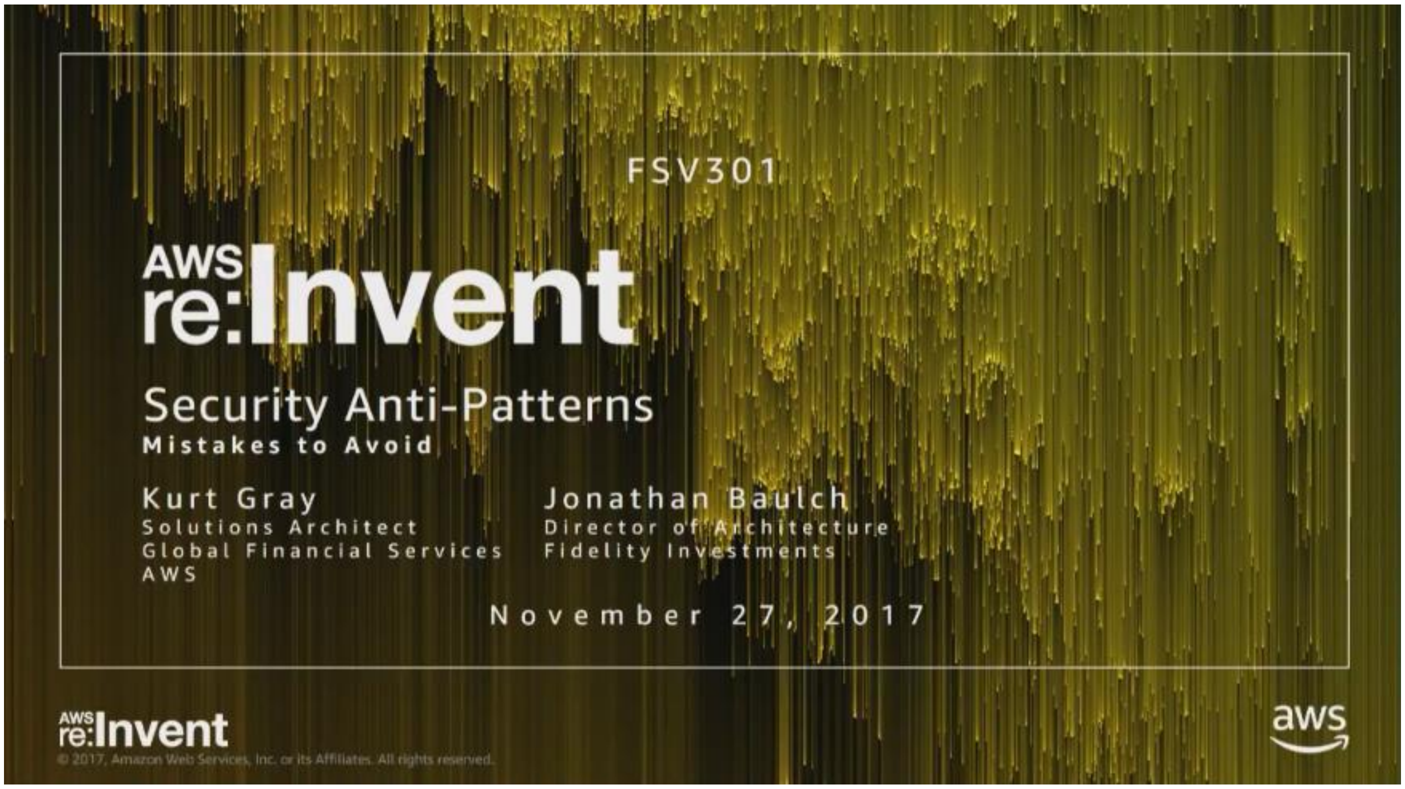


FSV301

Security Anti-Patterns

Mistakes to Avoid

Kurt Gray
Solutions Architect
Global Financial Services
AWS

Jonathan Baulch
Director of Architecture
Fidelity Investments

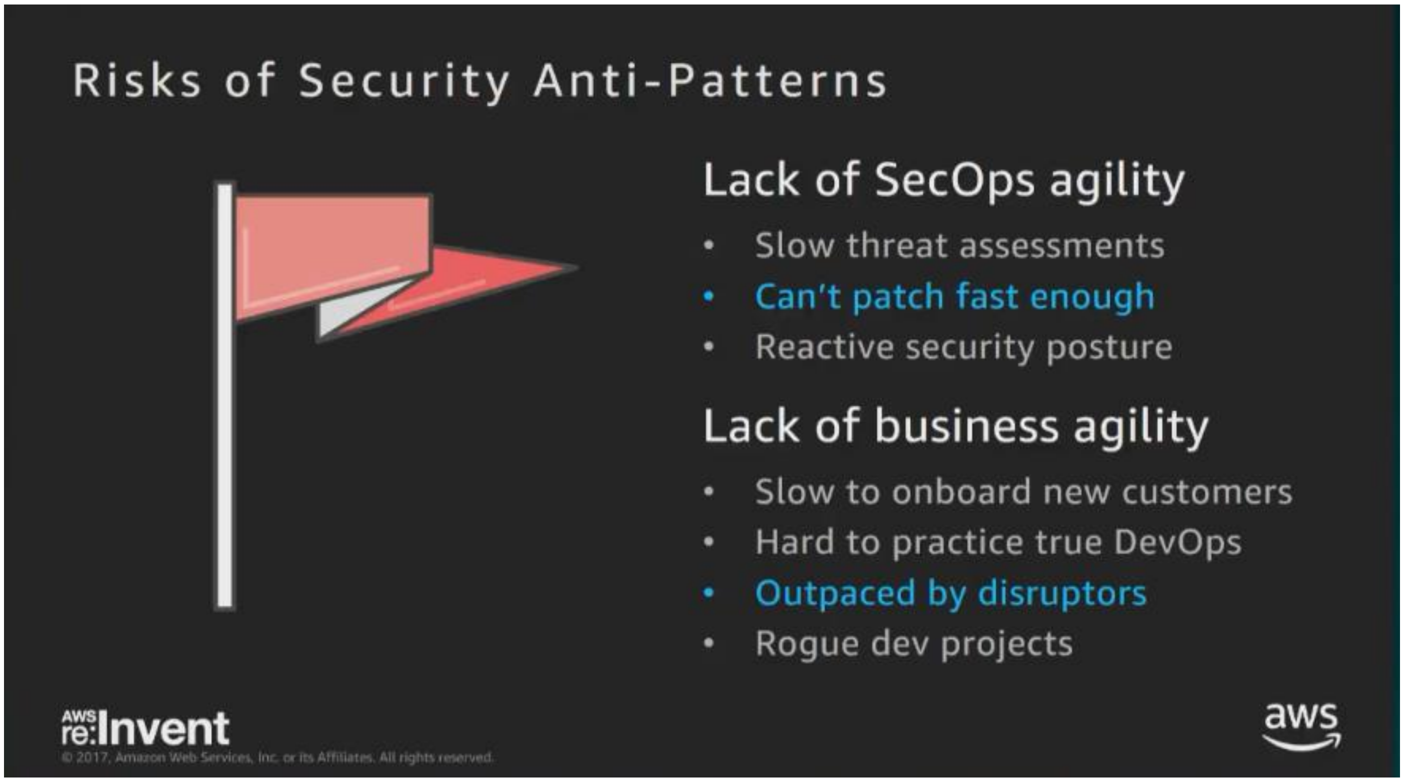
November 27, 2017

© 2017, Amazon Web Services, Inc. or its Affiliates. All rights reserved.

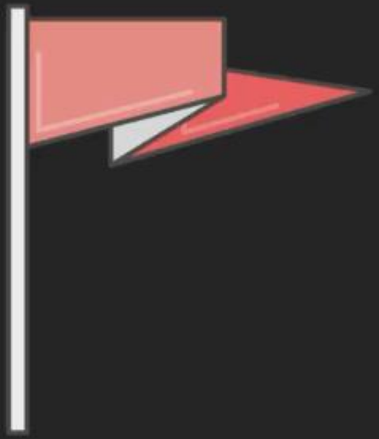
At AWS, security is job zero. Our infrastructure is architected for the most data-sensitive, financial services companies in the world. We have worked with global enterprises to meet their respective security requirements and have learned that there are best practices and pitfalls to avoid. In this session, we provide a guided tour of governance patterns to avoid – ones that may seem logical at first, but that actually impede your ability scale and realize business agility. We also cover best practices, such as setting up key preventative and detective controls for implementing 360-degrees of security coverage, practicing DevSecOps on a massive scale, and leveraging the AWS services (such as Amazon VPC, IAM, Amazon EMR, Amazon S3, Amazon CloudWatch, and AWS Lambda) to meet the most strict and robust enterprise security requirements.

Anti-Pattern: A common response to a recurring problem that is usually ineffective and risks being highly counterproductive

© 2017, Amazon Web Services, Inc. or its Affiliates. All rights reserved.



Risks of Security Anti-Patterns



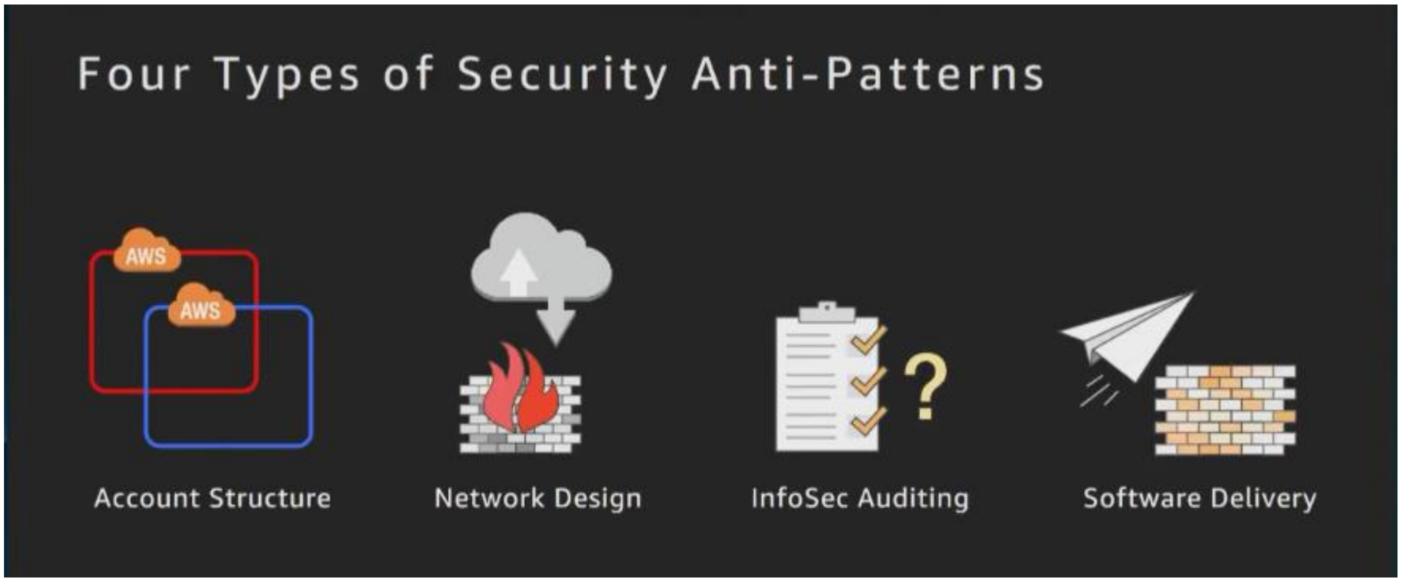
Lack of SecOps agility

- Slow threat assessments
- Can't patch fast enough
- Reactive security posture

Lack of business agility

- Slow to onboard new customers
- Hard to practice true DevOps
- Outpaced by disruptors
- Rogue dev projects

AWS re:Invent
© 2017, Amazon Web Services, Inc. or its Affiliates. All rights reserved.
aws



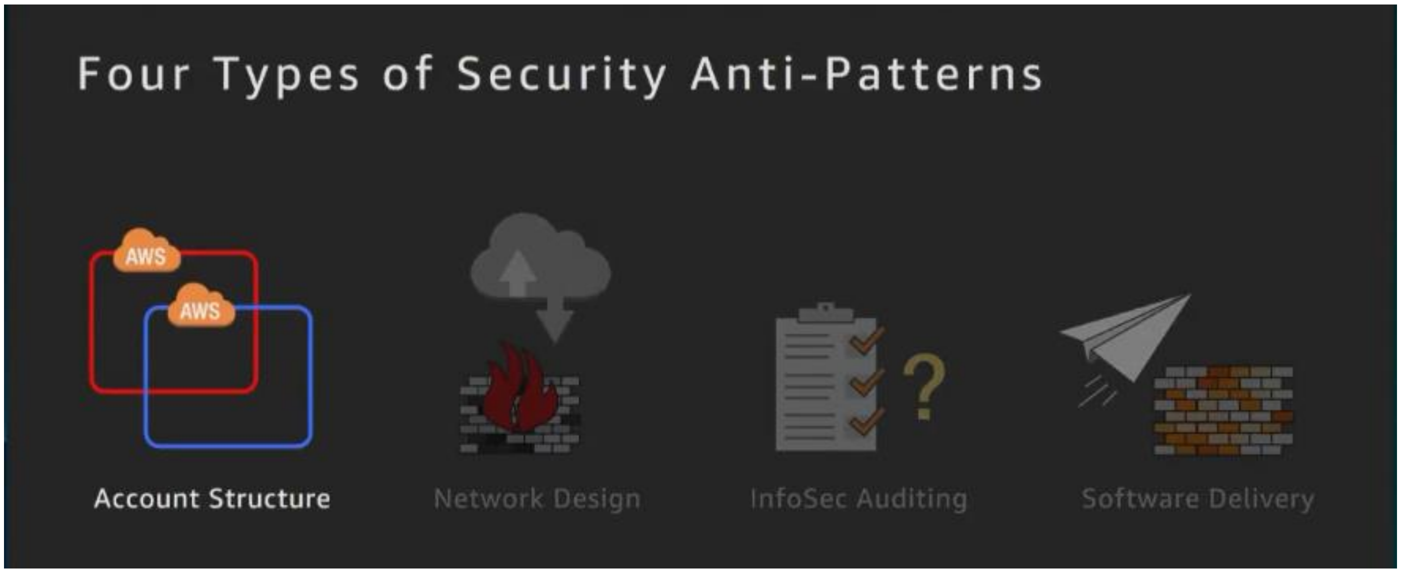
Four Types of Security Anti-Patterns

Account Structure

Network Design

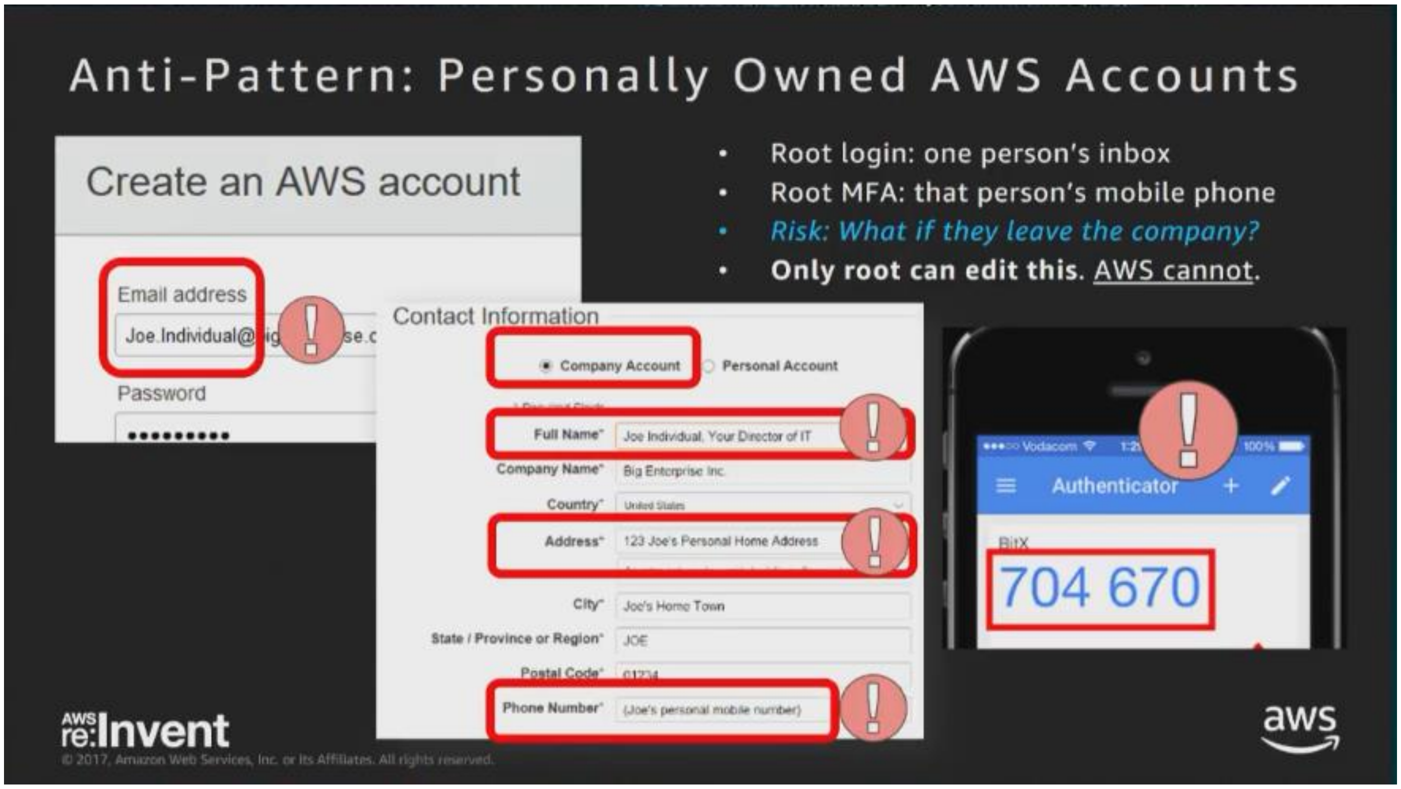
InfoSec Auditing

Software Delivery



Four Types of Security Anti-Patterns

- Account Structure
- Network Design
- InfoSec Auditing
- Software Delivery



Anti-Pattern: Personally Owned AWS Accounts

- Root login: one person's inbox
- Root MFA: that person's mobile phone
- *Risk: What if they leave the company?*
- **Only root can edit this.** AWS cannot.

Create an AWS account

Email address: Joe.Individual@ig se.c

Password: ••••••••••

Contact Information

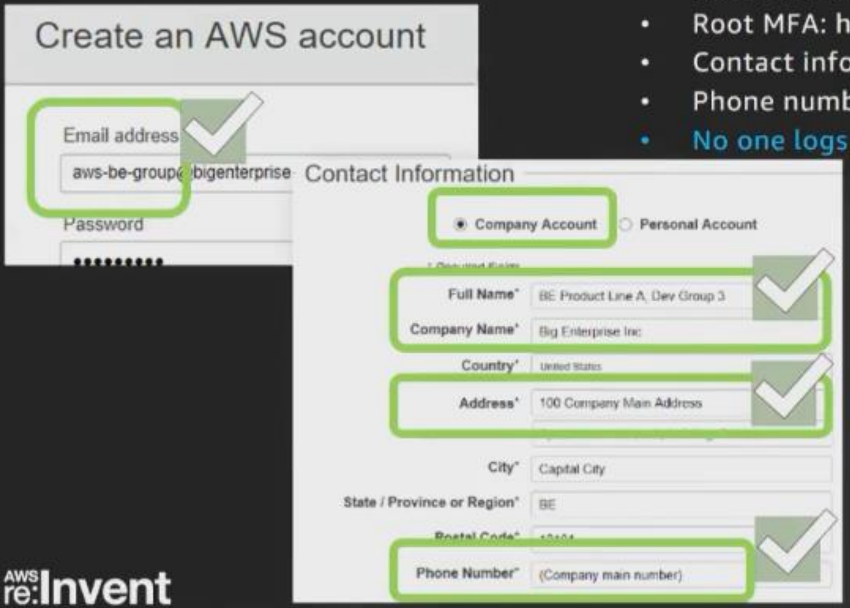
[✓ Company Account] Personal Account

Full Name*	Joe Individual, Your Director of IT
Company Name*	Big Enterprise Inc.
Country*	United States
Address*	123 Joe's Personal Home Address
City*	Joe's Home Town
State / Province or Region*	JOE
Postal Code*	01234
Phone Number*	(Joe's personal mobile number)

AWS re:Invent
© 2017, Amazon Web Services, Inc. or its Affiliates. All rights reserved.
aws

Best Practice: Group Contacts on All Accounts

- Root email: <u>team distribution list address</u>
- Root MFA: hardware device, in office safe
- Contact info: company street address
- Phone number: company main number
- No one logs into account root! Use IAM only!



Create an AWS account

Email address: aws-be-group@bigenterprise

Password: ••••••••••

Contact Information

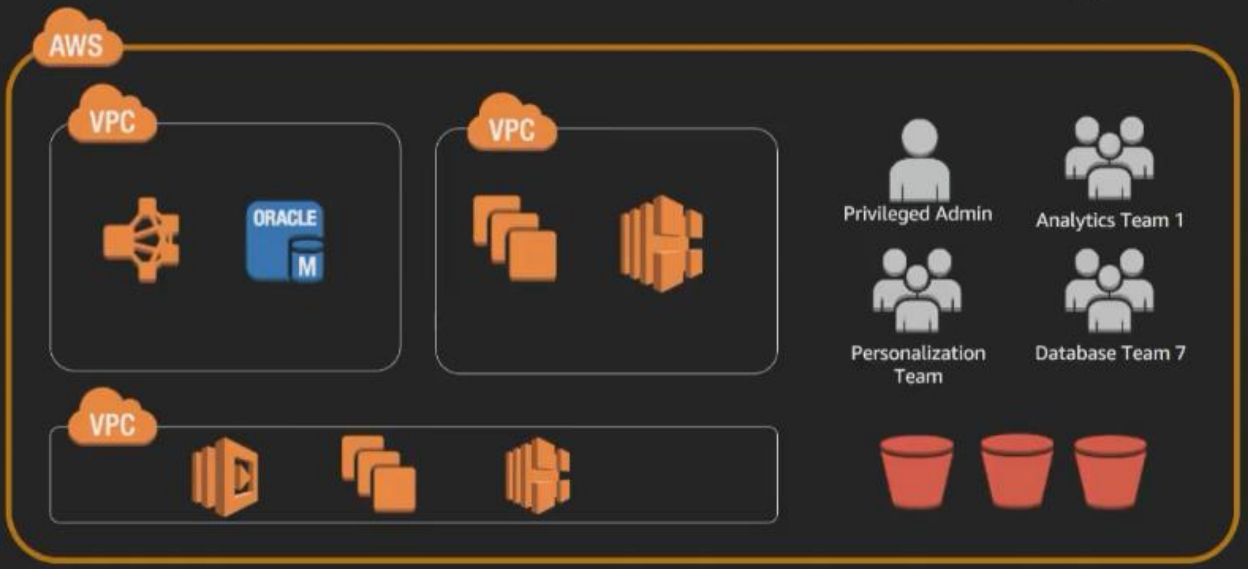
[✓ Company Account] Personal Account

Full Name*	BE Product Line A, Dev Group 3
Company Name*	Big Enterprise Inc
Country*	United States
Address*	100 Company Main Address
City*	Capital City
State / Province or Region*	BE
Postal Code*	[illegible]
Phone Number*	(Company main number)

re:Invent

© 2017, Amazon Web Services, Inc. or its Affiliates. All rights reserved.

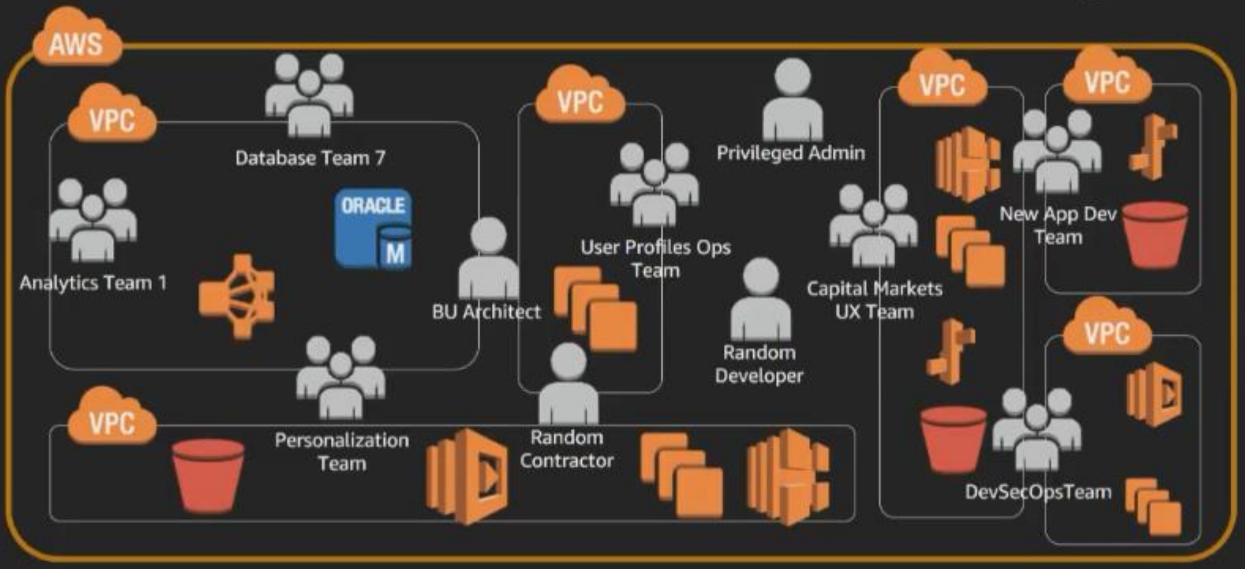
Anti-Pattern: AWS Account Overcrowding



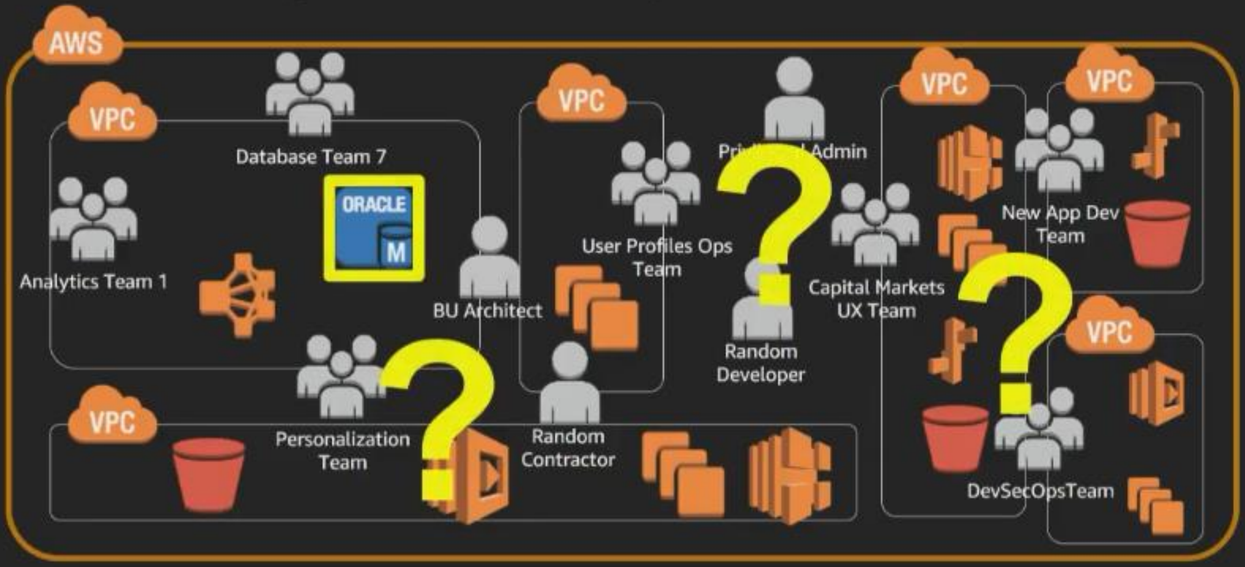
re:Invent

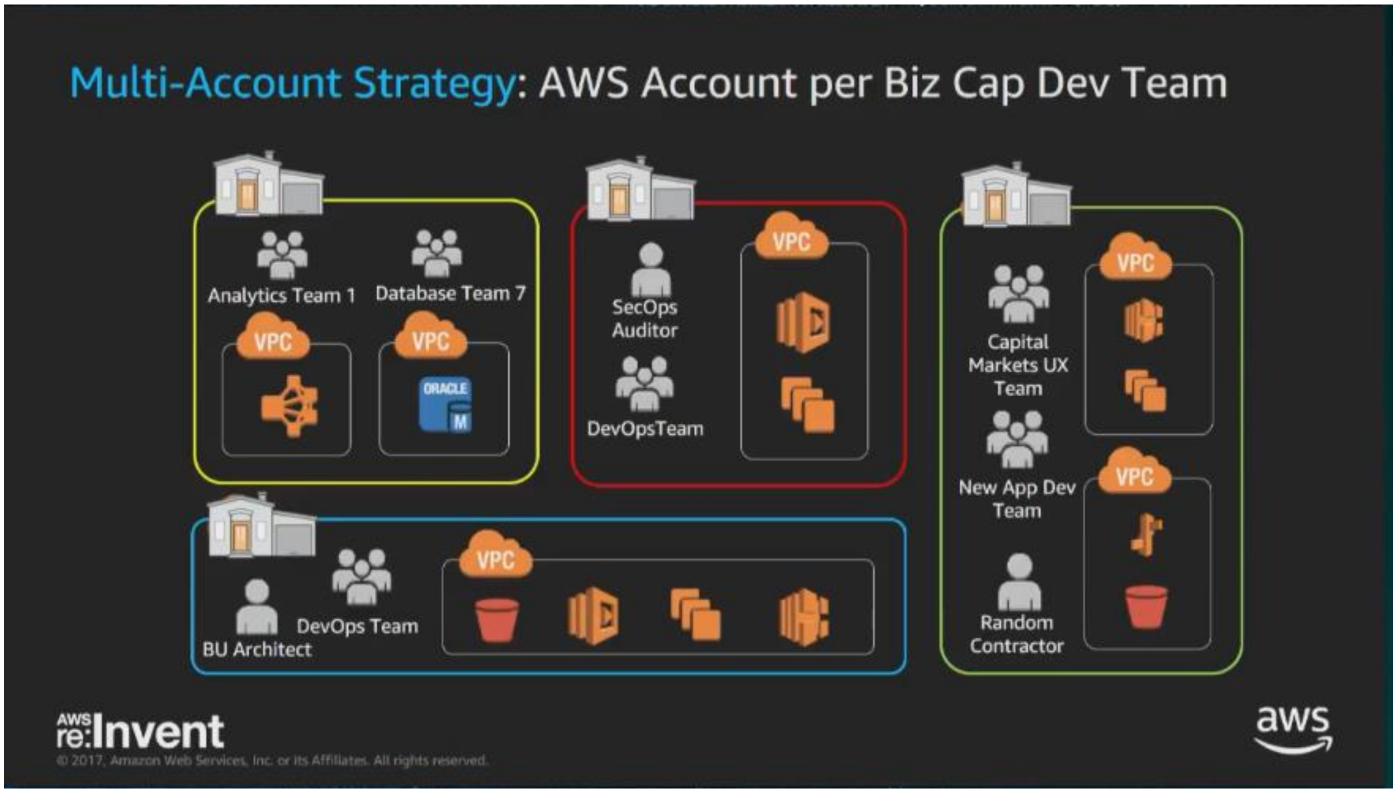
© 2017, Amazon Web Services, Inc. or its Affiliates. All rights reserved.

Anti-Pattern: AWS Account Overcrowding



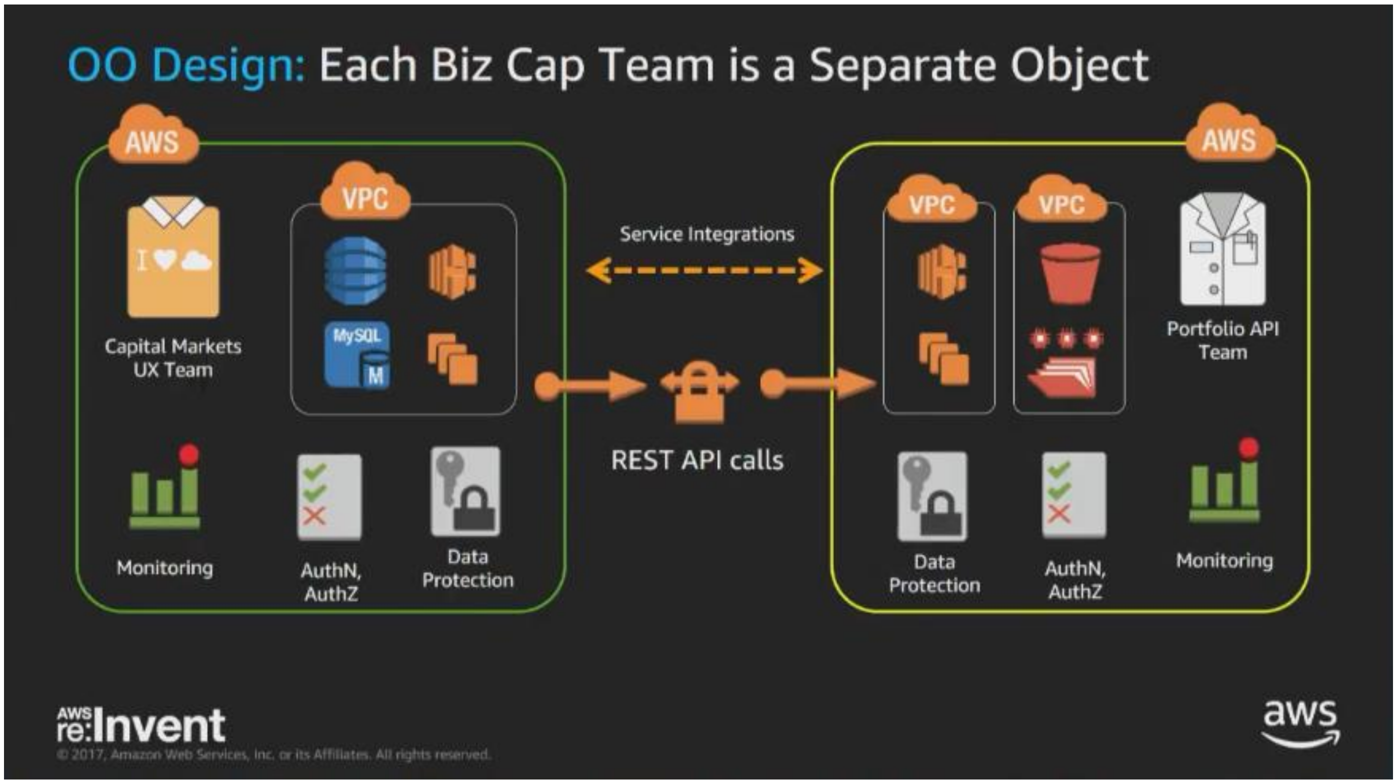
Risk: Ambiguous Security Boundaries





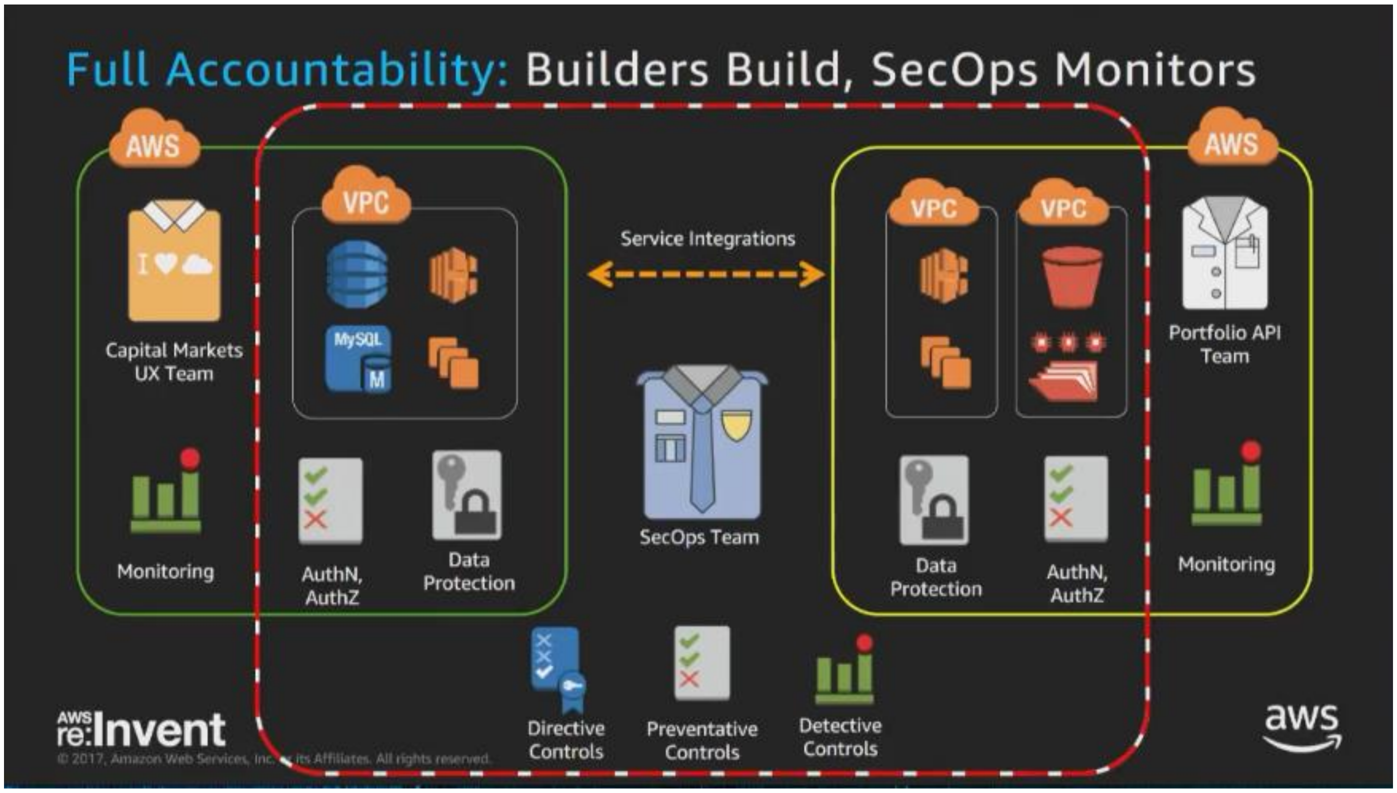
Multi-Account Strategy: AWS Account per Biz Cap Dev Team

AWS re:Invent
© 2017, Amazon Web Services, Inc. or its Affiliates. All rights reserved.
aws



OO Design: Each Biz Cap Team is a Separate Object

AWS re:Invent
© 2017, Amazon Web Services, Inc. or its Affiliates. All rights reserved.
aws

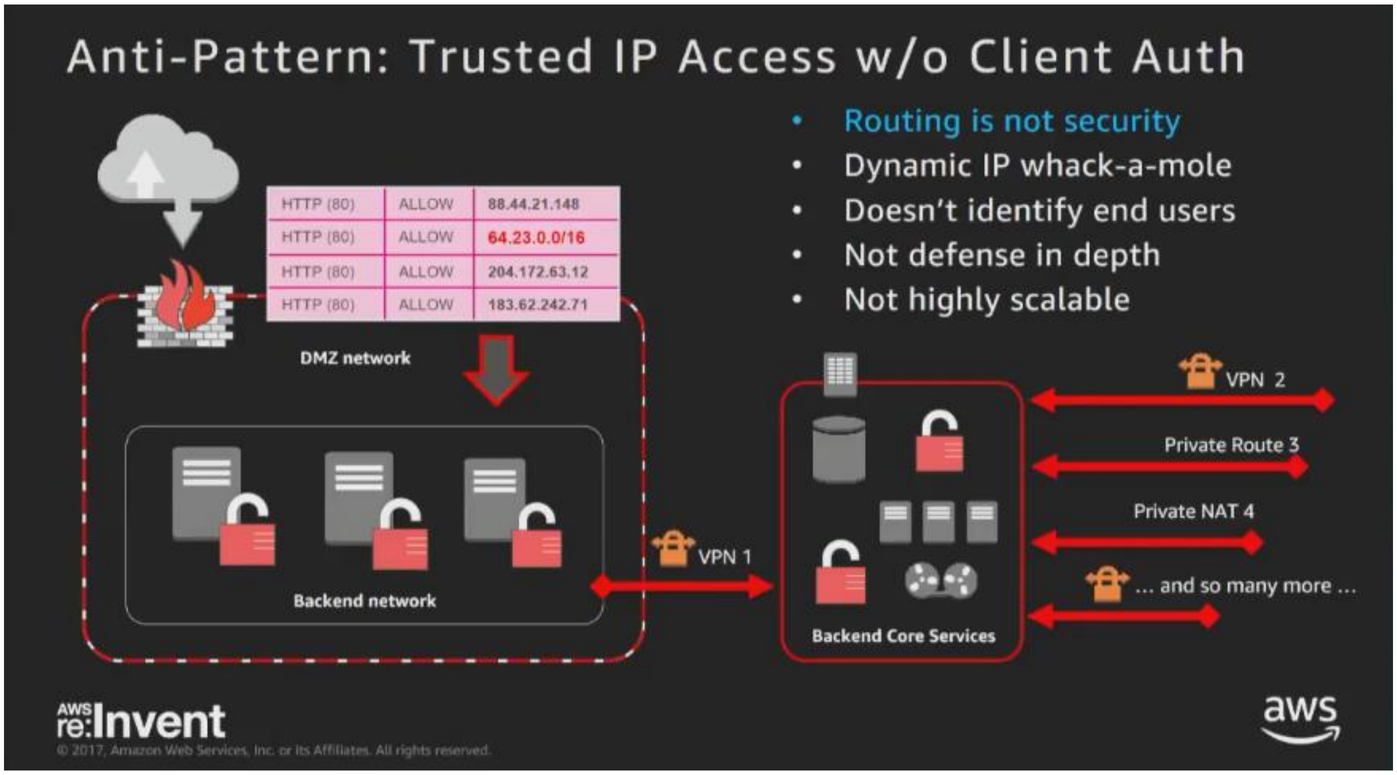


Full Accountability: Builders Build, SecOps Monitors

© 2017, Amazon Web Services, Inc. or its Affiliates. All rights reserved.

Four Types of Security Anti-Patterns

AWS
AWS
Account Structure
Network Design
InfoSec Auditing
Software Delivery

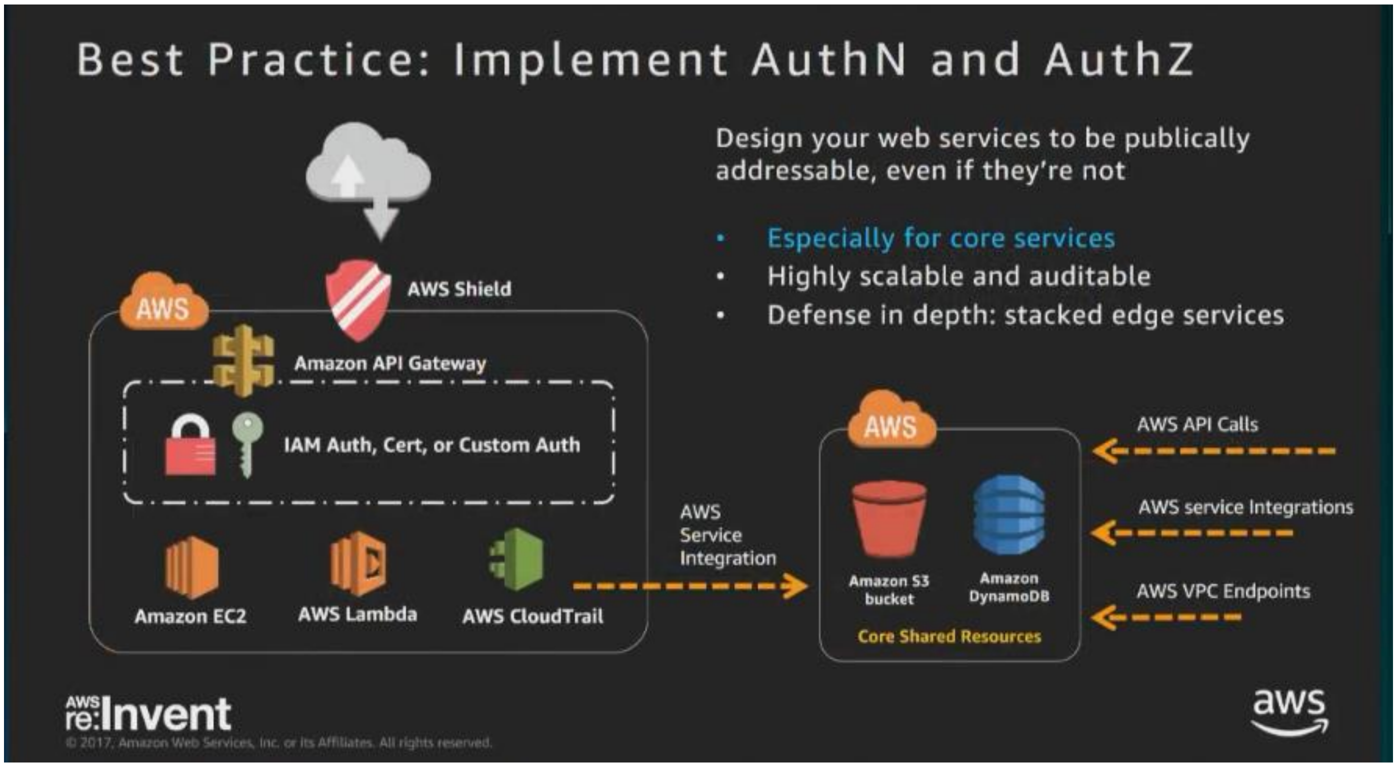


Anti-Pattern: Trusted IP Access w/o Client Auth

- Routing is not security
- Dynamic IP whack-a-mole
- Doesn't identify end users
- Not defense in depth
- Not highly scalable

HTTP (80)	ALLOW	88.44.21.148
HTTP (80)	ALLOW	64.23.0.0/16
HTTP (80)	ALLOW	204.172.63.12
HTTP (80)	ALLOW	183.62.242.71

AWS re:Invent
© 2017, Amazon Web Services, Inc. or its Affiliates. All rights reserved.
aws

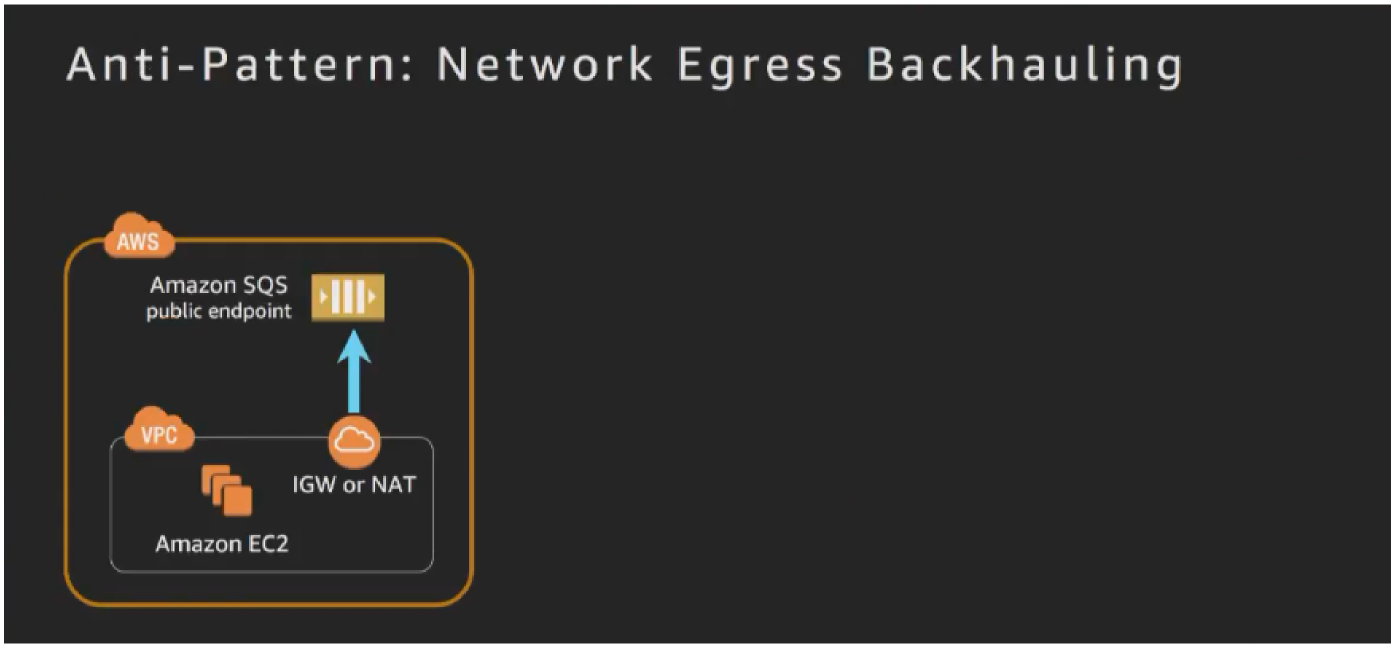


Best Practice: Implement AuthN and AuthZ

Design your web services to be publically addressable, even if they're not

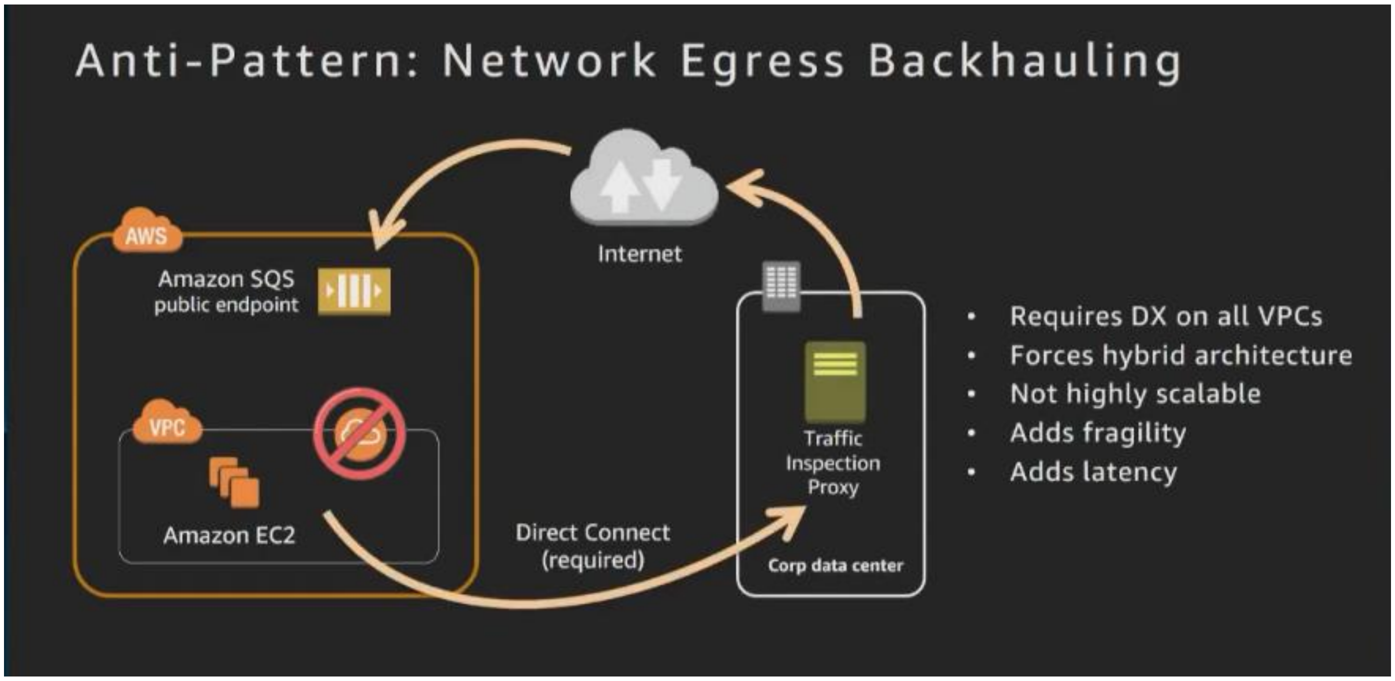
- Especially for core services
- Highly scalable and auditable
- Defense in depth: stacked edge services

AWS re:Invent
© 2017, Amazon Web Services, Inc. or its Affiliates. All rights reserved.
aws



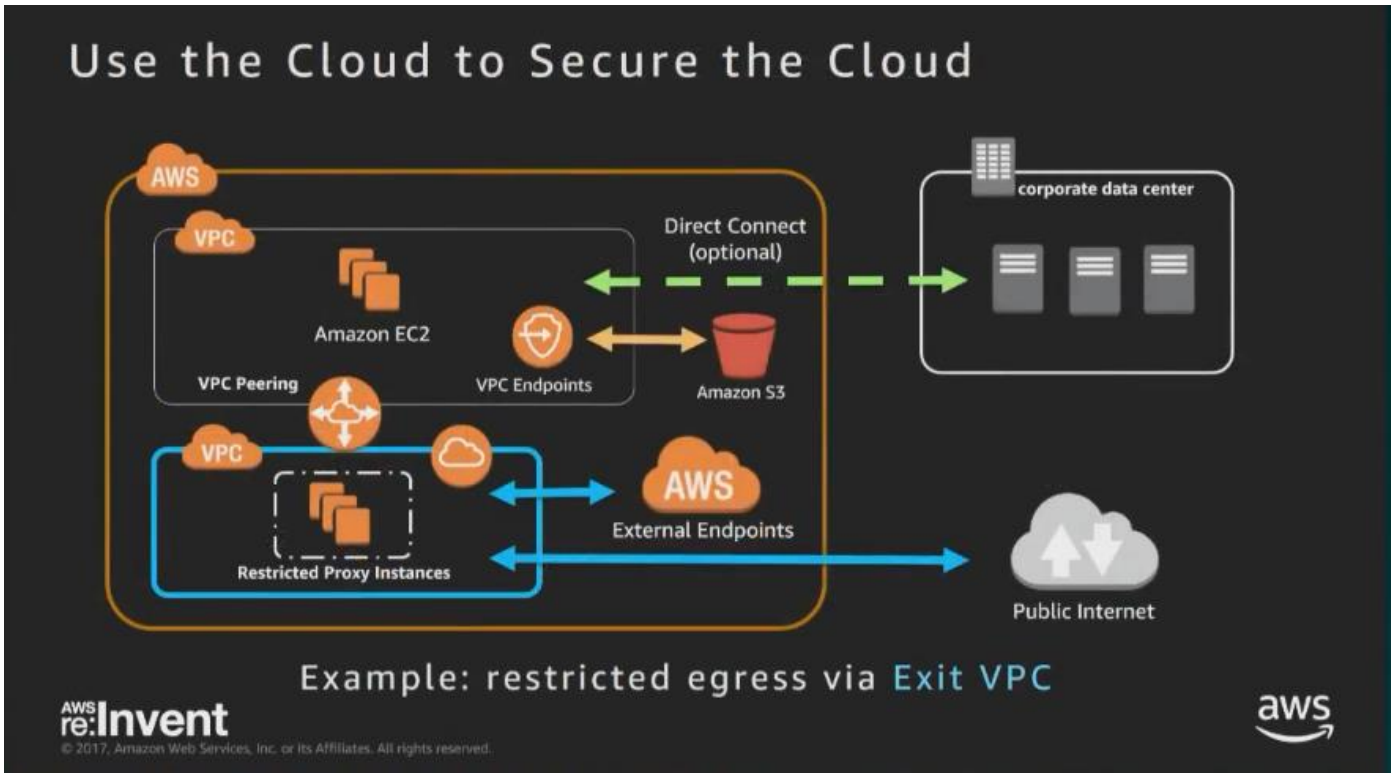
Anti-Pattern: Network Egress Backhauling

We need some Layer 7 controls on the NAT for traffic going out, a possible solution is below



Anti-Pattern: Network Egress Backhauling

- Requires DX on all VPCs
- Forces hybrid architecture
- Not highly scalable
- Adds fragility
- Adds latency



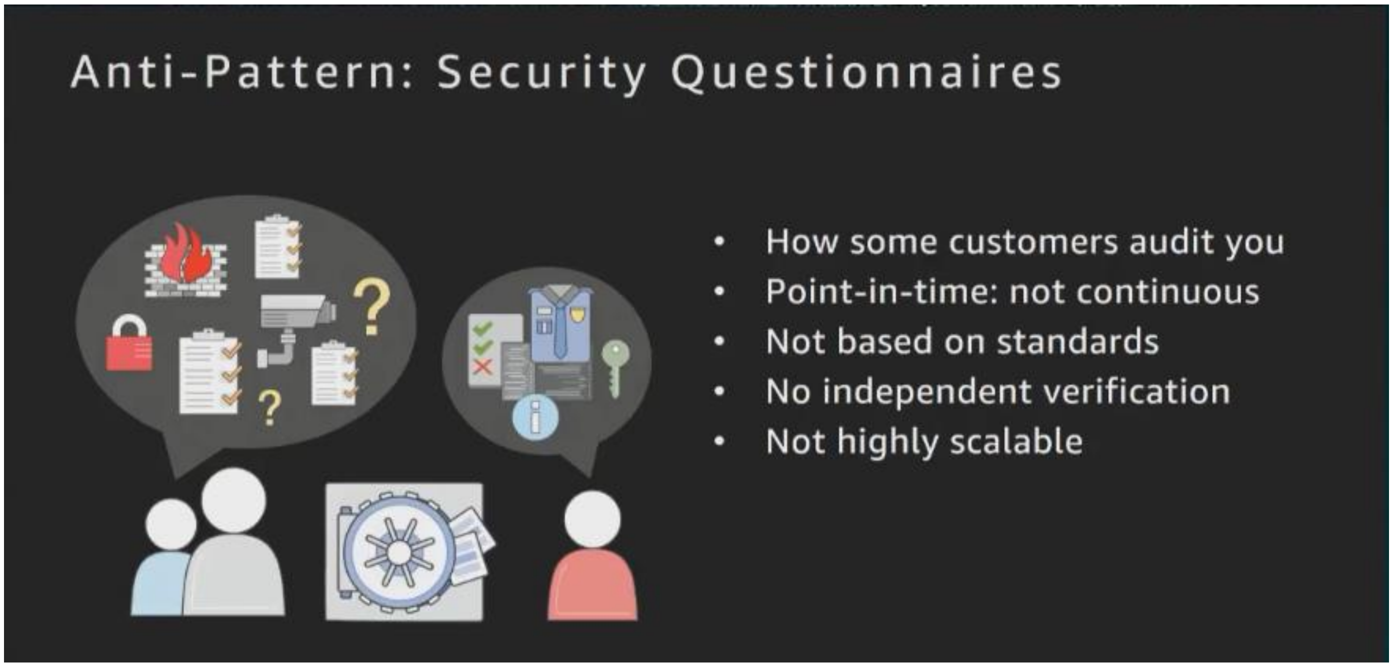
Use the Cloud to Secure the Cloud

Example: restricted egress via Exit VPC

© 2017, Amazon Web Services, Inc. or its Affiliates. All rights reserved.

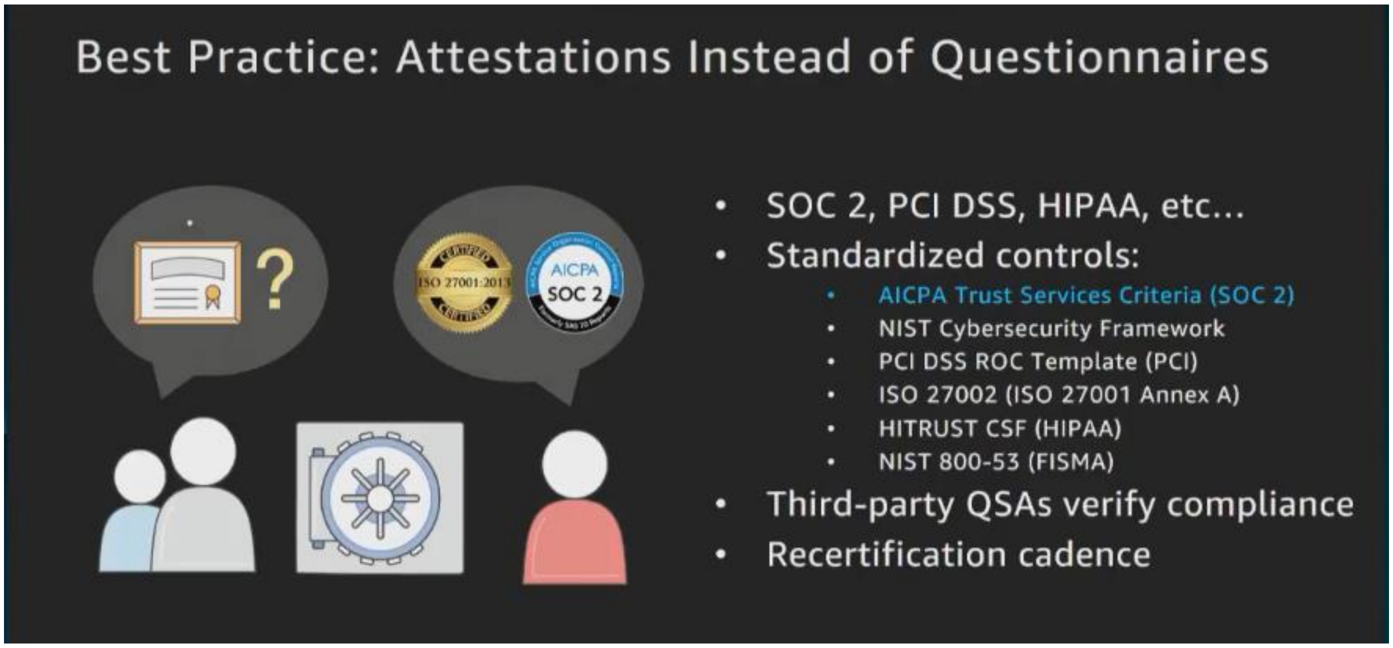
Use Egress VPCs (a pool of EC2 servers running proxy filtering) instead as above

Four Types of Security Anti-Patterns



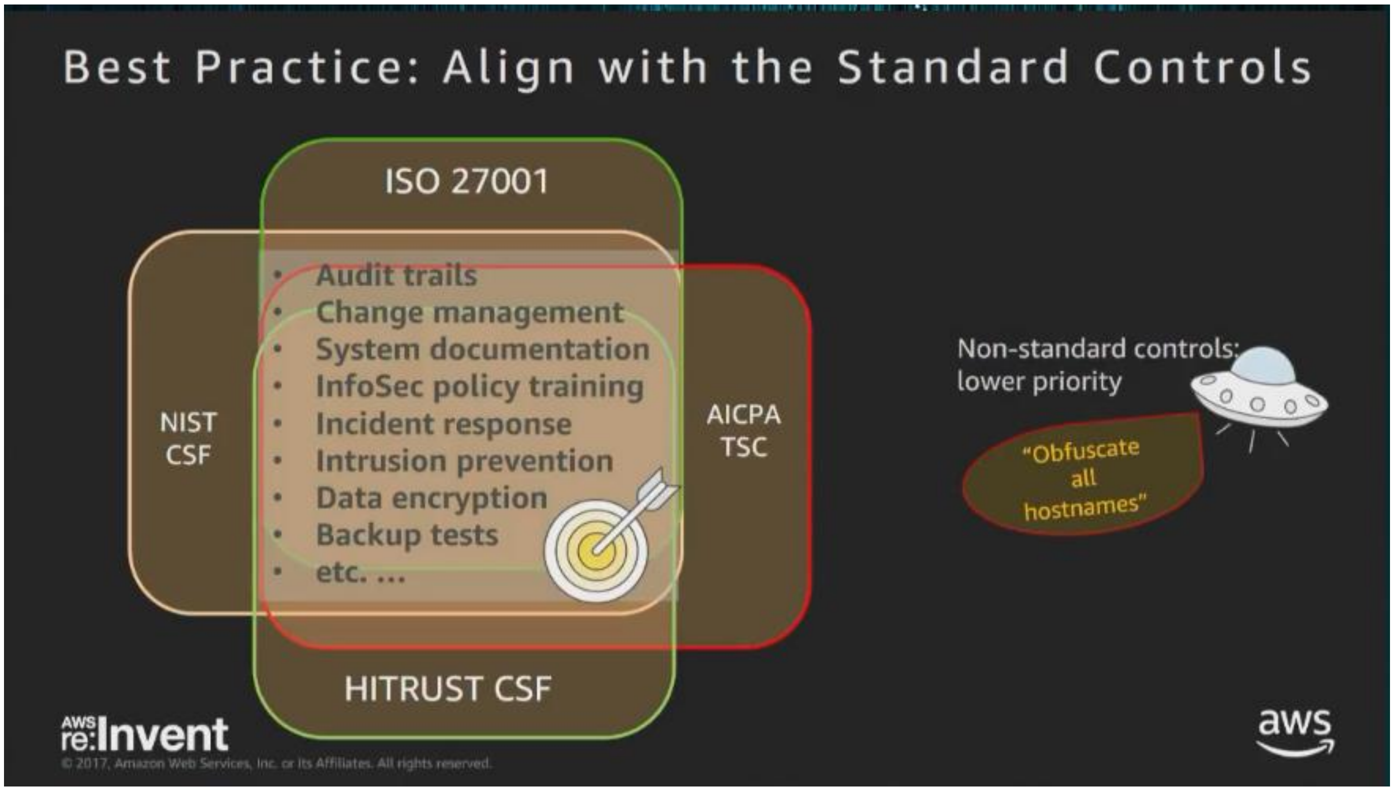
Anti-Pattern: Security Questionnaires

- How some customers audit you
- Point-in-time: not continuous
- Not based on standards
- No independent verification
- Not highly scalable



Best Practice: Attestations Instead of Questionnaires

- SOC 2, PCI DSS, HIPAA, etc…
- Standardized controls:
  - AICPA Trust Services Criteria (SOC 2)
  - NIST Cybersecurity Framework
  - PCI DSS ROC Template (PCI)
  - ISO 27002 (ISO 27001 Annex A)
  - HITRUST CSF (HIPAA)
  - NIST 800-53 (FISMA)
- Third-party QSAs verify compliance
- Recertification cadence

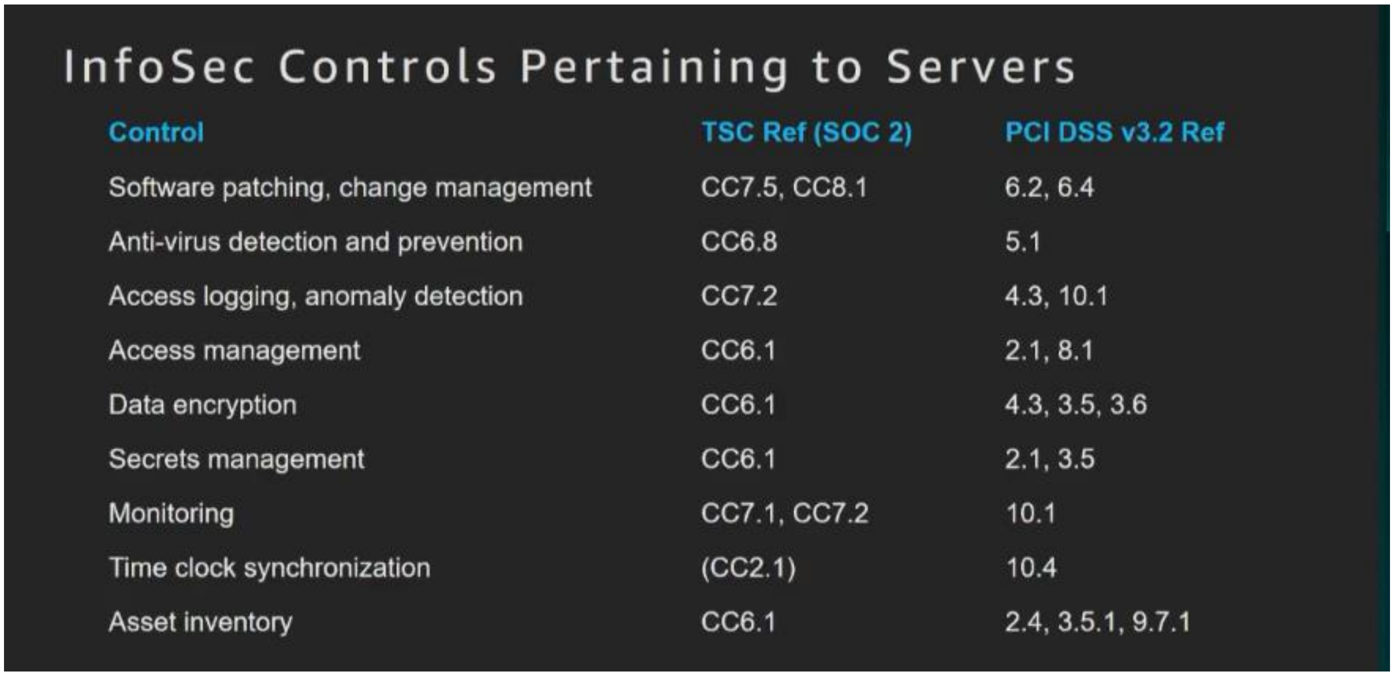


Best Practice: Align with the Standard Controls

ISO 27001
NIST CSF
AICPA TSC
HITRUST CSF
Non-standard controls: lower priority
"Obfuscate all hostnames"

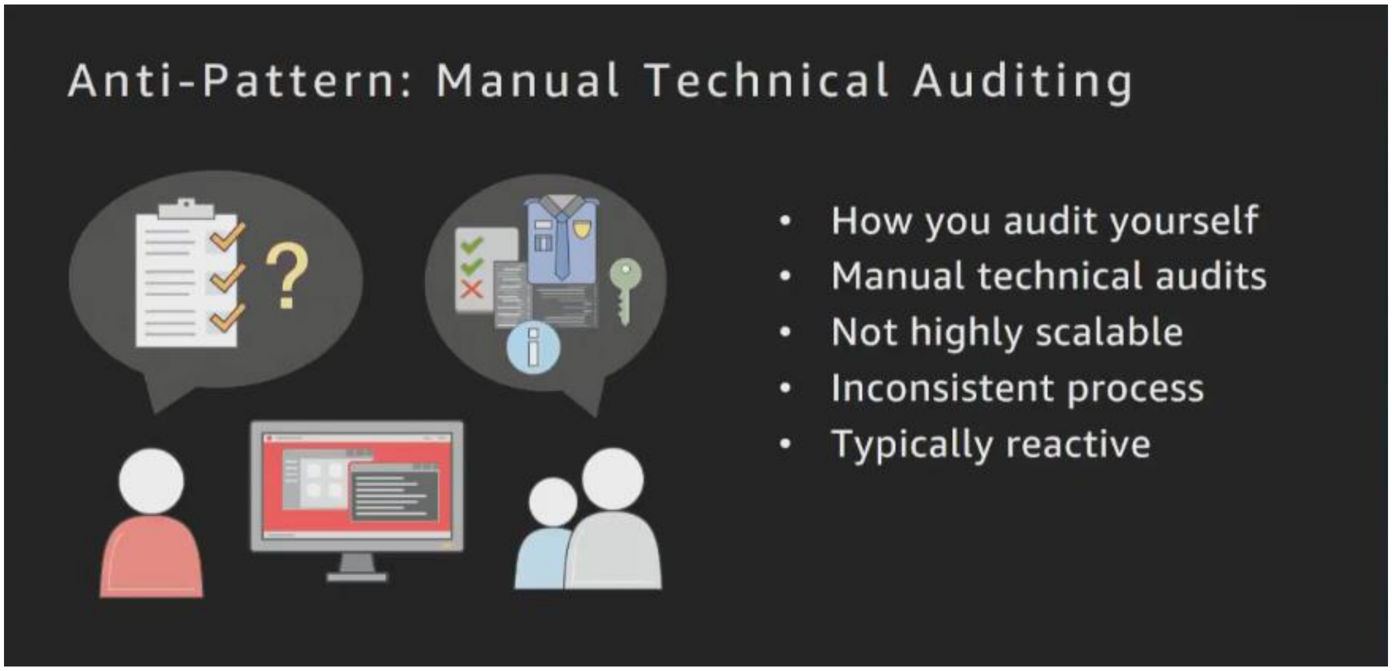
- Audit trails
- Change management
- System documentation
- InfoSec policy training
- Incident response
- Intrusion prevention
- Data encryption
- Backup tests
- etc. ...

© 2017, Amazon Web Services, Inc. or its Affiliates. All rights reserved.



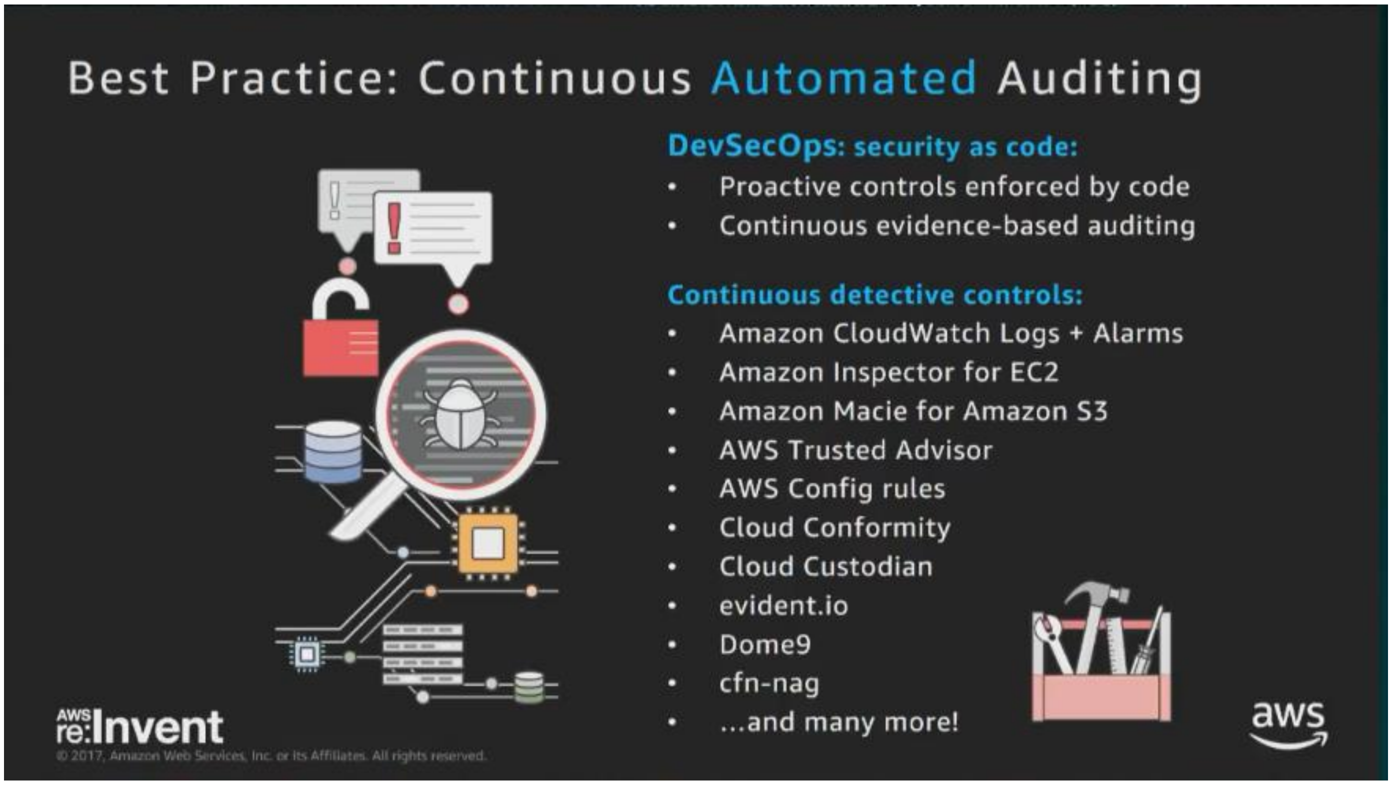
InfoSec Controls Pertaining to Servers

Control	TSC Ref (SOC 2)	PCI DSS v3.2 Ref
Software patching, change management	CC7.5, CC8.1	6.2, 6.4
Anti-virus detection and prevention	CC6.8	5.1
Access logging, anomaly detection	CC7.2	4.3, 10.1
Access management	CC6.1	2.1, 8.1
Data encryption	CC6.1	4.3, 3.5, 3.6
Secrets management	CC6.1	2.1, 3.5
Monitoring	CC7.1, CC7.2	10.1
Time clock synchronization	(CC2.1)	10.4
Asset inventory	CC6.1	2.4, 3.5.1, 9.7.1



Anti-Pattern: Manual Technical Auditing

- How you audit yourself
- Manual technical audits
- Not highly scalable
- Inconsistent process
- Typically reactive



Best Practice: Continuous Automated Auditing

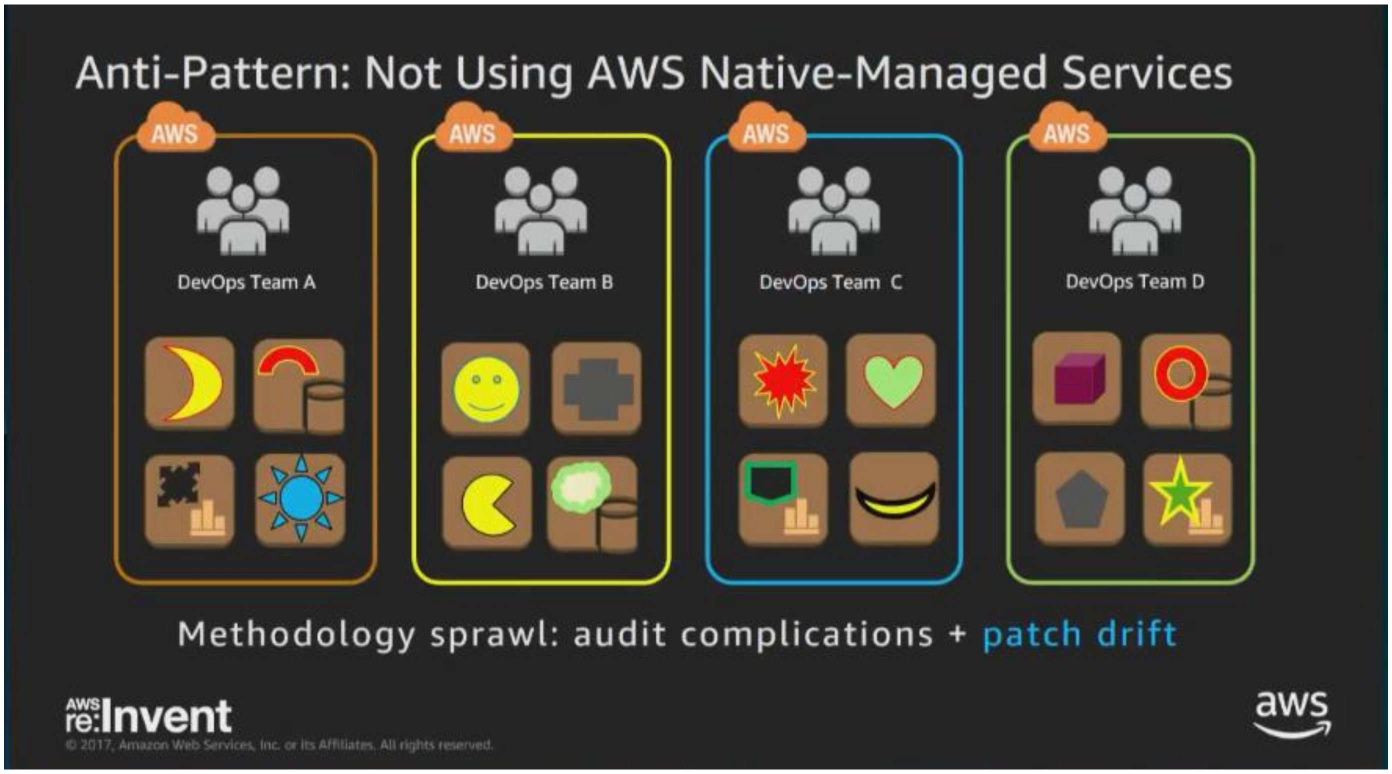
DevSecOps: security as code:

- Proactive controls enforced by code
- Continuous evidence-based auditing

Continuous detective controls:

- Amazon CloudWatch Logs + Alarms
- Amazon Inspector for EC2
- Amazon Macie for Amazon S3
- AWS Trusted Advisor
- AWS Config rules
- Cloud Conformity
- Cloud Custodian
- evident.io
- Dome9
- cfn-nag
- ...and many more!

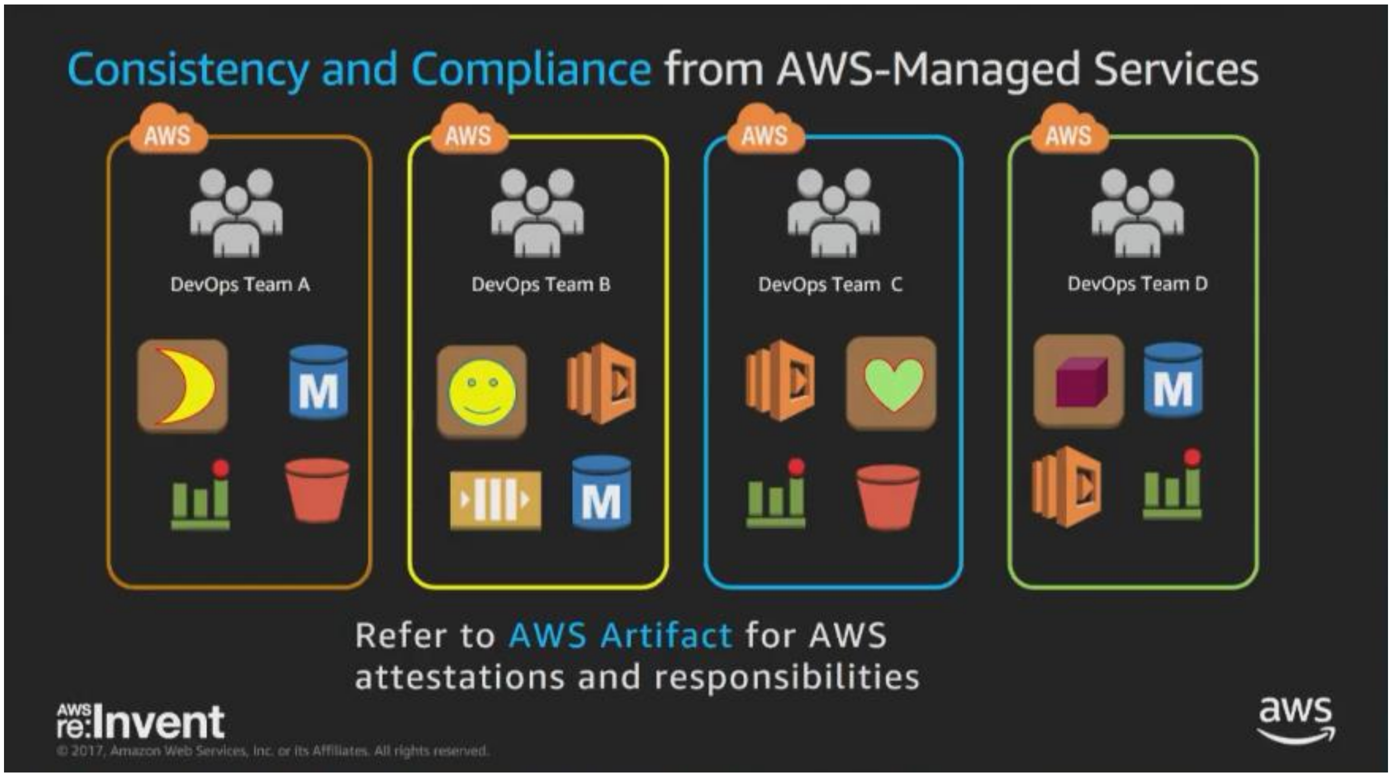
© 2017, Amazon Web Services, Inc. or its Affiliates. All rights reserved.



Anti-Pattern: Not Using AWS Native-Managed Services

Methodology sprawl: audit complications + patch drift

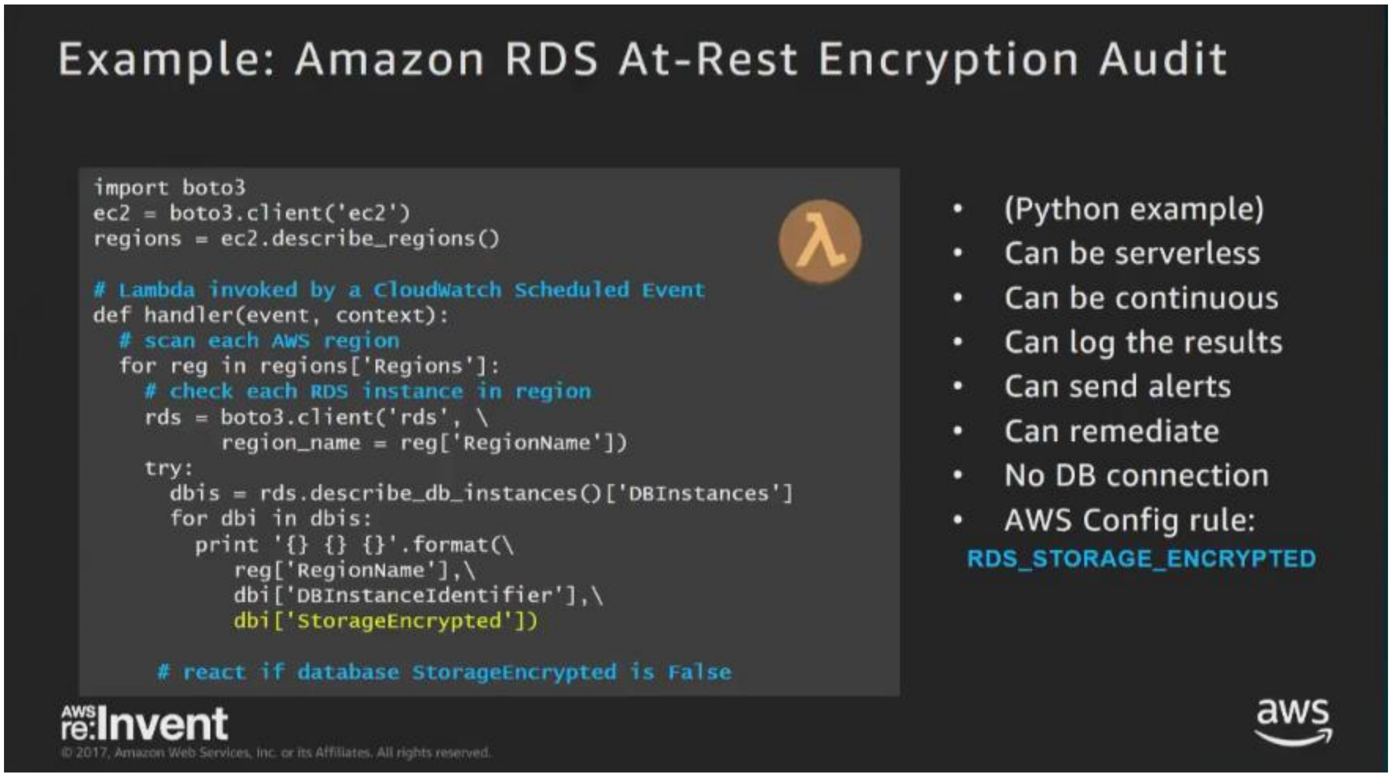
AWS re:Invent
© 2017, Amazon Web Services, Inc. or its Affiliates. All rights reserved.
aws



Consistency and Compliance from AWS-Managed Services

Refer to AWS Artifact for AWS attestations and responsibilities

AWS re:Invent
© 2017, Amazon Web Services, Inc. or its Affiliates. All rights reserved.
aws



Example: Amazon RDS At-Rest Encryption Audit

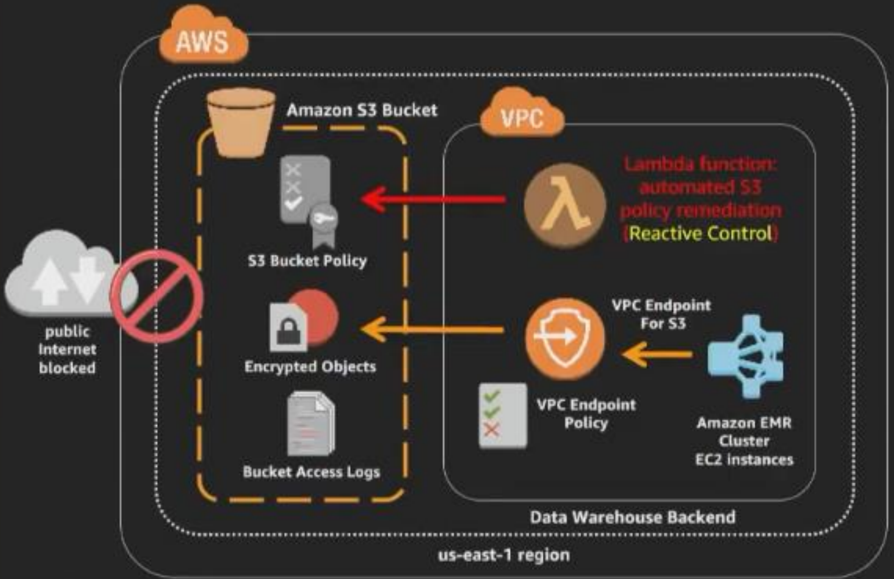
```
import boto3
ec2 = boto3.client('ec2')
regions = ec2.describe_regions()

# Lambda invoked by a CloudWatch Scheduled Event
def handler(event, context):
  # scan each AWS region
  for reg in regions['Regions']:
    # check each RDS instance in region
    rds = boto3.client('rds', \
          region_name = reg['RegionName'])
    try:
      dbis = rds.describe_db_instances()['DBInstances']
      for dbi in dbis:
        print '{} {} {}'.format(\
           reg['RegionName'],\
           dbi['DBInstanceIdentifier'],\
           dbi['StorageEncrypted'])

     # react if database StorageEncrypted is False
```

- (Python example)
- Can be serverless
- Can be continuous
- Can log the results
- Can send alerts
- Can remediate
- No DB connection
- AWS Config rule:
  RDS_STORAGE_ENCRYPTED

re:Invent

© 2017, Amazon Web Services, Inc. or its Affiliates. All rights reserved.

Example: Amazon S3 Bucket Security Controls



Preventative controls:

S3 Bucket Policy
- Deny request unless:
- From specific sourceVpce
- From specific sourceVpc
- (AND) has specific IAM role
- (AND) Server Side Encryption
- (AND) Secure Transport (SSL)
- (AND MFA Delete required)
- (AND Versioning Enabled)

VPC S3 Endpoint Policy
- Denies S3 request unless:
- Targeted to specific S3 buckets

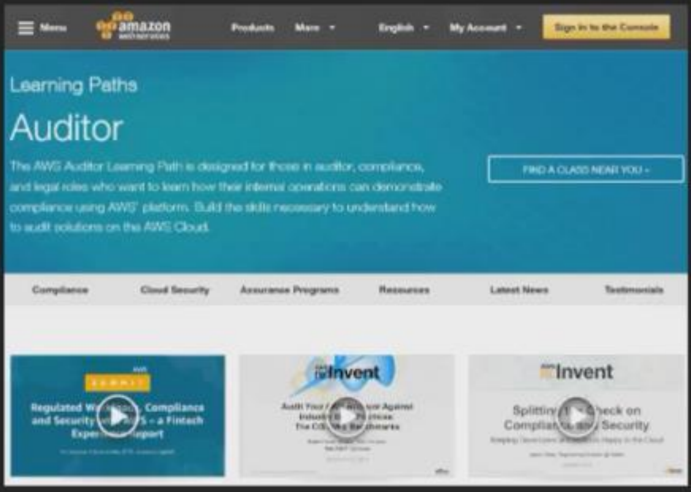
Detective controls:

AWS Config rules
- s3-bucket-logging-enabled
- s3-bucket-public-read-prohibited
- s3-bucket-public-write-prohibited
- s3-bucket-ssl-requests-only

re:Invent

© 2017, Amazon Web Services, Inc. or its Affiliates. All rights reserved.

Best Practice: Train Your Technical Auditors


Menu
amazon web services
Products
More
English
My Account
Sign In to the Console

Learning Paths

Auditor

The AWS Auditor Learning Path is designed for those in auditor, compliance, and legal roles who want to learn how their internal operations can demonstrate compliance using AWS' platform. Build the skills necessary to understand how to audit solutions on the AWS Cloud.

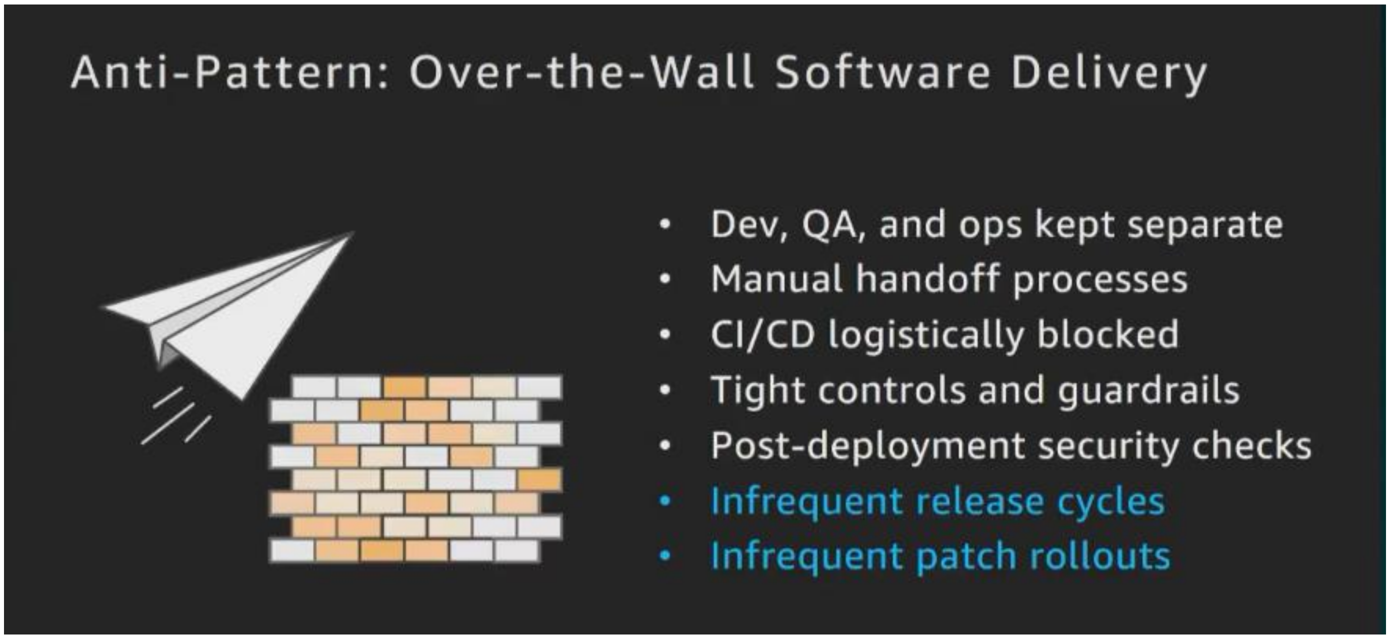
FIND A CLASS NEAR YOU »
Compliance
Cloud Security
Assurance Programs
Resources
Latest News
Testimonials
SUMMIT
Regulated Workloads, Compliance and Security with AWS – a Fintech Experience Report
re:Invent
Audit Your AWS Account Against Industry Best Practices: The CIS AWS Benchmarks
re:Invent
Splitting the Check on Compliance and Security

- AWS Auditor Learning Path
- AWS Tech Essentials
- Goal: DevSecOps

AWS re:Invent
© 2017, Amazon Web Services, Inc. or its Affiliates. All rights reserved.
aws

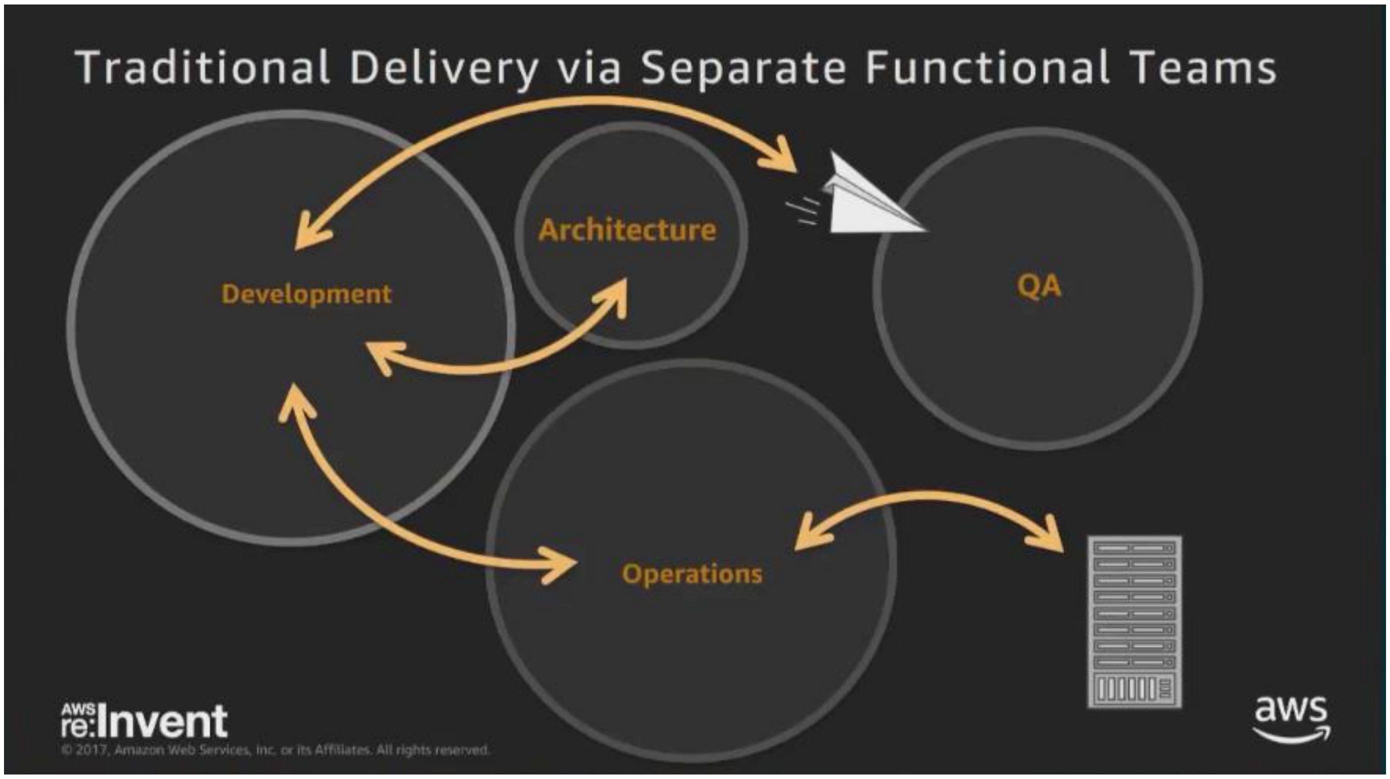
Four Types of Security Anti-Patterns

Account Structure   Network Design   InfoSec Auditing   Software Delivery



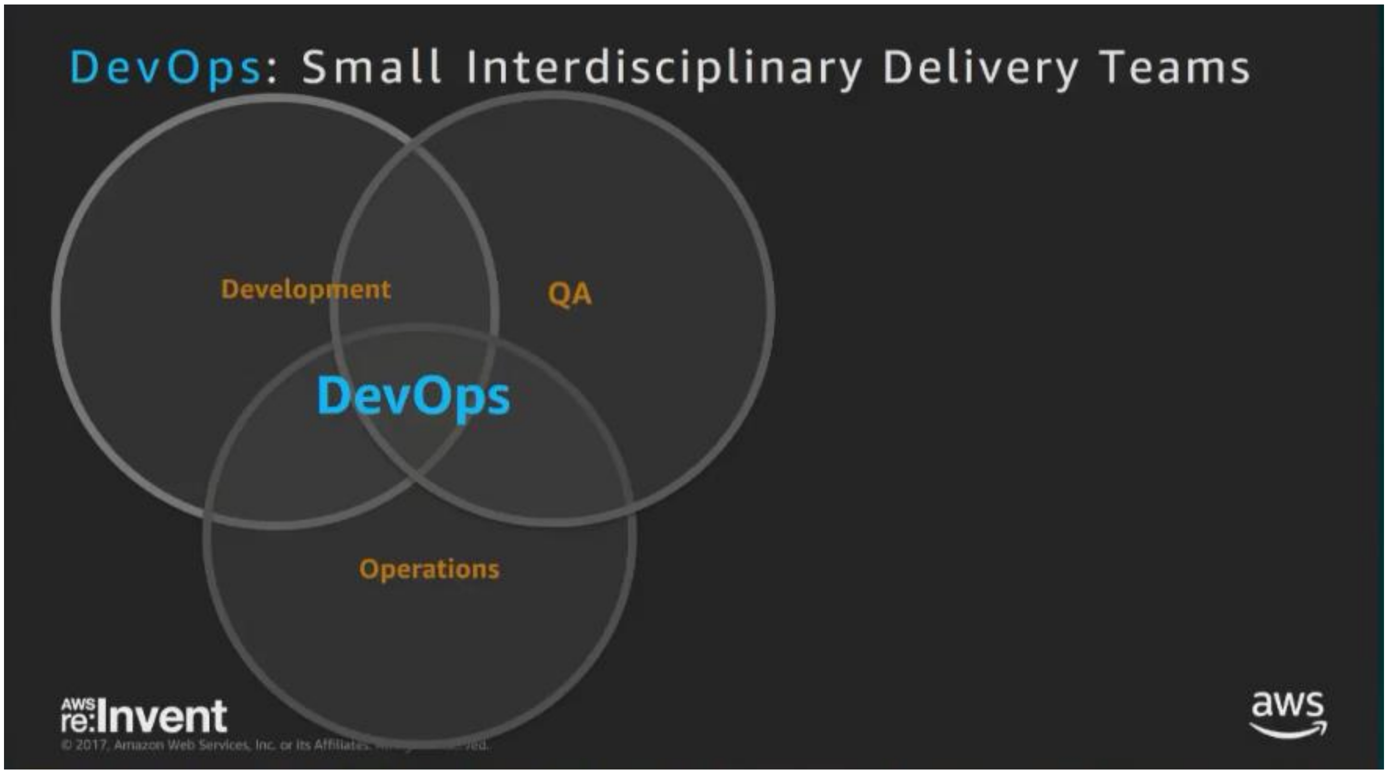
Anti-Pattern: Over-the-Wall Software Delivery

- Dev, QA, and ops kept separate
- Manual handoff processes
- CI/CD logistically blocked
- Tight controls and guardrails
- Post-deployment security checks
- Infrequent release cycles
- Infrequent patch rollouts



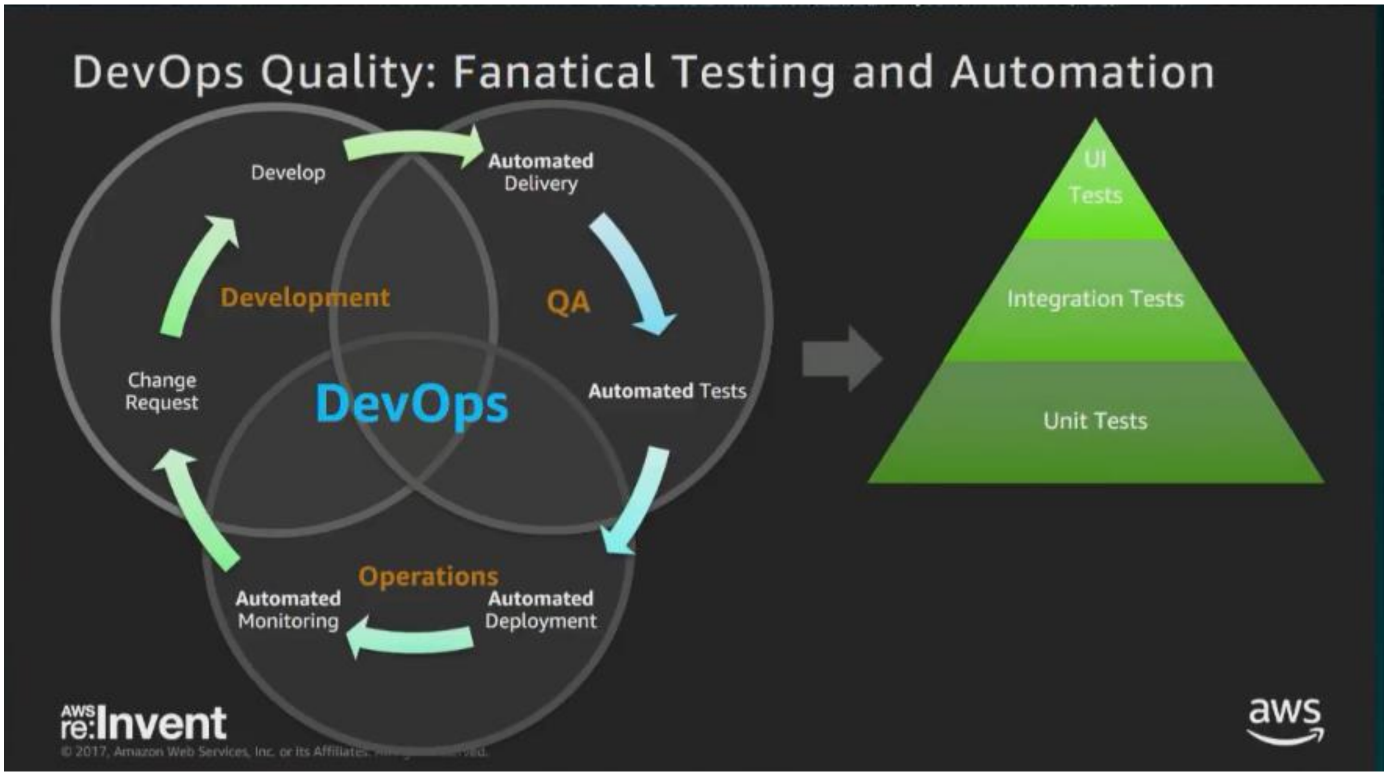
Traditional Delivery via Separate Functional Teams

AWS re:Invent
© 2017, Amazon Web Services, Inc. or its Affiliates. All rights reserved.
aws



DevOps: Small Interdisciplinary Delivery Teams

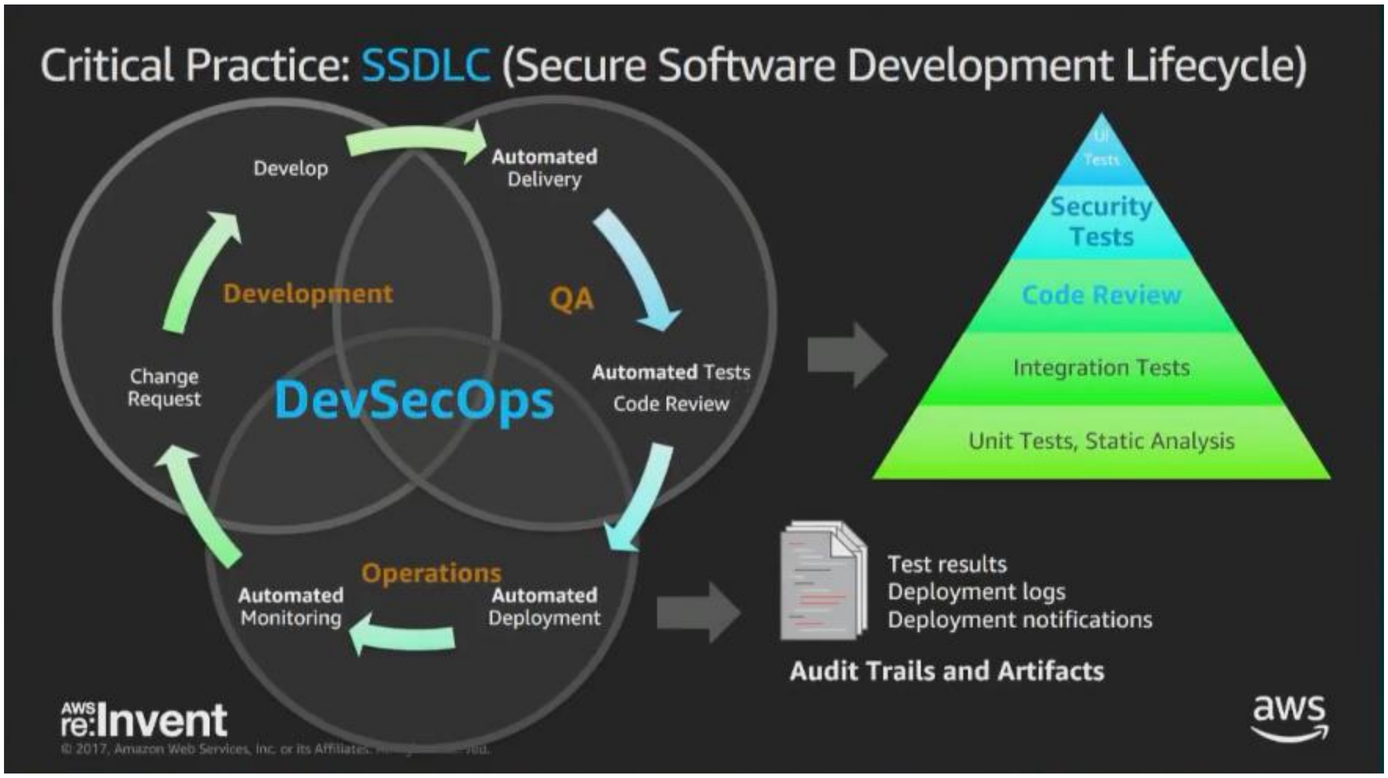
re:Invent
© 2017, Amazon Web Services, Inc. or its Affiliates.
aws



DevOps Quality: Fanatical Testing and Automation

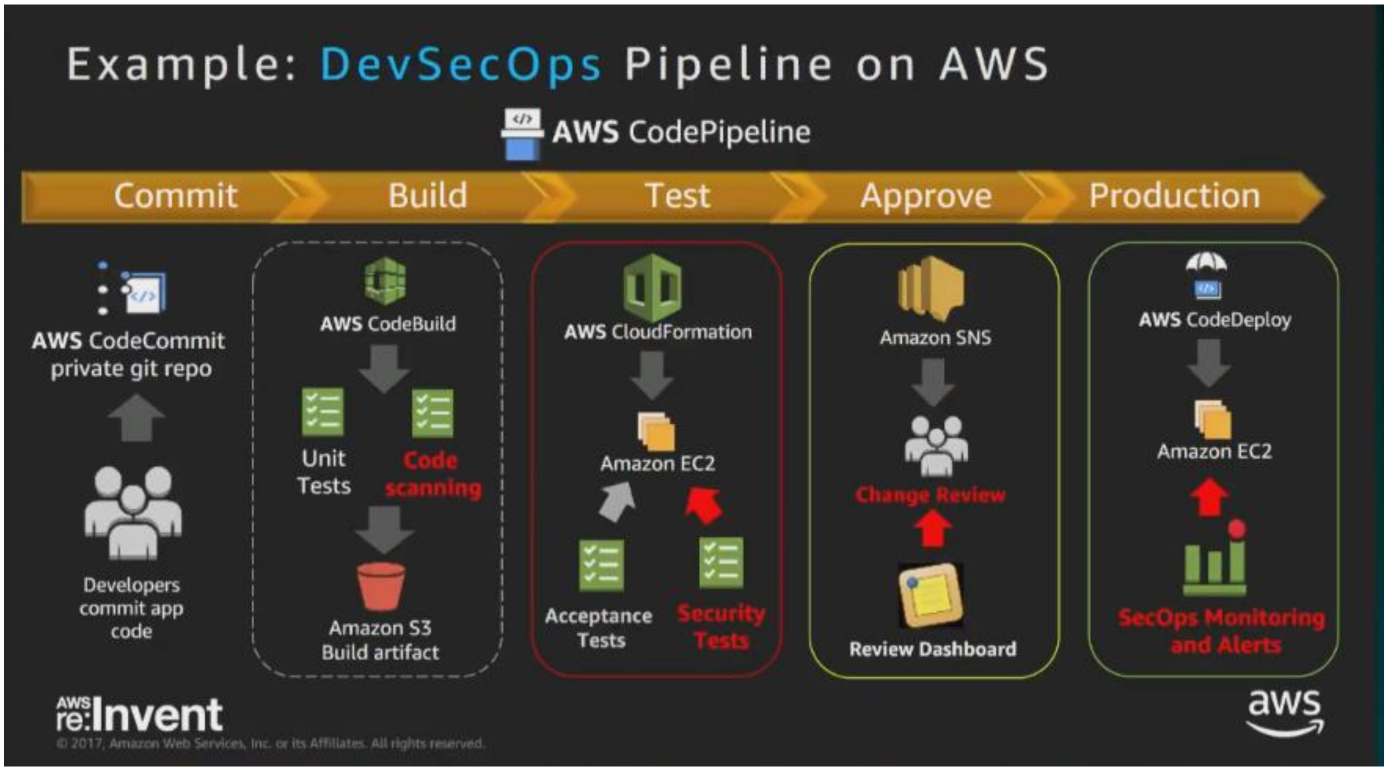
re:Invent
© 2017, Amazon Web Services, Inc. or its Affiliates.
aws

Critical Practice: SSDLC (Secure Software Development Lifecycle)



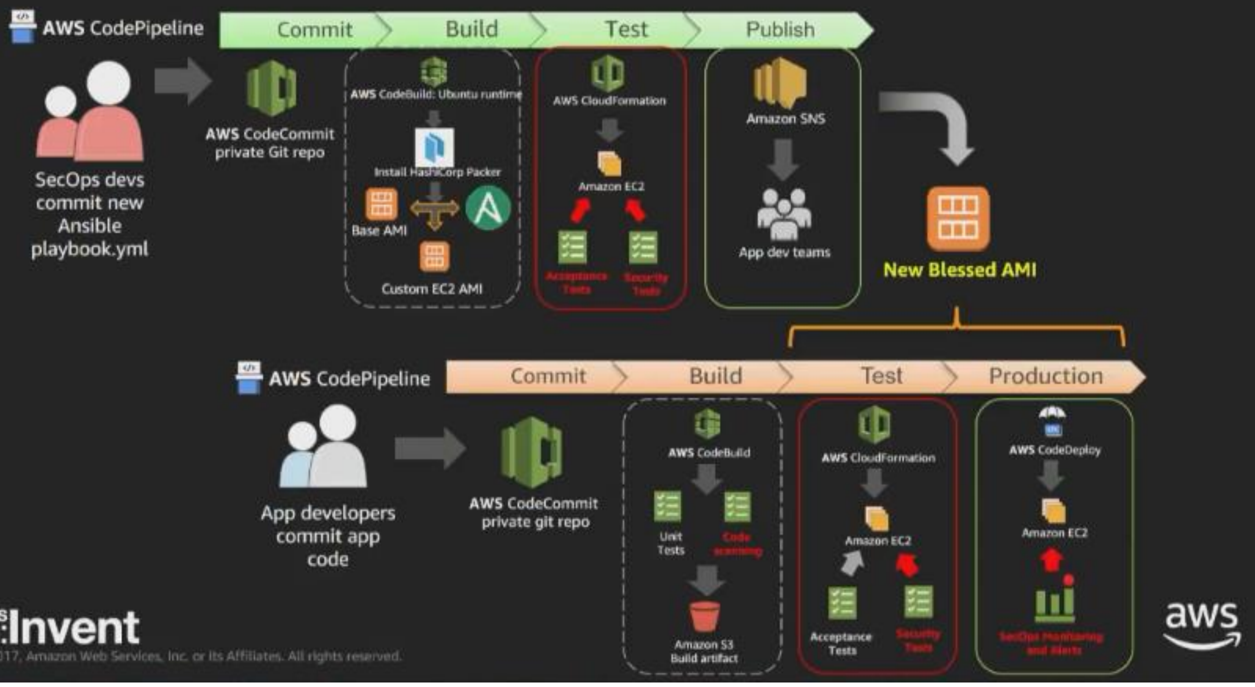
© 2017, Amazon Web Services, Inc. or its Affiliates.

Example: DevSecOps Pipeline on AWS



© 2017, Amazon Web Services, Inc. or its Affiliates. All rights reserved.

Example: Continuous and Routine OS Rehydration, Patching



AWS re:Invent

© 2017, Amazon Web Services, Inc. or its Affiliates. All rights reserved.

aws

DevSecOps at Fidelity Investments

Jonathan Baulch

Director of Architecture

Fidelity Investments

How Fidelity is Leveraging AWS

Fidelity Investments

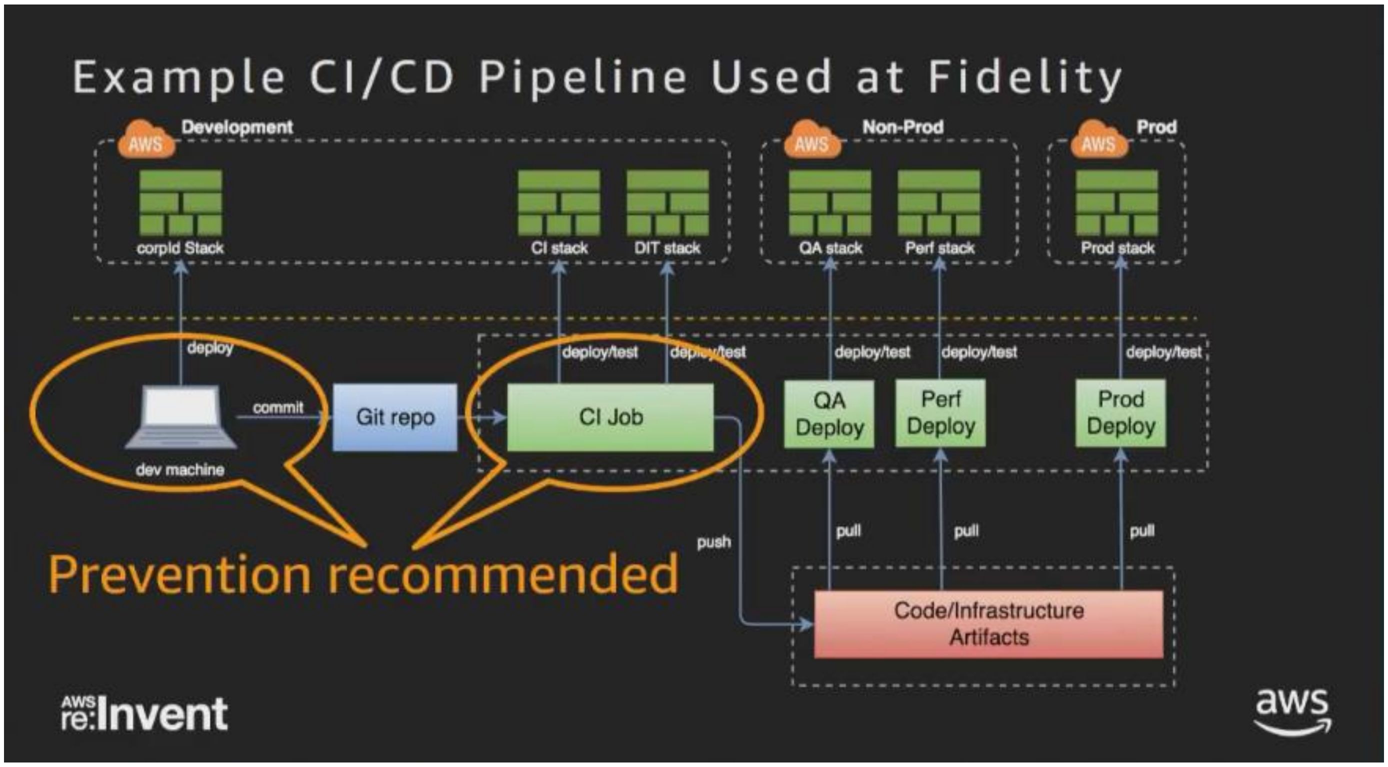
- Web application re-platform
- Big-data analytics
- Artificial Intelligence platform
- DevOps transformation

Layers of Security at Fidelity

Prevention

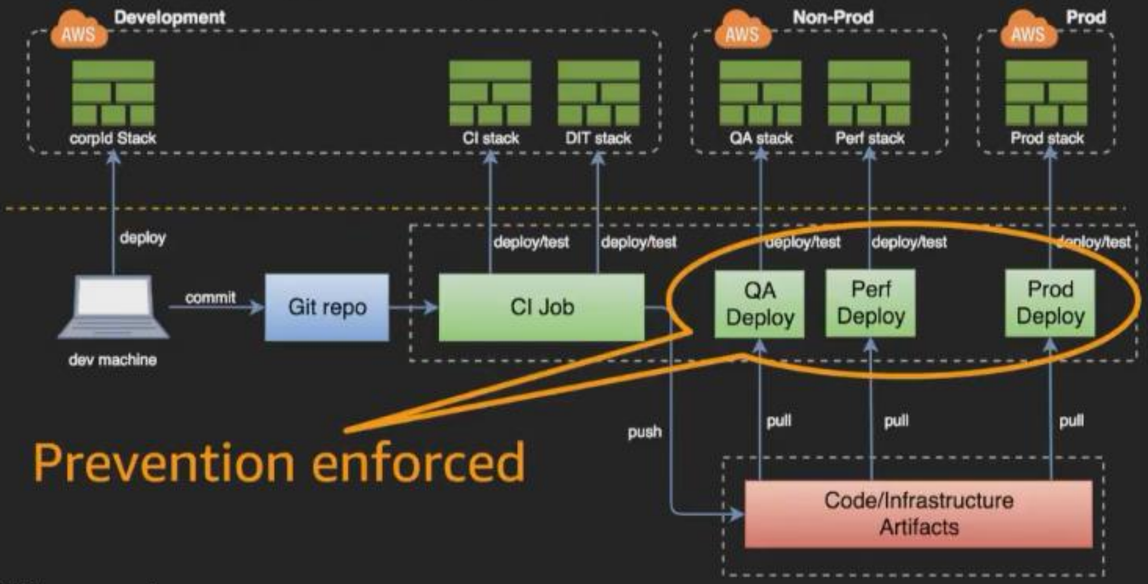
Detection

Remediation



Example CI/CD Pipeline Used at Fidelity

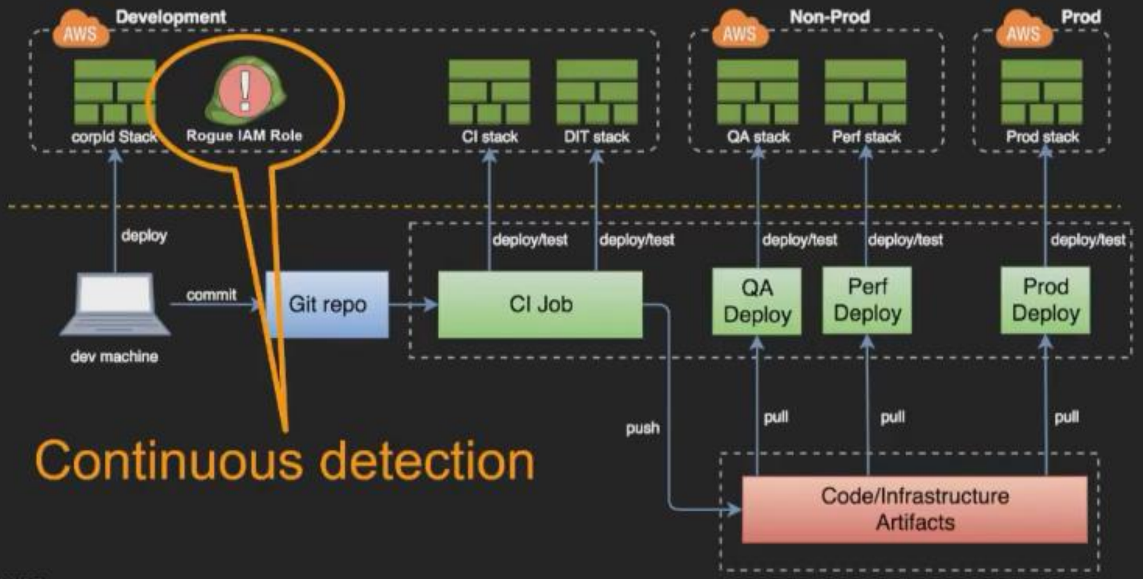
Example CI/CD Pipeline Used at Fidelity



re:Invent

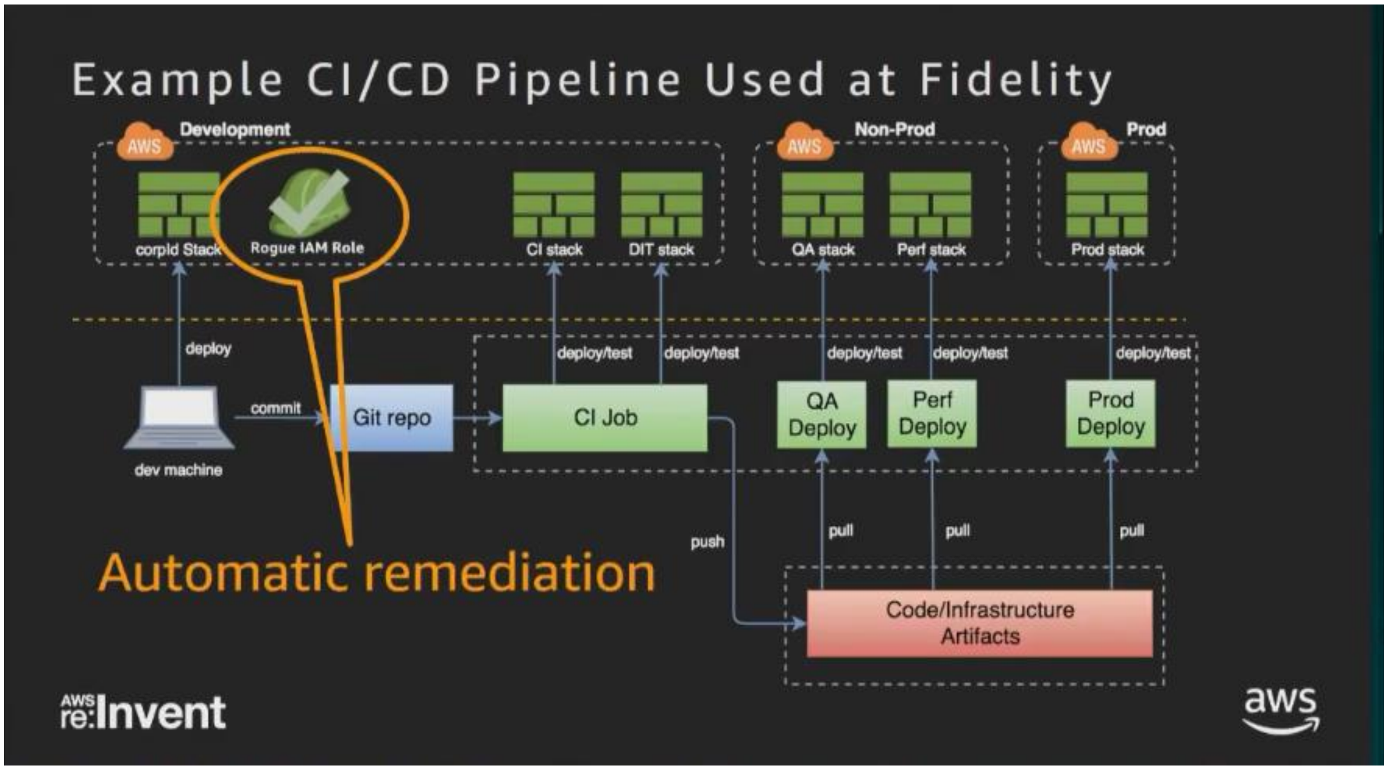
aws

Example CI/CD Pipeline Used at Fidelity

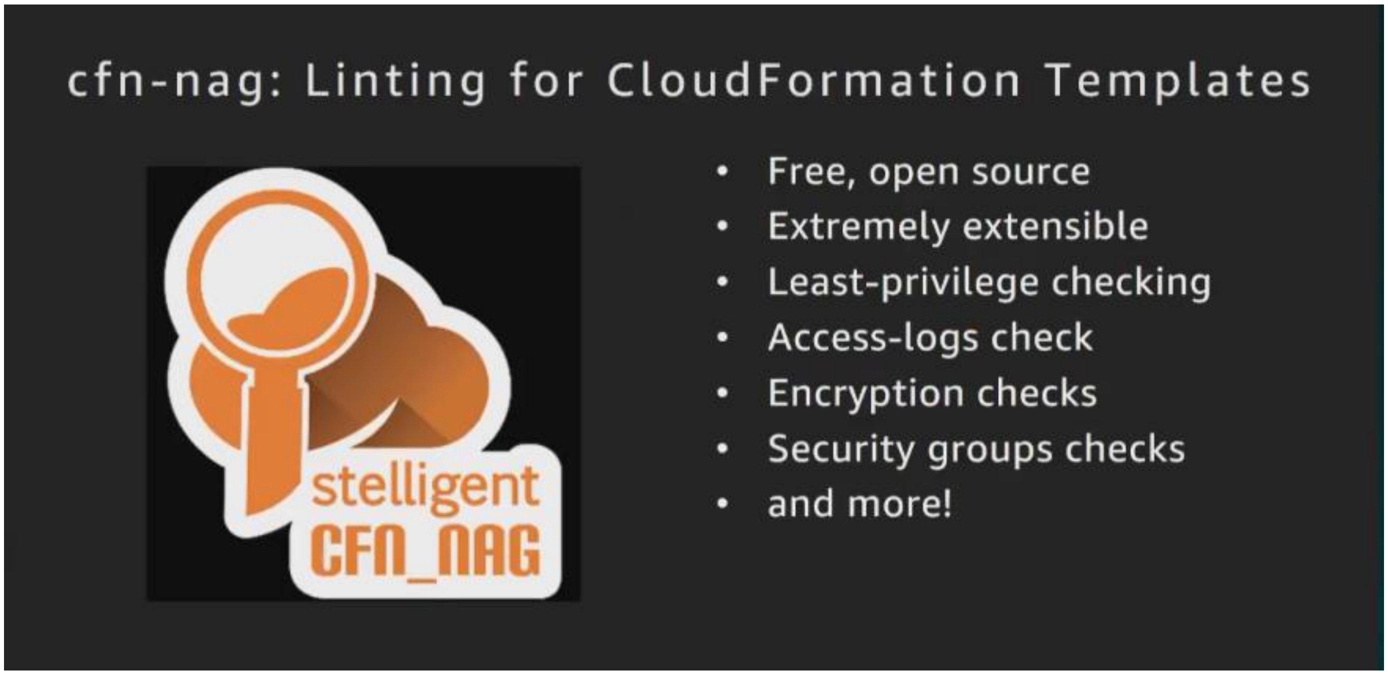


re:Invent

aws



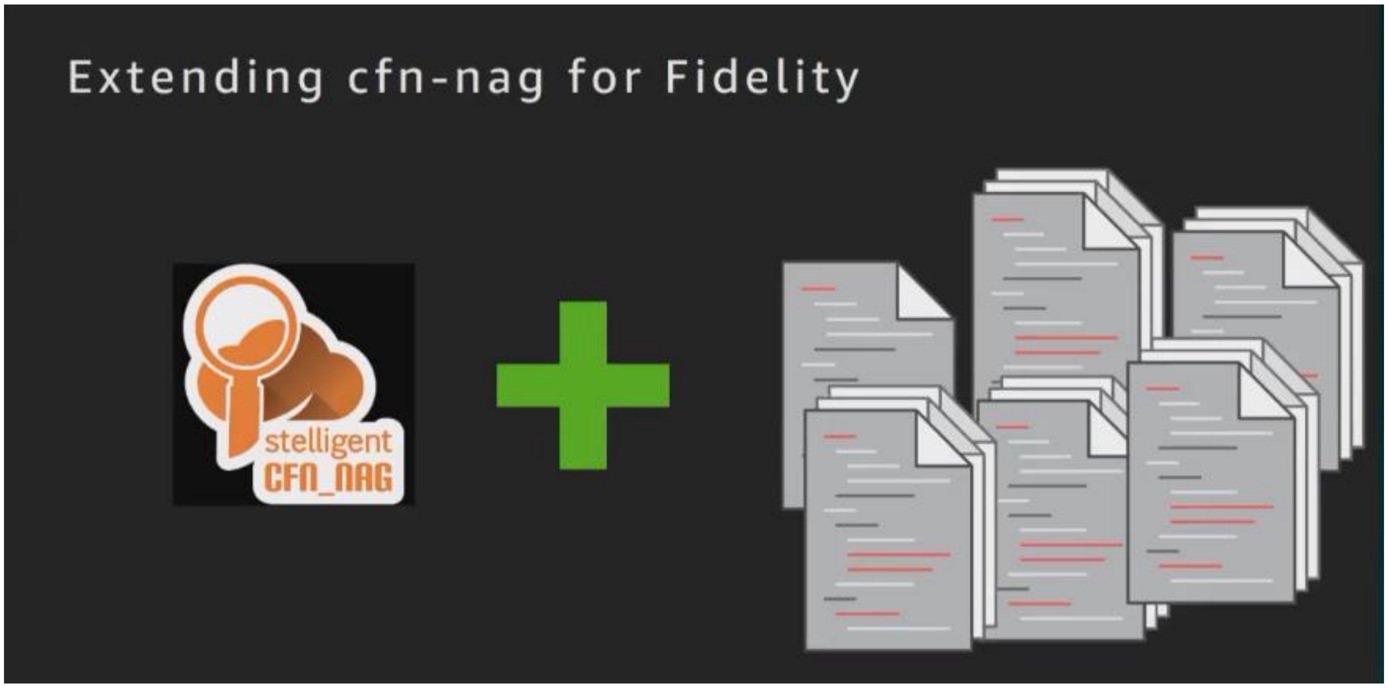
Example CI/CD Pipeline Used at Fidelity



cfn-nag: Linting for CloudFormation Templates

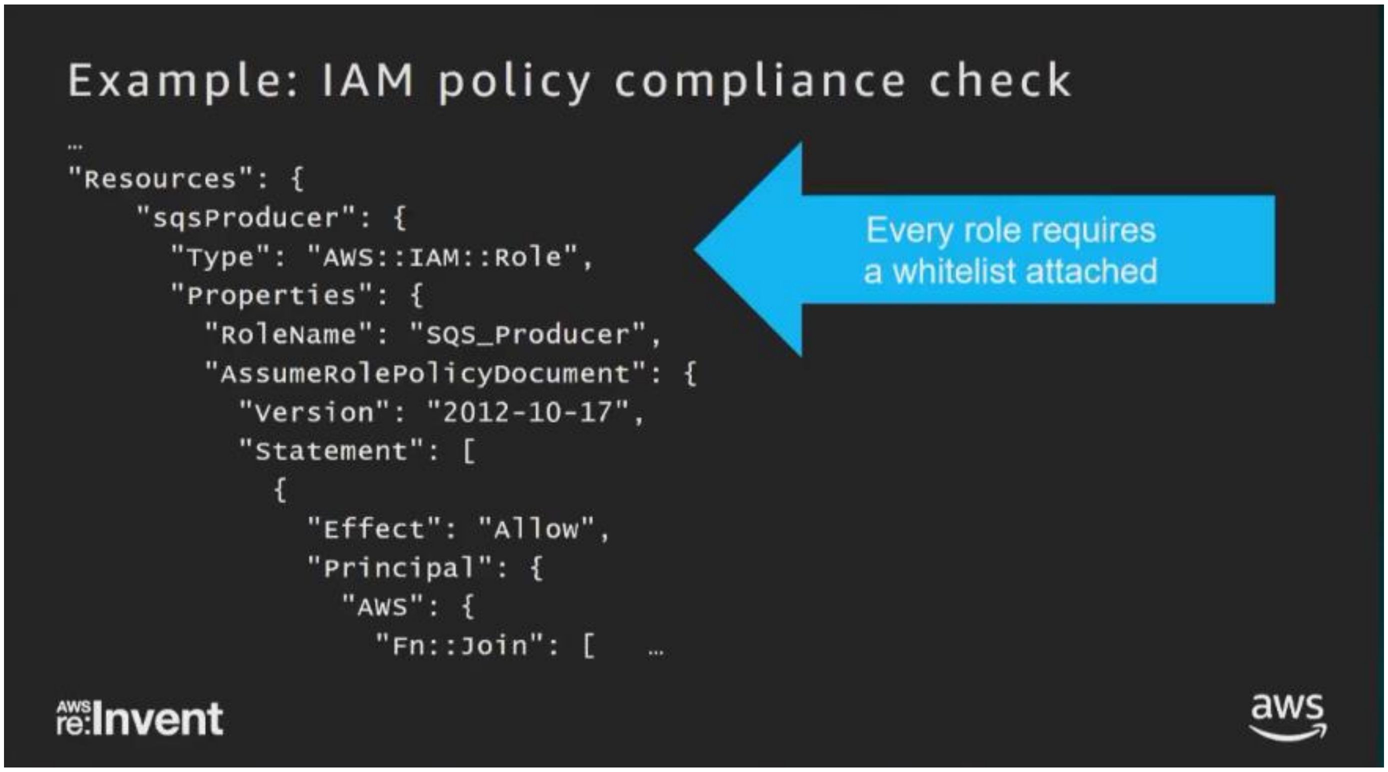
stelligent
CFN_NAG

- Free, open source
- Extremely extensible
- Least-privilege checking
- Access-logs check
- Encryption checks
- Security groups checks
- and more!



Extending cfn-nag for Fidelity

We created rules covering our AWS infrastructure to be used with cfn-nag rules.

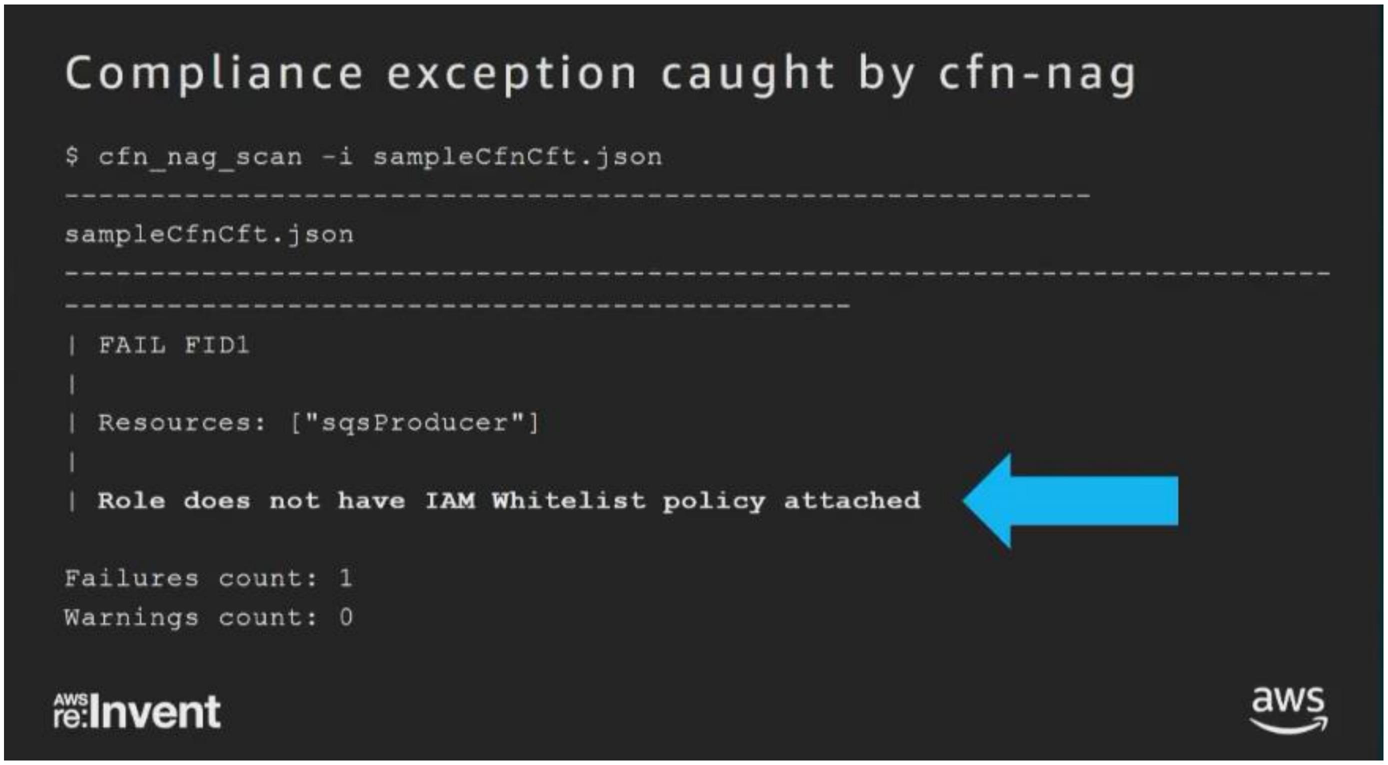


Example: IAM policy compliance check

```
…
"Resources": {
    "sqsProducer": {
      "Type": "AWS::IAM::Role",
      "Properties": {
        "RoleName": "SQS_Producer",
        "AssumeRolePolicyDocument": {
          "Version": "2012-10-17",
          "Statement": [
            {
              "Effect": "Allow",
              "Principal": {
                "AWS": {
                  "Fn::Join": [   …
```

Every role requires a whitelist attached

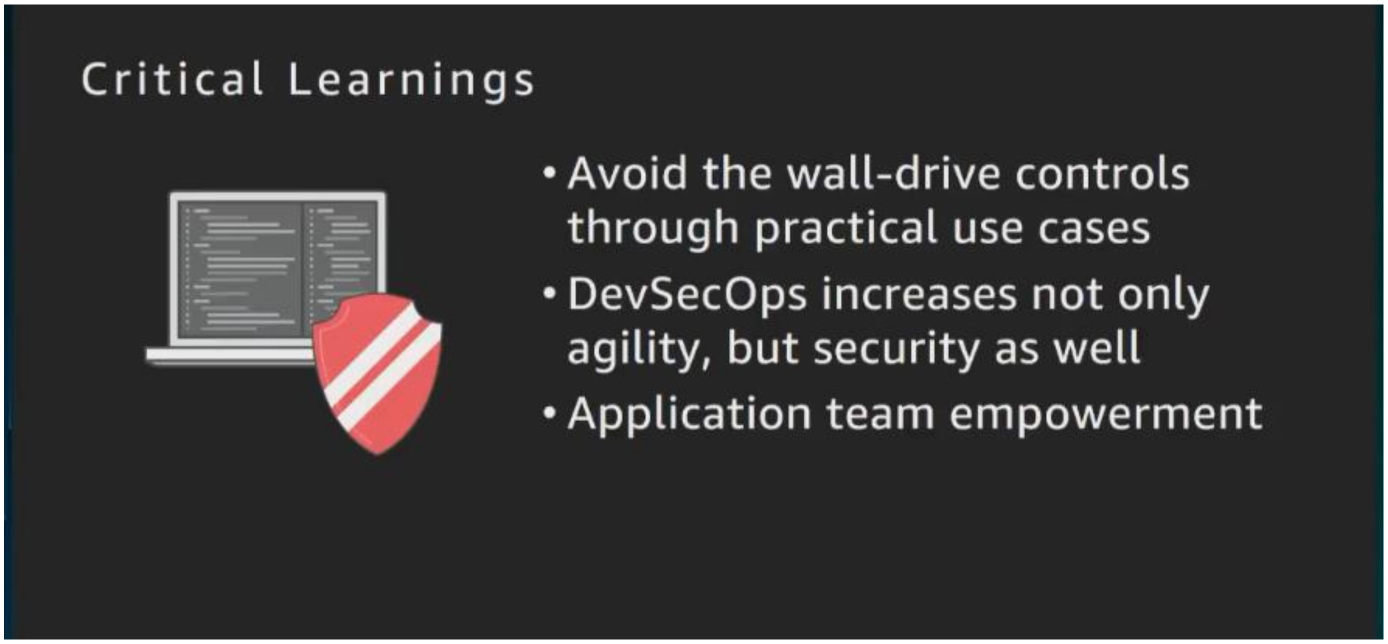
AWS re:Invent
aws



Compliance exception caught by cfn-nag

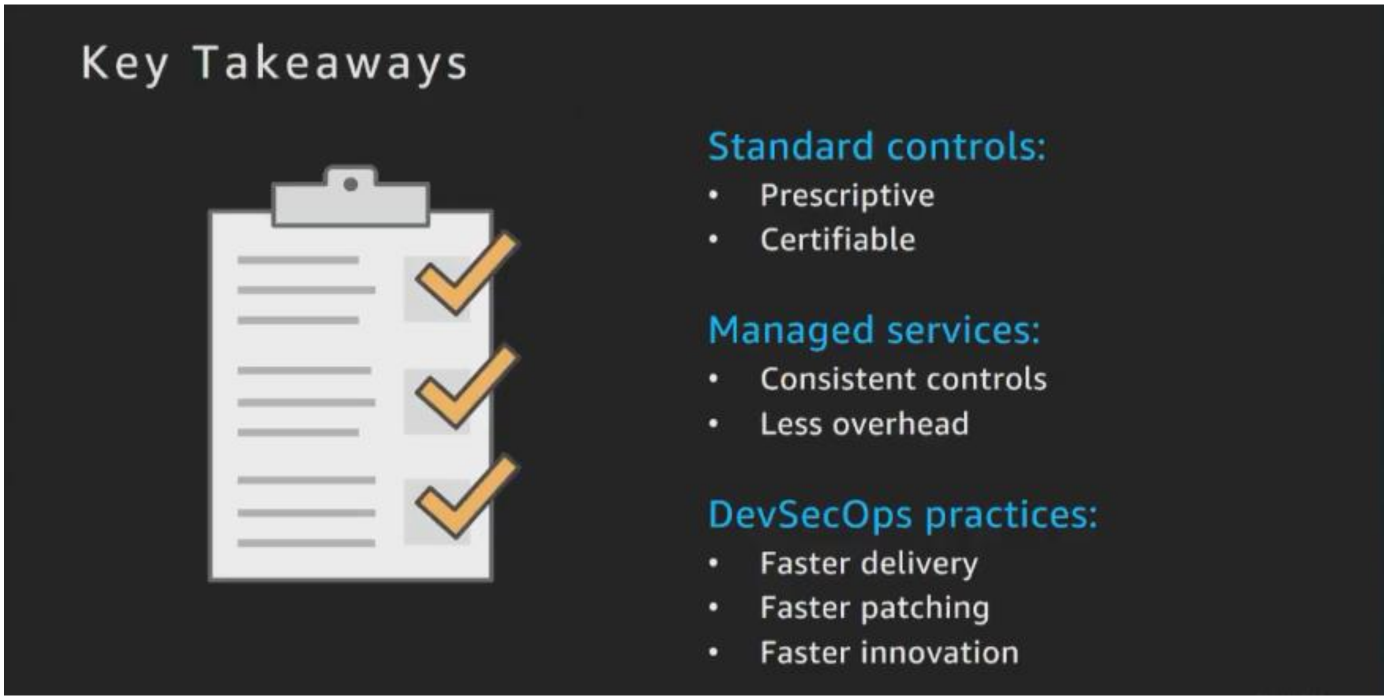
```
$ cfn_nag_scan -i sampleCfnCft.json
------------------------------------------------------------
sampleCfnCft.json
------------------------------------------------------------------------------
----------------------------------------------
| FAIL FID1
|
| Resources: ["sqsProducer"]
|
| Role does not have IAM Whitelist policy attached

Failures count: 1
Warnings count: 0
```

We execute cfn-nag on all of our CF templates before deployments



Critical Learnings

- Avoid the wall-drive controls through practical use cases
- DevSecOps increases not only agility, but security as well
- Application team empowerment



Key Takeaways

Standard controls:
- Prescriptive
- Certifiable

Managed services:
- Consistent controls
- Less overhead

DevSecOps practices:
- Faster delivery
- Faster patching
- Faster innovation

AWS re:Invent

Thank you!

Please submit your evaluations!

AWS re:Invent

© 2017, Amazon Web Services, Inc. or its Affiliates. All rights reserved.

aws