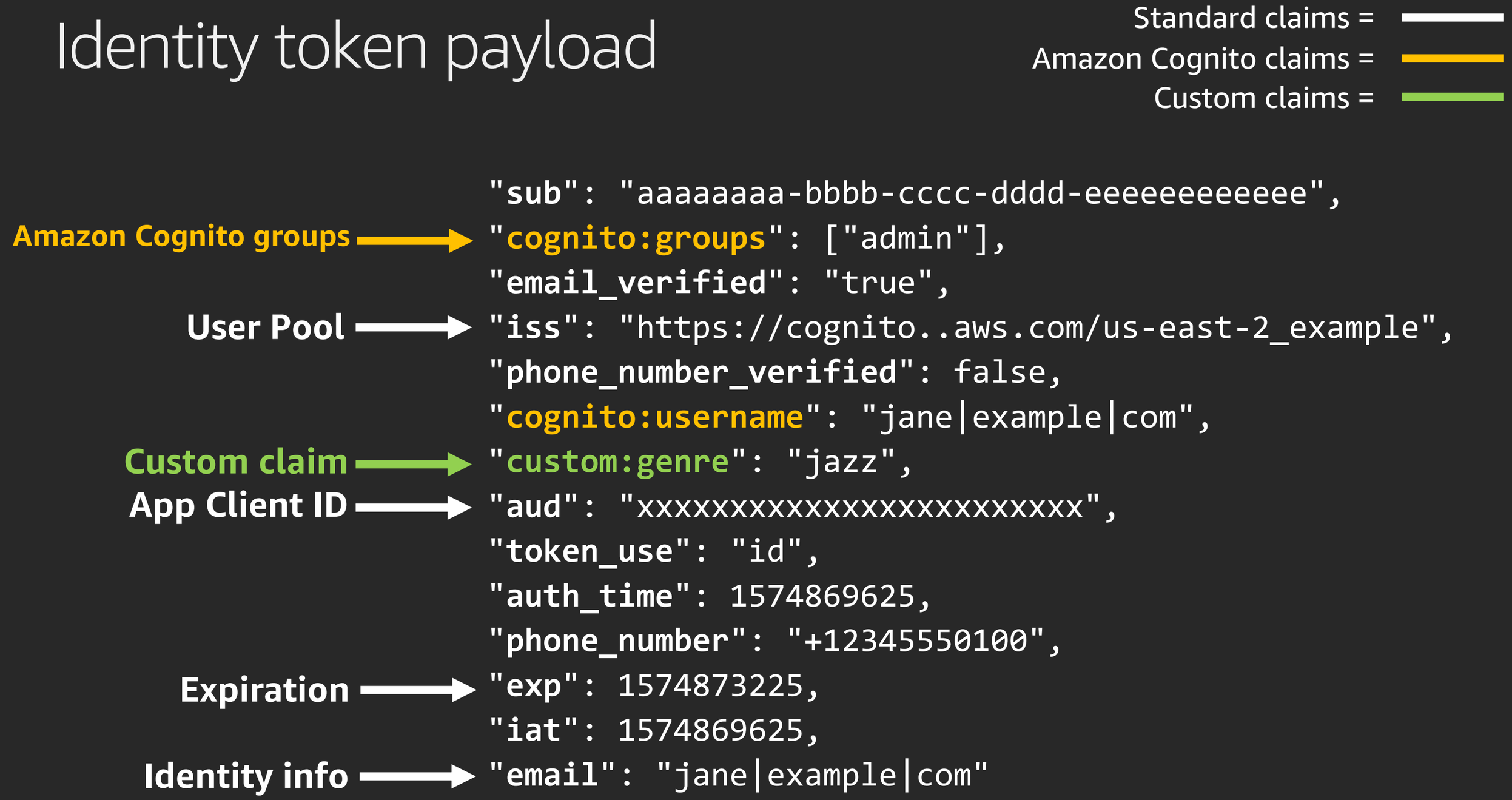
Payload

```
{
  "sub":"6f557368-a884-484e-b662-9fc69f3c3802",
  "aud":"6lkfs70rovkubirh1qtntvj012",
  "email_verified":true,
  "token_use":"id",
  "auth_time":1478449060,
  "iss":"https://cognito..aws.com/us-east-1_XMlUW9sUy",
  "cognito:username":”jane|example|com",
  "custom:genre":"jazz",
  "exp":1478452660,
  "given_name":"Test",
  "iat":1478449060,
  "phone_number":"+12345550100"
}
```

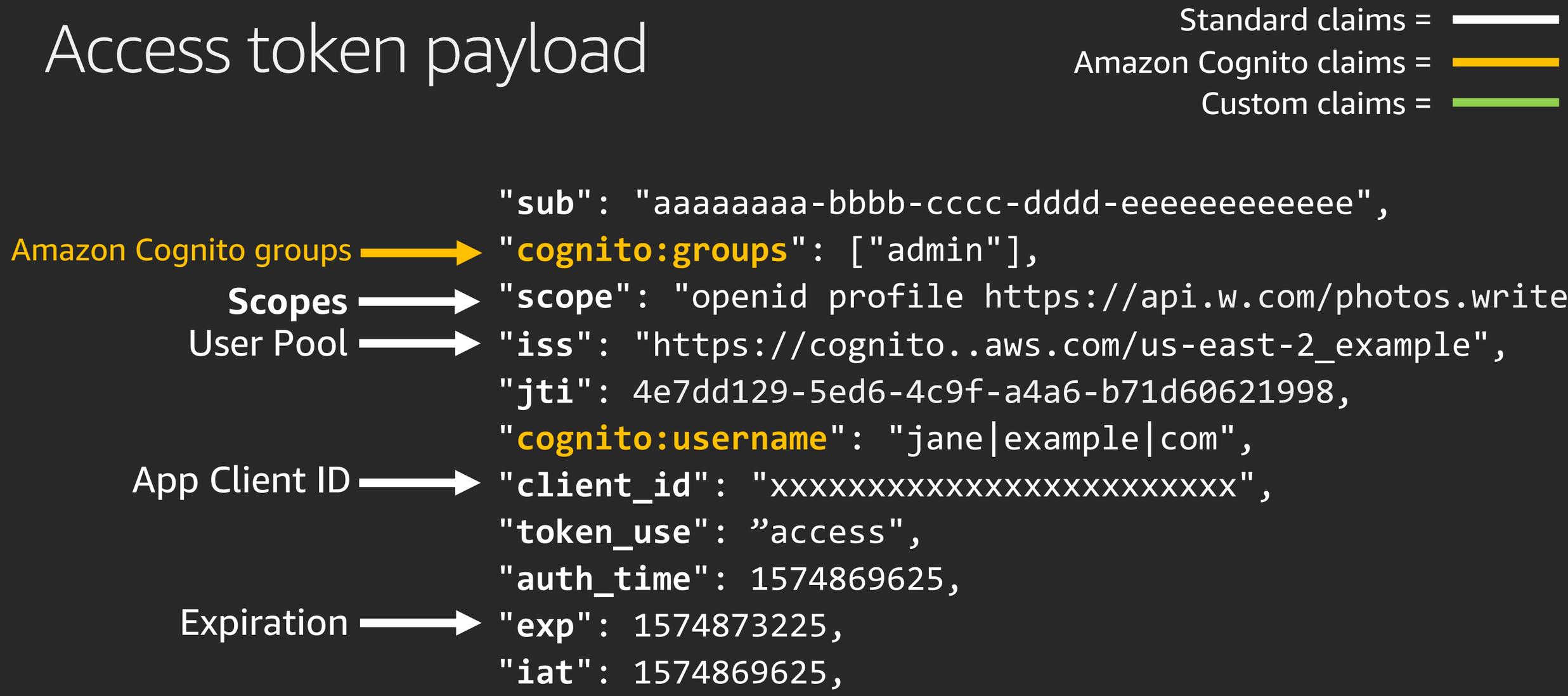
Signature

```
HMACSHA256(base64UrlEncode(header) + "." +
base64UrlEncode(payload), {secret});
```

Identity token payload



Access token payload



Module 0 and 1

Before you start…

- **Hash codes (use your assigned code)**
- **The CloudFormation stack has already been deployed. Make sure you read the Event Engine options in the instructions.**

Workshop guide

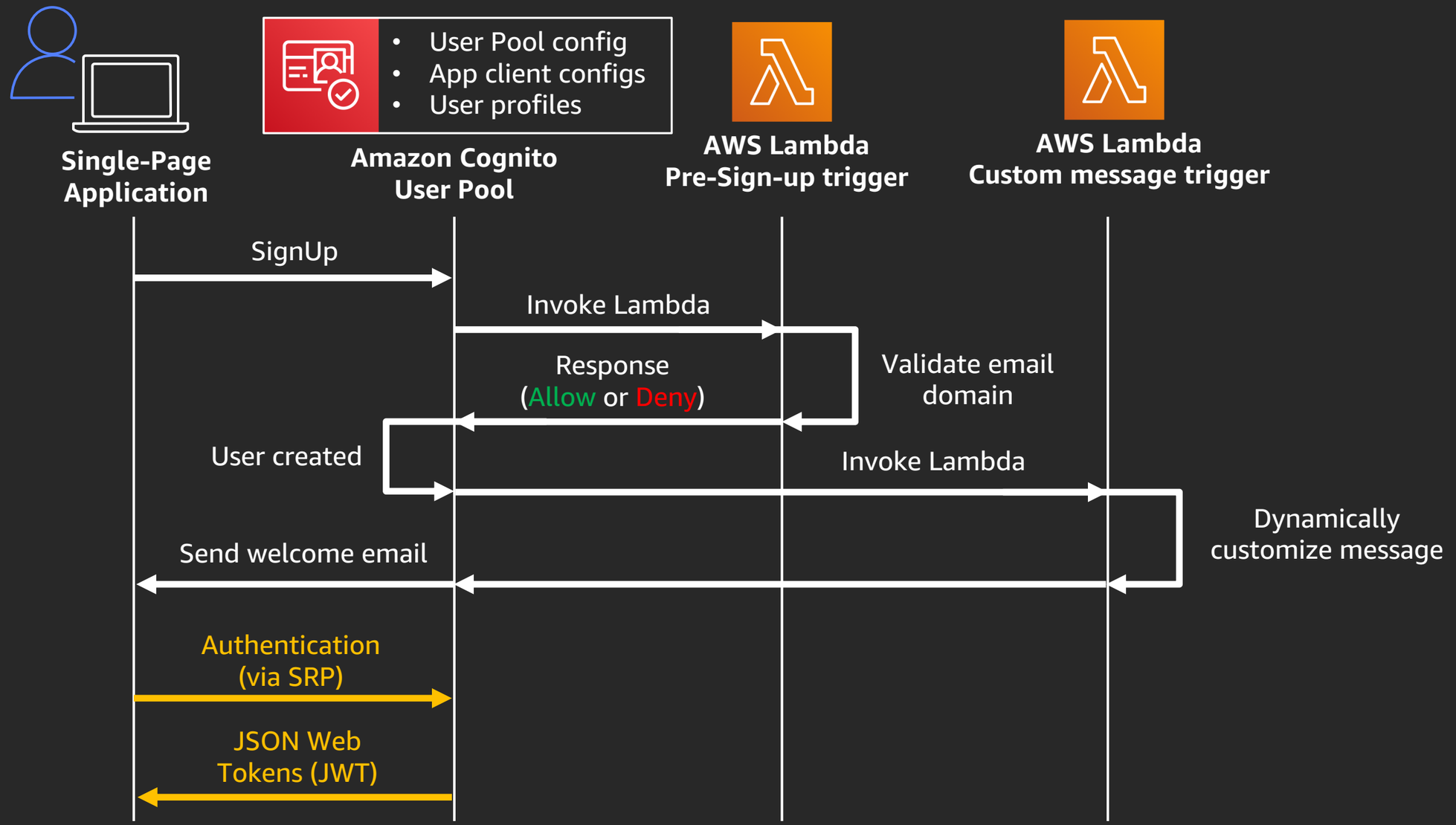
https://bit.ly/2Qsudtv

serverless-idm.awssecworkshops.com

Directions - 50 minutes:

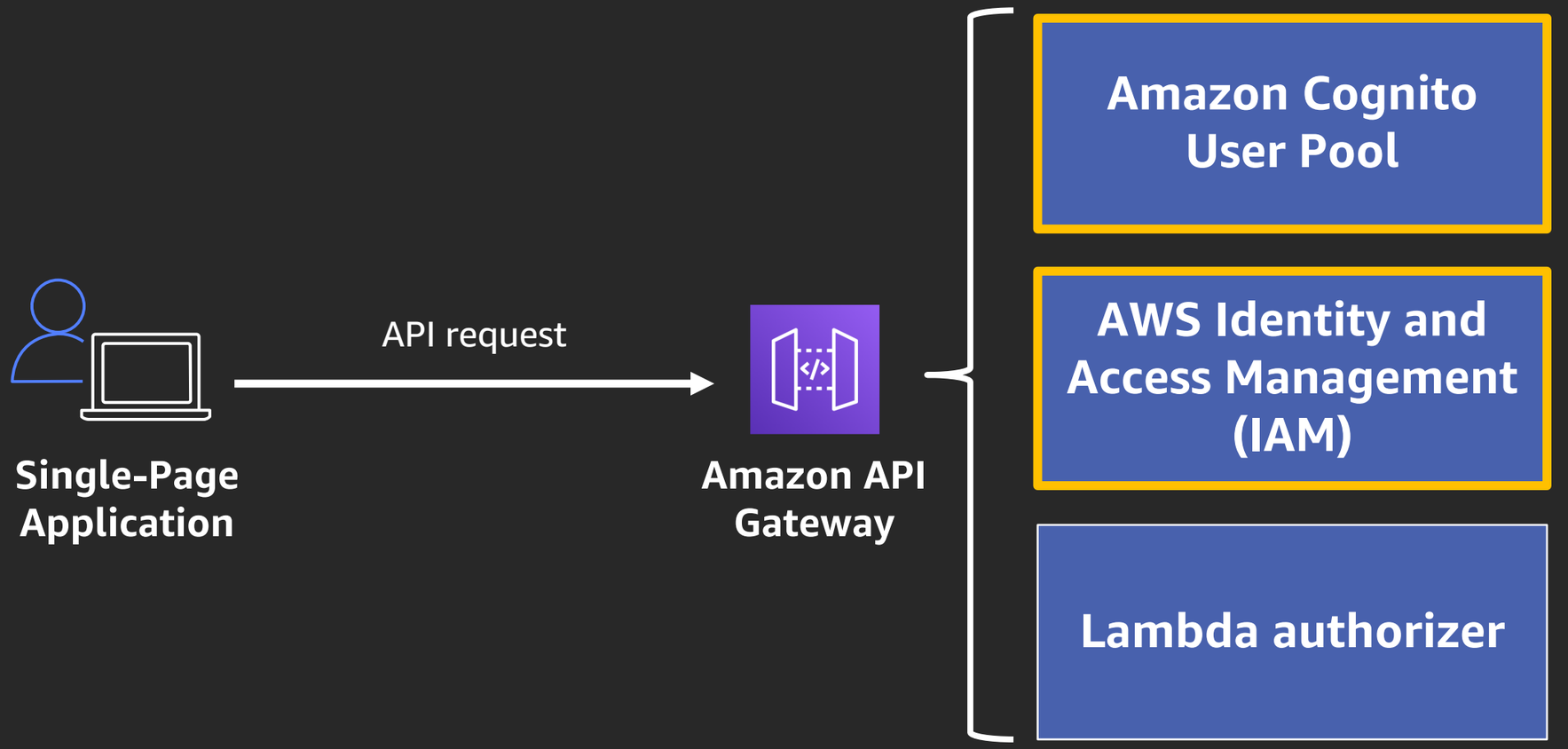
- **Scenario**
- **Module 0 – Environment setup**
- **Module 1 – User sign-up and sign-in**

Wild Rydes user sign-up and sign-in



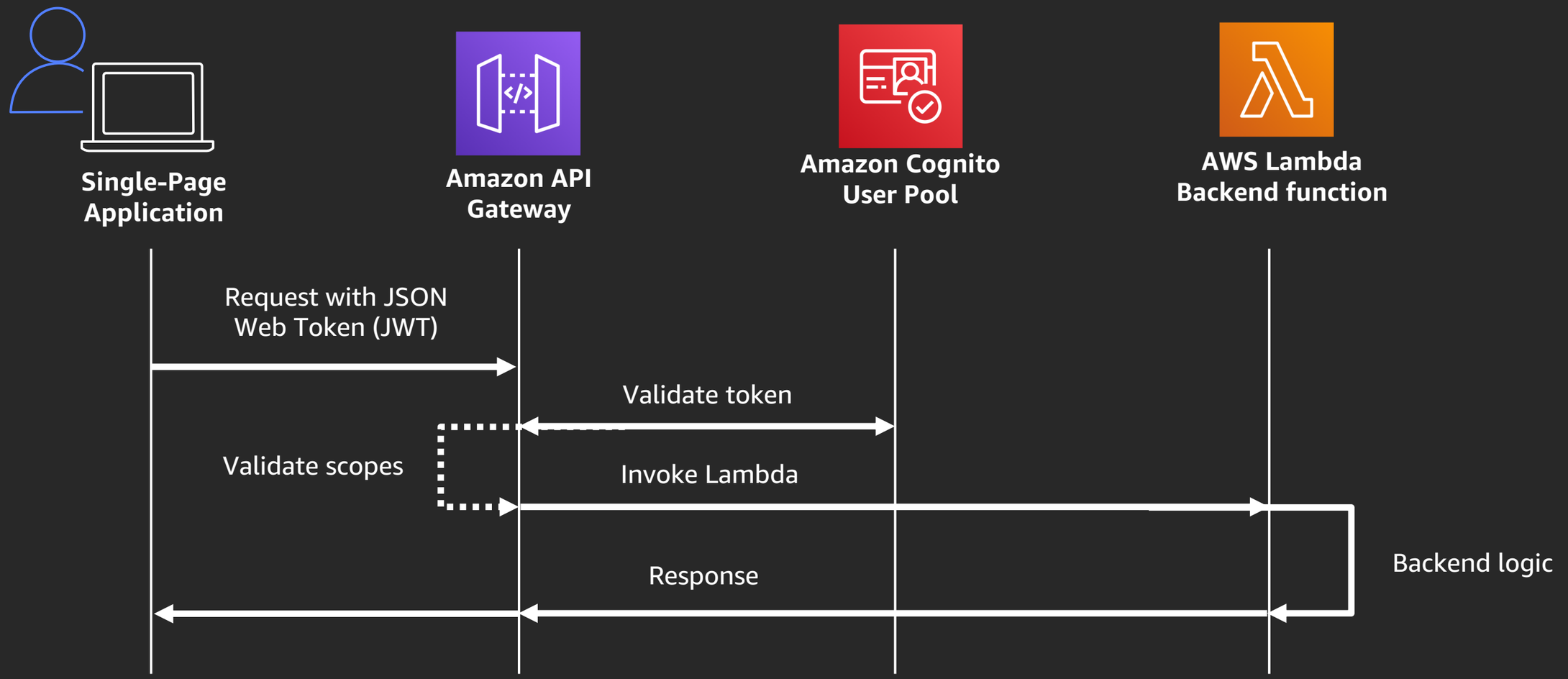
Backend authorization

Integrated authorization with Amazon API Gateway



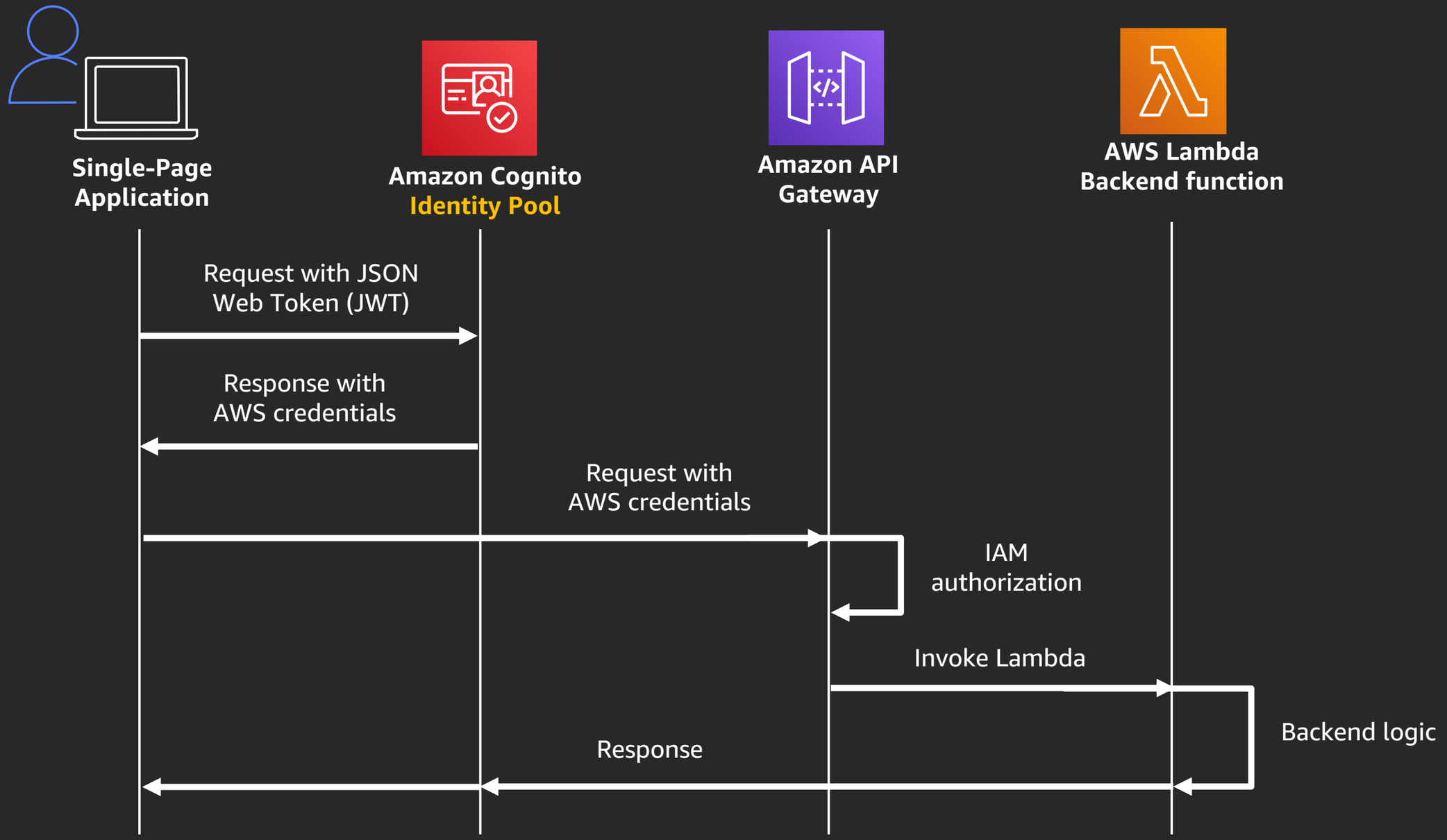
Integrated authorization with Amazon API Gateway

Amazon Cognito user pool



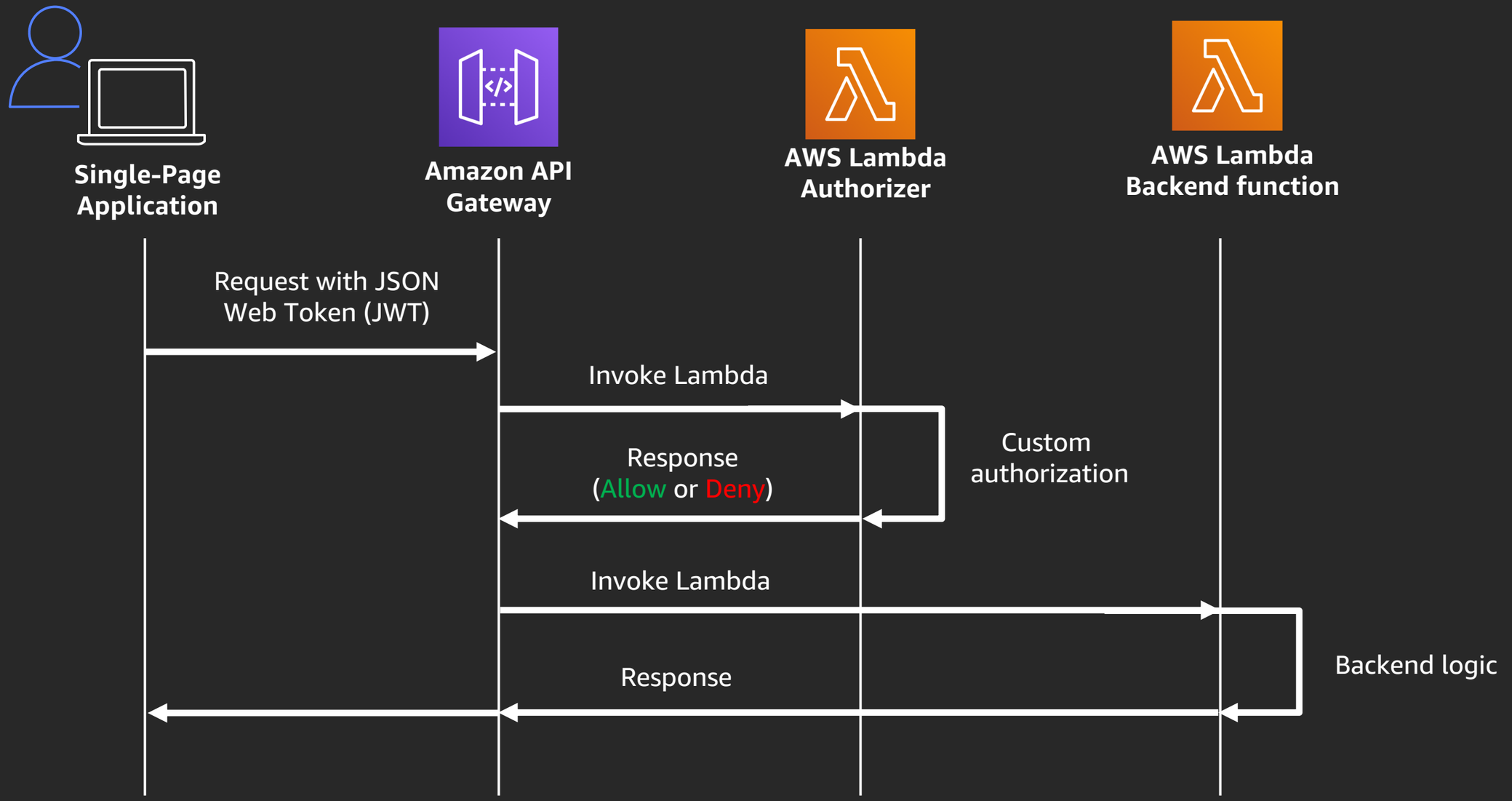
Integrated authorization with Amazon API Gateway

IAM



Integrated authorization with Amazon API Gateway

AWS Lambda authorizer



What if your application users need AWS credentials?

Difference between user pools and identity pools

Cognito User Pools

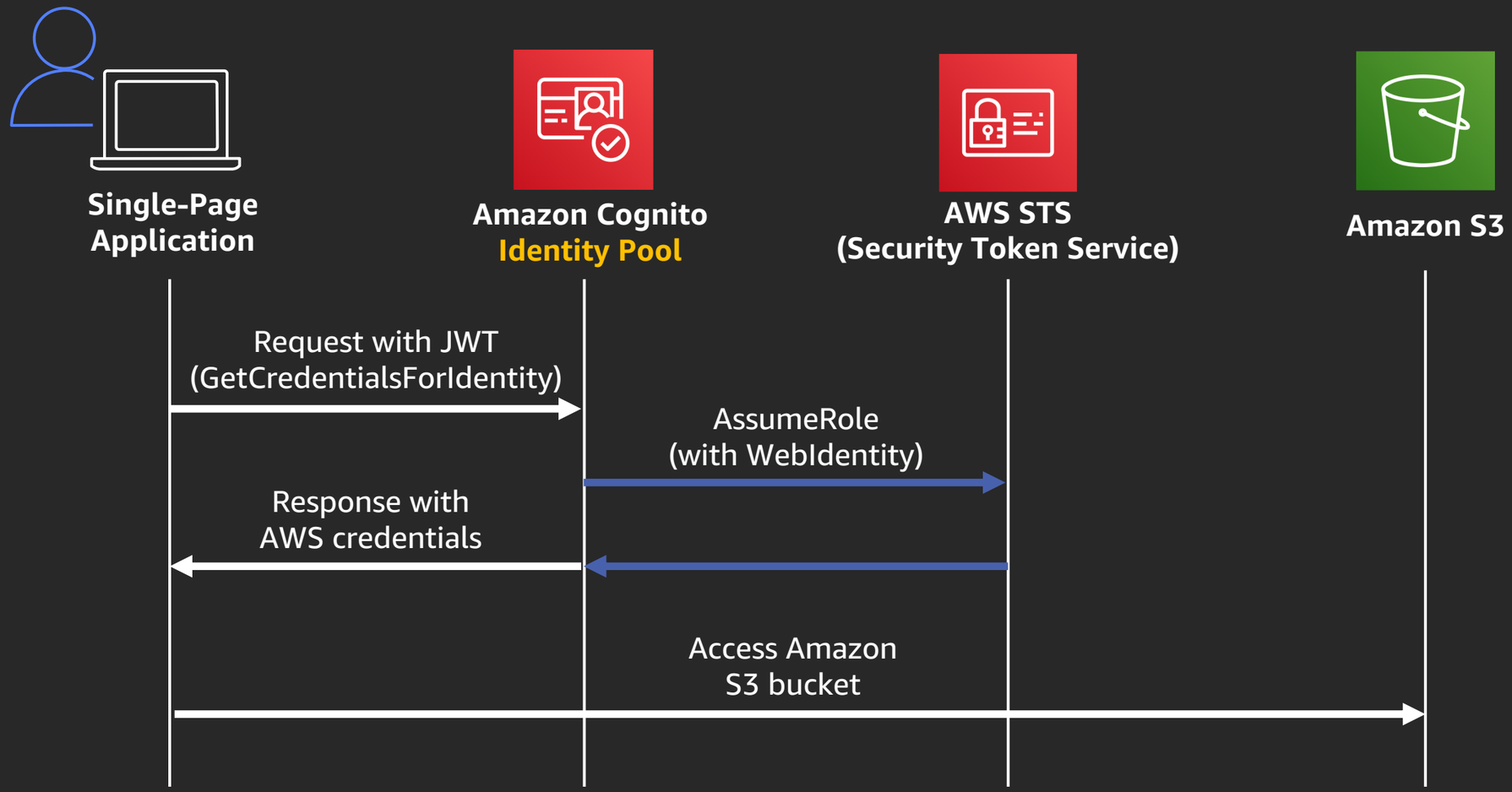
- Managed user directory
- Provides profiles to manage users
- Sign-up and sign-in user flows
- Provides OpenID Connect and OAuth2.0 standard tokens
- Priced per monthly active user

Cognito Identity Pools

- Vends temporary AWS credentials
- Supports authenticated and unauthenticated IAM roles
- Supports rules to map users to different IAM roles using groups
- Free

Integrated authorization with Amazon API Gateway

Enhanced authflow



Module 2 and 3

Workshop guide

https://bit.ly/2Qsudtv

serverless-idm.awssecworkshops.com

Directions - 40 minutes:

- Scenario
- Module 0 – Environment setup
- Module 1 – User sign-up and sign-in
- **Module 2 – Backend authorization**
- **Module 3 – Temporary AWS credentials**

Wrap-up

Which OAuth flow is used for this application?

How can you localize the language of the welcome message after sign-up?

Which token will contain the groups of a user?

Key takeaways

- Amazon Cognito user pools is a managed user directory that can be used in your applications for sign-up and sign-in user flows
- User flows can be customized according to your business requirements using Lambda triggers
- Amazon Cognito issues out standard JSON Web Tokens (JWT) that contain information that can be used for authentication and authorization
- API Gateway has native integration that allows for authorization using Amazon Cognito user pools, IAM, and Lambda authorizers
- Amazon Cognito identity pools allows your application end users to obtain temporary AWS credentials

Related breakouts

Sessions

MOB307 – Frontend web and cross-platform mobile development on AWS

Chalk talks

MOB315 – Breaking down the OAuth flow

SEC219 – Build the next great app with Amazon Cognito

Builder Sessions

SEC409 – Fine-grained access control for serverless apps

SVS330 – Build secure serverless mobile or web applications

ARC405 – Building multi-tenant-aware SaaS microservices

Workshops

MDS405 - UnicornFlix: Building a video-on-demand app with AWS

IOT402 – Building an AWS IoT-enabled drink dispenser

Learn security with AWS Training and Certification

Resources created by the experts at AWS to help you build and validate cloud security skills

30+ free digital courses cover topics related to cloud security, including Introduction to Amazon GuardDuty and Deep Dive on Container Security

Classroom offerings, like AWS Security Engineering on AWS, feature AWS expert instructors and hands-on activities

Validate expertise with the **AWS Certified Security - Specialty** exam

Visit aws.amazon.com/training/paths-specialty/

aws training and certification

Thank you!

AWS re:Invent

aws

Please complete the session survey in the mobile app.

AWS re:Invent

aws