RSAConference2020

San Francisco | February 24 – 28 | Moscone Center

HUMAN ELEMENT

SESSION ID: **CSV-R01**

Clearing the Clouds: Incident Response in AWS (Isn't as Bad as You Thought)

Kristy Westphal

VP, CSIRT

Disclaimer

- The views, opinions, and material presented by Kristy Westphal at this conference are solely based on her experience and opinions related to incident response.
- The content of this presentation does not reflect the views or opinions of MUFG Union Bank.

Agenda

- Why am I Here?
- AWS Architecture 101
- Incident Response Use Cases
- Acquiring Amazon Web Services (AWS) Skills
- 90 Day Plan for AWS Incident Response

Why am I here?

- Information security leader specializing in security assessments, operational risk, and program development
- Security is painful all around; hopefully I can help
- Let's share knowledge and make it less painful for all of us!
- Props to Pete Ehlke for helping make this preso really come to life

Why AWS incident response is important…

What this session is…and isn't…

- Think about cloud incident response differently
  - But not as the impossible mountain to climb
- Yes, this is only Amazon Web Services
  - But the approach can be applied to other providers
- We won't be doing in-depth AWS training
  - But you will have resources to do this yourself

Poll the Audience

- CSV-R01
- **Are you doing security incident response in AWS now?**

A. Yes

B. No

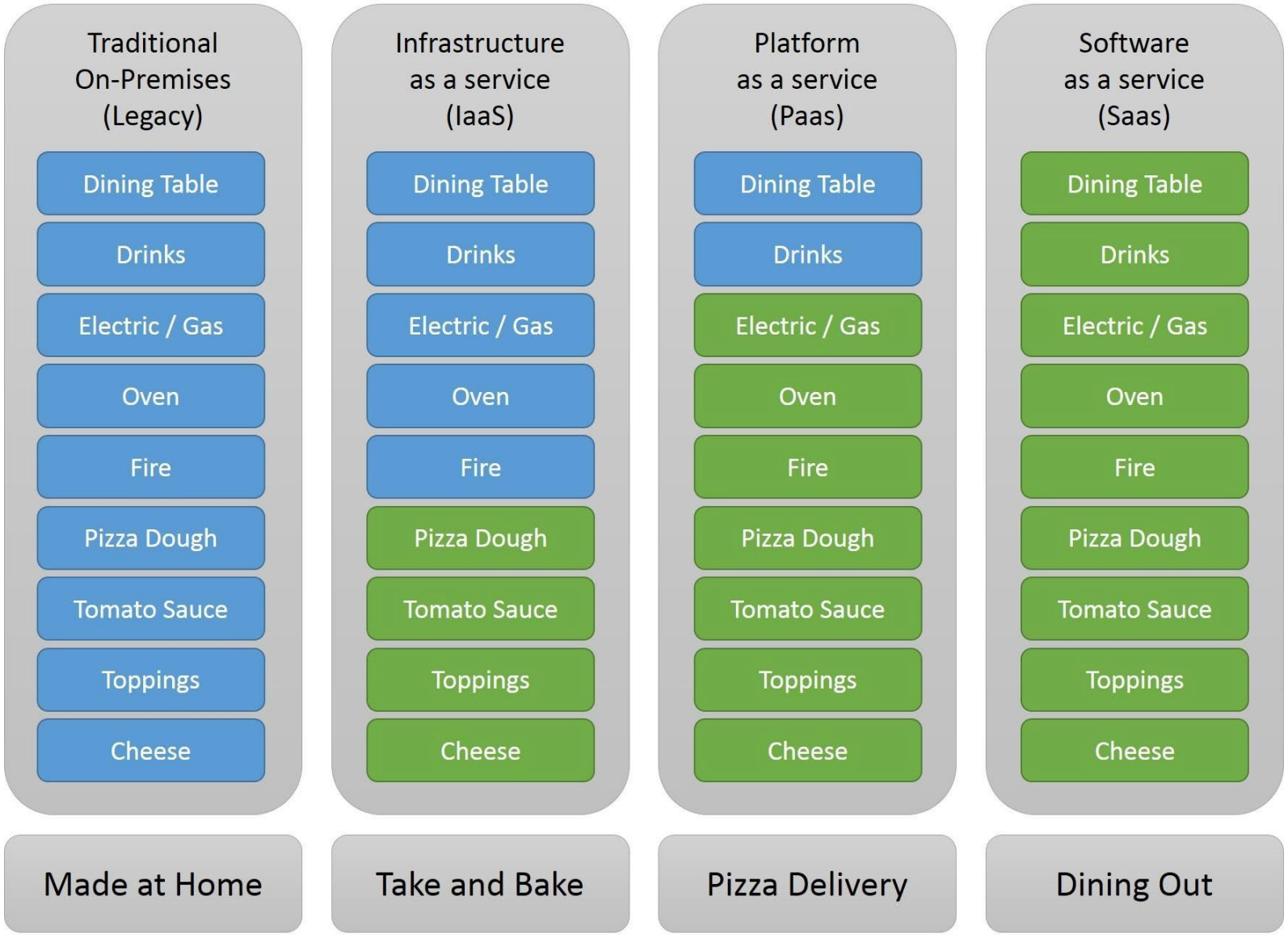
C. I Don't Know

https://rsa1-live.eventbase.com/polls?event=rsa2020&session=1454108104

RSAConference2020

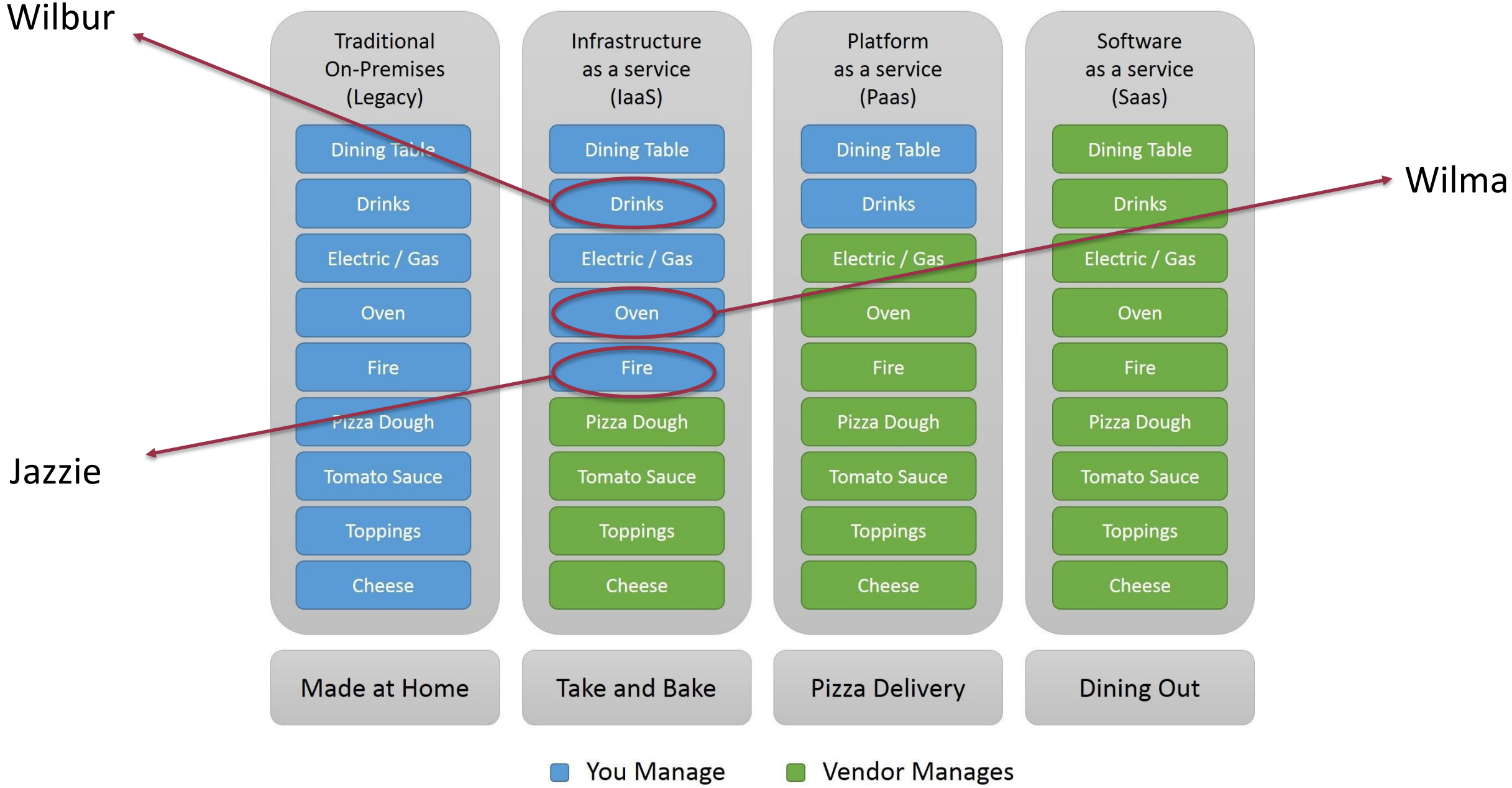
A Peek Into AWS Architecture

Pizza as a Service

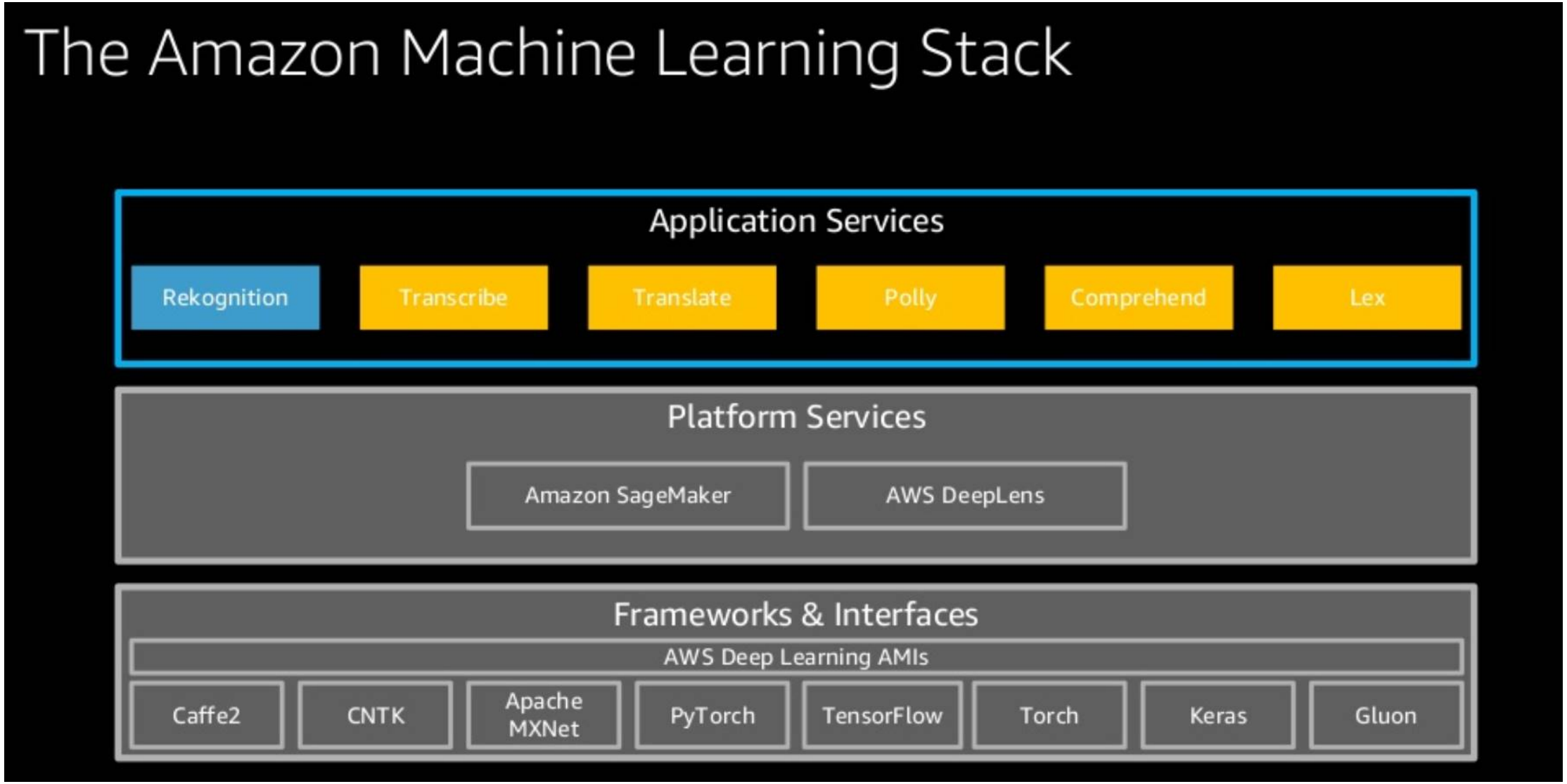


*Thank you Albert Barron for this example.

Pizza as a Service



In reality….


The Amazon Machine Learning Stack
Application Services
Rekognition
Transcribe
Translate
Polly
Comprehend
Lex
Platform Services
Amazon SageMaker
AWS DeepLens
Frameworks & Interfaces
AWS Deep Learning AMIs
Caffe2
CNTK
Apache MXNet
PyTorch
TensorFlow
Torch
Keras
Gluon

The five pillars of the AWS framework

- Operational Excellence
- Security
- Reliability
- Performance Efficiency
- Cost Optimization

https://d1.awsstatic.com/whitepapers/architecture/AWS_Well-Architected_Framework.pdf?did=wp_card&trk=wp_card

Apps Agile stuff Imported Imported (1)

aws Services Resource Groups

History

Console Home

Find a service by name or feature (for example, EC2, S3 or VM, storage).

Group A-Z

Compute
- EC2
- Lightsail
- ECR
- ECS
- EKS
- Lambda
- Batch
- Elastic Beanstalk
- Serverless Application Repository

Storage
- S3
- EFS
- FSx
- S3 Glacier
- Storage Gateway
- AWS Backup

Database
- RDS
- DynamoDB
- ElastiCache
- Neptune
- Amazon Redshift
- Amazon QLDB
- Amazon DocumentDB

Customer Enablement
- AWS IQ
- Support
- Managed Services

Blockchain
- Amazon Managed Blockchain

Satellite
- Ground Station

Management & Governance
- AWS Organizations
- CloudWatch
- AWS Auto Scaling
- CloudFormation
- CloudTrail
- Config
- OpsWorks
- Service Catalog
- Systems Manager
- Trusted Advisor
- Control Tower
- AWS License Manager
- AWS Well-Architected Tool
- Personal Health Dashboard

Analytics
- Athena
- EMR
- CloudSearch
- Elasticsearch Service
- Kinesis
- QuickSight
- Data Pipeline
- AWS Data Exchange
- AWS Glue
- AWS Lake Formation
- MSK

Security, Identity, & Compliance
- IAM
- Resource Access Manager
- Cognito
- Secrets Manager
- GuardDuty
- Inspector
- Amazon Macie
- AWS Single Sign-On
- Certificate Manager
- Key Management Service
- CloudHSM
- Directory Service
- WAF & Shield
- Artifact

Business Applications
- Alexa for Business
- Amazon Chime
- WorkMail

End User Computing
- WorkSpaces
- AppStream 2.0
- WorkDocs
- WorkLink

Internet Of Things
- IoT Core
- Amazon FreeRTOS
- IoT 1-Click
- IoT Analytics
- IoT Device Defender
- IoT Device Management
- IoT Events
- IoT Greengrass
- IoT SiteWise
- IoT Things Graph

Game Development
- Amazon GameLift

close

Where to focus IR efforts?

- Host
- IAM (a.k.a. role-based access)
- Data storage (e.g., S3)
- Persistence (e.g., when odd things change)
  - S3 bucket permissions
  - Security groups
  - Network gateways
  - EC2 instance ownership
  - Authorization failures

Breaking it down

Bad guy

Leverages

IAM

Attacks

EC2

S3

We detect via

CloudWatch

CloudTrail

Poll the Audience

- CSV-R01
- **What of the following do you see as roadblock to AWS Incident Response?**

A. Not enough authority

B. Knowledge of landscape

C. Lack of skills

https://rsa1-live.eventbase.com/polls?event=rsa2020&session=1454108104

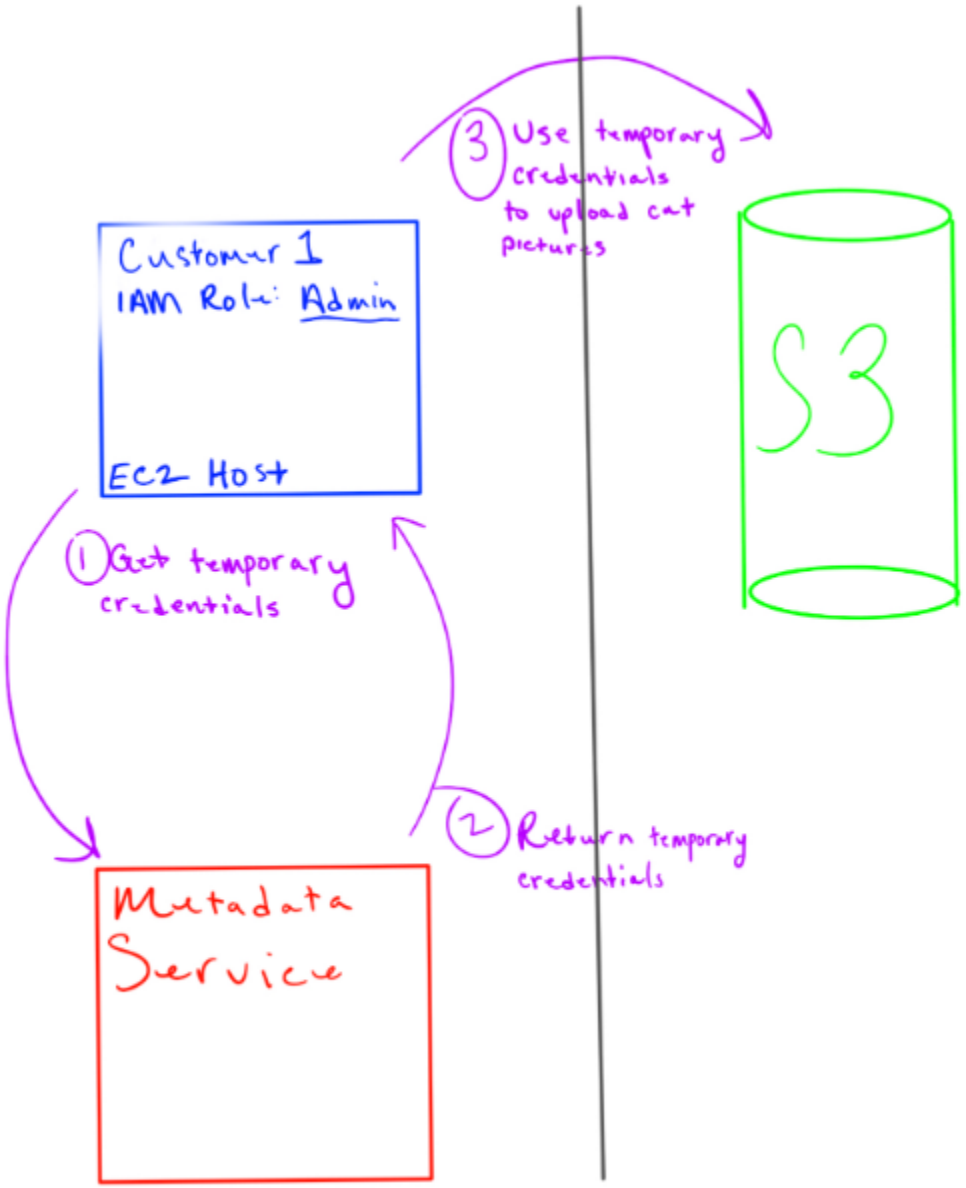
RSAConference2020

AWS Incident Response Use Cases

Use Case 1: What really happened at Capital One?

- An old vulnerability called Server Side Request Forgery (SSRF)
  - On the Web Application Firewall (WAF) - ModSecurity
- The WAF was misconfigured
  - Able to make a metadata service request
  - Which is how the attacker gained access to credentials
  - Credentials had access to whatever resource requested them
  - WAF assigned too much privilege
  - Could list information contained within S3 storage buckets
- AWS has since added additional authentication to the service

A visual


Customer 1
IAM Role: Admin
EC2 Host
3 Use temporary credentials to upload cat pictures
S3
1 Get temporary credentials
2 Return temporary credentials
Metadata Service

Why should you care?

- Cloud misconfigurations can have greater impact if exploited
  - Versus on premises misconfigurations
- A bad actor much more likely to access if internet-facing

Use Case 2: Oh no, not more!

- Ever heard of Code Spaces?
  - Maybe not since it's been dead since 2014
  - It was a site that hosted source code repositories and offered project management services
  - Mostly hosted on AWS
- An attacker gained access to their AWS console
  - Held it for ransom
  - When no payment, started deleting…everything
  - Elastic Block Storage (EBS) Snapshots, S3 data, some server instances

All eggs in one basket

- All the data that was deleted included backups
- Effectively put Code Spaces out of business

So how do I respond to that?

- Follow the breadcrumbs
  - Read the logs (Cloud Trail, Cloud Watch)
  - Understand the flow
  - Understand what your scope of response is
    - An internal shared responsibility model?
  - Understand how things are configured
    - And did they behave as expected?

amazon web services

Table Of Contents

Quick search

Search

Feedback

Did you find this page useful?
Do you have a suggestion?
Give us feedback or send us a pull request on GitHub.

User Guide

aws

Description

The AWS Command Line Interface is a unified tool to manage your AWS

Synopsis

```
aws [options] <command> <subcommand> [parameters]
```

Use *aws command help* for information on a specific command. Use *aws* available help topics. The synopsis for each command shows its parame parameters are shown in square brackets.

Options

`--debug` (boolean)

Turn on debug logging.

Available Services

- accessanalyzer
- acm
- acm-pca
- alexaforbusiness
- amplify
- apigateway
- apigatewaymanagementapi
- apigatewayv2
- appconfig
- application-autoscaling
- application-insights
- appmesh
- appstream
- appsync
- athena
- autoscaling
- autoscaling-plans
- backup
- batch
- budgets
- ce
- chime
- cloud9
- clouddirectory
- cloudformation
- cloudfront

Command line interface tips

- You'll have to install the environment
  - Looks like DOS! (seriously!)
- Run aws-configure
  - Consider – debug to get all the interactive detail
  - Do you have a proxy? May need to pass through/tunnel.
  - If you do tunnel, may need to import certs
    - set REQUESTS_CA_BUNDLE=full-path-to\[name of].pem
- Potential for automation

Some examples

- Want to copy a snapshot?

aws ec2 copy-snapshot \

- region us-east-1 \

- source-region us-west-2 \

- source-snapshot-id snap-06687767178 9bd71b \

- description "This is my copied snapshot."

Select Command Prompt - aws help topics

```
AWS CLI Topic Guide
^^^^^^^^^^^^^^^^^^^


Description
***********

This is the AWS CLI Topic Guide. It gives access to a set of topics
that provide a deeper understanding of the CLI. To access the list of
topics from the command line, run "aws help topics". To access a
specific topic from the command line, run "aws help [topicname]",
where "topicname" is the name of the topic as it appears in the output
from "aws help topics".


Available Topics
****************


General
=======

* config-vars: Configuration Variables for the AWS CLI

* return-codes: Describes the various return codes of the AWS CLI


S3
-- More --
```

Another example

- Change a security group?

```
aws ec2 authorize-security-group-ingress \\
    - group-name MySecurityGroup \
    - protocol tcp \
    - port 22 \
    - cidr 203.0.113.0/24
```

Other options

- Crash cart
  - What tools might you want to have available?
    - Depends upon incident response scope
    - Remnux for malware analysis
    - Other gems?
- Breakglass
  - May be more palatable for Incident Response to have emergency only access
- Lambda
  - What can you automate through scripting?
- Who contacts Amazon in case of an incident?

github.com/awslabs?utf8=✓&q=security&type=&language=

Agile stuff | Imported | Imported (1)

Why GitHub? | Enterprise | Explore | Marketplace | Pricing | Search / | Sign in | Sign up

Amazon Web Services - Labs

AWS Labs

Seattle, WA | http://amazon.com/aws/ | Verified

Repositories 448 | Packages 1 | People 109 | Projects

security | Type: All | Language: All

11 results for repositories matching **security**

Clear filter

aws-security-benchmark

Open source demos, concept and guidance related to the AWS CIS Foundation framework.

Python | 209 | 501 | 28 | 18 | Updated on Sep 25

aws-waf-security-automations

This solution automatically deploys a single web access control list (web ACL) with a set of AWS WAF rules designed to filter common web-based attacks.

Top languages

Python | Java | JavaScript | Go | Shell

Most used topics

aws | serverless | deep-learning | machine-learning | lambda

github.com/ThreatResponse

Agile stuff | Imported | Imported (1)

Why GitHub? | Enterprise | Explore | Marketplace | Pricing | Search / | Sign in | Sign up

ThreatResponse

A Free Open Source Security Suite for Hardening and Responding in AWS

Ashland, OR | http://www.threatresponse.cloud | jo[illegible]@threatresponse.cloud

Repositories 39 | Packages | People 5 | Projects

Find a repository... | Type: All | Language: All

margaritashotgun

Remote Memory Acquisition Tool

Python | MIT | 25 | 133 | 11 (1 issue needs help) | 2 | Updated on Oct 22

aws_ir

Python installable command line utiltity for mitigation of host and key compromises.

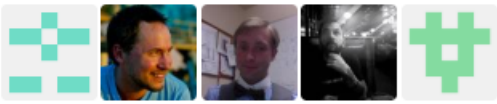
Python | MIT | 43 | 214 | 12 (3 issues need help) | 1 | Updated on Feb 28

Top languages

Python | CSS | Shell | JavaScript | HTML

People

5 >



Use Case 3: Let's Run Through An Incident End To End

- Your Macie service has identified odd behavior via a key pair
  - Key pair is accessing an account it's never logged into before
  - Is this a security incident?
- Work through four stages of an Incident:
  - Prepare
  - Detect
  - Contain
  - Post-Incident

Run It Down…

- Detect
  - How would you determine what IAM User the access keys belonged to?
    - How were the access keys used?
  - Which log would access to keys show up in? (access, use)
    - What would the log tell you?
    - How do you sift through all those events?
  - Would you have an alert any where when they were accessed?
    - Depends on the action
  - What regions were the keys used in?
    - Were they limited to specific regions?

A quick demo…

```
kmw@rsa-demo:~/src/rsa-demo (master.)$
```

kmw | kmw | 2:1 | 1:zsh- | 2:demo* | 19:21:24 | 18-Feb-20 | rsa-demo

Wrapping Up….

- Contain
  - What actions need to be taken to mitigate?
  - Who has the permission to do it?
    - Incident Response Team?
    - IAM Team?
  - Who needs to be notified?
    - Application Team?
    - End User?
- Post-Incident
  - How did the credentials get posted?
    - Was it posted from within your network?
    - Code for the demo available here: https://github.com/kameenan/RSAC2020

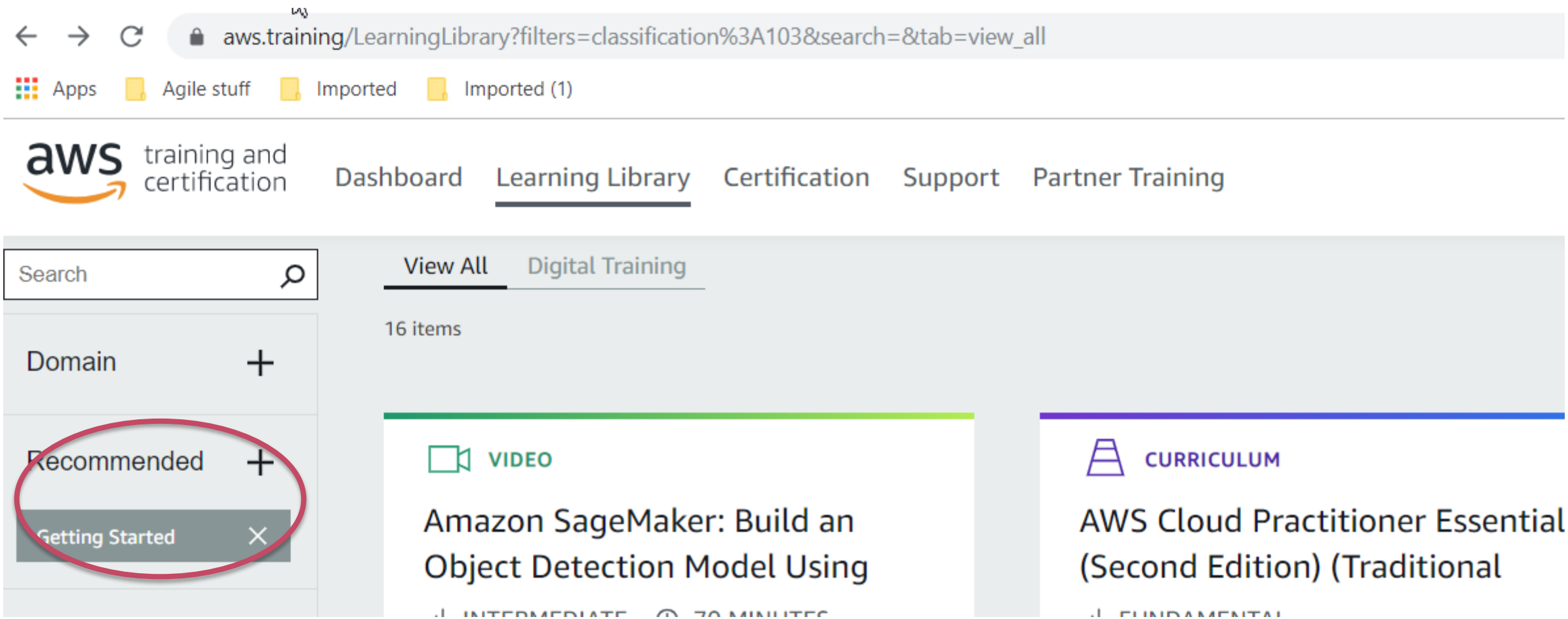
RSAConference2020

Acquiring AWS Skills

Start of available resources

- https://d1.awsstatic.com/whitepapers/aws_security_incident_response.pdf
- https://d1.awsstatic.com/whitepapers/Security/AWS_Security_Best_Practices.pdf
- https://d0.awsstatic.com/whitepapers/AWS_CAF_Security_Perspective.pdf
- https://www.aws.training/ (need to set up an account)
- https://aws.amazon.com/free (get your own hands on!)
- https://www.coursera.org/ (search for AWS [topic])
- https://github.com/jlevy/og-aws (awesome find)
- https://aws.amazon.com/

AWS training



E-LEARNING

AWS Security Fundamentals (Second Edition)

FREE COURSE

You must subscribe to Free Digital Training to take this course.

SUBSCRIBE

FUNDAMENTAL 2 HOURS ENGLISH

ABOUT

Description

In this self-paced course, you will learn fundamental AWS cloud security concepts, including AWS access control, data encryption methods, and how network access to your AWS infrastructure can be secured. We will address and your security responsibility in the AWS cloud and the different security-oriented services available.

Intended Audience

This course is intended for:

- IT business-level professionals interested in cloud security practices

Youtube. OMG.

- Go there
- Search for Amazon Web Service
- Find the very latest from re:Invent
- Learn a ton of stuff!

asecure.cloud

Agile stuff | Imported | Imported (1)

beta

Home | What's New | FAQ | No Items in Stack | Browse

Get Started: Find Configuration

Search All Content | Search

OR

Browse by

Security Control

- Config Rules
- Auto Remediation
- Amazon GuardDuty
- Amazon Inspector
- Security Hub
- Billing and Cost
- S3 Bucket Policies
- CloudWatch Alarms & Rules
- Logging & Monitoring
- Service Control Policies
- AWS Systems Manager
- Security Groups & NACLs
- KMS
- IAM Policies

Configuration Packages

- Deploy Amazon VPC
- Enable Logging Services
- Threat Detection
- Monitoring & Compliance
- Auto Remediation Rules
- EC2 Patch Management
- Common SCPs Package
- CIS AWS Benchmark

Service

- VPC
- S3
- EC2
- IAM
- CloudFormation
- Lambda
- EMR
- DynamoDB
- RDS

Security Strategy Guides

- EC2 Security Strategy
- S3 Security Strategy

Solutions, Guides & Tools

- Security Solutions
- Security Tools

Apply – 90-day AWS incident response plan

- 30 days:
  - Identify gaps between existing plan and what we've discussed
  - Begin acquiring needed skills
- 60 days:
  - Confirm architecture of AWS (or other cloud provider) environment
  - Acquire sandbox environment
  - Document processes to support
- 90 days:
  - Consider automation for certain tasks
  - Test out processes
    - Table tops!

RSAConference2020

Thank you!

Keep the conversation going!

kmwestphal@cox.net