
AWS
re:Invent
NOV. 29 – DEC. 3, 2021 | LAS VEGAS, NV

DOP313

Best practices for securing your software delivery lifecycle

Curtis Rissi
Principal Solutions Architect – Containers & App Modernization
AWS

Sundeep Goswami
Principal 5G Solutions Consultant
Dish Network

© 2021, Amazon Web Services, Inc. or its affiliates. All rights reserved.

“Good intentions never work, you need good mechanisms to make anything happen.”

Jeff Bezos

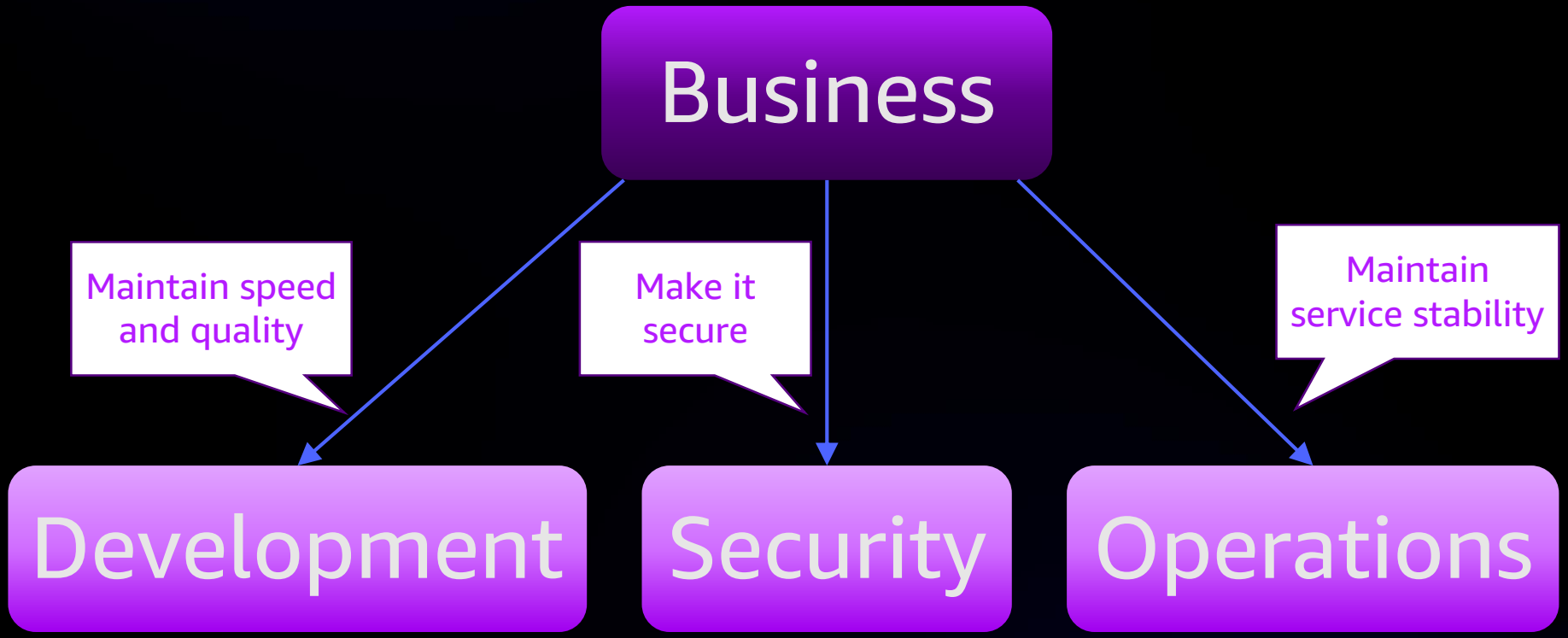
© 2021, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Putting security in DevOps

(It was technically there all along)

© 2021, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Competing forces for DevOps



© 2021, Amazon Web Services, Inc. or its affiliates. All rights reserved.

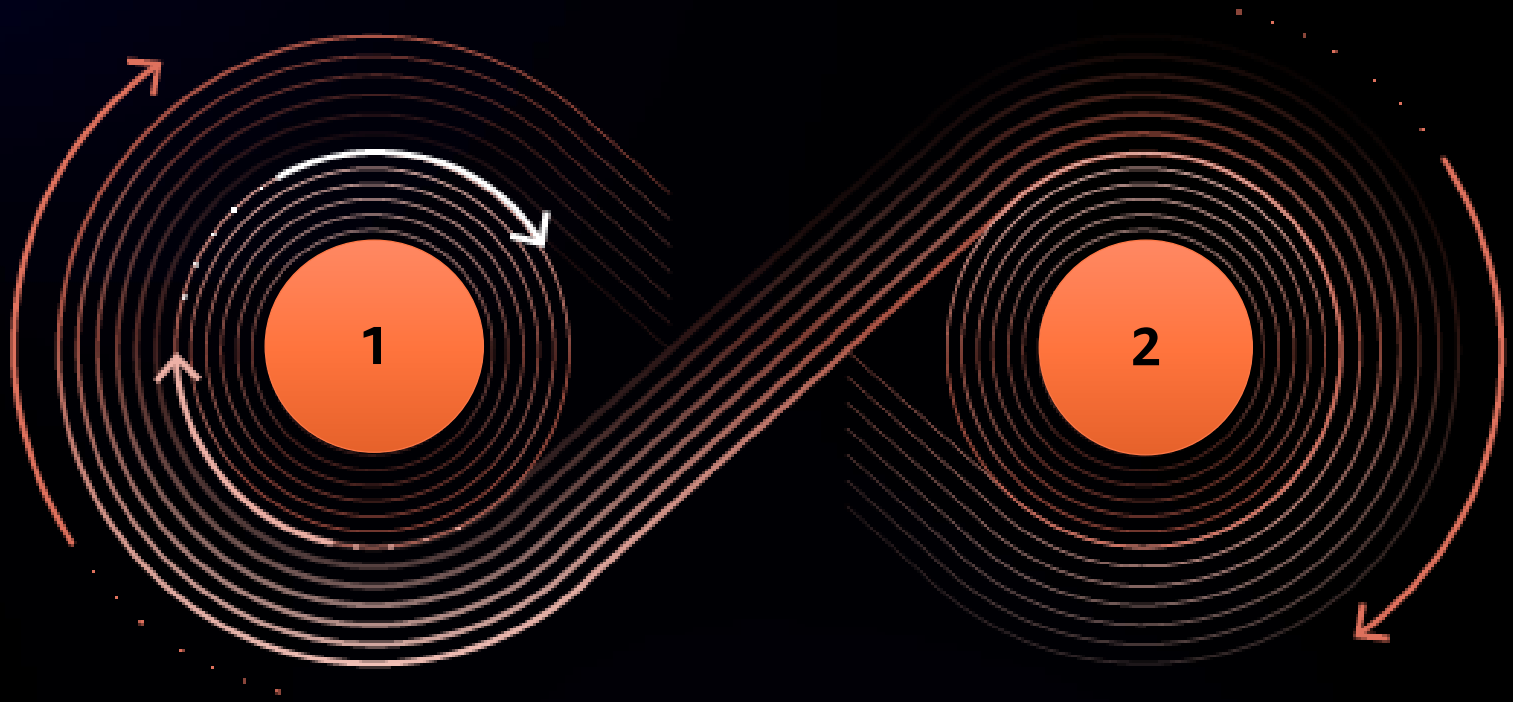
Why DevSecOps?

1. Security – Reduced risk
2. Speed – Compliant workloads
3. Collaboration – Buy-in from all the teams (including security)
4. Reliability – Consistency and repeatability of security validation
5. Scale – Security at scale
6. Agility – Match the business's pace of innovation
7. Automation – Better operational controls

© 2021, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Security of and in the pipeline

© 2021, Amazon Web Services, Inc. or its affiliates. All rights reserved.



Security of the CI/CD pipeline

Access roles

Hardening build servers/nodes

Security in the CI/CD pipeline

Artifact validation

Static code analysis

aws

© 2021, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Security of the pipeline

The DevOps pipeline is a composition of services that has infrastructure, users, data components, etc.

Follow best practices that are applicable to any infrastructure setup

- Identity and Access Management: Least privilege / MFA / Read-only access
- Data Protection: Encryption at-rest and in-transit, integrity checking
- Governance and Auditing: Who? What? When?
- Logging and Monitoring: Trigger events, auto responses, etc.
- Pipeline as Code: Infrastructure as a Code

© 2021, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Security in the pipeline

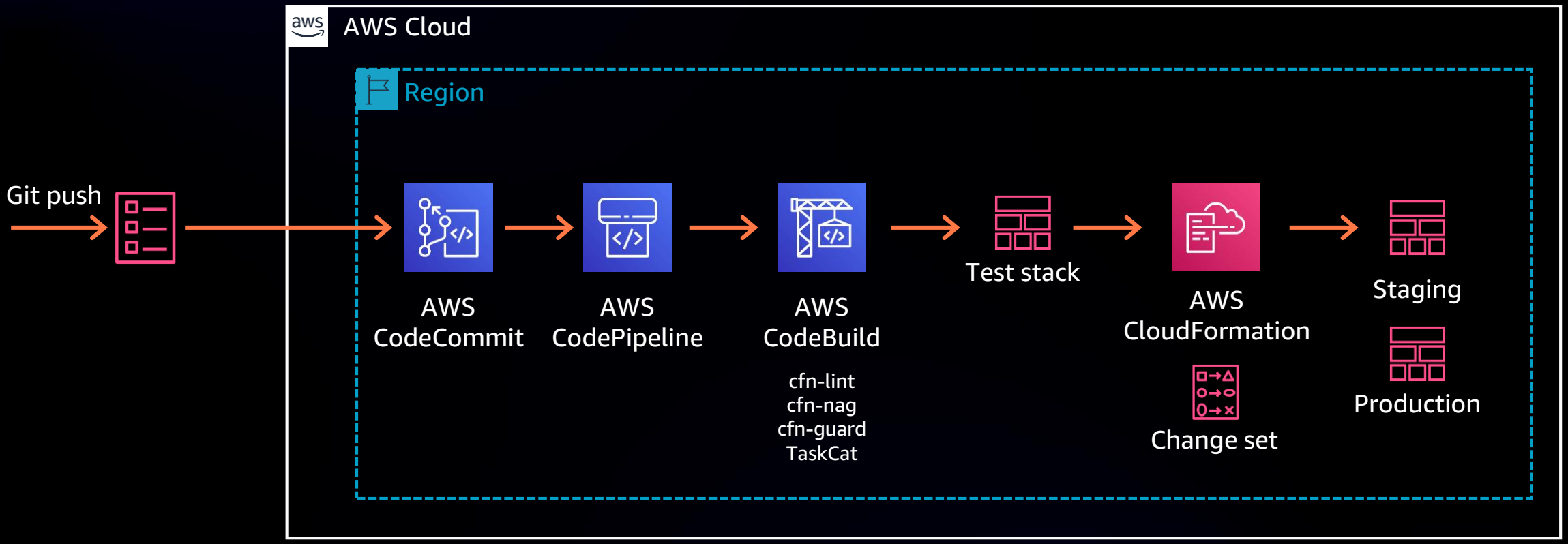
- Protect Sensitive Information (PII)
- Static Application Security Testing (SAST)
- Software Composition Analysis (SCA)
- Dynamic Application Security Testing (DAST)
- Runtime Application Self Protection (RASP)
- Interactive Application Security Testing (IAST)

© 2021, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Patterns

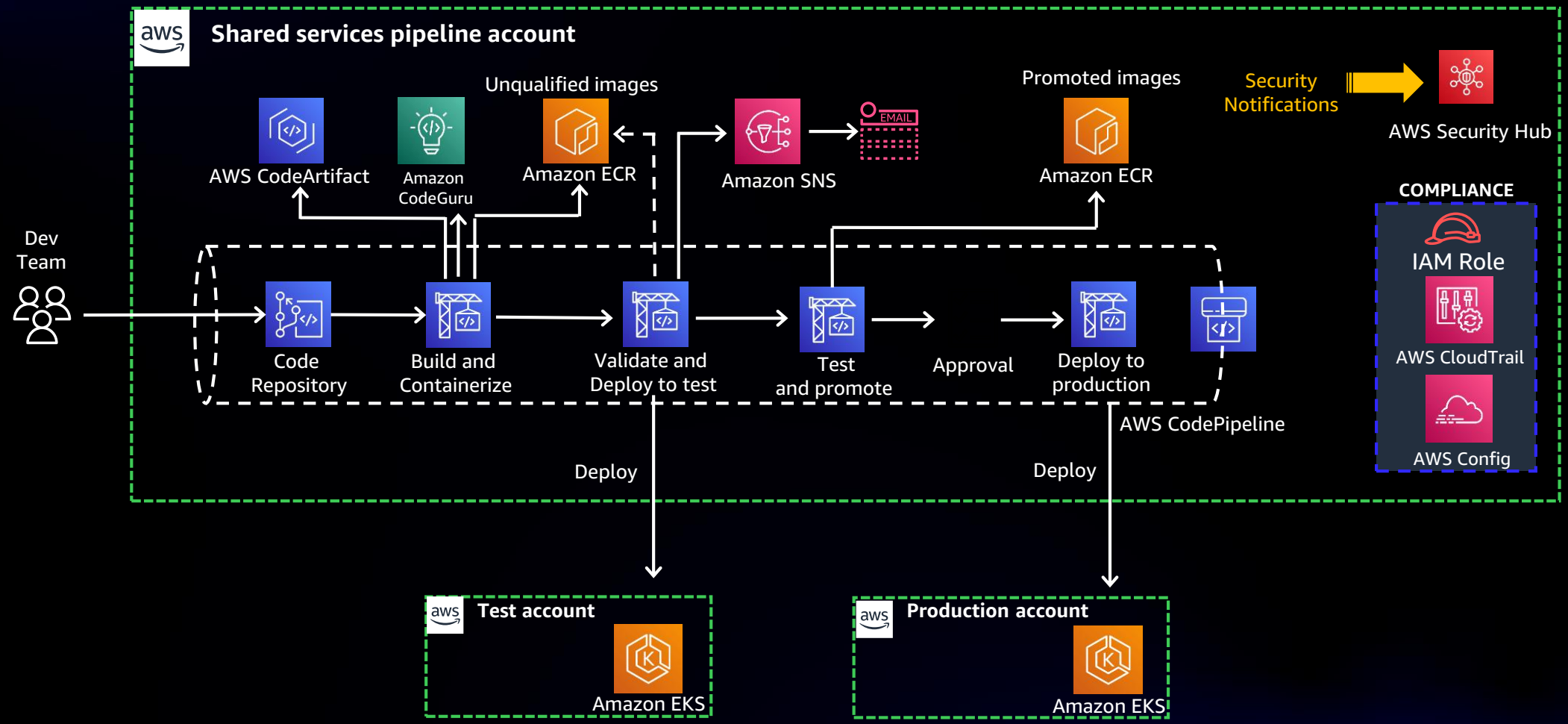
© 2021, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Integrating security of the pipeline



© 2021, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Integrating security in the pipeline



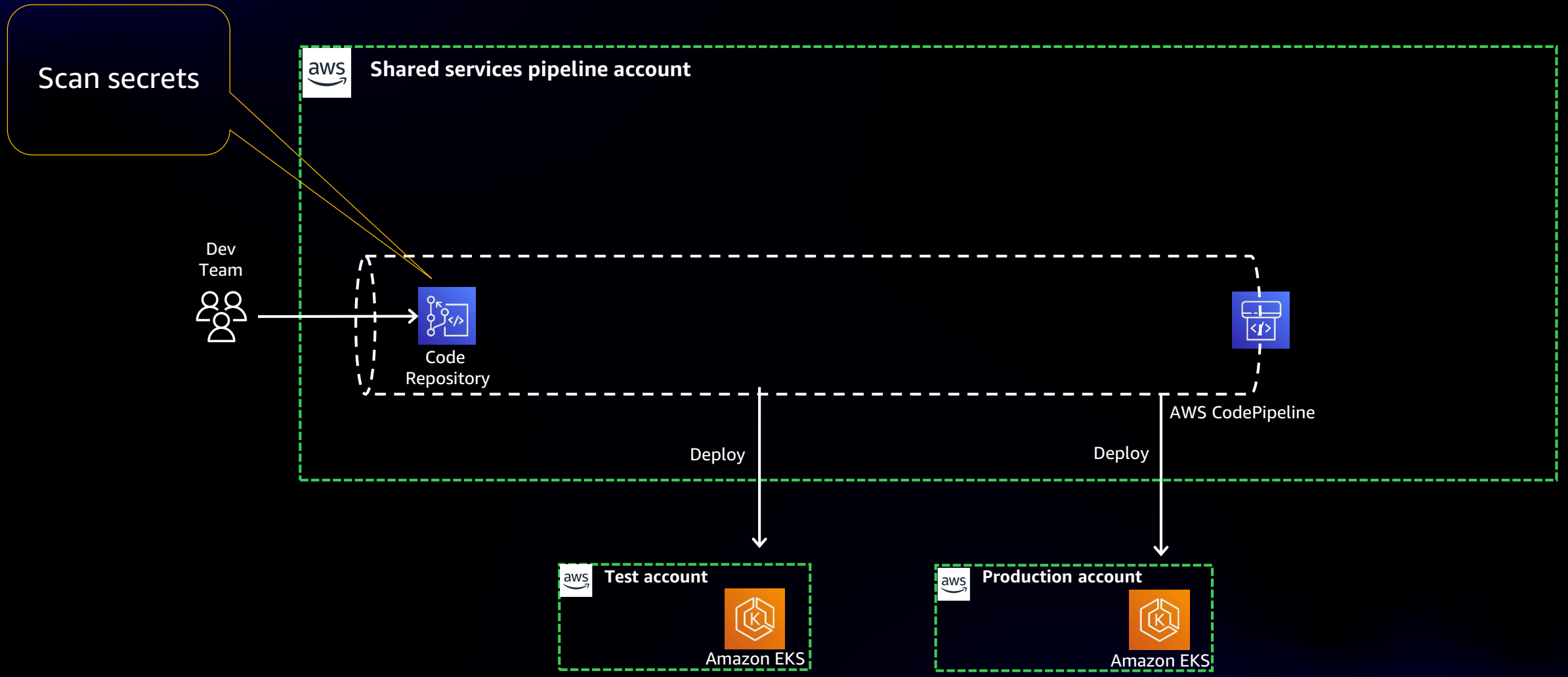
© 2021, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Security of your continuous delivery pipeline

- CodePipeline supports resource-based permissions in many API calls
  - `"Resource": "arn:aws:codepipeline:us-east-2:111222333444:myPipeline”`
- Examples of resource-based policies
  - Manual approvals
  - Enable or disable stage transitions
  - Tags can be used as well
- CodePipeline works through a service role
  - Required when you first create a pipeline
  - Default role allows CodePipeline to interact with all AWS services it integrates with
  - This role is updated as new integration are added
  - You can lock this down to be based on least privilege principle
- Use Parameter Store when using AWS CloudFormation with CodePipeline to keep credentials secure

© 2021, Amazon Web Services, Inc. or its affiliates. All rights reserved.

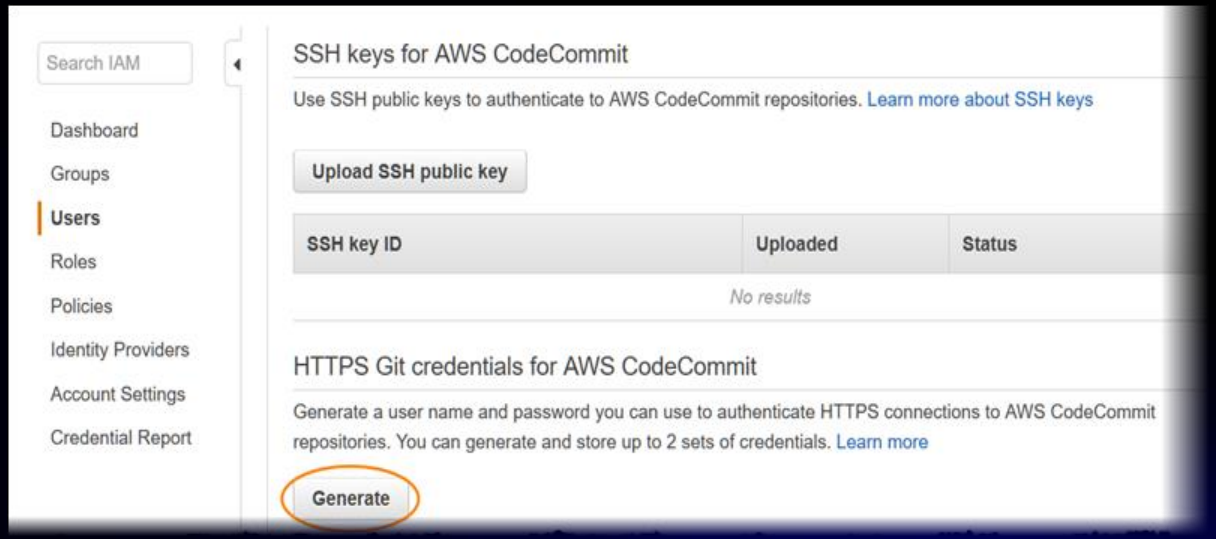
Integrating security in the pipeline



© 2021, Amazon Web Services, Inc. or its affiliates. All rights reserved.

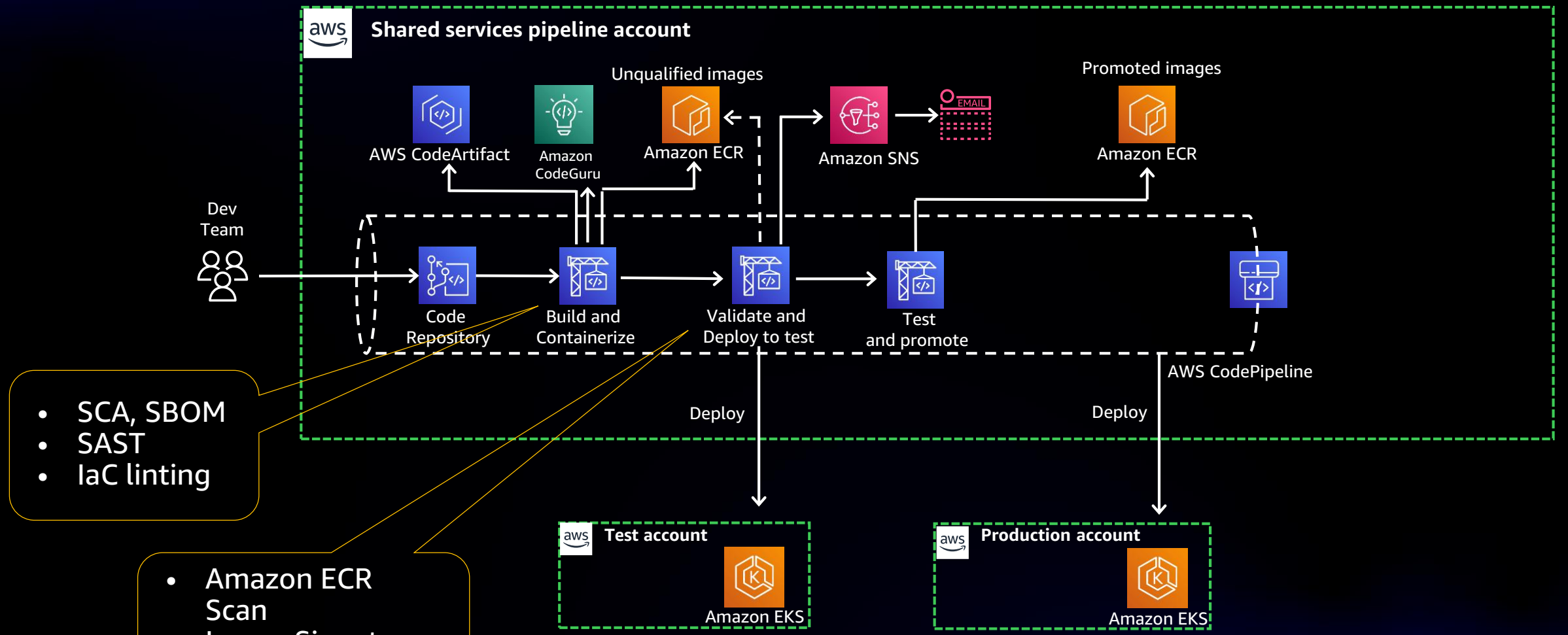
Security in CodeCommit

- Primary resource is a repository
  - "Resource": "arn:aws:codecommit:us-west-2:111111111111:MyDemoRepo”
- All CodeCommit permissions except ListRepositories can be scoped to a resource
- You can connect to a repository two ways
  - HTTPS
    - Credentials tied to AWS IAM user
    - With credential helper and AWS CLI
      - Can be done through an IAM role
  - SSH Key
    - Must be tied to an IAM user



© 2021, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Integrating security in the pipeline



© 2021, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Security in CodeBuild

- Build artifacts are all encrypted with AWS KMS CMKs
  - CodeBuild can also use a CMK to encrypt artifacts it places in Amazon S3
- Primary resource in CodeBuild is a project
  - "Resource": "arn:aws:codebuild:us-east-2:123456789012:project/myBuildProject”
- Resource-level permissions for some CodeBuild actions
  - Examples
    - Create or delete project
    - Start or stop build
- Tags can be used
- For building Docker images and putting into Amazon ECR, make sure to enable image scanning on push on the Amazon ECR repository

© 2021, Amazon Web Services, Inc. or its affiliates. All rights reserved.

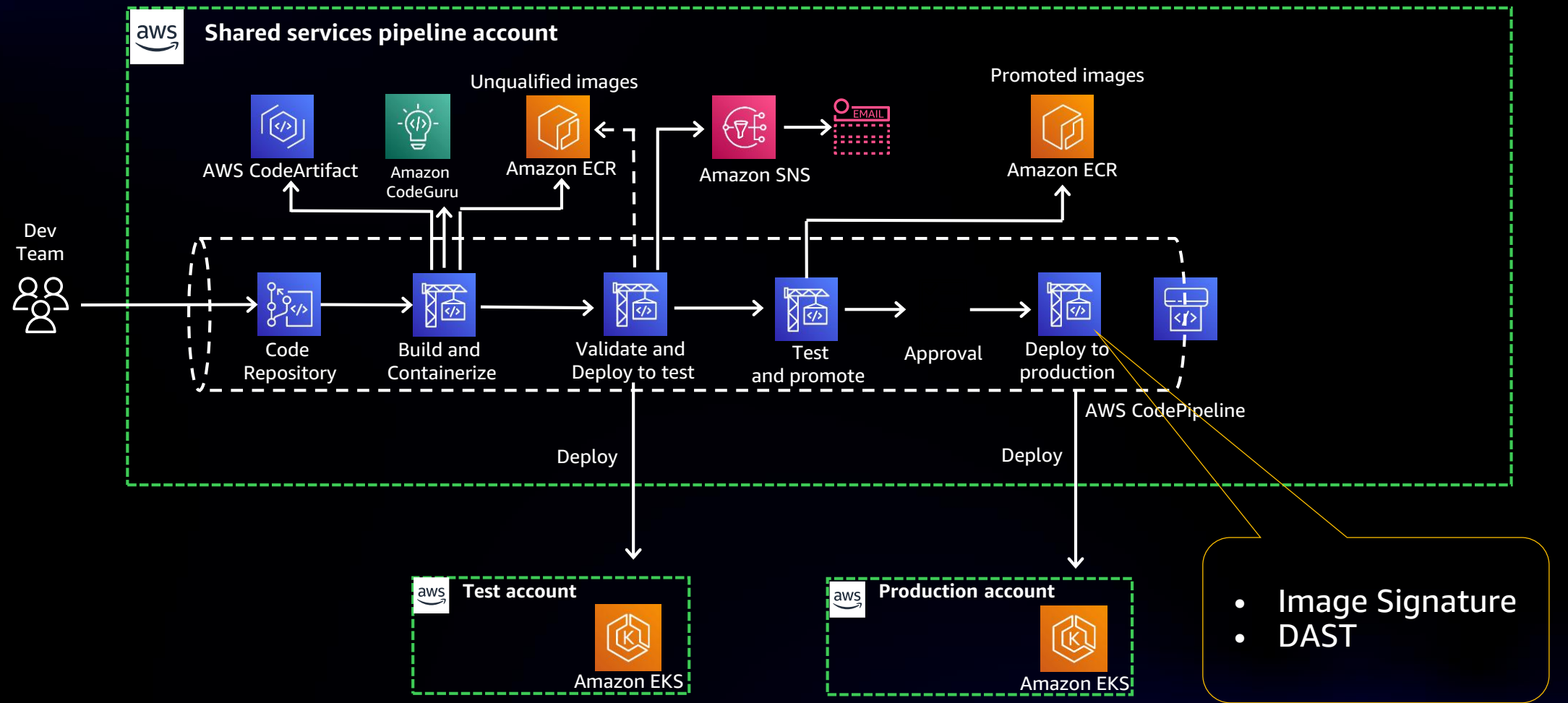
Security in CodeBuild

- CodeBuild projects can run in Amazon VPC
  - Security groups can be used to lock down what a project can communicate with
  - Service endpoint can be used to restrict what a project can talk to outside of its subnet
  - By default, projects run outside of Amazon VPC
- Use AWS Systems Manager Parameter Store to store sensitive credentials

```
env:
  variables:
    JAVA_HOME: "/usr/lib/jvm/java-8-openjdk-amd64"
  parameter-store:
    LOGIN_PASSWORD: /CodeBuild/dockerLoginPassword
```

© 2021, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Integrating security in the pipeline



© 2021, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Security in CodeDeploy

- Primary resource in CodeDeploy is a deployment group
  - `"Resource": "arn:aws:codedeploy:us-west-2:123456789012:deploymentgroup/myDeploymentGroup”`
- Some resources support resource-level permissions
- You can use AWS Identity and Access Management (IAM) to control access to
  - Applications and application revisions
  - Deployments
  - Deployment configurations
  - Instances and on-premises instances

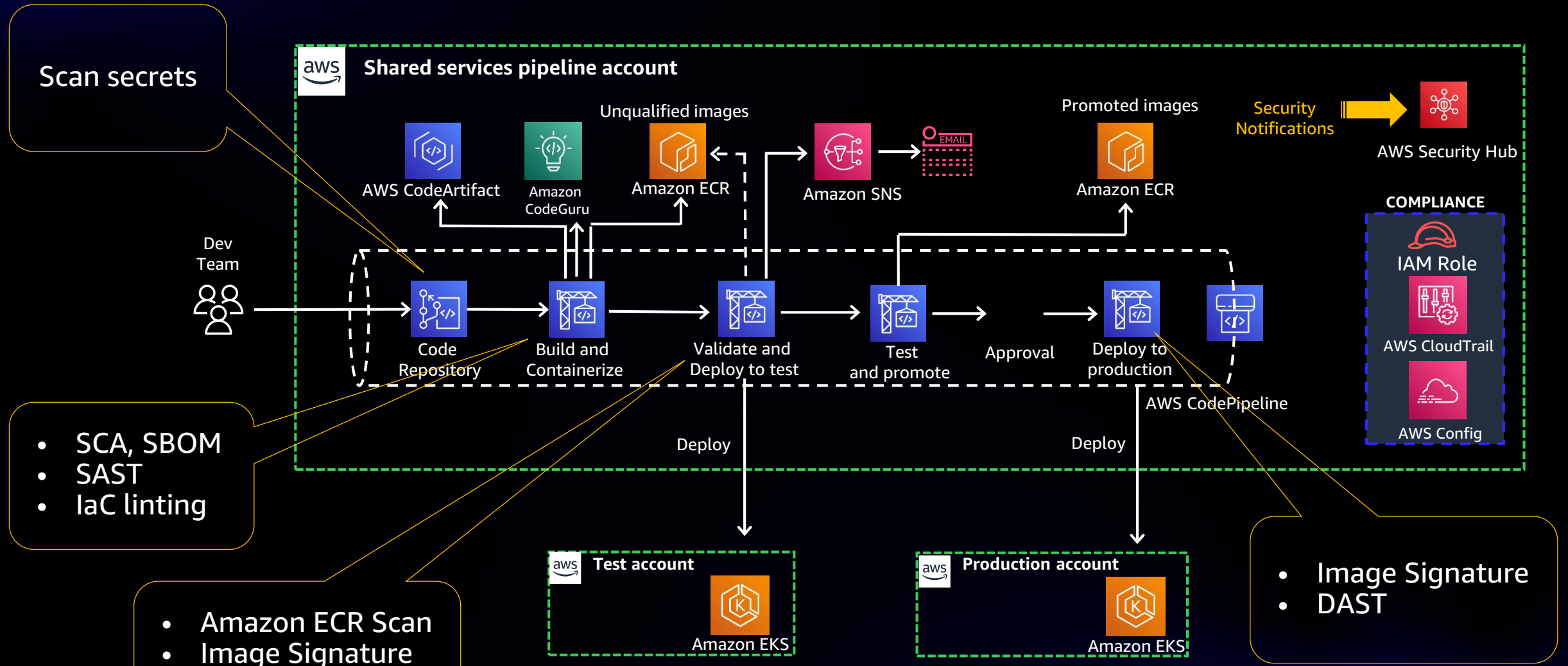
© 2021, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Security in CodeDeploy

- Instance running CodeDeploy agent must have an IAM instance profile
  - Allow instance to access Amazon S3 buckets or GitHub repos where artifacts are stored
- The IAM service role is used by CodeDeploy
  - This allows CodeDeploy to do things like read Amazon EC2 tags and interact with ELBs
- appspec.yml file can use the Systems Manager Parameter Store for secure credentials

© 2021, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Integrating security in the pipeline



© 2021, Amazon Web Services, Inc. or its affiliates. All rights reserved.

DevOps for DISH 5G with emphasis on security

© 2021, Amazon Web Services, Inc. or its affiliates. All rights reserved.

4. DISH's vision

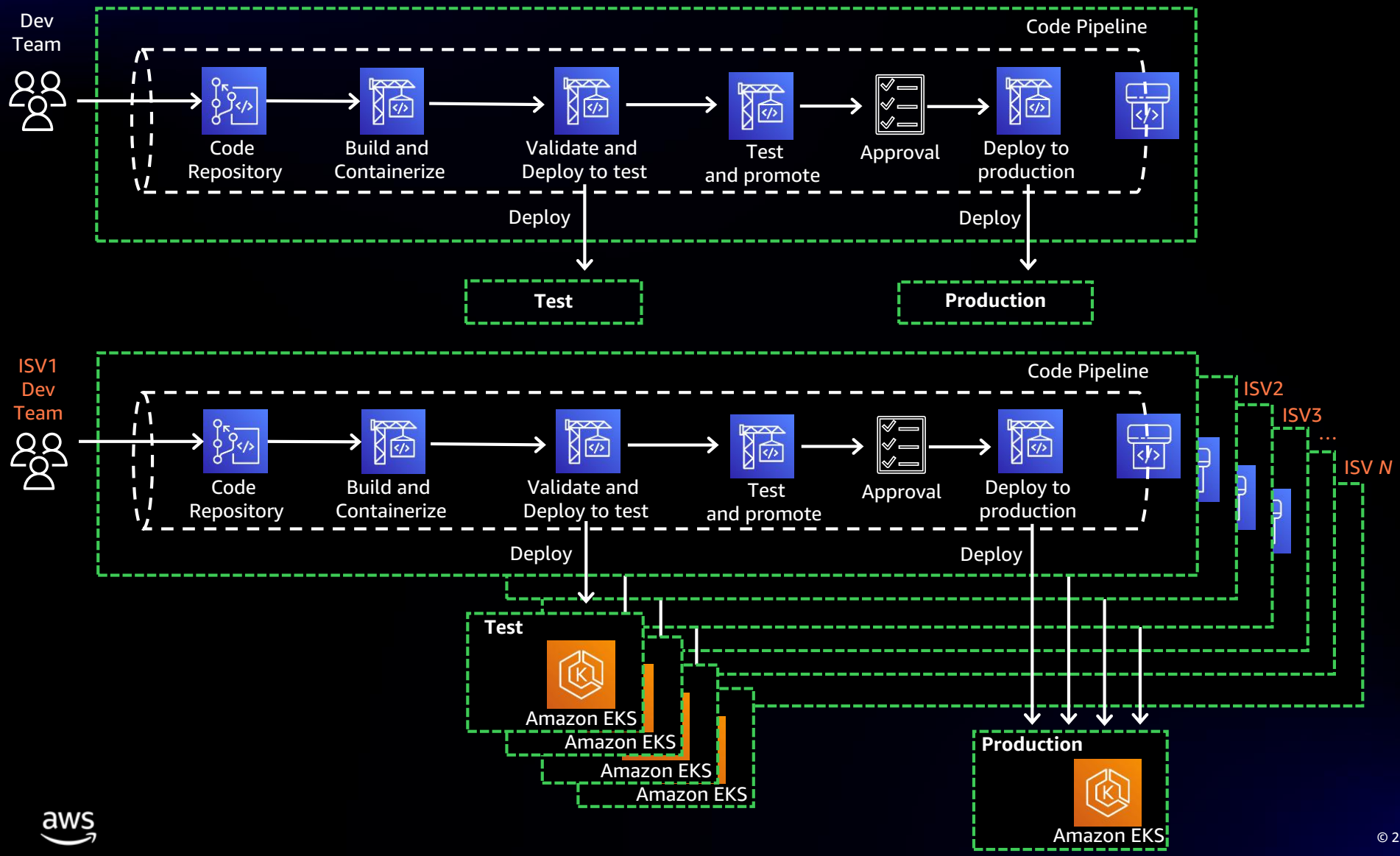
Humans will never fly!

Electric cars are a fad!

A fully containerized 5G network, in AWS . . . with DevSecOps? Nah!

© 2021, Amazon Web Services, Inc. or its affiliates. All rights reserved.

4. Telco DevOps



Complexities of DevOps are . . .

Exponentially amplified in the telco context

© 2021, Amazon Web Services, Inc. or its affiliates. All rights reserved.

4. Some important lessons learned

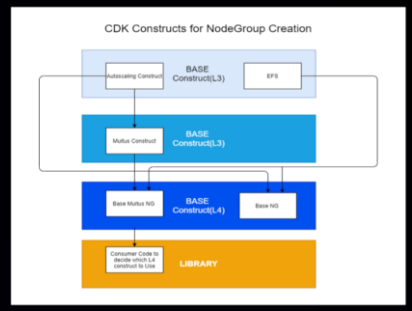
Suppliers do . . . and did, what's best for them!



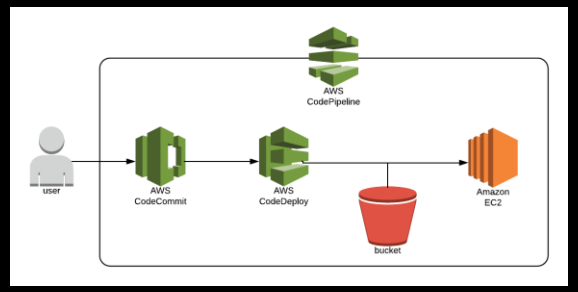
But this is not sustainable at scale



Standardize



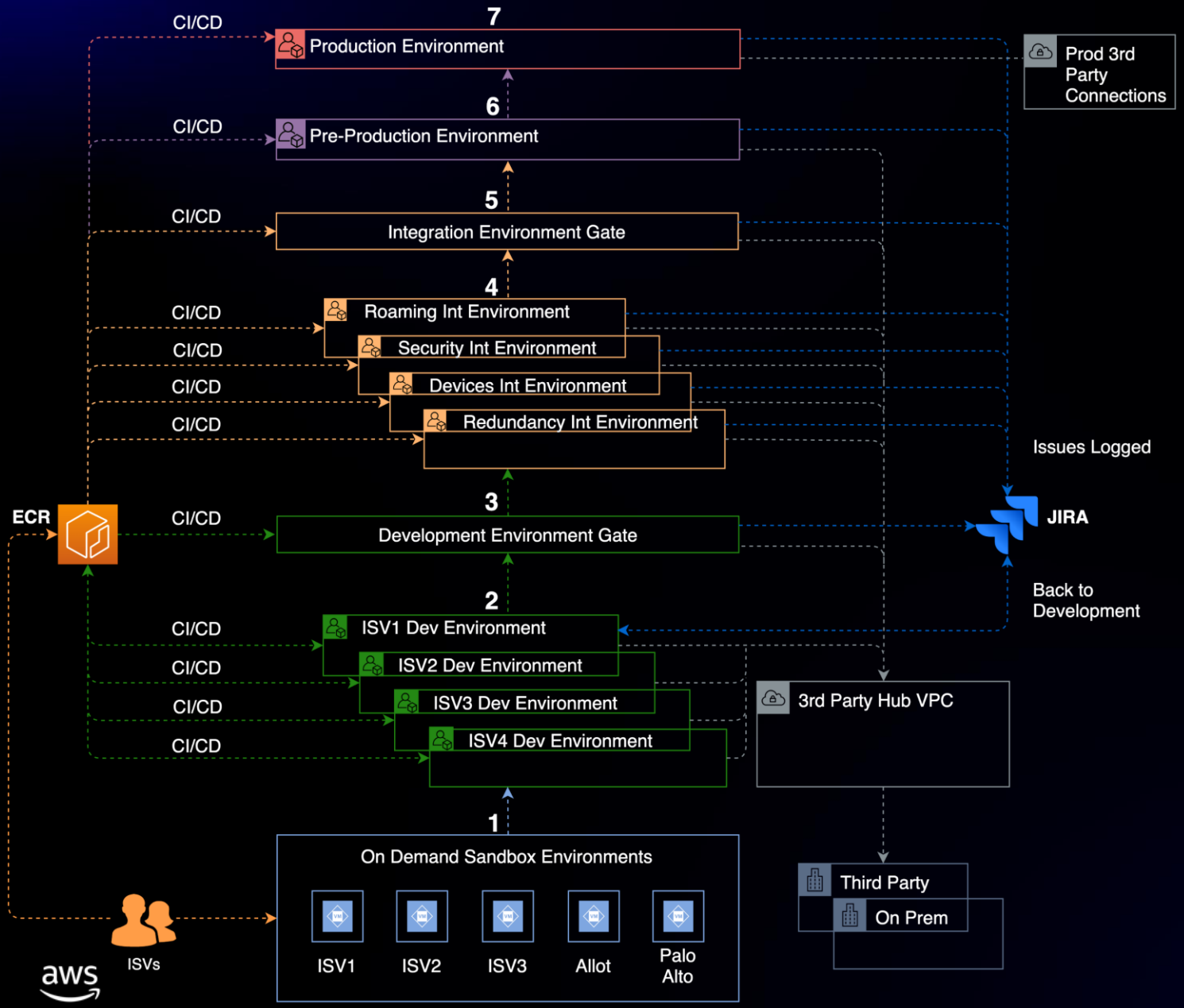
Abstract



Automate

© 2021, Amazon Web Services, Inc. or its affiliates. All rights reserved.

4. Solving for DISH DevSecOps



Hierarchy of environments – all leveraging AWS DevOps tools, e.g., CodeBuild, CodeDeploy, CloudFormation, Amazon ECR, etc.

CI/CD pipelines in every environment with centralized issue logging and tracking

Least-trust model – no human touches production

External connectivity limited to only environments that require it

Security VPCs

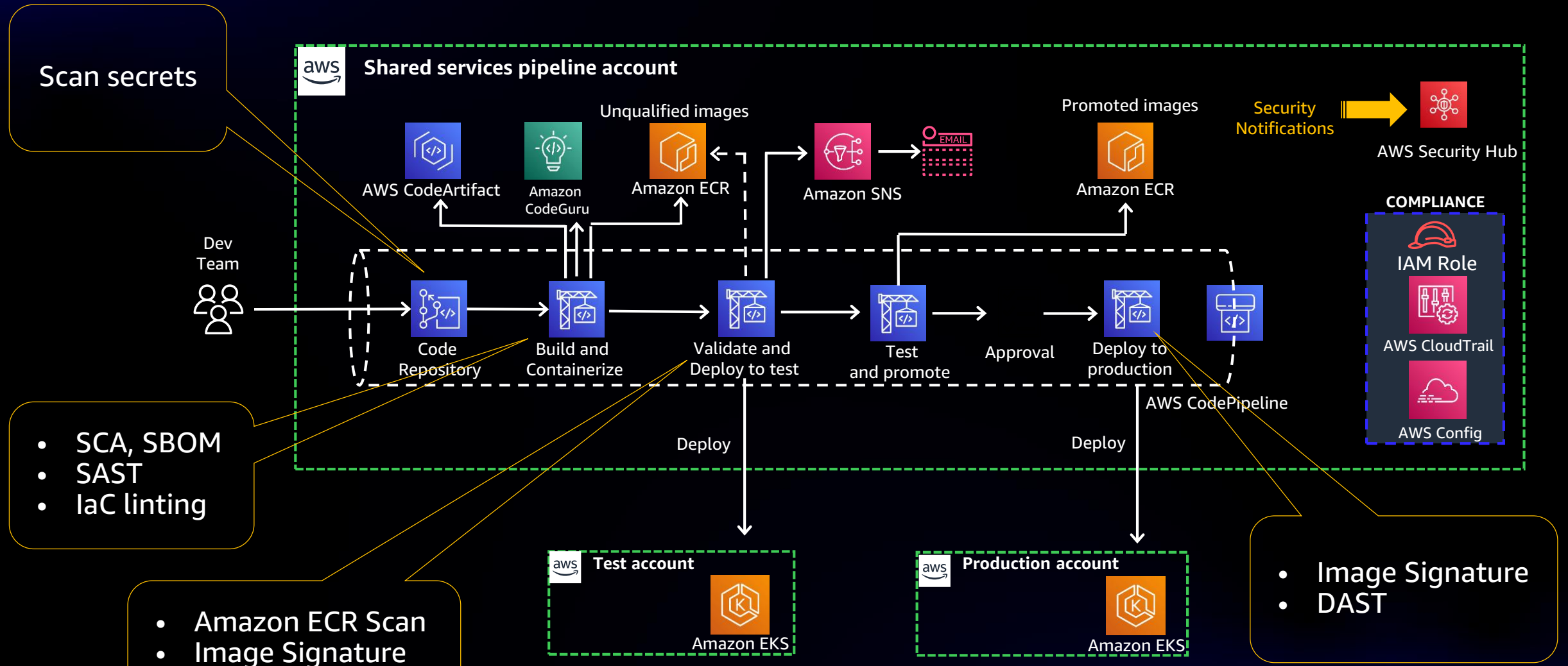
Additional components to secure the network – Prisma Security, Defender, FW, DDoS

© 2021, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Where should you get started?

© 2021, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Integrating security in the pipeline



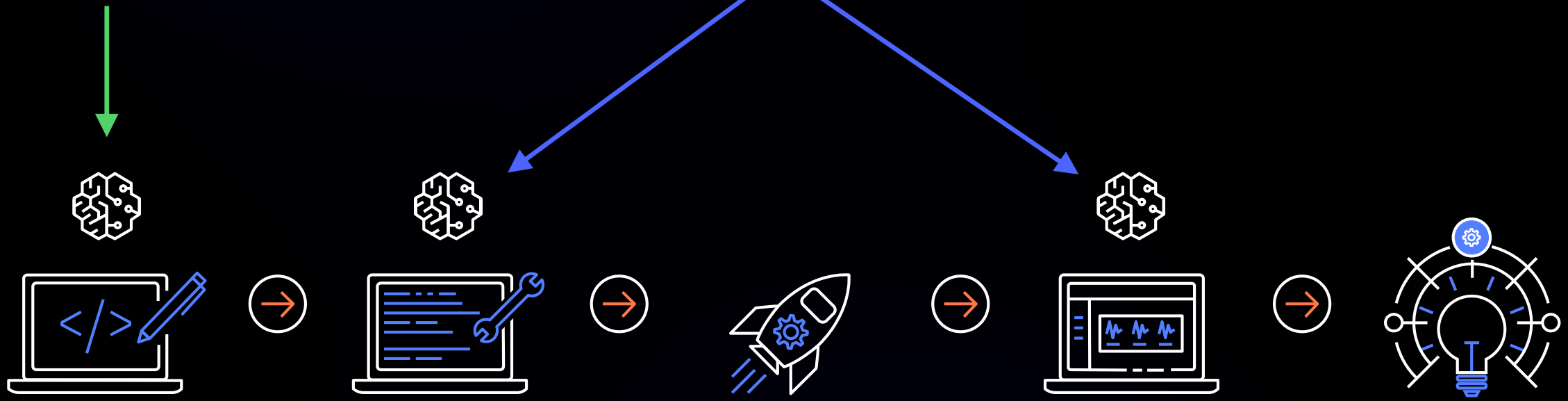
© 2021, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Amazon CodeGuru

USING ML TO CODE REVIEW AND OPTIMIZE HIGH-PERFORMING APPLICATIONS

CodeGuru Reviewer

CodeGuru Profiler



CODING:

Built-in code reviews with actionable recommendations

BUILD & TEST:

Detect and optimize the expensive lines of code pre-production

DEPLOY

MEASURE:

Easily identify performance and cost improvements in production environment

© 2021, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Categories of security code issues addressed by Reviewer

- **AWS SDK Best Practices:** Following security best practices when using AWS APIs
- **Java Crypto Library Best Practices:** Best practices for common Java cryptography libraries
- **Secure Web Applications:** Not sanitizing user-supplied inputs can result in several security issues (e.g., cross-site scripting, SQL injection, LDAP injection, path traversal injection)
- **Apply Amazon's AppSec Best Practices:** Developed in collaboration with AWS security, bringing internal security expertise to our customers
- **Sensitive Information Leak:** Leakage of sensitive information (e.g., logging of credit card number) leads to compliance issues

© 2021, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Detecting sensitive information leaks

Code

```
try {
     updateJobStatus(context.getAwsRequestId(),
                     request.getAwsAccountId(),
                     request.getPredictorName(),
                     request.getInternalStatus());
 } catch (ValidationException e) {
     log.error(NON_RETRIABLE_LIST_ERROR_MESSAGE, e);
     throw e;
 } catch (InternalServerException e) {
     log.warn(RETRIABLE_LIST_ERROR_MESSAGE, e);
     retries++;
     continue;
 }
```

Fix

```
try {
        updateJobStatus(context.getAwsRequestId(),
                                request.getAwsAccountId(),
                                request.getPredictorName(),
                                request.getInternalStatus());
      } catch (ValidationException e) {
        log.error(NON_RETRIABLE_LIST_ERROR_MESSAGE, redact(e));
                throw redact(e);
      } catch (InternalServerException e) {
        log.warn(RETRIABLE_LIST_ERROR_MESSAGE, redact(e));
                retries++;
                continue;
      }
```

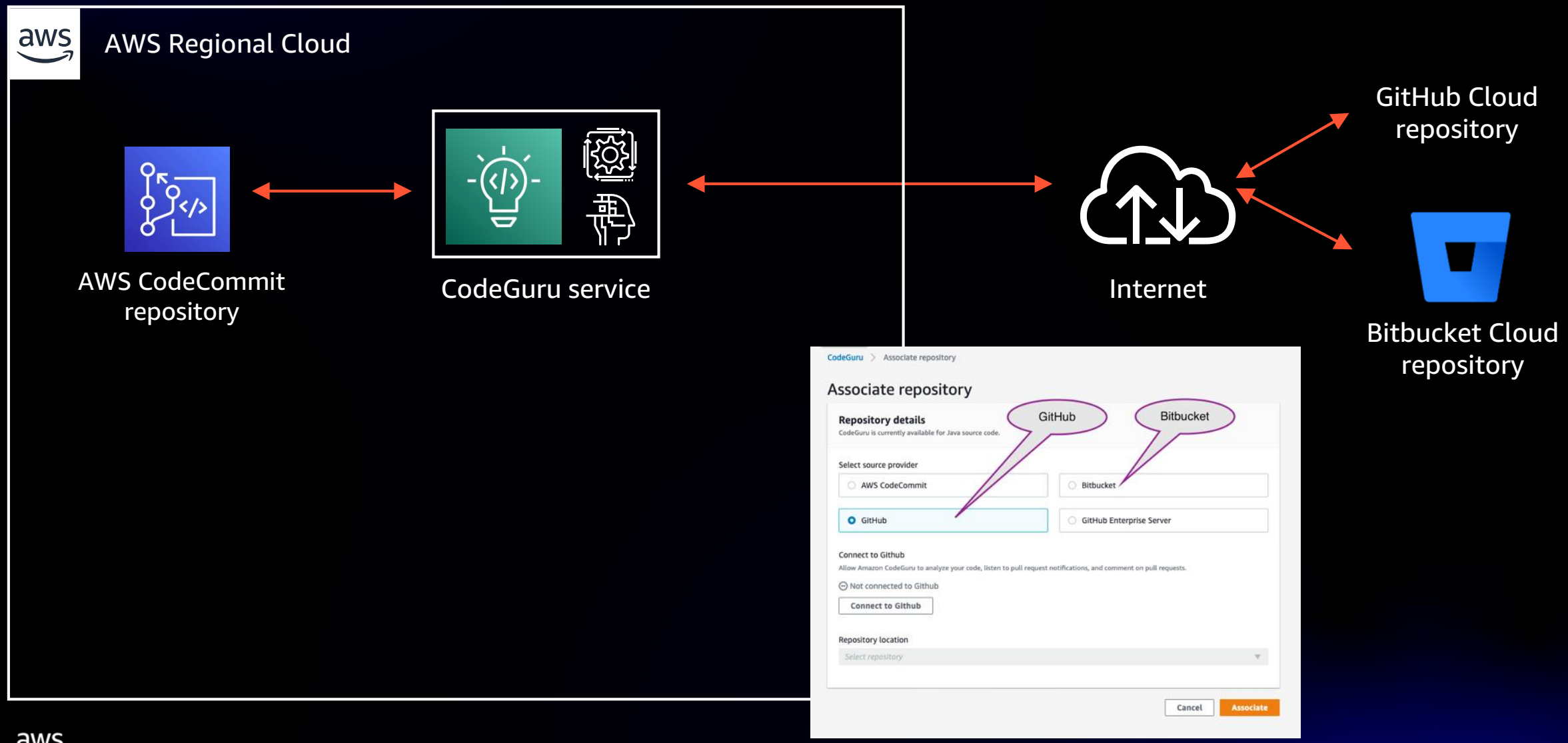
Recommendation

This code contains potential unintended disclosure of information in the error handling for the following call: 'getAwsAccountId()'. You are handling this error with catch clauses. There are methods available that could be added to handle sensitive data, like masking and redaction.

For more information about unintended disclosure of information, see https://cwe.mitre.org/data/definitions/209.html

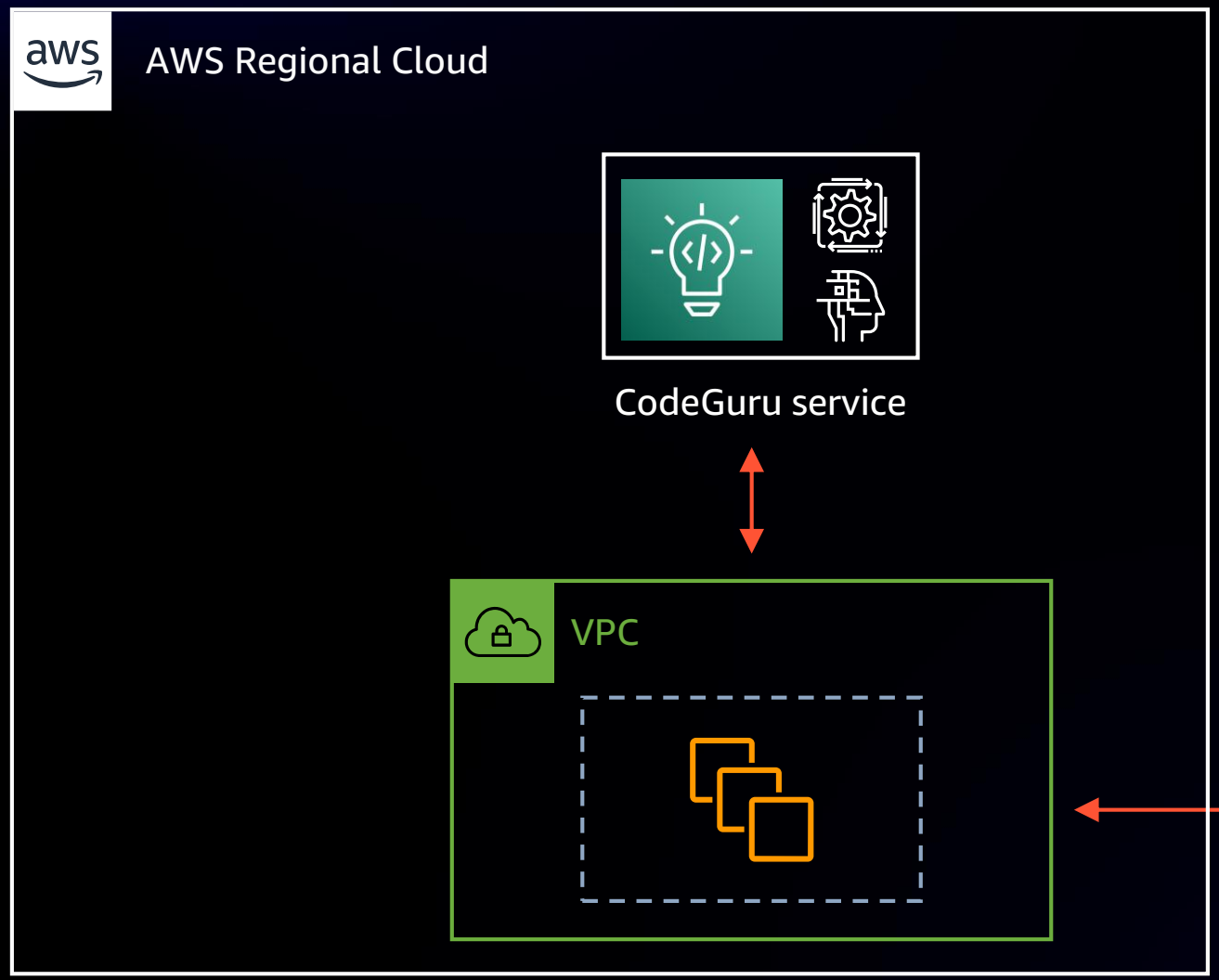
© 2021, Amazon Web Services, Inc. or its affiliates. All rights reserved.

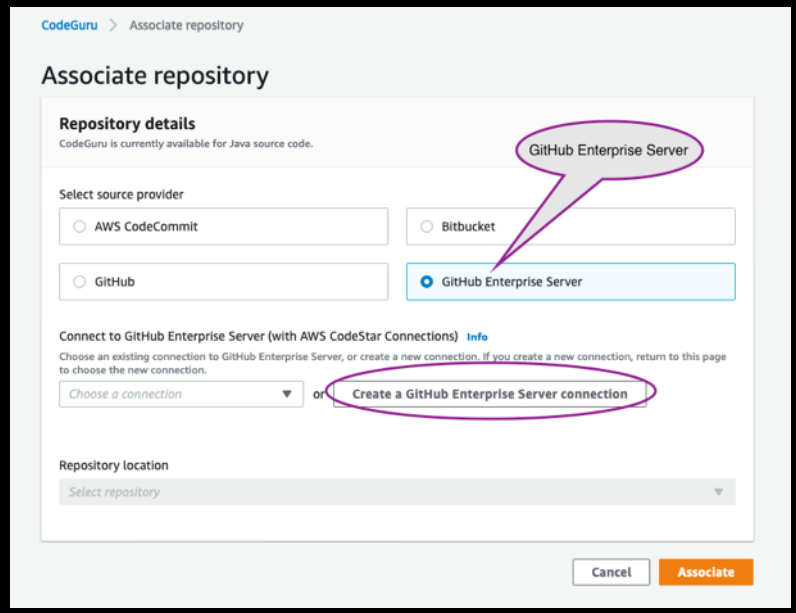
CodeGuru repository support

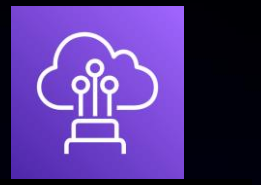


© 2021, Amazon Web Services, Inc. or its affiliates. All rights reserved.

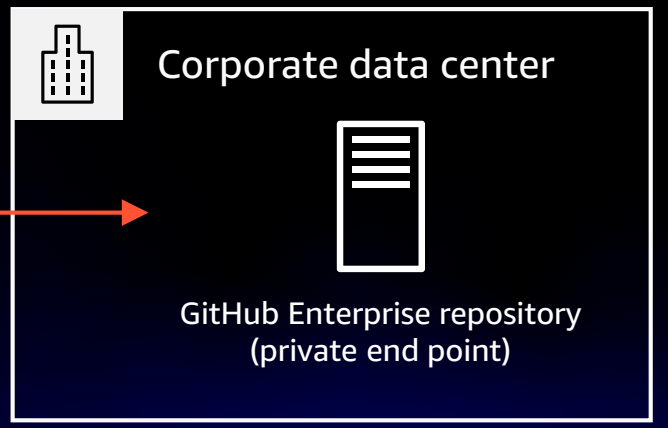
CodeGuru repository support





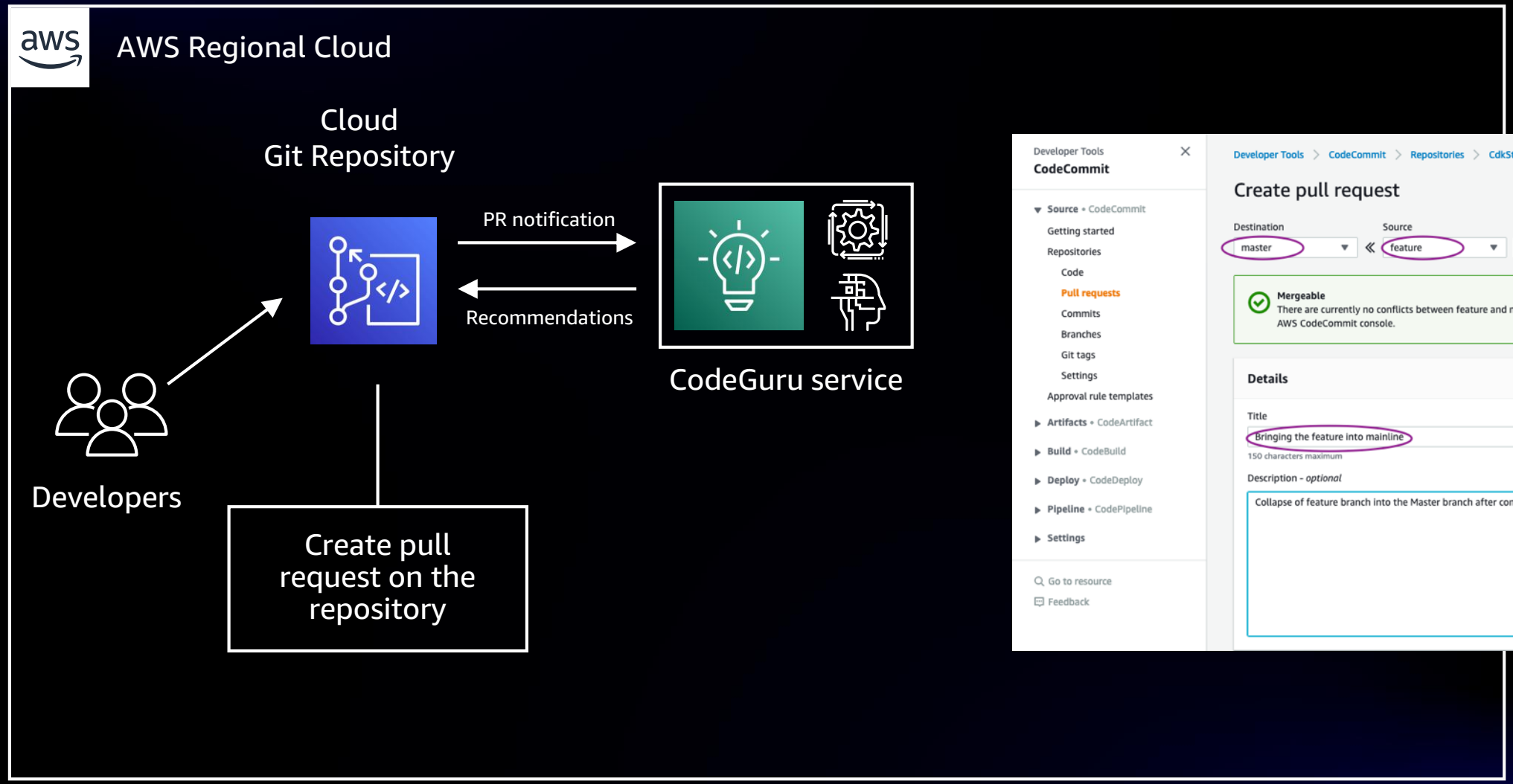


AWS Direct Connect connectivity



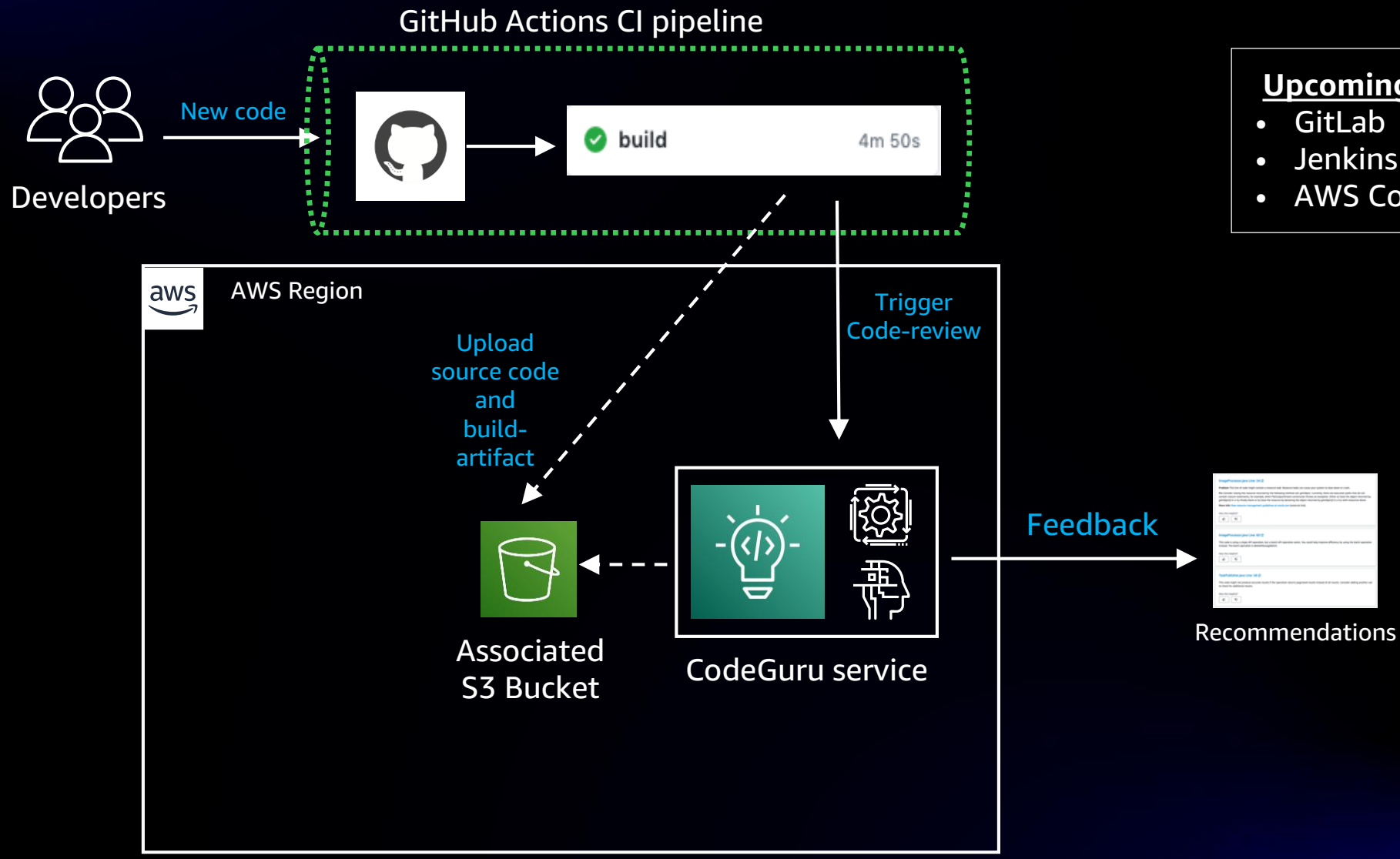
© 2021, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Supported workflows: 1. Pull request



© 2021, Amazon Web Services, Inc. or its affiliates. All rights reserved.

2. CodeGuru using GitHub Actions CI pipeline



Upcoming CI pipeline Support:

- GitLab
- Jenkins
- AWS CodePipeline

© 2021, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Conclusion

Call to action

Get started with Amazon CodeGuru (free for first 90 days)
https://aws.amazon.com/codeguru/

Automate your security code reviews

- Introduce CodeGuru in your Pull request workflow
- Establish continuous feedback to developers
- Iterate though code improvements
- Maintain a high bar of code quality for your applications

Sign up for our "Continuous security and compliance for your CI/CD pipeline" workshop at re:Invent

- Look for Session ID DOP401

aws

© 2021, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Reference

Sample application code:

github.com/aws-samples/amazon-codeguru-reviewer-sample-app

Useful blogs and videos:

- CodeGuru thread concurrency blog: https://go.aws/3EFEgRD
- CodeGuru security analysis: https://go.aws/2YbD2Ow
- Validate IAM policies in CloudFormation templates using IAM Access Analyzer: https://go.aws/3k4Iz13
- Amazon's approach to security during development: https://go.aws/3ELwTbF

Useful workshops:

- Integrating security into your container pipeline: https://bit.ly/31iOCZm

© 2021, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Thank you!

Curtis Rissi

@boringgeek

Sundeep Goswami

linkedin.com/in/sundeepgoswami/

© 2021, Amazon Web Services, Inc. or its affiliates. All rights reserved.