You make **possible**

The Past, Present, and Future of Cloud

Pete Johnson, Principal Architect, Cisco Systems
@nerdguru

BRKCLD-2808

Barcelona | January 27-31, 2020

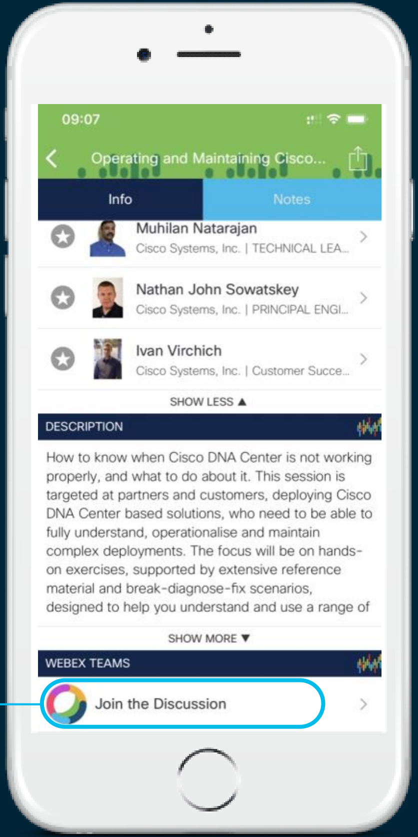
Cisco Webex Teams

Questions?

Use Cisco Webex Teams to chat
with the speaker after the session

How

1. Find this session in the Cisco Events Mobile App
2. Click “Join the Discussion”
3. Install Webex Teams or go directly to the team space
4. Enter messages/questions in the team space



09:07

Operating and Maintaining Cisco...

Muhilan Natarajan
Cisco Systems, Inc. | TECHNICAL LEA...

Nathan John Sowatskey
Cisco Systems, Inc. | PRINCIPAL ENGI...

Ivan Virchich
Cisco Systems, Inc. | Customer Succe...

DESCRIPTION

How to know when Cisco DNA Center is not working properly, and what to do about it. This session is targeted at partners and customers, deploying Cisco DNA Center based solutions, who need to be able to fully understand, operationalise and maintain complex deployments. The focus will be on hands-on exercises, supported by extensive reference material and break-diagnose-fix scenarios, designed to help you understand and use a range of

WEBEX TEAMS

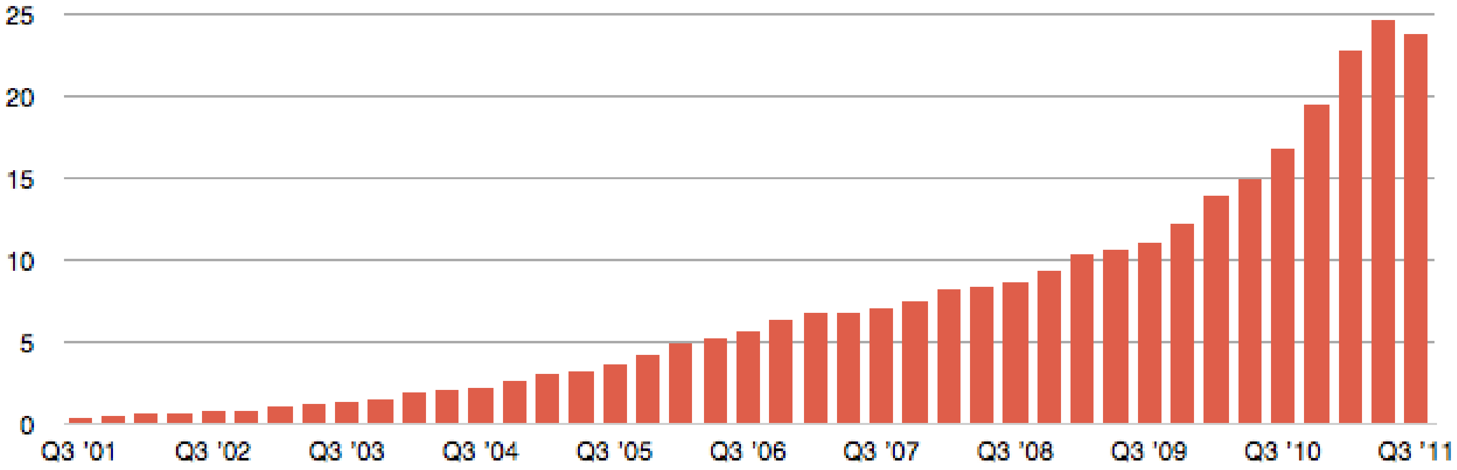
Agenda

- The Past
  - What has driven cloud adoption?
- The Present
  - AWS Shared Responsibility Model
  - Microservices and K8s
- The Future
  - Serverless
  - Serverless in the Datacenter
  - No Code
  - Edge Clusters

The Past

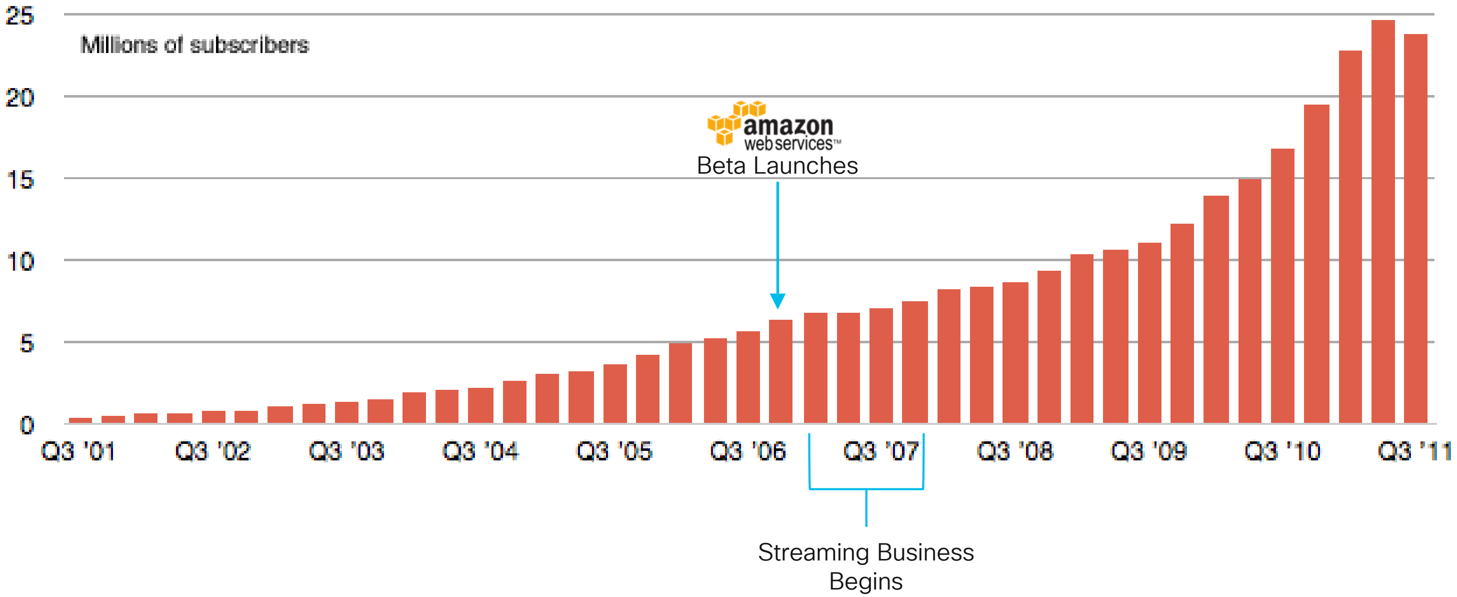
What has driven cloud adoption?

What’s This?


25
20
15
10
5
0
Q3 '01
Q3 '02
Q3 '03
Q3 '04
Q3 '05
Q3 '06
Q3 '07
Q3 '08
Q3 '09
Q3 '10
Q3 '11

First 10 Years Subscriber Growth

NETFLIX

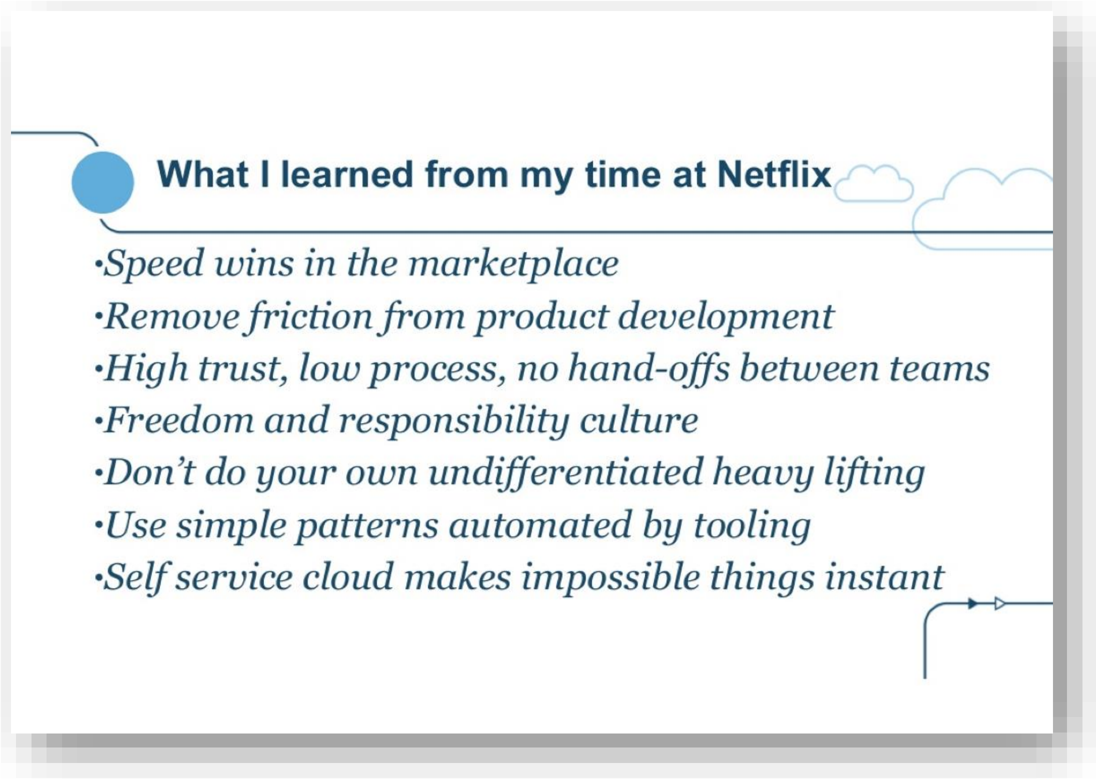


http://www.fromedome.com/2011/10/netflix-10-years/

Who Led This? Adrian Cockcroft

- Netflix Director of Web Engineering
  - 2007-2010
- Netflix Cloud Architect
  - 2010-2014
- Battery Ventures Technology Fellow
  - 2014-2016
- AWS VP Cloud Architecture Strategy
  - 2016-

Adrian’s Battery Venture pitch began with . . .

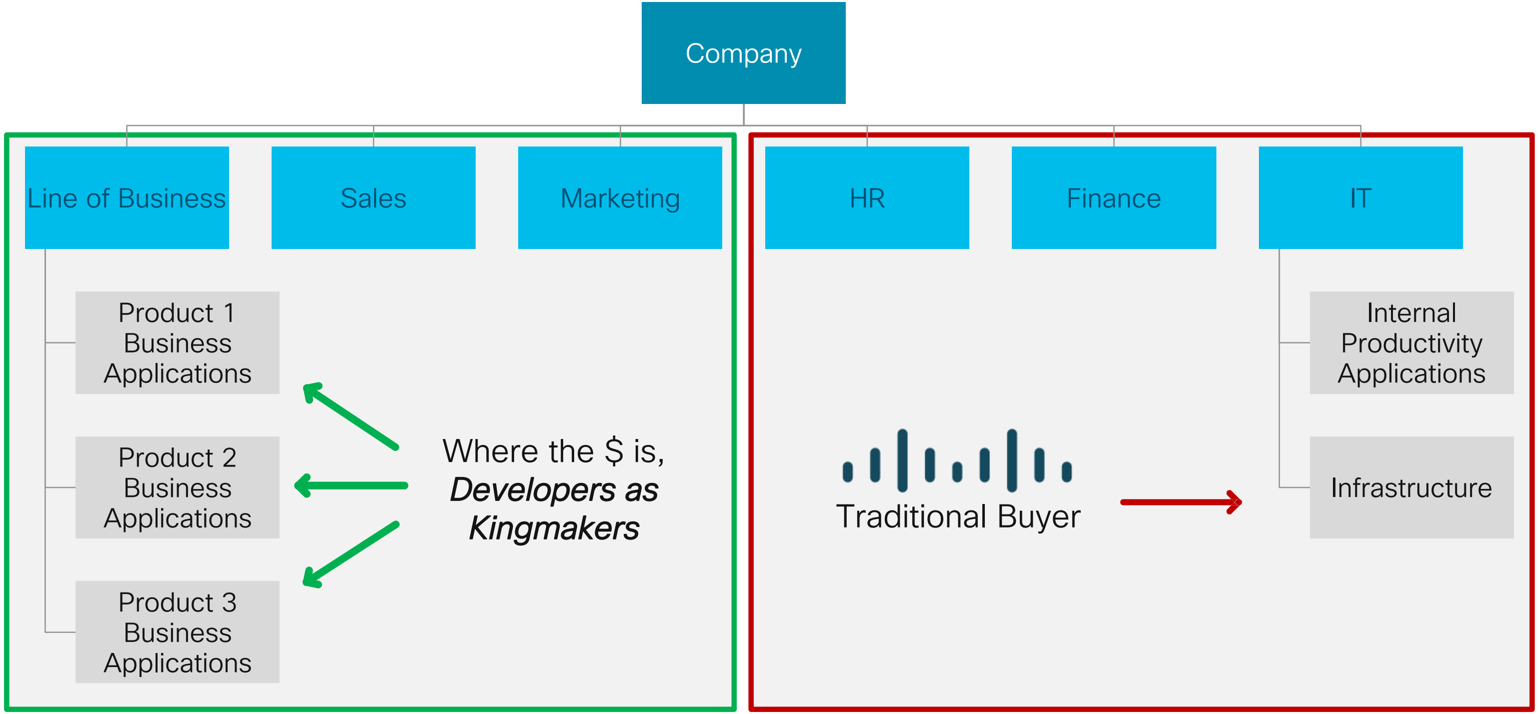


What I learned from my time at Netflix

- *Speed wins in the marketplace*
- *Remove friction from product development*
- *High trust, low process, no hand-offs between teams*
- *Freedom and responsibility culture*
- *Don’t do your own undifferentiated heavy lifting*
- *Use simple patterns automated by tooling*
- *Self service cloud makes impossible things instant*

http://www.slideshare.net/adriancockcroft/monktoberfest-fast-delivery

Standard Company Structure



Company

Line of Business

Sales

Marketing

HR

Finance

IT

Product 1 Business Applications

Product 2 Business Applications

Product 3 Business Applications

Where the $ is, ***Developers as Kingmakers***

Traditional Buyer

Internal Productivity Applications

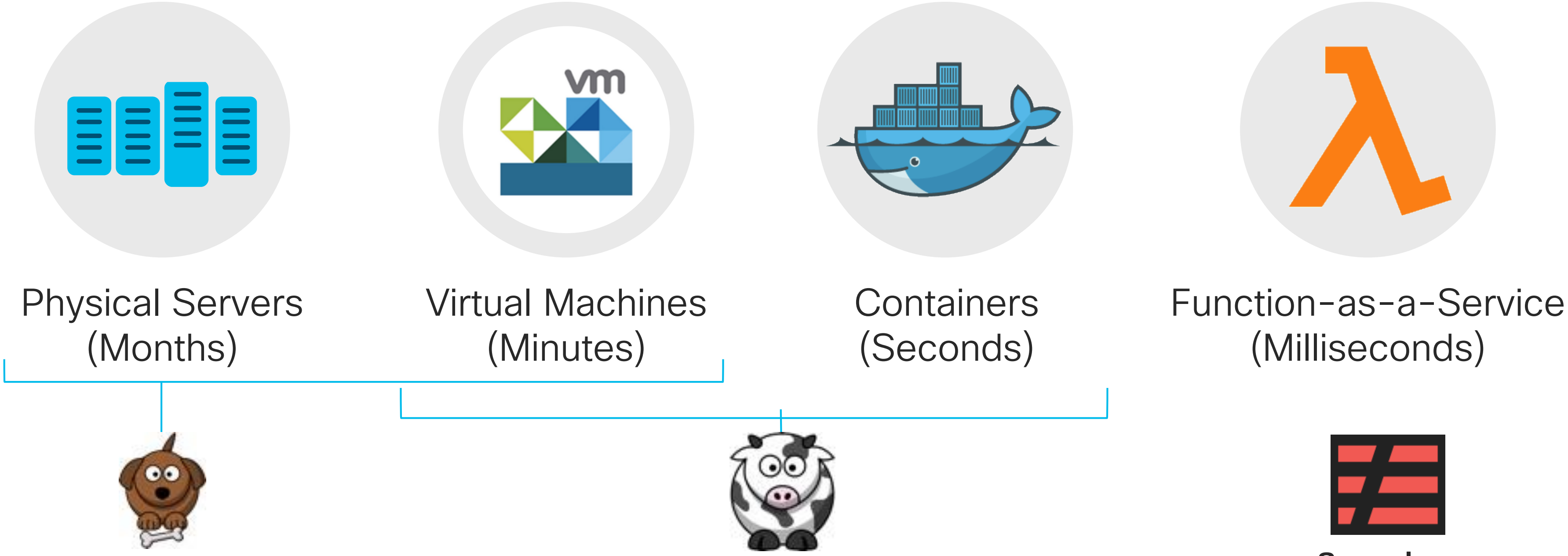
Infrastructure

Revenue Producers:
Optimized for innovation speed

Cost Centers:
Optimized for cost reduction

Application Architecture Approaches

Given time to create a new unit of compute



Physical Servers
(Months)

Virtual Machines
(Minutes)

Containers
(Seconds)

Function-as-a-Service
(Milliseconds)

Pets/Mode 1/Monoliths
Go to great lengths to
keep compute alive

Cattle/Mode 2/Microservices
Create and destroy compute
frequently

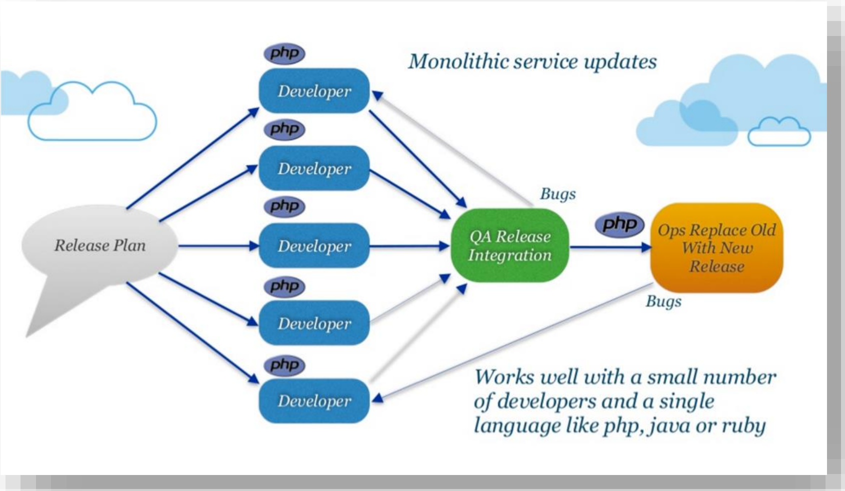
Serverless
Smaller and less coupled

Scarcity Has Changed

Used To Be

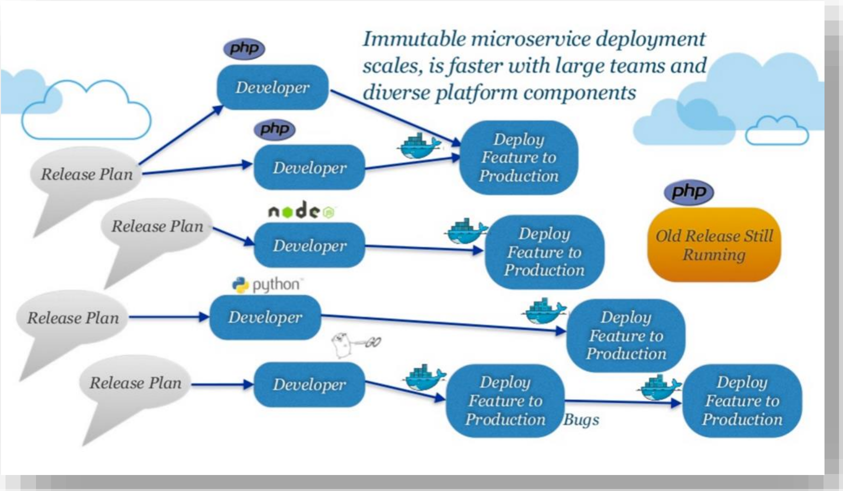
Is Now

Adrian: Monolithic vs Microservices



Monolithic service updates

Works well with a small number of developers and a single language like php, java or ruby



Immutable microservice deployment scales, is faster with large teams and diverse platform components

php
Developer
php
Developer
Release Plan
Deploy Feature to Production
php
Old Release Still Running
Release Plan
node
Developer
Deploy Feature to Production
python
Release Plan
Developer
Deploy Feature to Production
Release Plan
Developer
Deploy Feature to Production
Bugs
Deploy Feature to Production

All In The Name of More Iterations, More Innovation

http://www.slideshare.net/adriancockcroft/monktoberfest-fast-delivery

What Has Driven Cloud Adoption?

More Iterations,
More Innovation

Agenda

- ~~The Past~~
  - ~~What has driven cloud adoption?~~
- The Present
  - AWS Shared Responsibility Model
  - Microservices and K8s
- The Future
  - Serverless
  - Serverless in the Datacenter
  - No Code
  - Edge Clusters

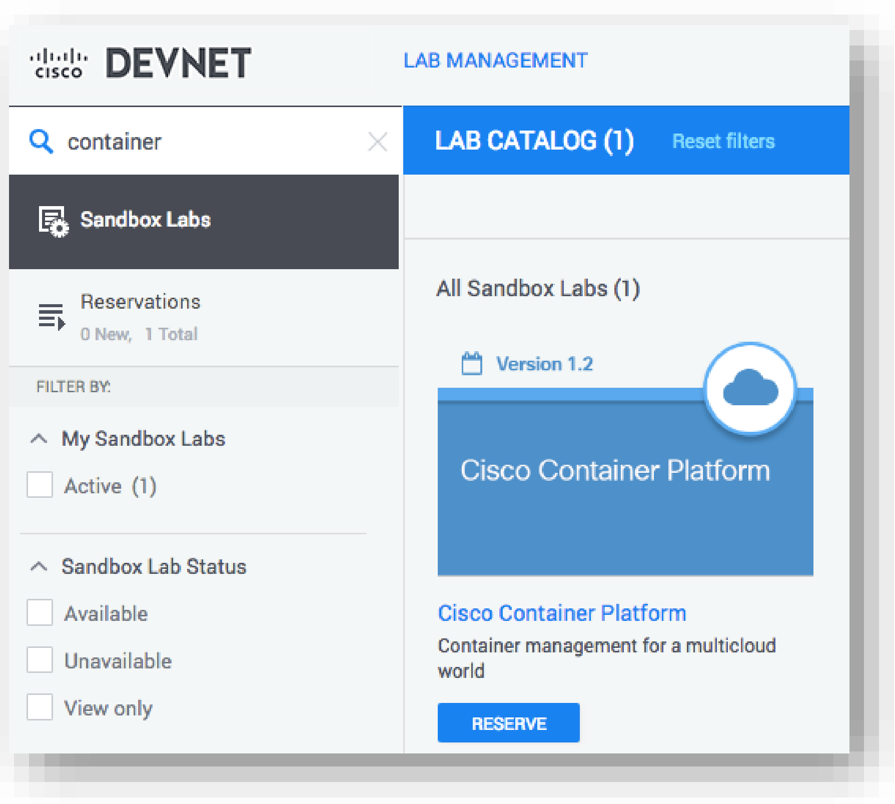
Demo #1:
DevNet Sandbox

K8s Live Demo

Explore yourself:

devnetsandbox.cisco.com

Search for “container”



cisco DEVNET LAB MANAGEMENT

container

LAB CATALOG (1) Reset filters

Sandbox Labs

Reservations
0 New, 1 Total

FILTER BY:

My Sandbox Labs

□ Active (1)

Sandbox Lab Status

□ Available

□ Unavailable

□ View only

All Sandbox Labs (1)

Version 1.2

Cisco Container Platform

Cisco Container Platform

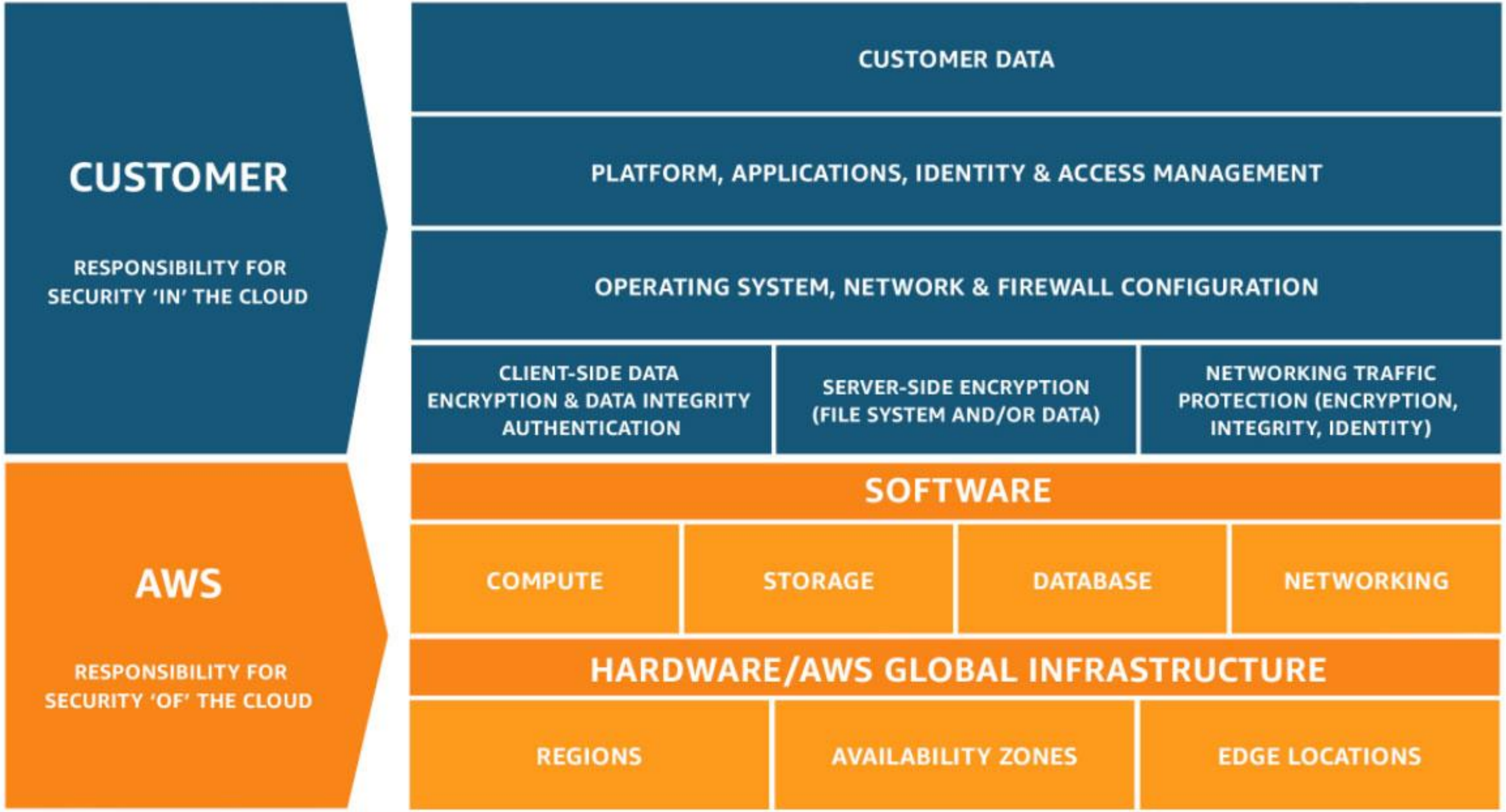
Container management for a multicloud world

RESERVE

The Present

AWS Shared Responsibility Model

The AWS Shared Responsibility Model



CUSTOMER

RESPONSIBILITY FOR SECURITY 'IN' THE CLOUD

CUSTOMER DATA

PLATFORM, APPLICATIONS, IDENTITY & ACCESS MANAGEMENT

OPERATING SYSTEM, NETWORK & FIREWALL CONFIGURATION

CLIENT-SIDE DATA ENCRYPTION & DATA INTEGRITY AUTHENTICATION

SERVER-SIDE ENCRYPTION (FILE SYSTEM AND/OR DATA)

NETWORKING TRAFFIC PROTECTION (ENCRYPTION, INTEGRITY, IDENTITY)

AWS

RESPONSIBILITY FOR SECURITY 'OF' THE CLOUD

SOFTWARE

COMPUTE

STORAGE

DATABASE

NETWORKING

HARDWARE/AWS GLOBAL INFRASTRUCTURE

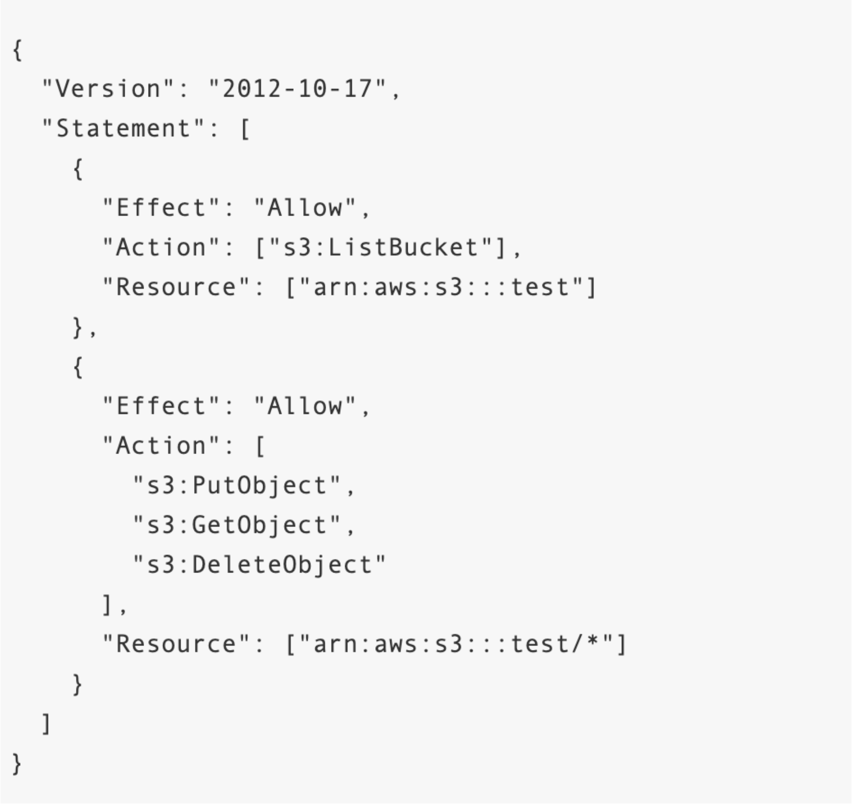
REGIONS

AVAILABILITY ZONES

EDGE LOCATIONS

https://aws.amazon.com/compliance/shared-responsibility-model/

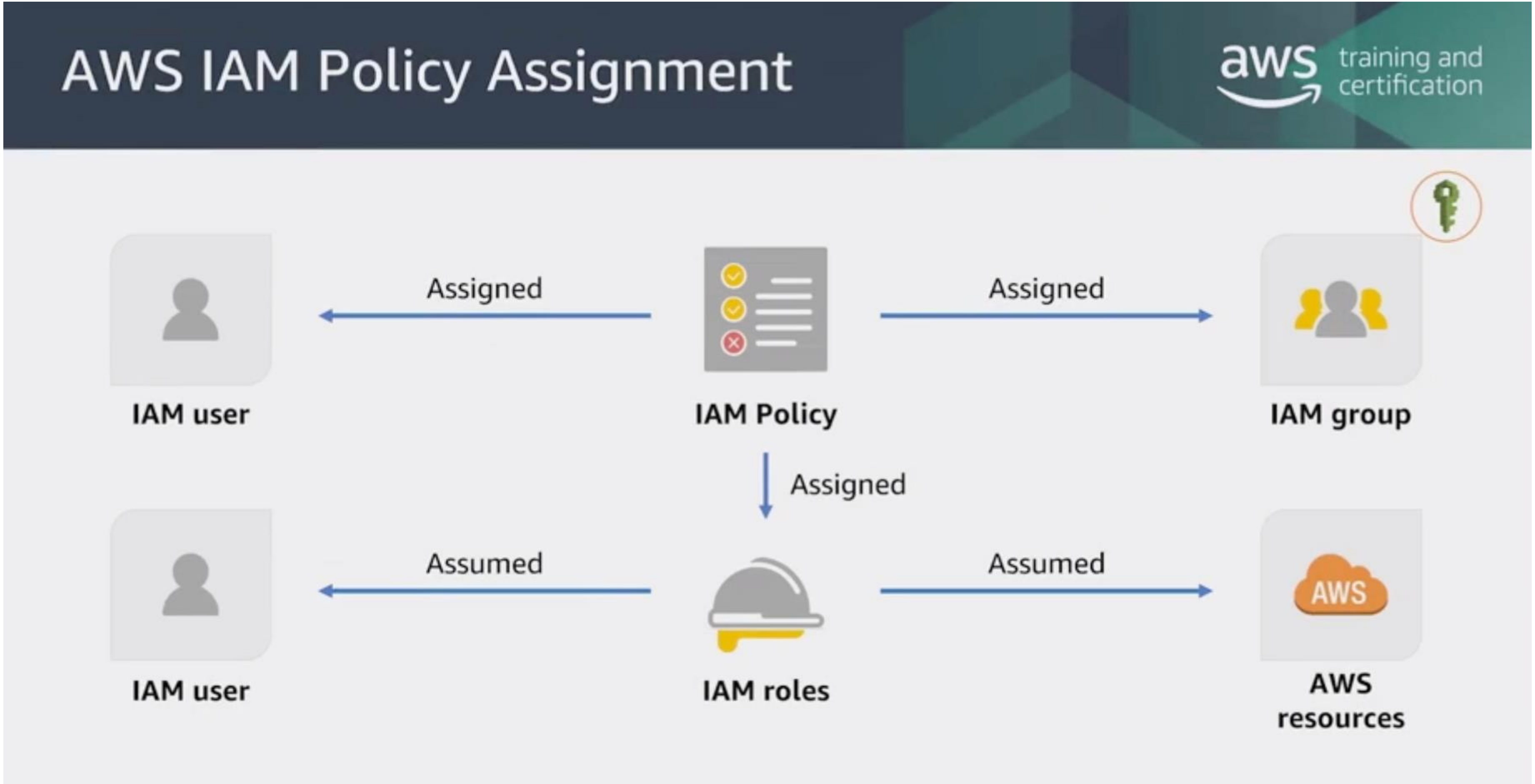
Sample IAM policy



```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": ["s3:ListBucket"],
      "Resource": ["arn:aws:s3:::test"]
    },
    {
      "Effect": "Allow",
      "Action": [
        "s3:PutObject",
        "s3:GetObject",
        "s3:DeleteObject"
      ],
      "Resource": ["arn:aws:s3:::test/*"]
    }
  ]
}
```

https://aws.amazon.com/blogs/security/writing-iam-policies-how-to-grant-access-to-an-amazon-s3-bucket/

AWS IAM Policy Assignment

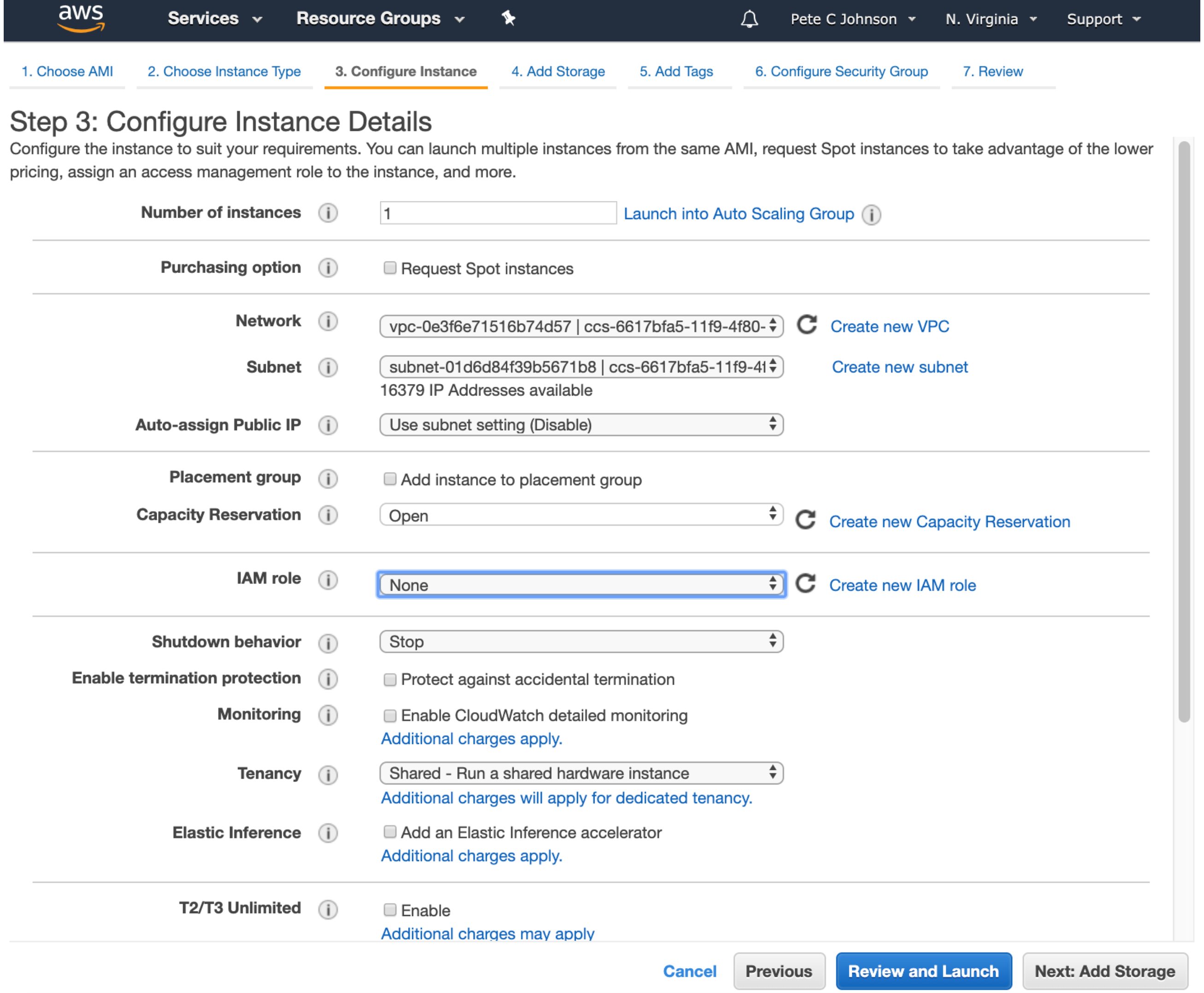


aws training and certification

IAM user ← Assigned — IAM Policy — Assigned → IAM group

IAM Policy ↓ Assigned IAM roles

IAM user ← Assumed — IAM roles — Assumed → AWS resources

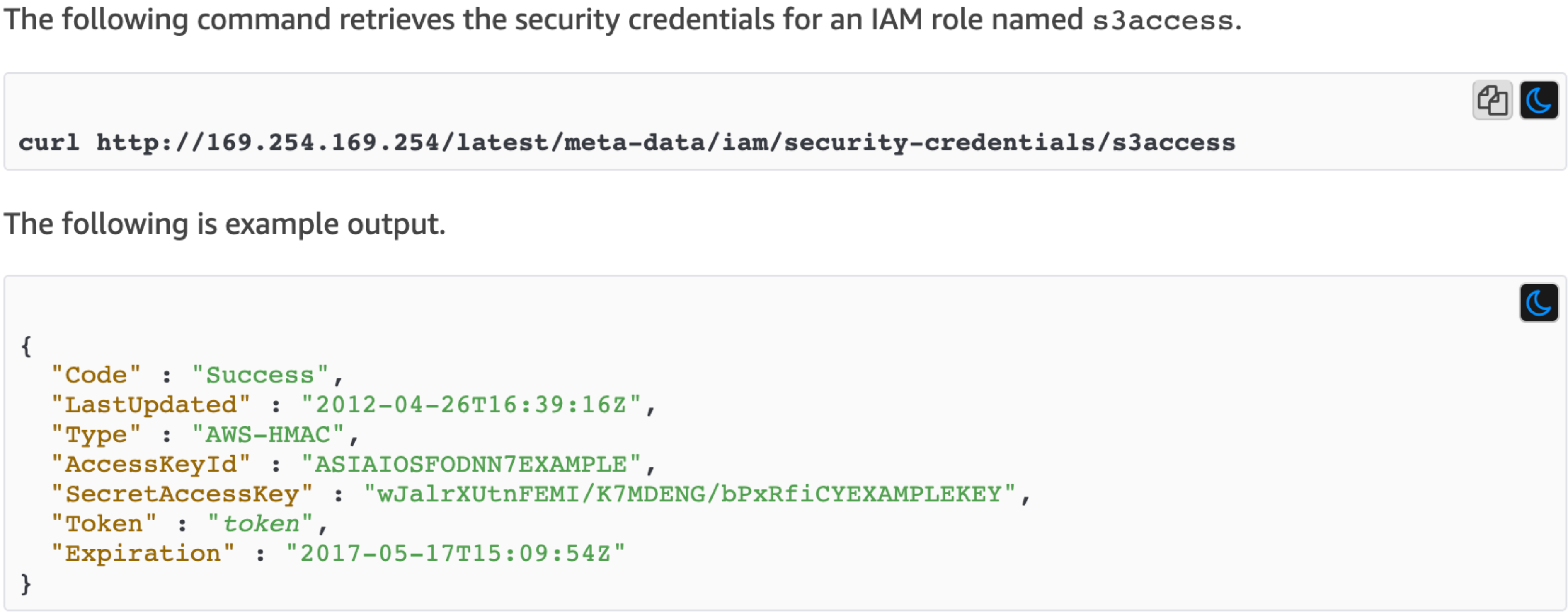


Step 3: Configure Instance Details

Configure the instance to suit your requirements. You can launch multiple instances from the same AMI, request Spot instances to take advantage of the lower pricing, assign an access management role to the instance, and more.

Number of instances	1
Purchasing option	□ Request Spot instances
Network	vpc-0e3f6e71516b74d57 | ccs-6617bfa5-11f9-4f80-
Subnet	subnet-01d6d84f39b5671b8 | ccs-6617bfa5-11f9-4f 16379 IP Addresses available
Auto-assign Public IP	Use subnet setting (Disable)
Placement group	□ Add instance to placement group
Capacity Reservation	Open
IAM role	None
Shutdown behavior	Stop
Enable termination protection	□ Protect against accidental termination
Monitoring	□ Enable CloudWatch detailed monitoring Additional charges apply.
Tenancy	Shared - Run a shared hardware instance Additional charges will apply for dedicated tenancy.
Elastic Inference	□ Add an Elastic Inference accelerator Additional charges apply.
T2/T3 Unlimited	□ Enable Additional charges may apply

How does an instance get its profile creds?



The following command retrieves the security credentials for an IAM role named `s3access`.

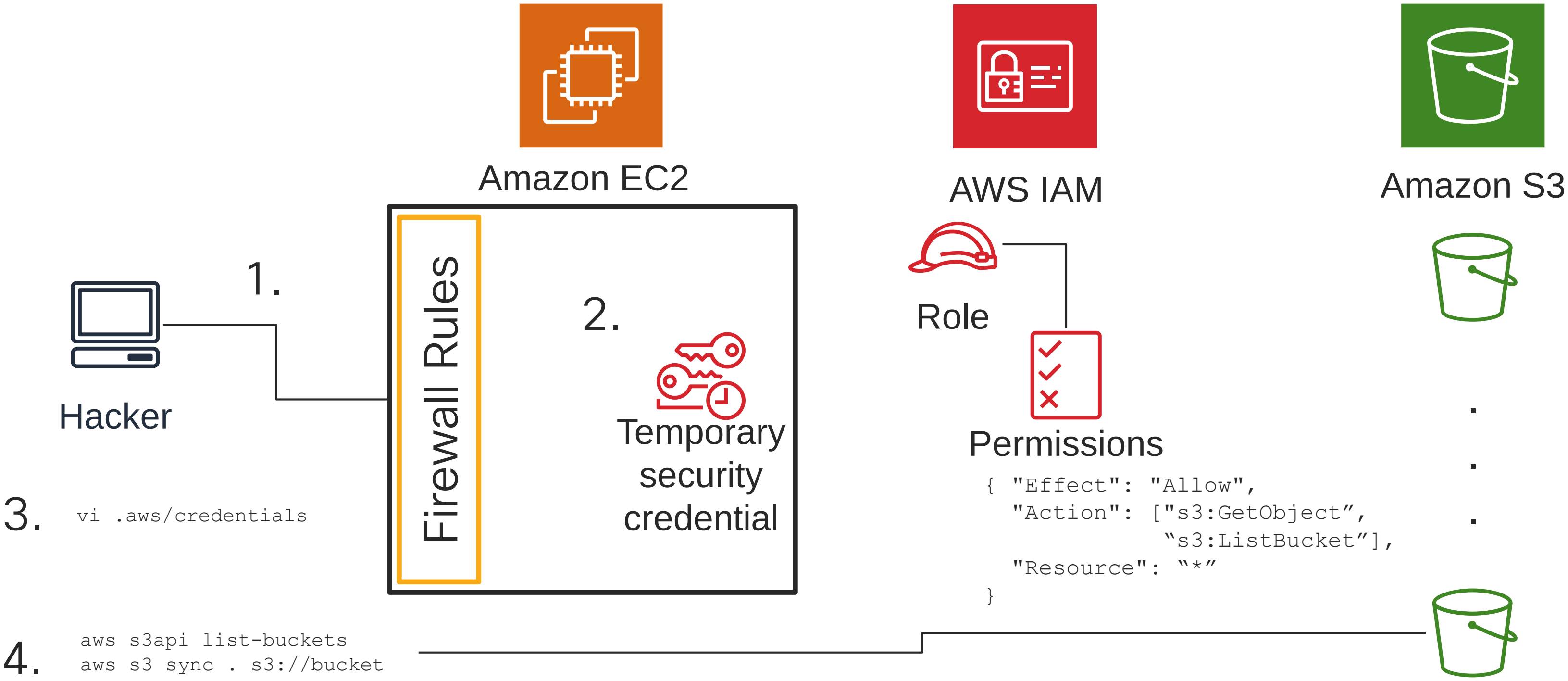
```
curl http://169.254.169.254/latest/meta-data/iam/security-credentials/s3access
```

The following is example output.

```
{
  "Code" : "Success",
  "LastUpdated" : "2012-04-26T16:39:16Z",
  "Type" : "AWS-HMAC",
  "AccessKeyId" : "ASIAIOSFODNN7EXAMPLE",
  "SecretAccessKey" : "wJalrXUtnFEMI/K7MDENG/bPxRfiCYEXAMPLEKEY",
  "Token" : "token",
  "Expiration" : "2017-05-17T15:09:54Z"
}
```

https://docs.aws.amazon.com/AWSEC2/latest/UserGuide/iam-roles-for-amazon-ec2.html

A Common Breach Tale



```
{ "Effect": "Allow",
  "Action": ["s3:GetObject”,
             “s3:ListBucket”],
  "Resource": “*”
}
```

3.

```
vi .aws/credentials
```

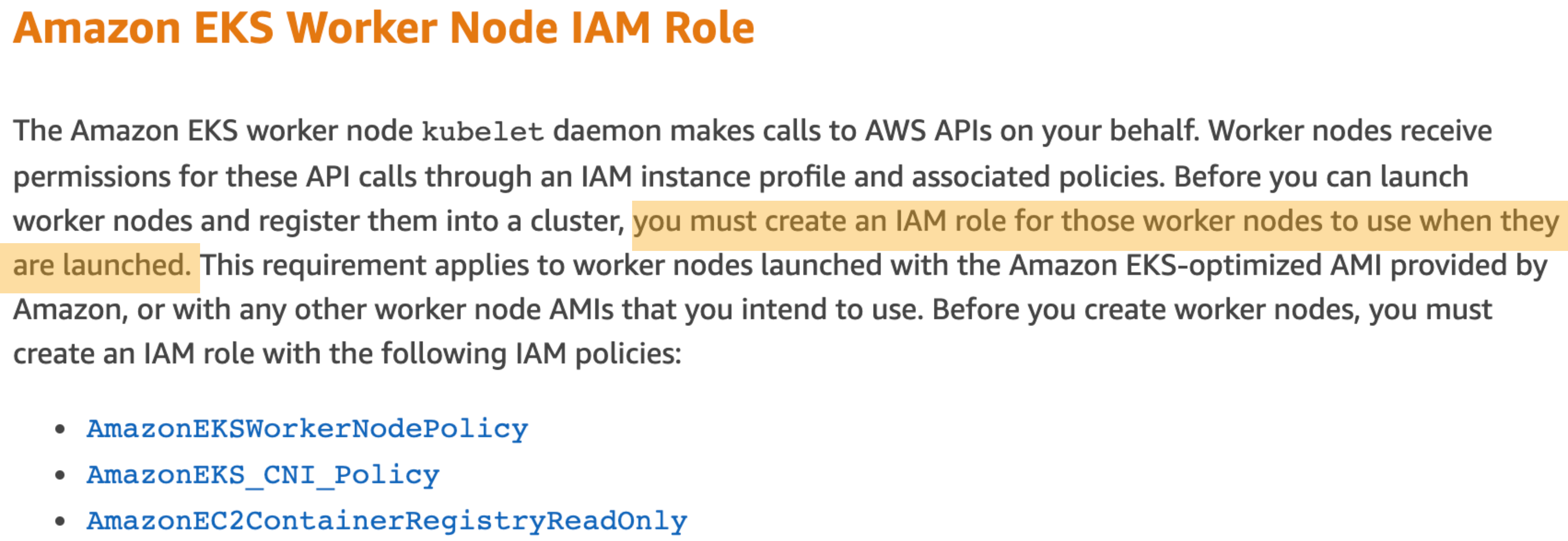
4.

```
aws s3api list-buckets
aws s3 sync . s3://bucket
```

Common Issues

- Misconfigured firewall rules (and lack of virtual firewall)
- Too broadly granted IAM policy/Too flat an application architecture
- No StealthwatchCloud to detect rogue logins
- Could have rolled their own secrets management instead of relying on one published publicly

It’s not any better for EKS, in fact, it’s worse



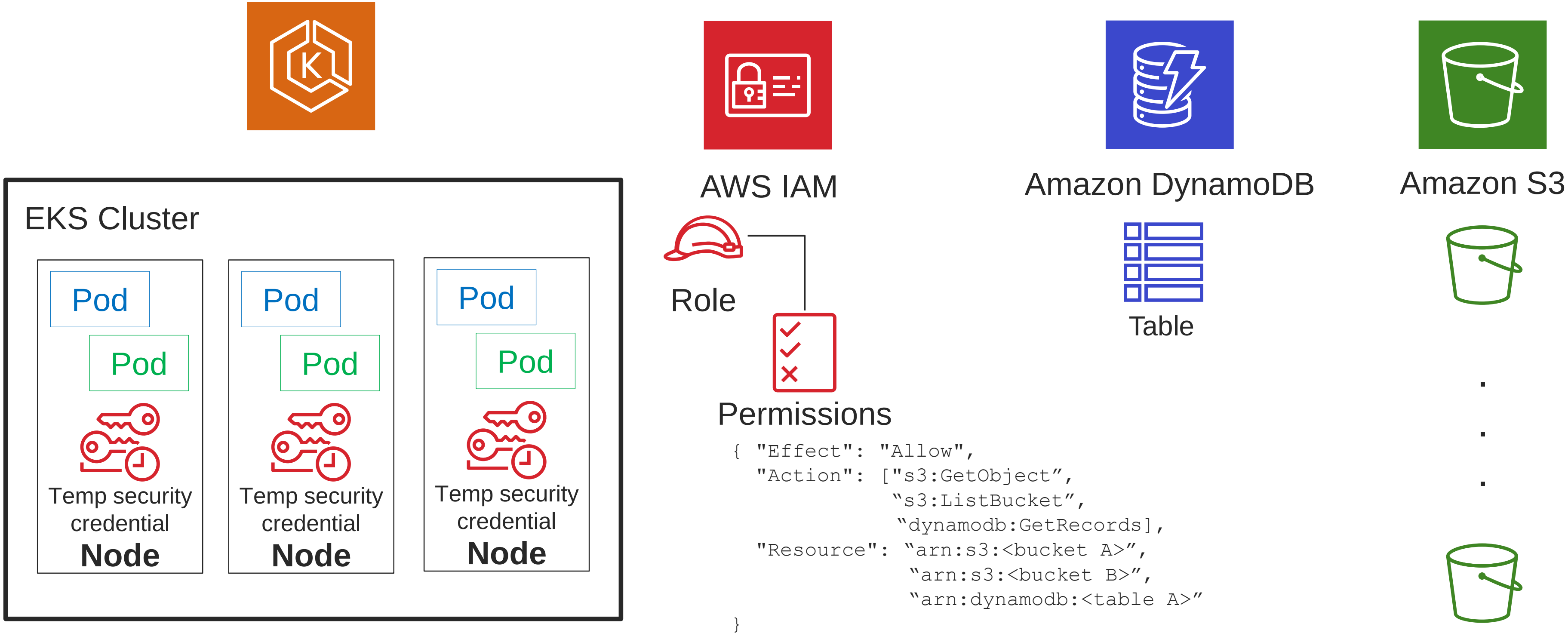
Amazon EKS Worker Node IAM Role

The Amazon EKS worker node `kubelet` daemon makes calls to AWS APIs on your behalf. Worker nodes receive permissions for these API calls through an IAM instance profile and associated policies. Before you can launch worker nodes and register them into a cluster, you must create an IAM role for those worker nodes to use when they are launched. This requirement applies to worker nodes launched with the Amazon EKS-optimized AMI provided by Amazon, or with any other worker node AMIs that you intend to use. Before you create worker nodes, you must create an IAM role with the following IAM policies:

- `AmazonEKSWorkerNodePolicy`
- `AmazonEKS_CNI_Policy`
- `AmazonEC2ContainerRegistryReadOnly`

https://docs.aws.amazon.com/eks/latest/userguide/worker_node_IAM_role.html

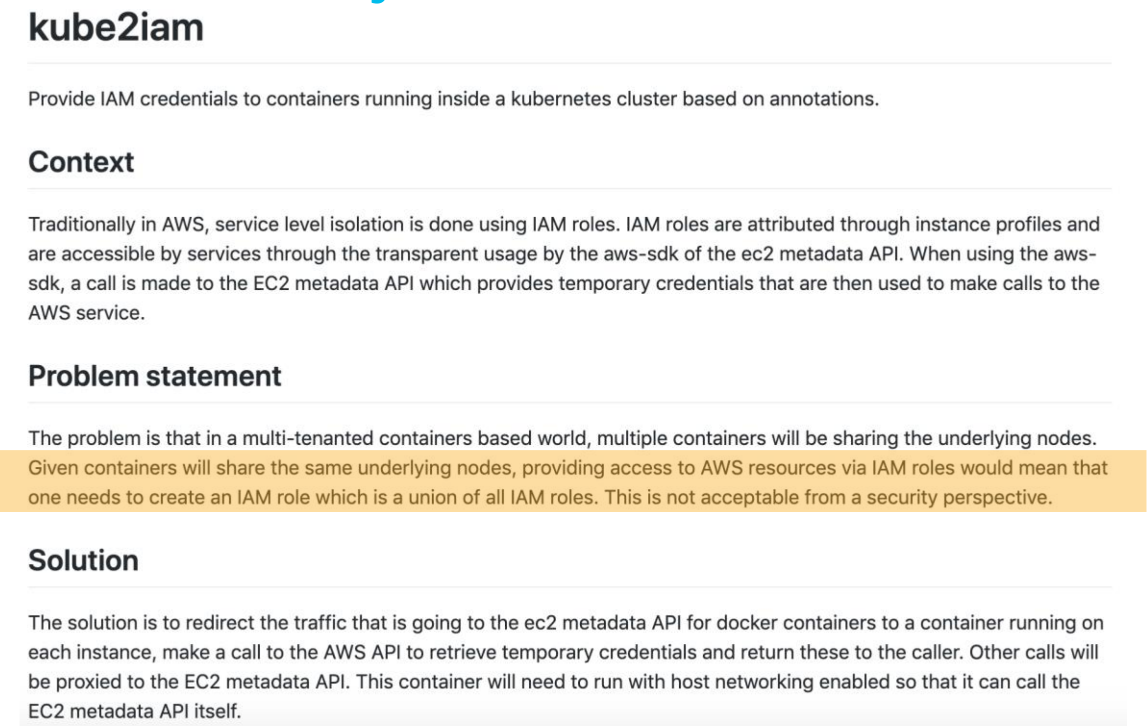
IAM Roles and EKS Clusters



```
{ "Effect": "Allow",
  "Action": ["s3:GetObject”,
             “s3:ListBucket”,
             “dynamodb:GetRecords],
  "Resource": “arn:s3:<bucket A>”,
              “arn:s3:<bucket B>”,
              “arn:dynamodb:<table A>”
}
```

* At least K8s has secrets management

Open Source Help: https://github.com/jtblin/kube2iam



kube2iam

Provide IAM credentials to containers running inside a kubernetes cluster based on annotations.

Context

Traditionally in AWS, service level isolation is done using IAM roles. IAM roles are attributed through instance profiles and are accessible by services through the transparent usage by the aws-sdk of the ec2 metadata API. When using the aws-sdk, a call is made to the EC2 metadata API which provides temporary credentials that are then used to make calls to the AWS service.

Problem statement

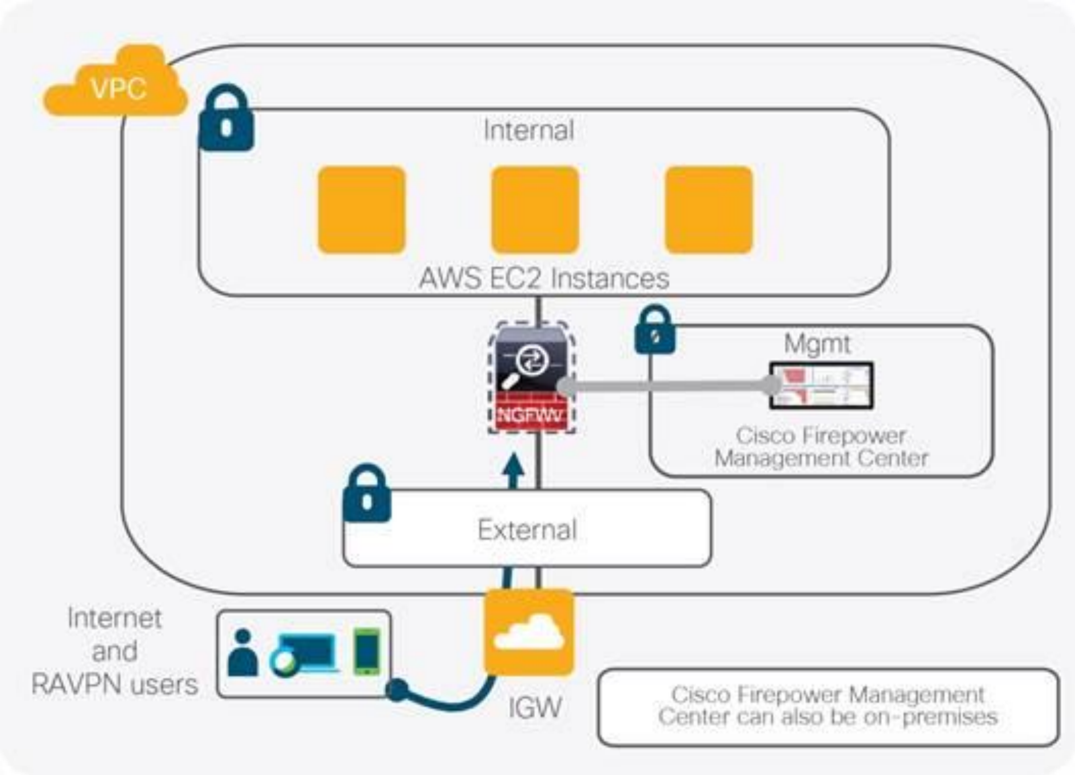
The problem is that in a multi-tenanted containers based world, multiple containers will be sharing the underlying nodes. Given containers will share the same underlying nodes, providing access to AWS resources via IAM roles would mean that one needs to create an IAM role which is a union of all IAM roles. This is not acceptable from a security perspective.

Solution

The solution is to redirect the traffic that is going to the ec2 metadata API for docker containers to a container running on each instance, make a call to the AWS API to retrieve temporary credentials and return these to the caller. Other calls will be proxied to the EC2 metadata API. This container will need to run with host networking enabled so that it can call the EC2 metadata API itself.

https://medium.com/merapar/securing-iam-access-in-kubernetes-cfbcc6954de

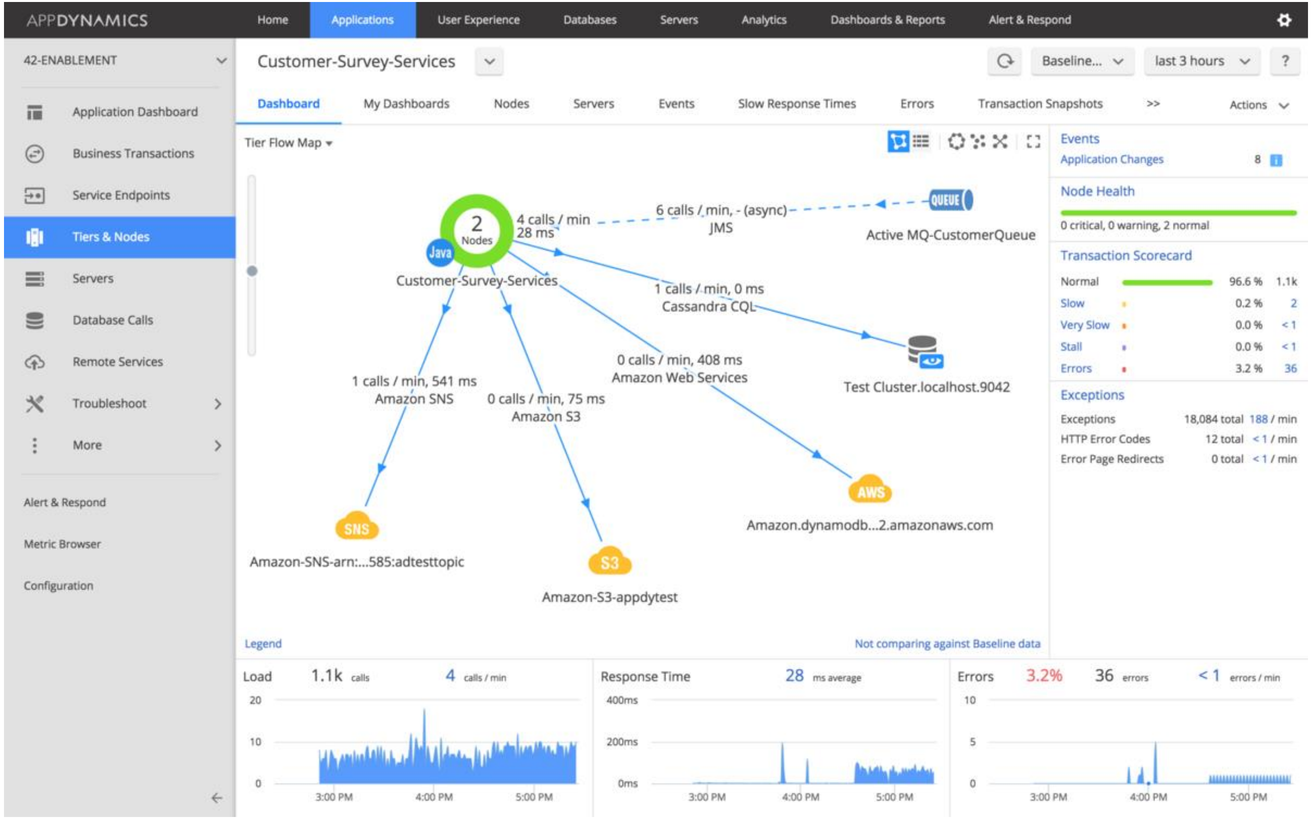
Cisco Help #1: Virtual Firewall



VPC
Internal
AWS EC2 Instances
NGFWv
Mgmt
Cisco Firepower Management Center
External
Internet and RAVPN users
IGW
Cisco Firepower Management Center can also be on-premises

https://www.cisco.com/c/en/us/products/collateral/security/adaptive-security-virtual-appliance-asav/white-paper-c11-740505.html

Cisco Help #2: App Dynamics



Transaction Scorecard

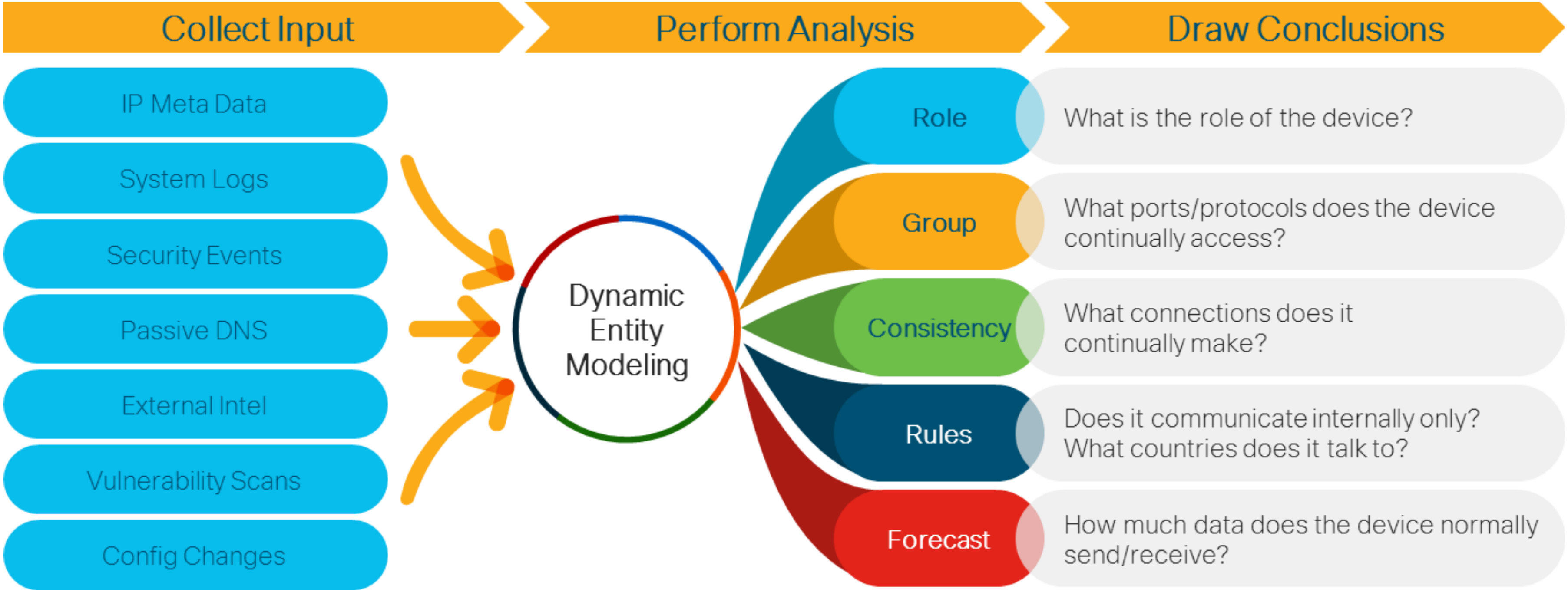
Normal	96.6 %	1.1k
Slow	0.2 %	2
Very Slow	0.0 %	< 1
Stall	0.0 %	< 1
Errors	3.2 %	36

Exceptions

Exceptions	18,084 total	188 / min
HTTP Error Codes	12 total	< 1 / min
Error Page Redirects	0 total	< 1 / min

https://www.appdynamics.com/solutions/cloud-monitoring/

Cisco Help #3: Stealthwatch Cloud



Collect Input

- IP Meta Data
- System Logs
- Security Events
- Passive DNS
- External Intel
- Vulnerability Scans
- Config Changes

Perform Analysis

Dynamic Entity Modeling

- Role
- Group
- Consistency
- Rules
- Forecast

Draw Conclusions

- What is the role of the device?
- What ports/protocols does the device continually access?
- What connections does it continually make?
- Does it communicate internally only? What countries does it talk to?
- How much data does the device normally send/receive?

https://blogs.cisco.com/security/stealthwatch-cloud-securing-the-public-cloud-without-undercutting-it

Shared Responsibility Model: Takeaways

- Shared Responsibility Model limits AWS liability

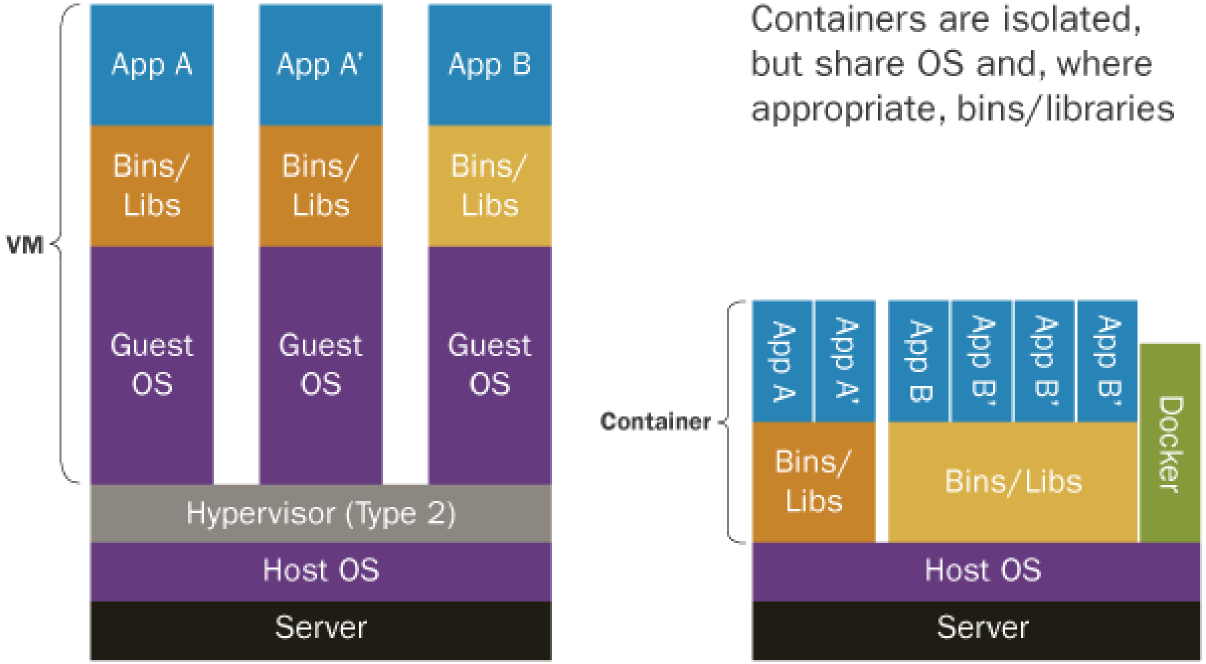
- Cisco products help with the customer part of that Shared Responsibility Model
  - Virtual firewalls (better than firewall rules)
  - App D (better than PowerPoint for complex app architectures)
  - Stealthwatch Cloud (better for noticing unusual access behavior)

The Present

Microservices, K8s

Demo #2:
CCP Tenant Cluster

Comparing VMs and Containers



Containers are isolated, but share OS and, where appropriate, bins/libraries

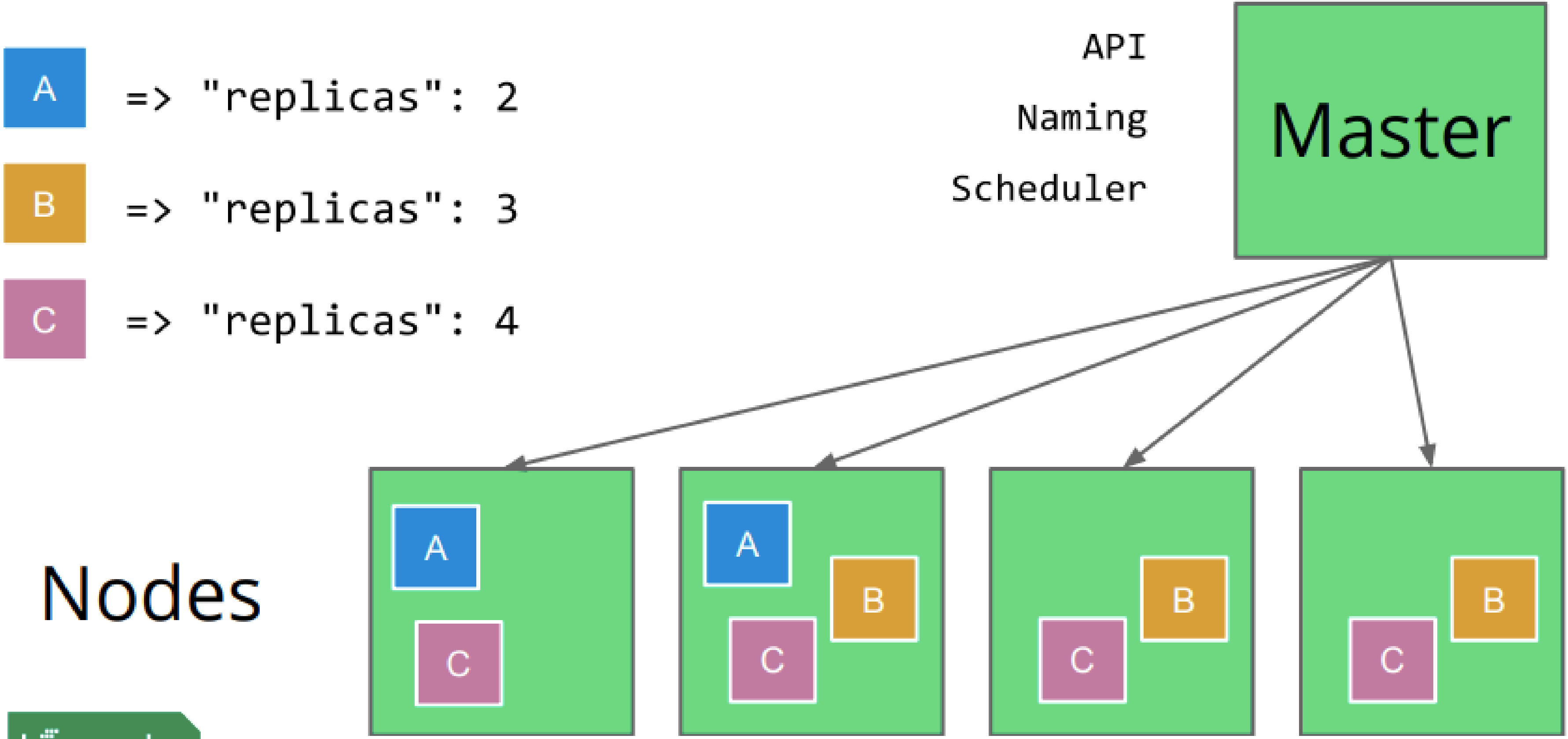
https://www.microcontrollertips.com/containerization-differs-virtual-machines-faq/

What is Kubernetes?

- Open source container cluster manager
- Used as a backend in Google's App Engine
- Runs on Private and Public Clouds, and even on Bare metal

Source: https://www.slideshare.net/ZoharStolar/introduction-to-containers-running-dockers-using-kubernetes

KUBERNETES ARCHITECTURE



A => "replicas": 2

B => "replicas": 3

C => "replicas": 4

API

Naming

Scheduler

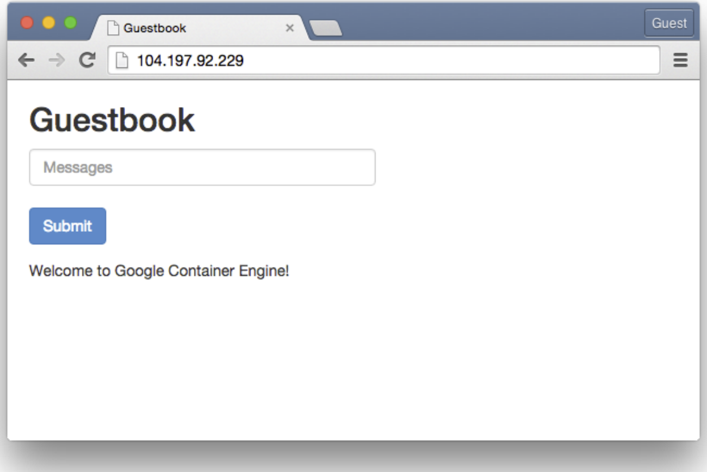
Master

Nodes

Source: https://www.slideshare.net/ZoharStolar/introduction-to-containers-running-dockers-using-kubernetes

Demo #3:
K8s Guestbook

Demo: Guestbook Application Architecture

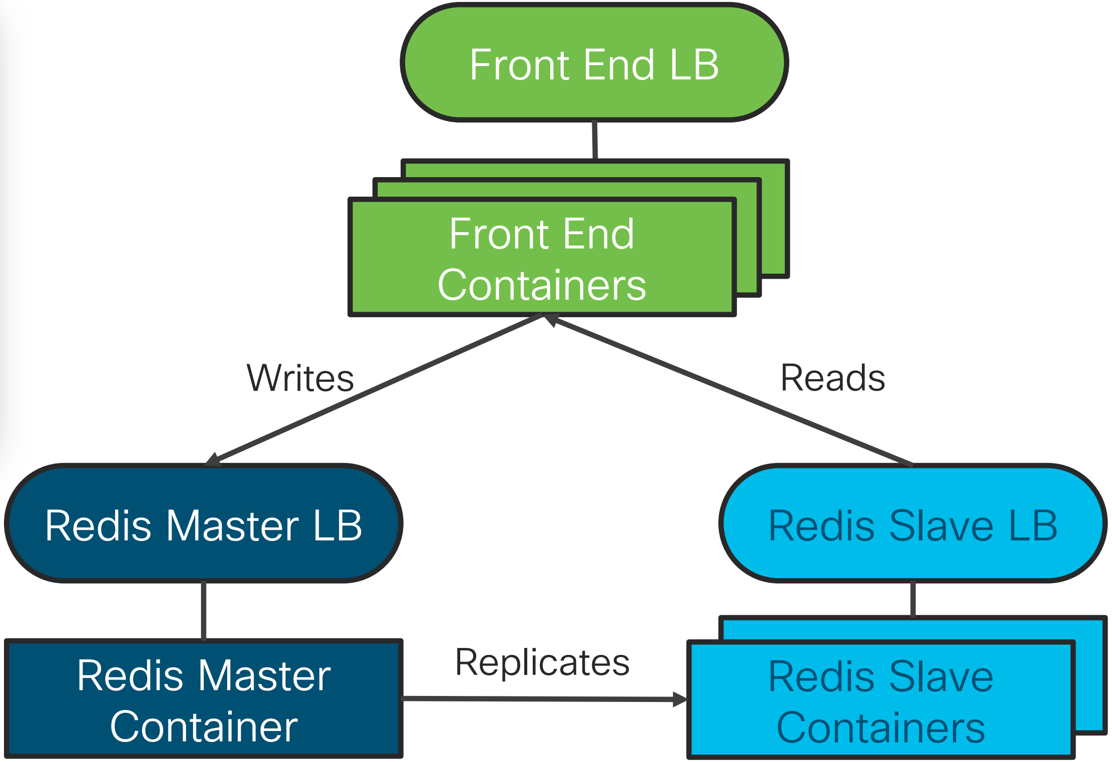


Guestbook

Messages

Submit

Welcome to Google Container Engine!



K8s Broad Support

GKE = Google Hosted and Managed K8s

EKS = AWS Hosted and Managed K8s

CCP = Cisco automated install of K8s

Agenda

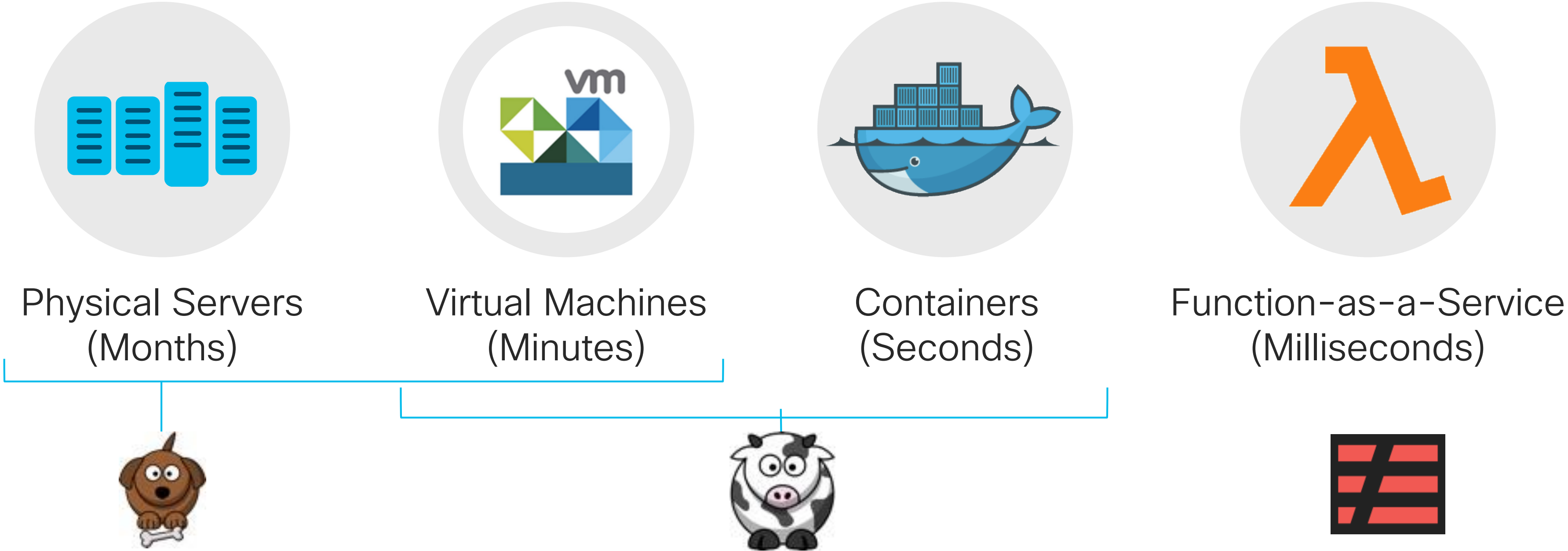
- ~~The Past~~
  - ~~What has driven cloud adoption?~~
- ~~The Present~~
  - ~~AWS Shared Responsibility Model~~
  - ~~Microservices and K8s~~
- The Future
  - Serverless
  - Serverless in the Datacenter
  - No Code
  - Edge Clusters

The Future

Serverless

Application Architecture Approaches

Given time to create a new unit of compute



Physical Servers (Months)

Virtual Machines (Minutes)

Containers (Seconds)

Function-as-a-Service (Milliseconds)

Pets/Mode 1/Monoliths
Go to great lengths to keep compute alive

Cattle/Mode 2/Microservices
Create and destroy compute frequently

Serverless
Smaller and less coupled

Some Terminology & Technology Maturity

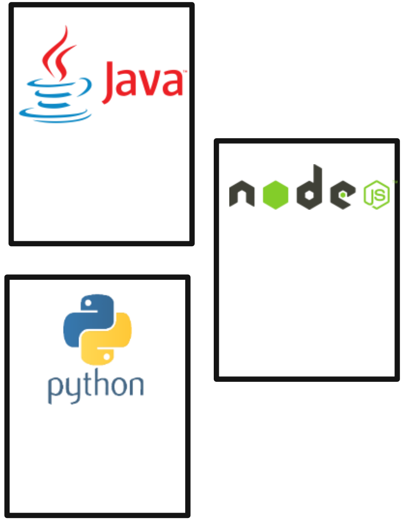
Serverless = The application architecture approach

FaaS = The underpinnings that make it possible

Serverless is to FaaS as Microservices are to Containers

Serverless 2020 ~= Cloud 2012

How FaaS Runtimes Work

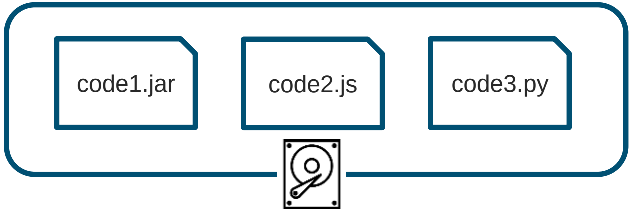


Standby containers
w/ language runtimes
but no app code

Event
Gateway

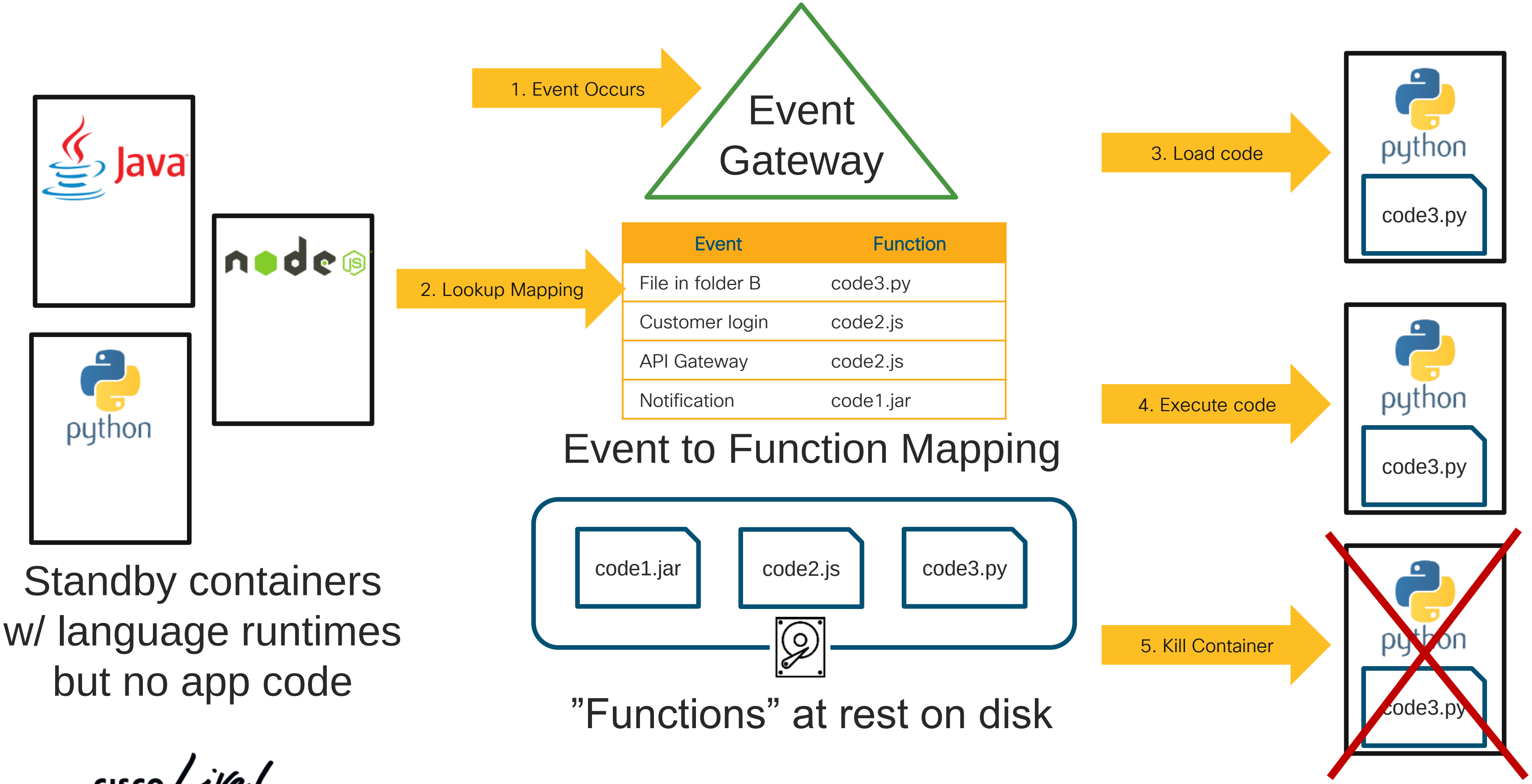
Event	Function
File in folder B	code3.py
Customer login	code2.js
API Gateway	code2.js
Notification	code1.jar

Event to Function Mapping



”Functions” at rest on disk

How FaaS Runtimes Work



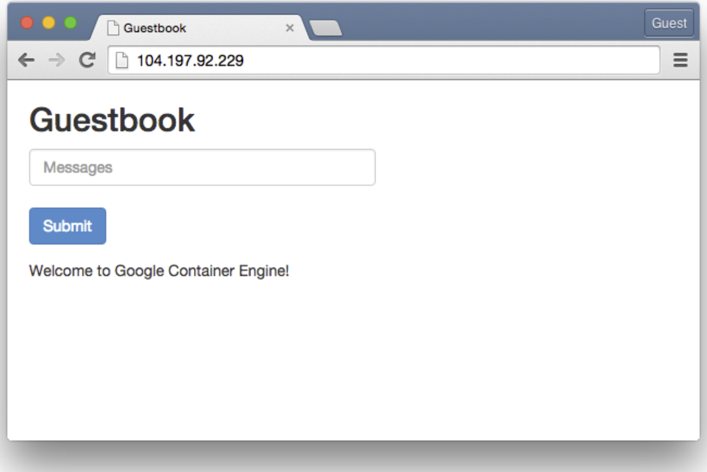
Event	Function
File in folder B	code3.py
Customer login	code2.js
API Gateway	code2.js
Notification	code1.jar

Event to Function Mapping

Standby containers w/ language runtimes but no app code

”Functions” at rest on disk

Guestbook Application Architecture



Guestbook

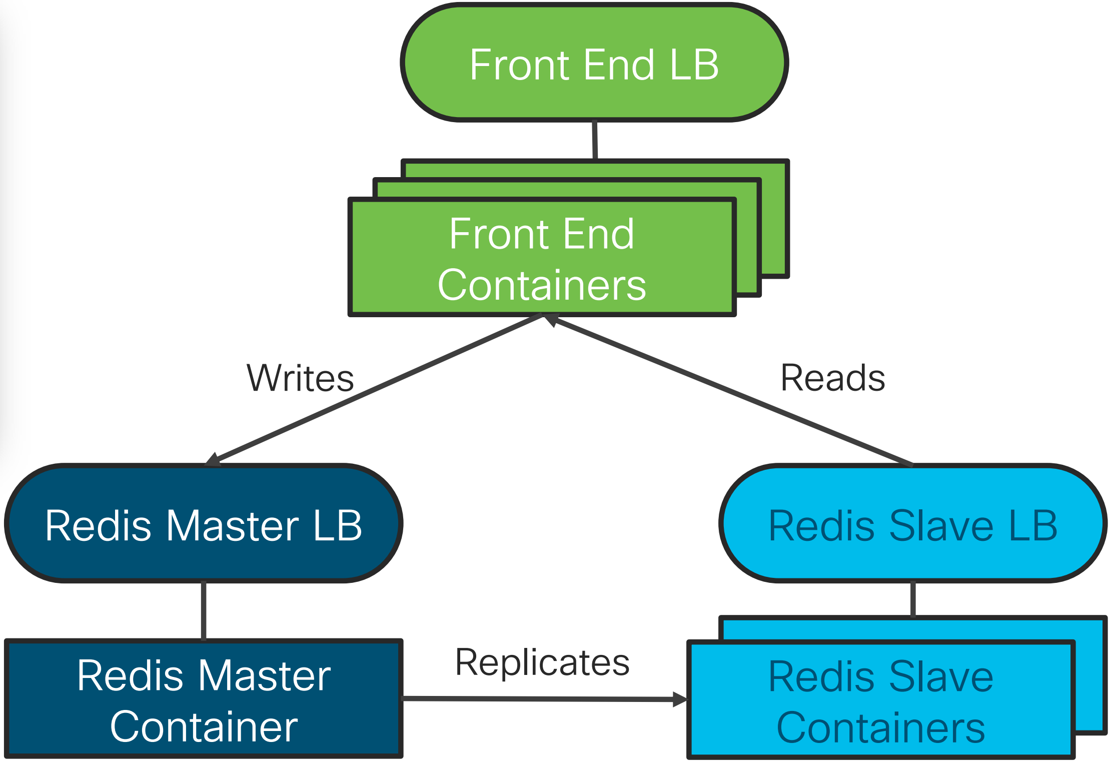
104.197.92.229

Guestbook

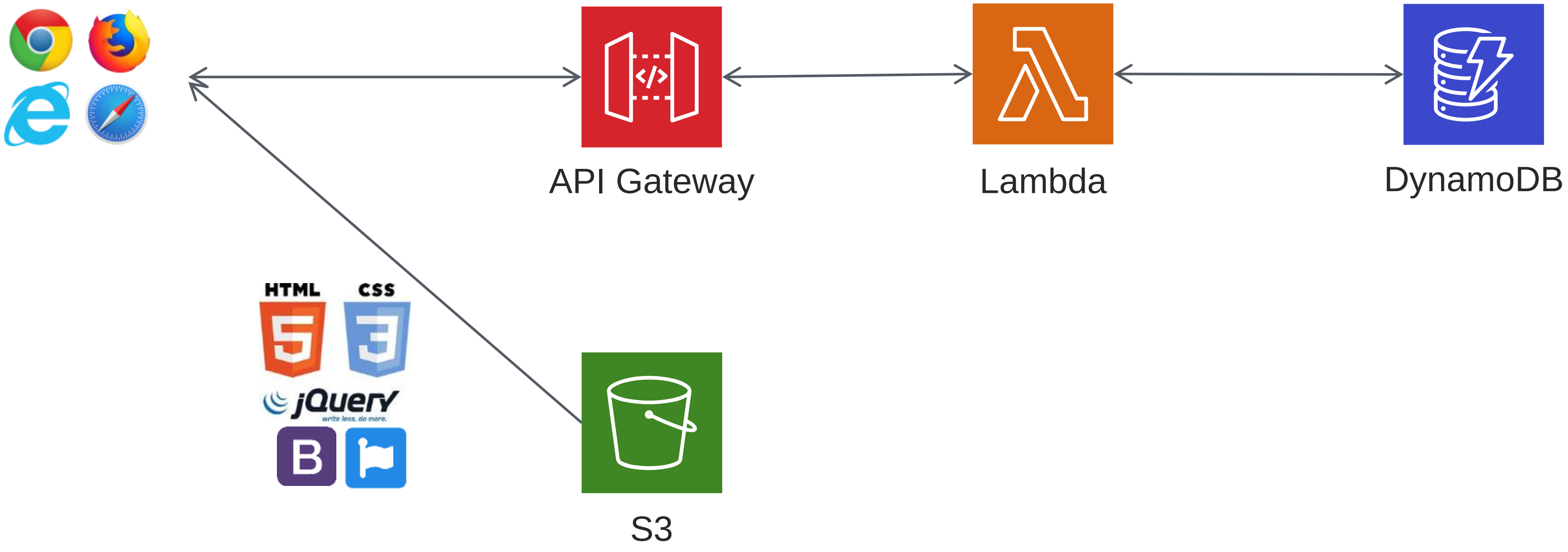
Messages

Submit

Welcome to Google Container Engine!



Demo: Serverless Application Architecture

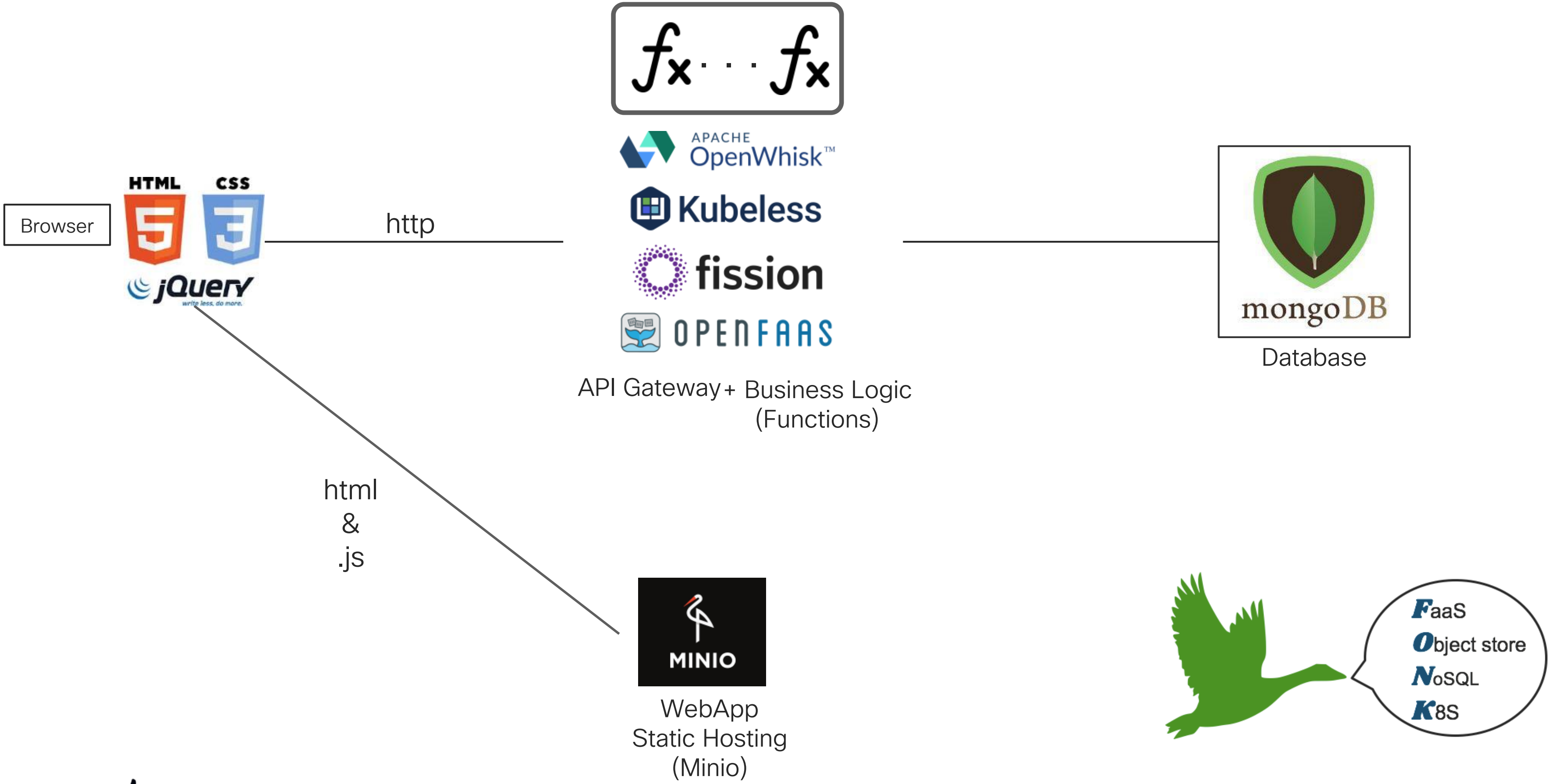


Demo #4:
Serverless Guestbook

The Future

Serverless in the Datacenter

FONK Guestbook



Browser

http

html & .js

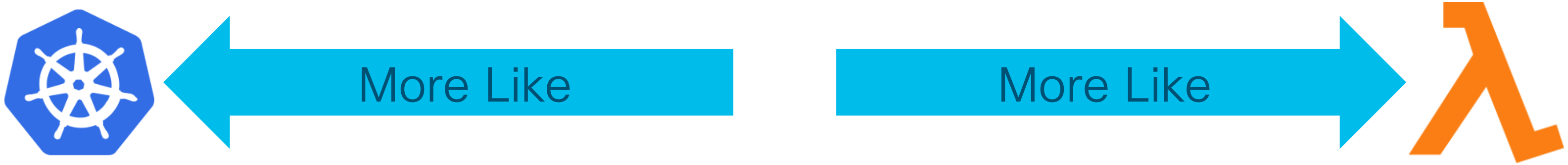
API Gateway + Business Logic (Functions)

Database

WebApp Static Hosting (Minio)

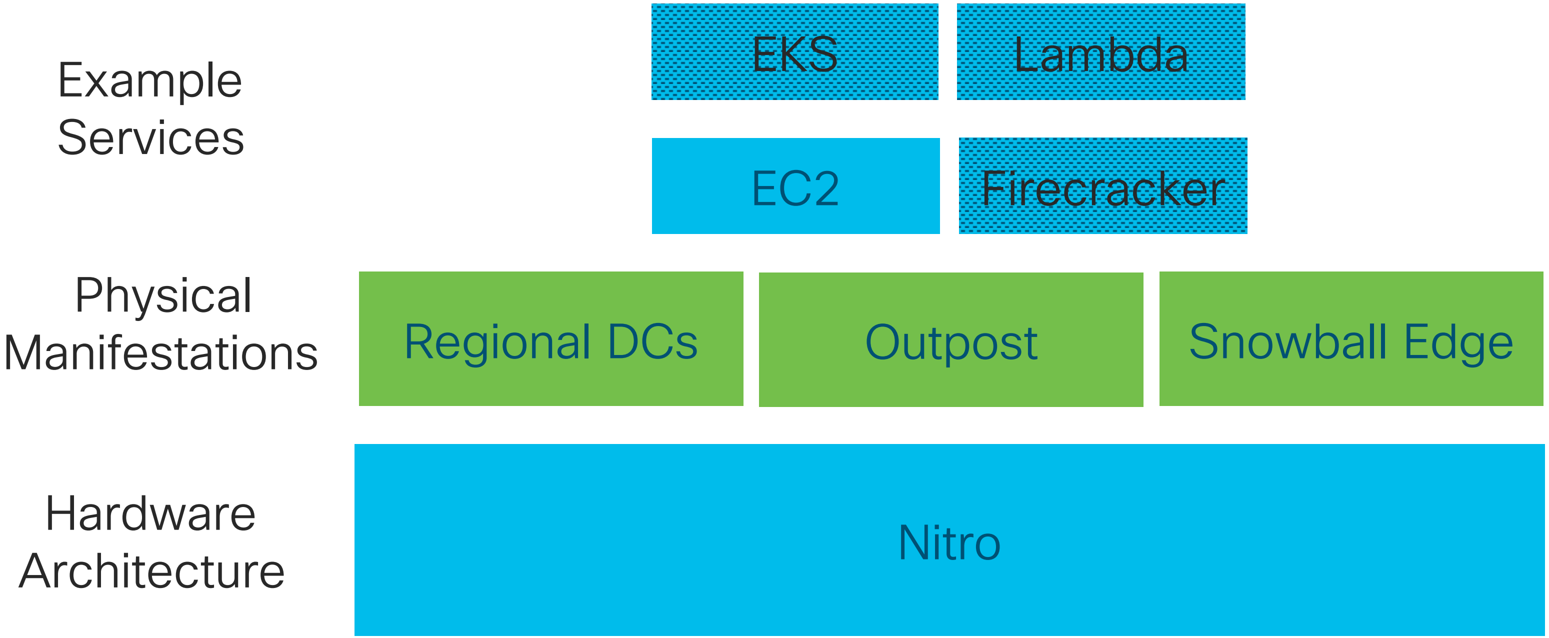
FaaS on K8S Landscape findings from fonk-apps.io

	Knative	OPENFAAS	fn	fission	Kubeless	APACHE OpenWhisk™
Dockerfile	Required	Hidden	Hidden	Hidden	Hidden	Hidden
Image Repo	Required	Required	Required	None	None	None
Local Docker	Required	Required	Required	None	None	None
Base Image	Required	Required	Required	Required	None	None


More Like

More Like

AWS Stack



Example Services

EKS

Lambda

EC2

Firecracker

Physical Manifestations

Regional DCs

Outpost

Snowball Edge

Hardware Architecture

Nitro

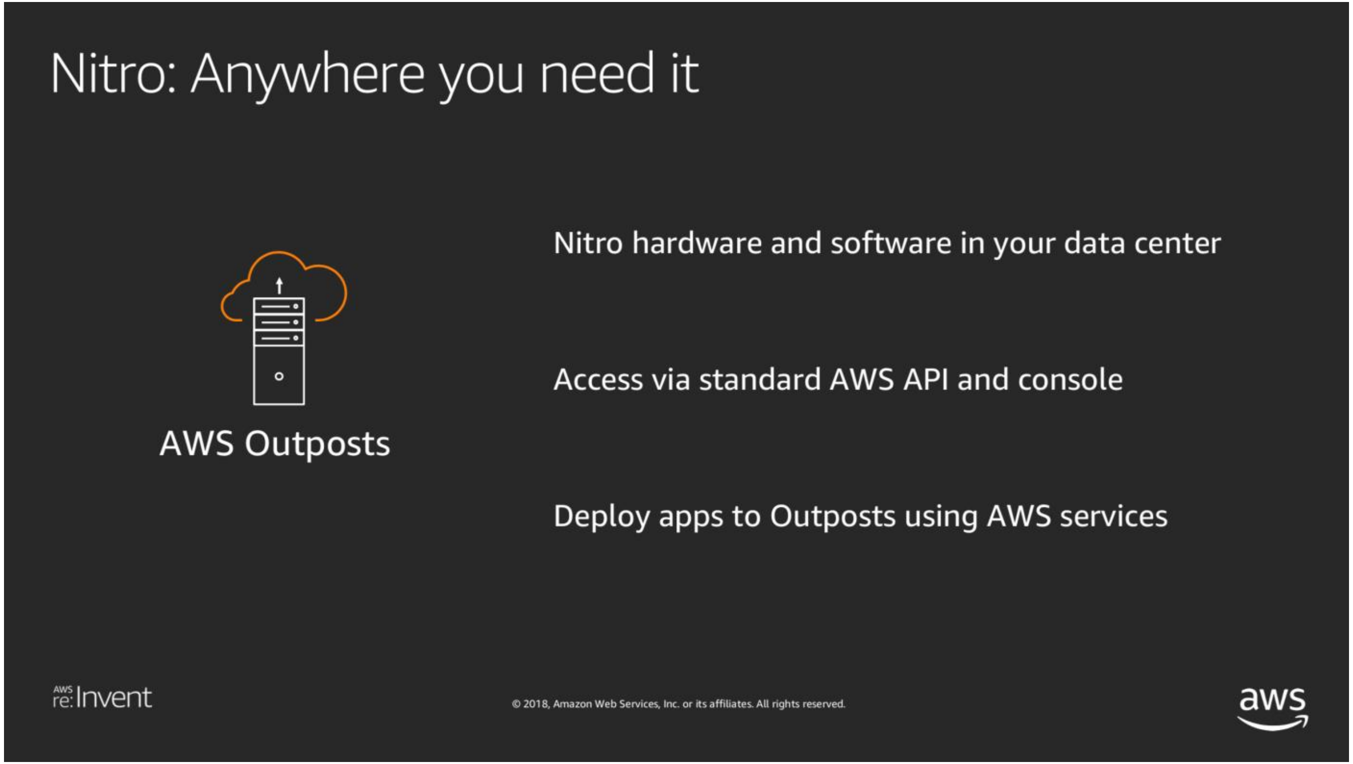
What’s a Snowball Edge?

- Storage Optimized Version
  - 100 TB of storage
  - 24 vCPUs
  - 1 TB SSD for pre-processing and large scale data transfer
- Compute Optimized Version
  - 52 vCPUs, an optional GPU
  - 7.68 TB NVMe SSD
  - 42 TB of storage for machine learning workloads
- EC2, Greengrass

https://ctovision.com/a-new-aws-snowball-edge-provides-the-power-of-the-cloud-in-disconnected-environments/

What is Outposts?

- 80” cabinet and smaller sizes shipped to customer DC within some latency threshold to an AWS AZ
- Control plane stays in AWS AZ
- Expected to offer EC2 and EKS, could offer Firecracker and Lambda
- Priced similar to reserved instances, but with equipment to return



Nitro: Anywhere you need it

AWS Outposts

Nitro hardware and software in your data center

Access via standard AWS API and console

Deploy apps to Outposts using AWS services

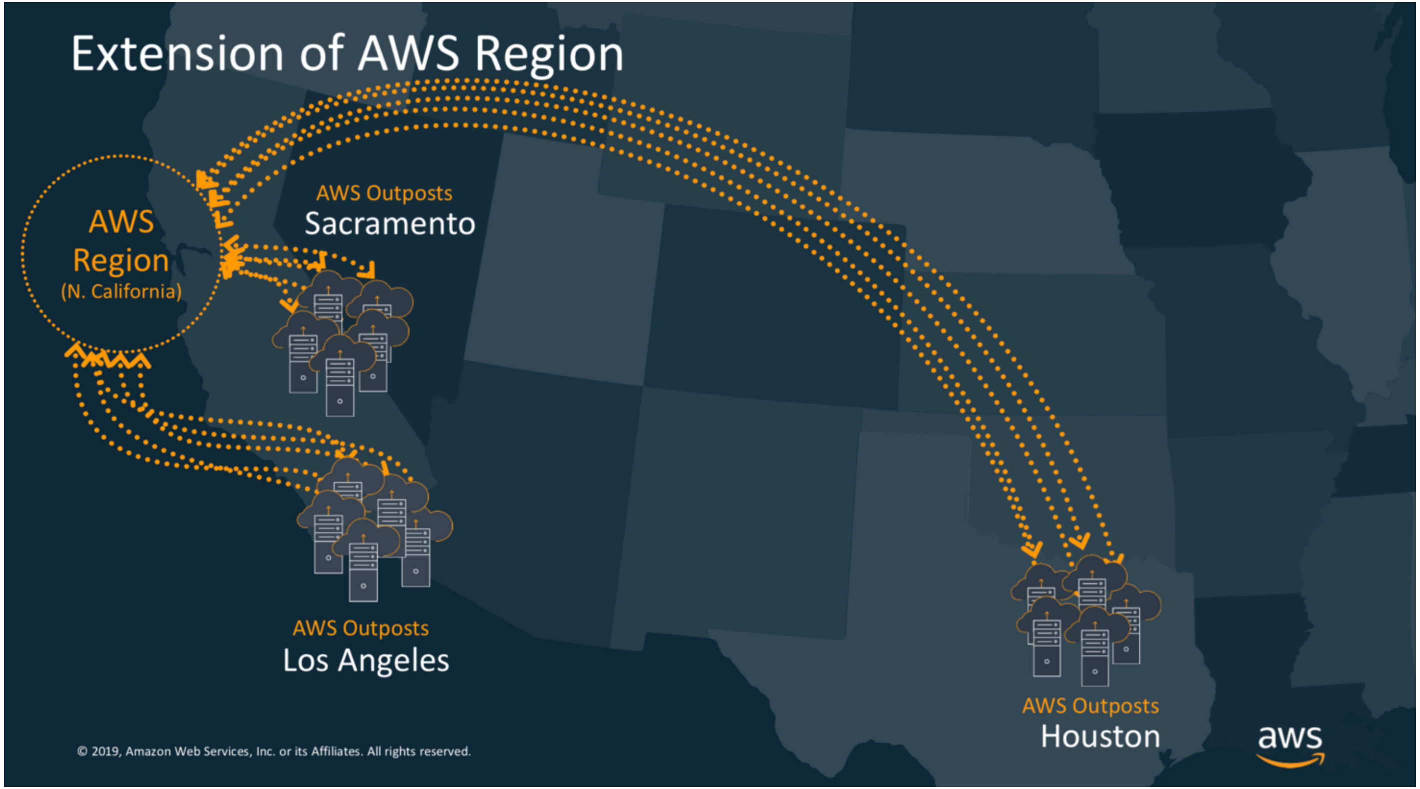
AWS re:Invent

aws

Shows up in AWS Console similar to an AZ

https://www.youtube.com/watch?v=e8DVmwj3OEs

Outpost’s Bet on Latency



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

https://pages.awscloud.com/Introduction-to-AWS-Outposts_2019_0319-CMP_OD.html

What’s Next? The Serverless Datacenter Race

FaaS on ☸

VS

AWS Outposts

AWS Snowball Edge

maturity

acceptance
and
latency

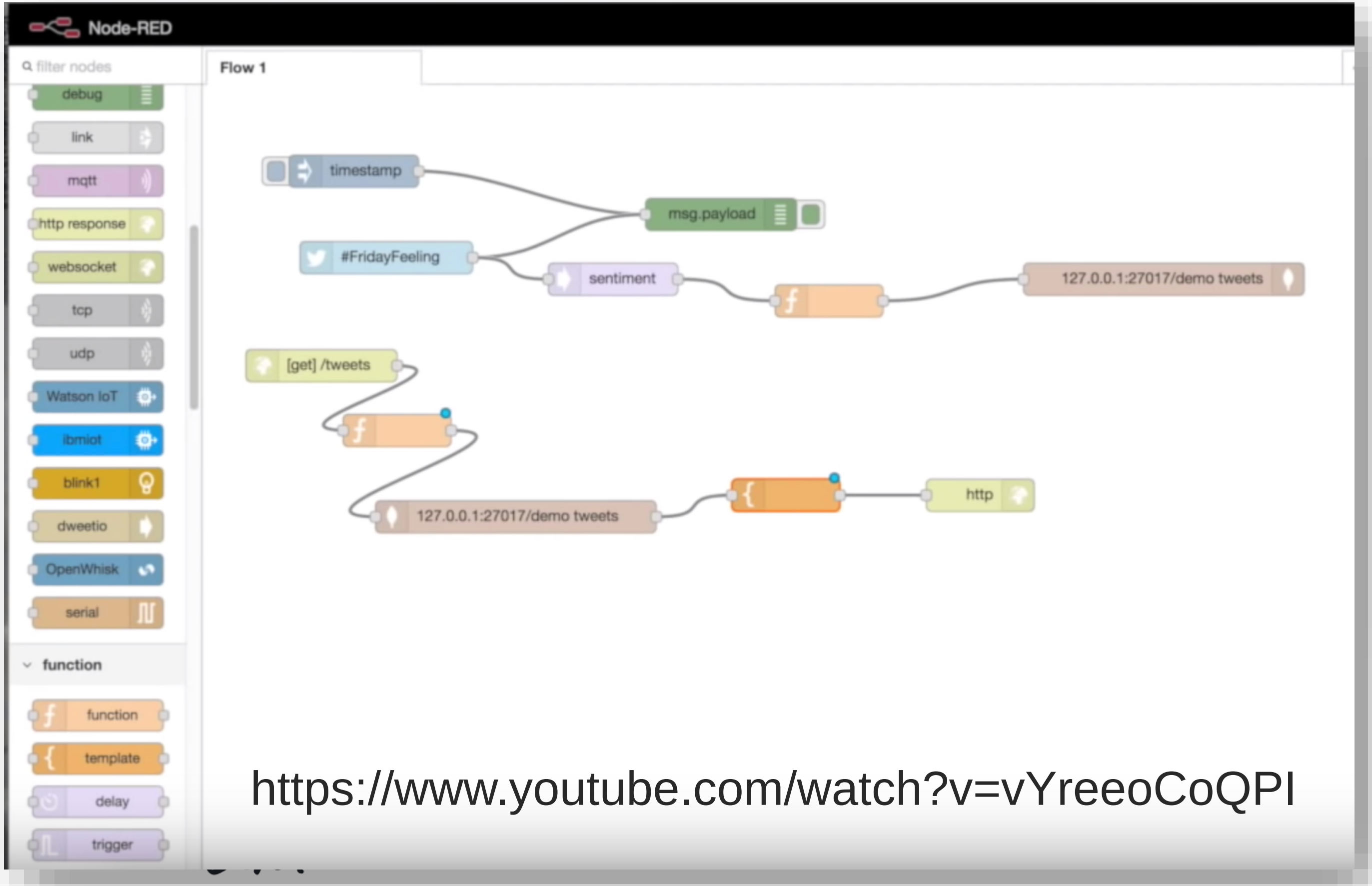
The Future

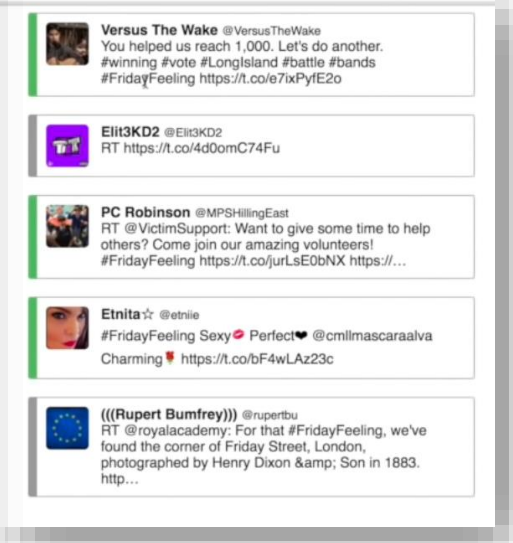
No Code

Demo #5:
Node-Red Video

No Code Movement: Beyond Serverless

Node-Red example (Linux Foundation Project, 1.0 release Sept 30, 2019)





Versus The Wake @VersusTheWake
You helped us reach 1,000. Let's do another. #winning #vote #LongIsland #battle #bands #FridayFeeling https://t.co/e7ixPyfE2o

Elit3KD2 @Elit3KD2
RT https://t.co/4d0omC74Fu

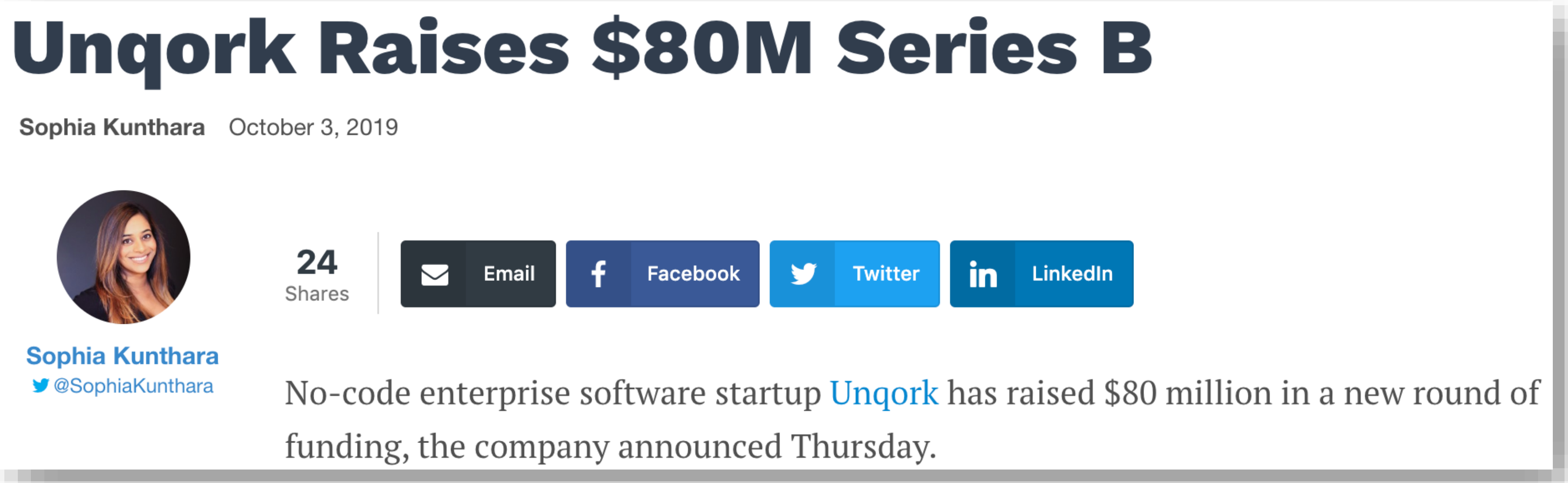
PC Robinson @MPSHillingEast
RT @VictimSupport: Want to give some time to help others? Come join our amazing volunteers! #FridayFeeling https://t.co/jurLsE0bNX https://…

Etnita☆ @etniie
#FridayFeeling Sexy💋 Perfect❤ @cmllmascaraalva Charming🌹 https://t.co/bF4wLAz23c

(((Rupert Bumfrey))) @rupertbu
RT @royalacademy: For that #FridayFeeling, we've found the corner of Friday Street, London, photographed by Henry Dixon &amp; Son in 1883. http…

https://www.youtube.com/watch?v=vYreeoCoQPI

Startup in the No Code Space



Unqork Raises $80M Series B

Sophia Kunthara October 3, 2019

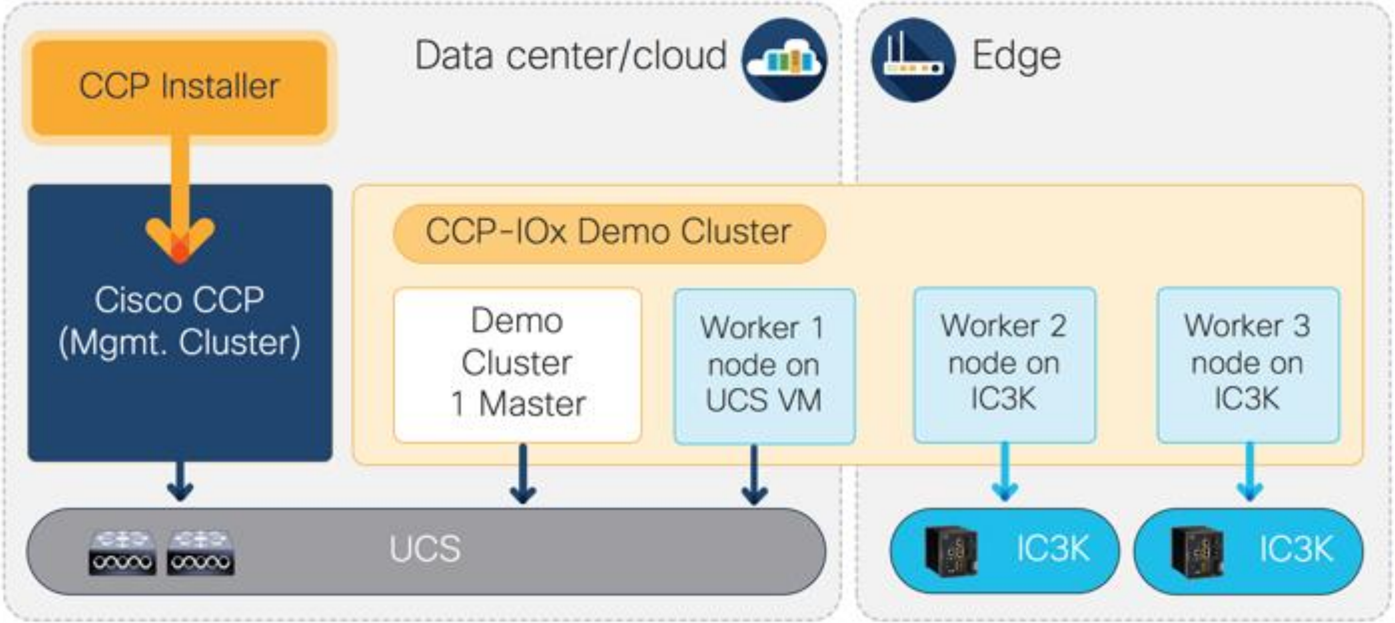
No-code enterprise software startup Unqork has raised $80 million in a new round of funding, the company announced Thursday.

https://news.crunchbase.com/news/unqork-raises-80m-series-b/

The Future

Edge Clusters

Cisco 10x – CCP Demo Setup



Complete your online session survey

- Please complete your session survey after each session. Your feedback is very important.
- Complete a minimum of 4 session surveys and the Overall Conference survey (starting on Thursday) to receive your Cisco Live t-shirt.
- All surveys can be taken in the Cisco Events Mobile App or by logging in to the Content Catalog on ciscolive.com/emea.

Cisco Live sessions will be available for viewing on demand after the event at ciscolive.com.

Continue your education

Demos in the Cisco Showcase

Walk-In Labs

Meet the Engineer 1:1 meetings

Related sessions

Thank you

You make **possible**