AWS re:Invent

DOP316-R

Securing your CI/CD tools and environments

Leo Zhadanovsky
Principal Solutions Architect
Amazon Web Services

AWS re:Invent

© 2019, Amazon Web Services, Inc. or its affiliates. All rights reserved.

aws

Agenda

Overview of AWS developer tools

Integrating security testing into your continuous delivery pipeline

Security of your continuous delivery pipeline

Security in AWS CodeBuild, AWS CodeCommit, AWS CodeDeploy

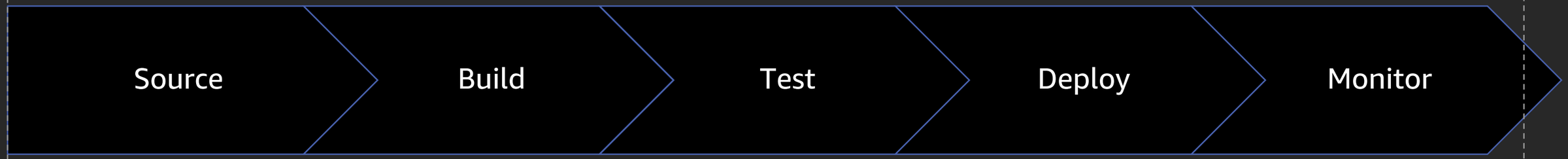
Q&A

AWS code services

Software release steps:

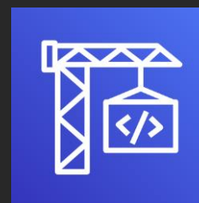


AWS CodeStar





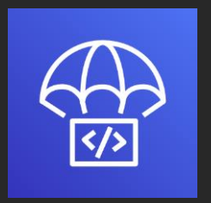
AWS CodeCommit



AWS CodeBuild



AWS CodeBuild + third party



AWS CodeDeploy



AWS X-Ray



AWS CodePipeline



Amazon CloudWatch

Integrating security into your continuous delivery pipeline

- CodeBuild projects and custom actions can be used to integrate security testing
  - Custom actions work via either Lambda functions or instance-based plugins
  - All three options have VPC support, and can test resources in private subnets
- Example use cases
  - Use Lambda Function custom action to trigger a third-party load test
  - Use Lambda Function custom action to trigger AWS Inspector to assess an AMI being built via CodeBuild and CodePipeline

Integrating security into your continuous delivery pipeline



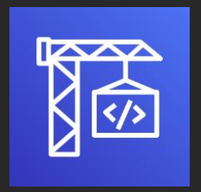
AWS CodePipeline

Release automation



AWS CodeCommit

1. Commits to master



AWS CodeBuild

2. TaskCat and cfn_nag



cfn_nag



AWS CodeBuild

3. Copy to Amazon S3 and execute AWS CloudFormation

Deploy with AWS CloudFormation



Amazon S3



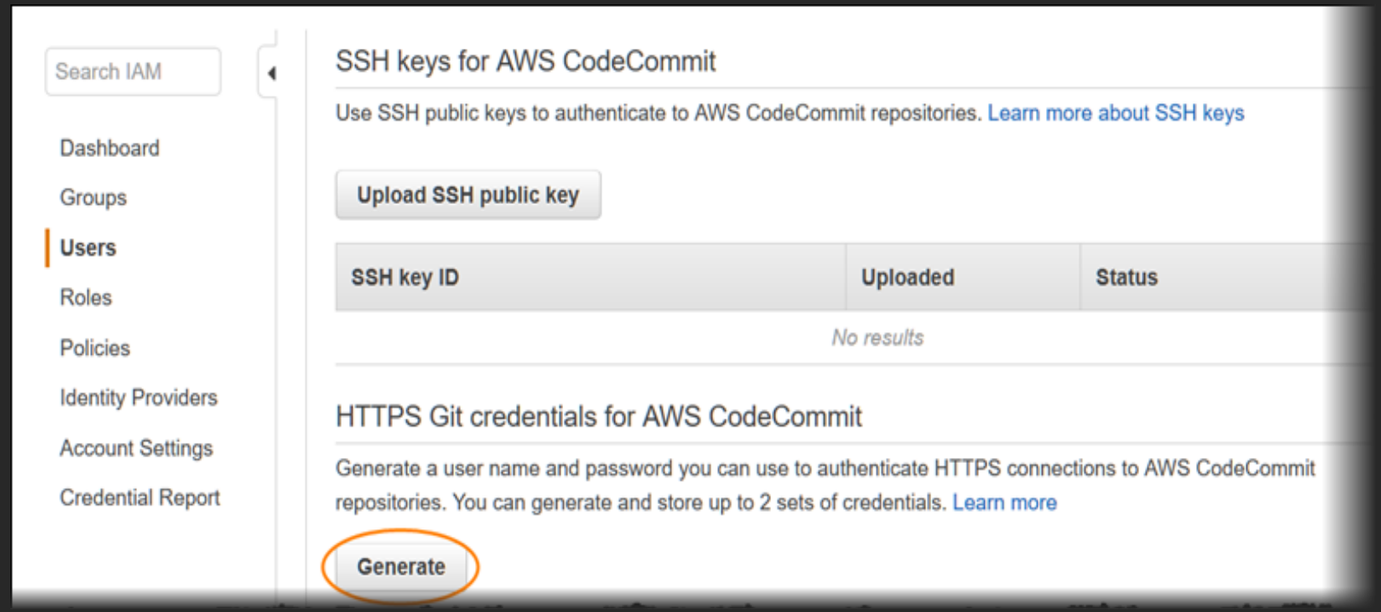
AWS CloudFormation

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

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



Security in CodeBuild

- Build artifacts are all encrypted with AWS Key Management Service 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 ECR, make sure to enable image scanning on push on the ECR repository

Security in CodeBuild

- CodeBuild projects can run in Amazon Virtual Private Cloud
  - 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
```

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 IAM to control access to
  - Applications and application revisions
  - Deployments
  - Deployment configurations
  - Instances and on-premises instances

Security in CodeDeploy

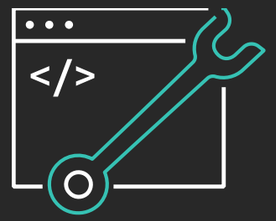
- Instance running CodeDeploy agent must have an IAM instance profile
  - Allow instance to access S3 buckets or GitHub repos where artifacts are stored
- The IAM service role is used by CodeDeploy
  - This allows CodeDeploy to do things like read Amazon Elastic Compute Cloud tags and interact with ELBs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

Q&A

© 2019, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Learn DevOps with AWS Training and Certification

Resources created by the experts at AWS to propel your organization and career forward



Take free digital training to learn best practices for developing, deploying, and maintaining applications



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



Validate expertise with the **AWS Certified DevOps Engineer - Professional** or **AWS Certified Developer - Associate** exams

Visit aws.amazon.com/training/path-developing/

© 2019, Amazon Web Services, Inc. or its affiliates. All rights reserved.

aws training and certification

Thank you!

Leo Zhadanovsky

@leozh

AWS re:Invent

© 2019, Amazon Web Services, Inc. or its affiliates. All rights reserved.

aws



Please complete the session survey in the mobile app.

© 2019, Amazon Web Services, Inc. or its affiliates. All rights reserved.