
AWS
re:Invent

CON209

Getting up and running with Amazon EKS

Mike Stefaniak
Senior Product Manager
AWS

Sudheer Adumulla
Senior Manager
Verizon

Vamsi Kalluri
DMTS
Verizon

AWS re:Invent

© 2020, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Agenda

EKS – Managed Kubernetes in 3 phases

EKS – Tools for getting started

Verizon – Who we are

Verizon – K8s migration strategy

Verizon – K8s platform challenges and tunings



© 2020, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Amazon EKS: A 3-phase story

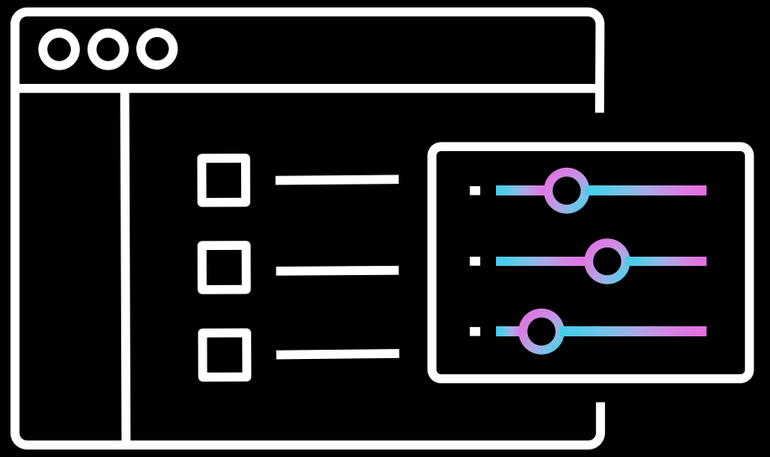
Phase 1

Management of the Kubernetes control plane



Phase 2

Management of the Kubernetes data plane



Phase 3

Management of operational add-ons

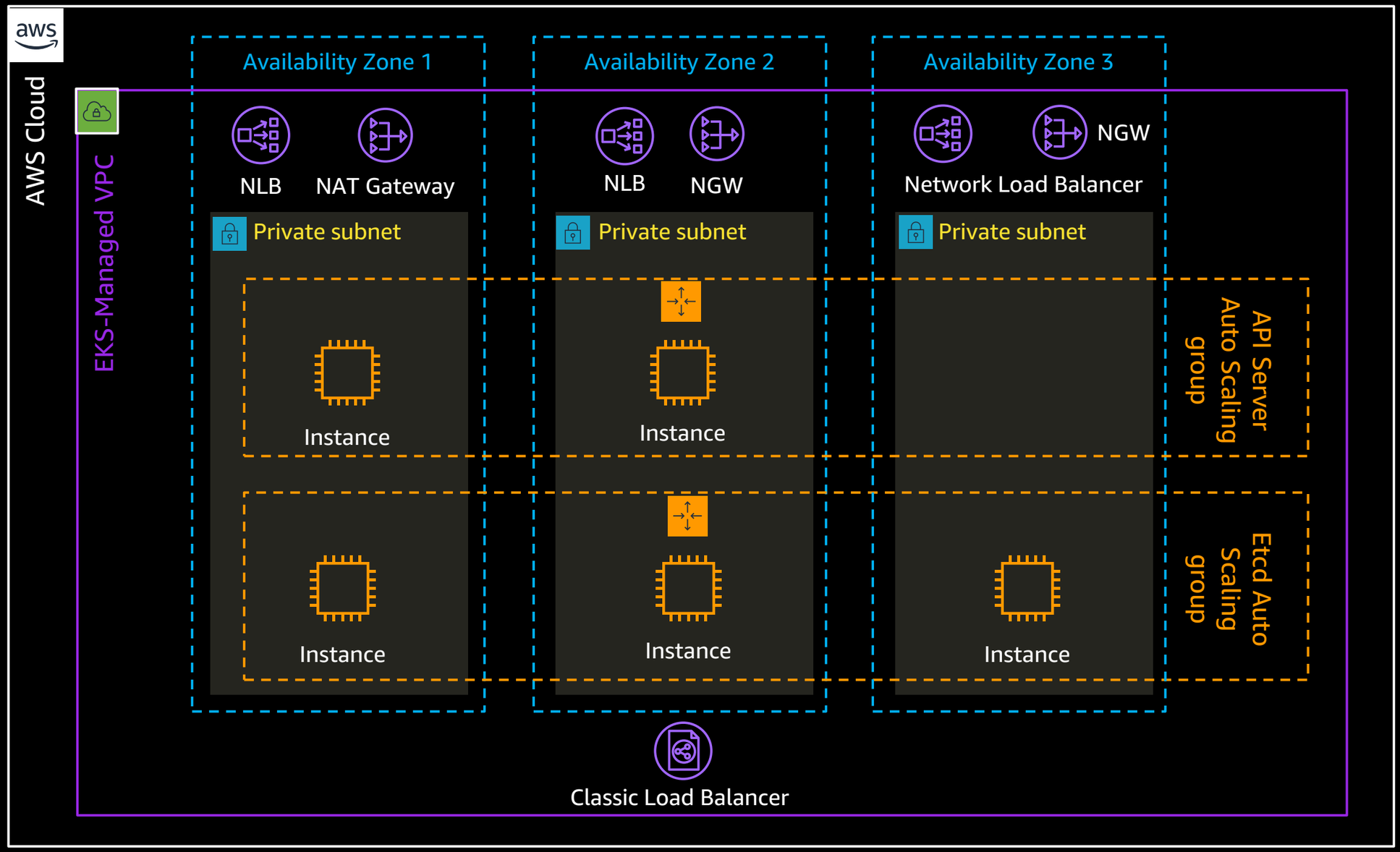
EKS Phase 1: Managed control plane



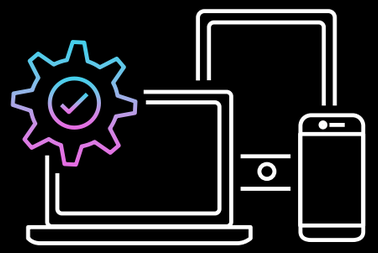
AWS re:Invent

© 2020, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Issue: Running a highly available control plane



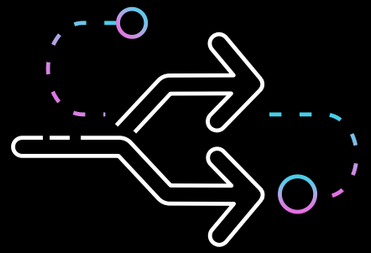
Managed control plane



Production workloads

Highly available, auto-scaled

99.95% Service Level Agreement



Native Kubernetes

Native and upstream conformant

Integration tests with community tooling

Seamless integrations

Monitoring

Logging

Ingress

Security

EKS Phase 2: Managed data plane



AWS re:Invent

© 2020, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Issue: Providing compute capacity for unpredictable workloads

Problem:

To run applications, you need to separately create and manage the nodes for a Kubernetes cluster. This includes understanding how to configure multiple AWS services such as Amazon EC2 and Auto Scaling groups, selecting EC2 instance types, and interacting with multiple AWS services to add nodes to a cluster.

You also need to manually update nodes to apply security fixes or to adopt new Kubernetes versions. To be efficient, you need to add or remove nodes in response to application scaling demands.

Amazon EKS managed node groups



Nodes with a single command

Use EKS Console, APIs, eksctl, CloudFormation, or Terraform

Add Kubernetes labels directly to nodes



Stay up to date

Launch with the latest EKS-optimized AMIs

Get the latest updates with a single command and rolling updates



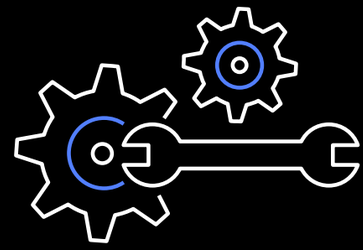
High availability

Run across multiple Availability Zones

Tagged for cluster autoscaler discovery

Integrated configuration health checks

Nodes automatically cordon and drain



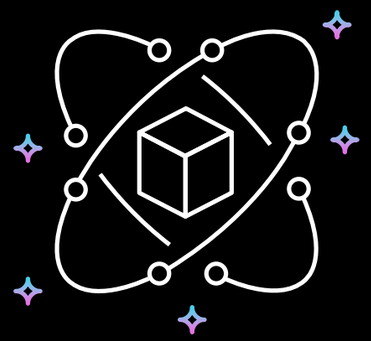
Customizable and cost effective

Use EC2 launch templates to modify worker nodes

Bring your own OS with custom AMI support

Fully integrated with EC2 Spot

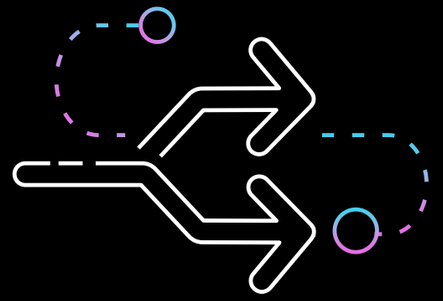
Amazon EKS on AWS Fargate



Bring existing pods

You don’t need to change your existing pods

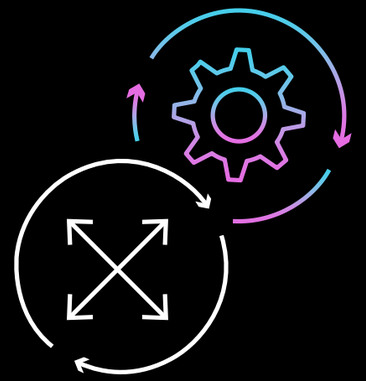
AWS Fargate works with existing workflows and services that run on Kubernetes



Production ready

Launch pods quickly, and easily run pods across multiple Availability Zones for high availability

Each pod runs in an isolated compute environment

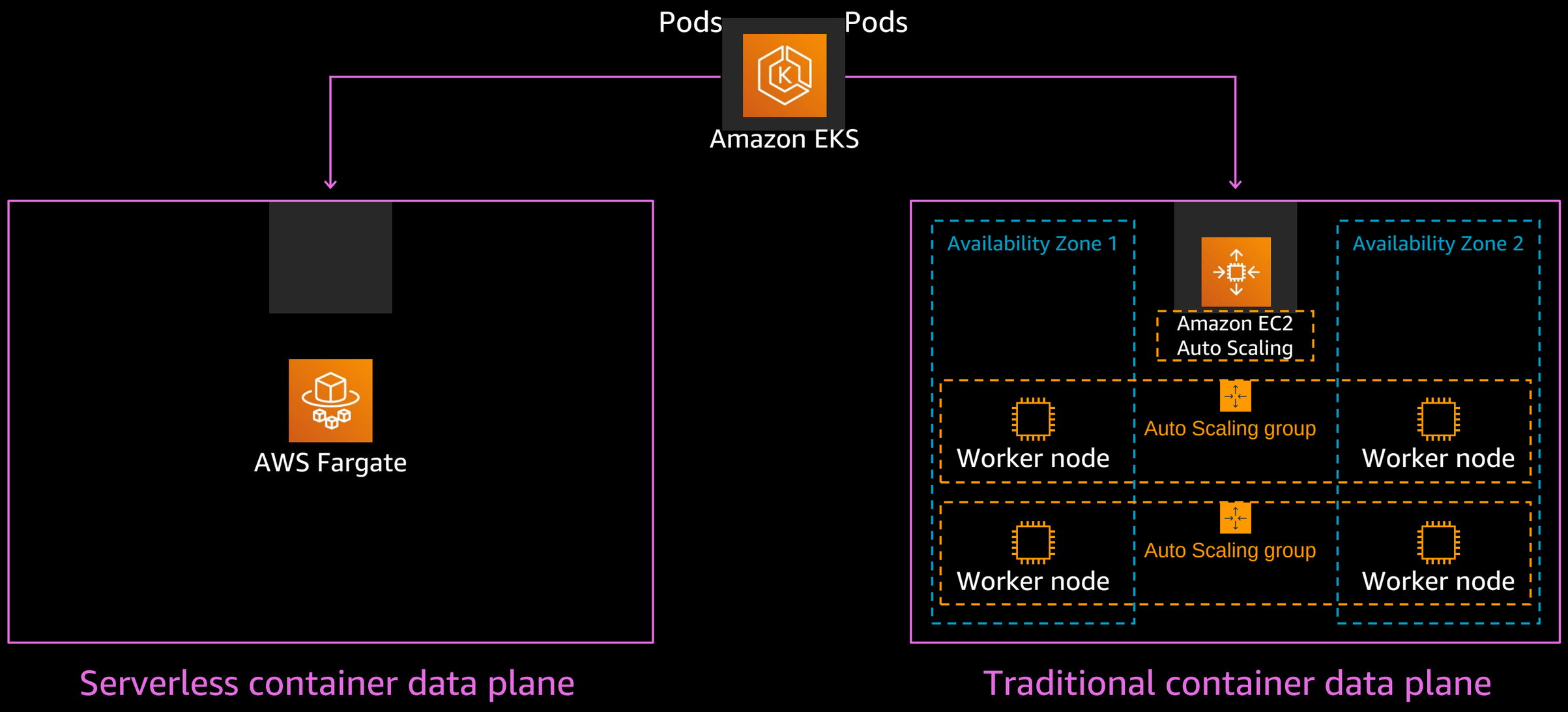


Right-sized and integrated

Only pay for the resources you need to run your pods

Includes native AWS integrations for networking, security, storage, and logging

EKS data plane options: Mix and match



Serverless container data plane

Traditional container data plane

EKS Phase 3: Add-ons



AWS re:Invent

© 2020, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Issue: Production workloads require operational tooling

Problem:

You need to install the prerequisite operational software for a cluster before running applications. This can include logging, storage, networking, observability, and scaling add-ons. Curating the catalog of operational software is time-consuming, and keeping it all up to date requires a lot of work.

EKS add-ons



One-click management

Install desired add-ons as part of cluster creation

Upgrade cluster control plane, worker nodes, and add-ons directly through EKS API



Secure and reliable

All software includes the latest security patches and bug fixes, and is validated by AWS to work with EKS

Specify fine-grained IAM role for each add-on



Flexible

Configure add-ons to meet your requirements

Networking now; storage, ingress, scaling, logging, and GitOps in the future

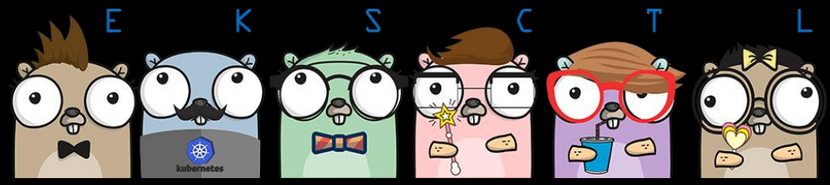
Tools for getting started



AWS re:Invent

© 2020, Amazon Web Services, Inc. or its affiliates. All rights reserved.

eksctl: The official CLI for EKS





One-click cluster creation

Automatically provision networking and IAM prerequisites

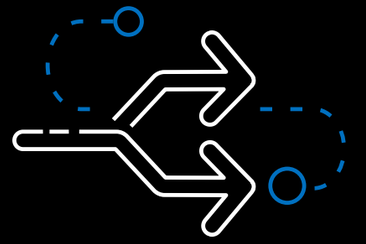
Start a cluster with managed node groups or Fargate



Utility commands

Easily apply IAM Roles to service accounts

Keep cluster add-ons up to date



GitOps enabled

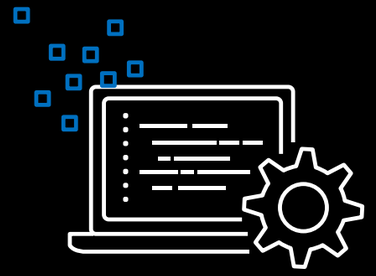
Use Git as the single source of truth for Kubernetes resources

Begin with Quick Start profiles and tailor to your needs

```
eksctl create cluster
```

Cloud Development Kit for Kubernetes

cdk8s



Go from code to config

Define Kubernetes applications and architectures using familiar programming languages

Generate well-formatted YAML for your applications every time



Define best practices

Compose and share reusable components through code libraries

Use the same tools to build your application and its configuration



Run everywhere

cdk8s runs locally and generates YAML you can deploy to any cluster anywhere

Standardize across on-premises and any cloud

How Verizon runs its core applications on Amazon EKS at scale



AWS re:Invent

© 2020, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Verizon: Who we are

Verizon is one of the world's leading providers of technology, communications, information, and entertainment products and services. The company offers data, voice, and video services and solutions on its award-winning networks and platforms, delivering on customers' demand for mobility, reliable network connectivity, security, and control.

Verizon was the first company in the world to launch a commercial 5G mobile network with a commercially available 5G-enabled smartphone. The company's operating structure focuses on three customer-facing areas: consumer, business, and media.



© Verizon 2020, All Rights Reserved. Information contained herein is provided AS IS and subject to change without notice. All trademarks used herein are property of their respective owners.

verizon

NorthStar architecture

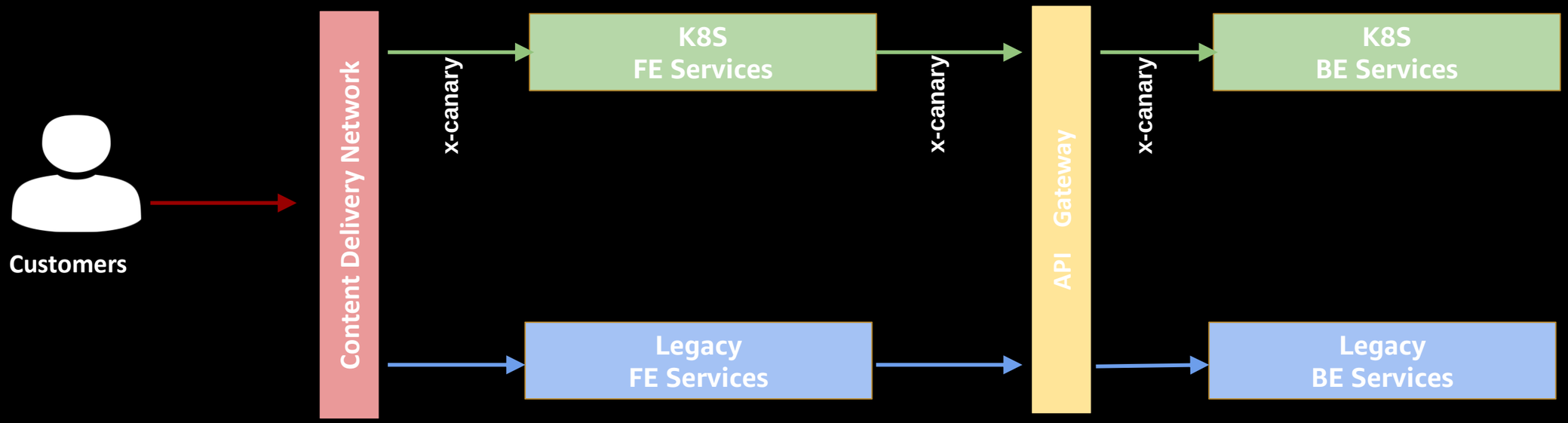
- Common digital experience platform
- Personalized omni-channel experience
- Customer-centric over channel-centric
- Enable consistent, reusable services
- API-first approach with reusable design patterns
- Standard tech stack for interchangeable resource pool
- DevSecOps pipelines and process
- Standardized monitoring at each level – Infra/App
- AIOps by design



© Verizon 2020, All Rights Reserved. Information contained herein is provided AS IS and subject to change without notice. All trademarks used herein are property of their respective owners.

verizon

K8S migration using canary solution



- Canary request header inserted on CDN engine based on user location
- API Gateway is used for routing to Legacy versus K8S based on header
- K8S traffic enabled in phased approach inserting canary header
- Canary header passed along through transaction lifecycle for routing

© Verizon 2020, All Rights Reserved. Information contained herein is provided AS IS and subject to change without notice. All trademarks used herein are property of their respective owners.

verizon

K8S platform challenges

Cluster scaling and pod performance

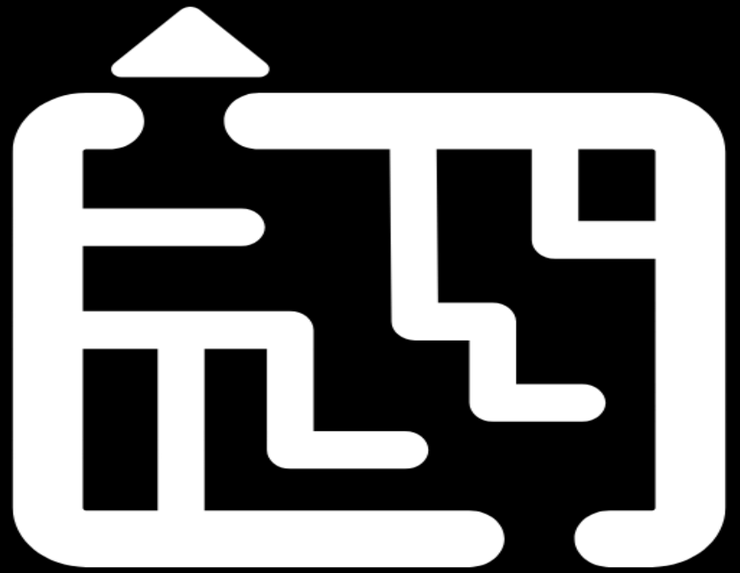
- Monolithic to microservice migration
- Pod resources tuning
- Data plane scaling
- Control plane scaling
- Interservice calls load balancing

Log shipping

- Log shipping agent
- Log buffer and stream
- Log shipping to multiple repositories
- Log repository scaling

Network

- Capacity planning for subnets
- DNS queries latency
- Ephemeral port usage
- Load balancer scaling



© Verizon 2020, All Rights Reserved. Information contained herein is provided AS IS and subject to change without notice. All trademarks used herein are property of their respective owners.

verizon

K8S platform tunings – Cluster

Cluster builds and CI/CD pipelines

- Automate cluster builds and operations
- High available repositories
- Security vulnerability scans
- K8S native services (ConfigMaps, secrets)



Cluster scaling and performance

- Cluster over provisioner
- Elastic worker nodes using cluster autoscaler
- Actuator tunings for self-healing
- Control plane upsizing

Interservice calls tunings

- Use IPVS instead of iptables
- Pod anti-affinity for deployment

© Verizon 2020, All Rights Reserved. Information contained herein is provided AS IS and subject to change without notice. All trademarks used herein are property of their respective owners.

verizon

K8S platform tunings – Network

K8S network tunings

- Right sizing and dedicated subnets
- Node local DNS
- UDP protocol for DNS queries
- CNI config to use random ephemeral ports
- Prefetch and cache DNS queries



Load balancer tunings

- Target type : IP (resource limit issue)
- Target registration and deregistration
- LB LCU and access logs
- Service-level health checks

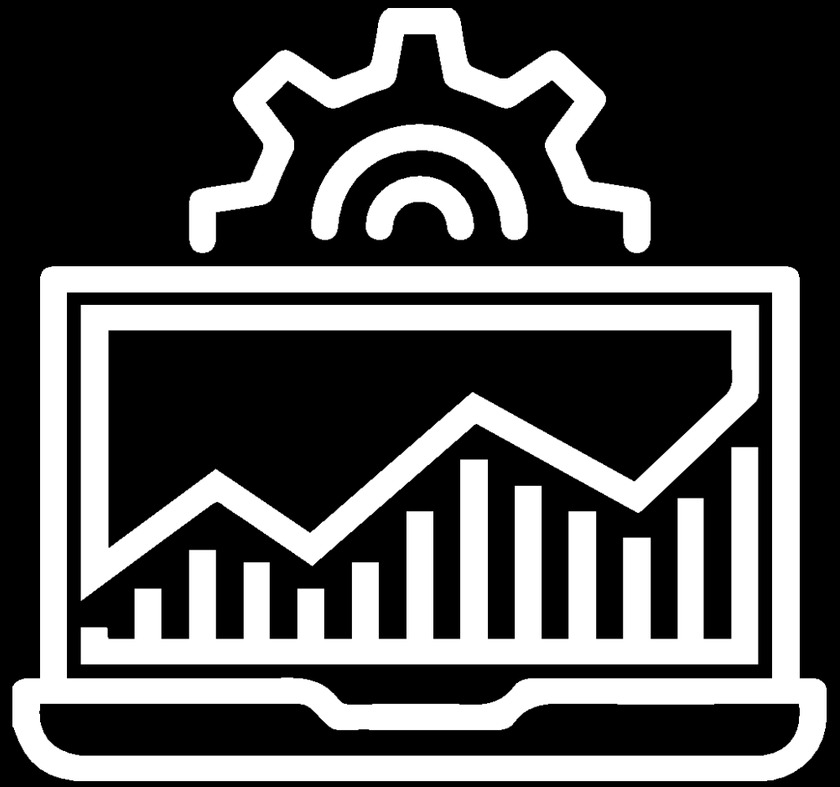
© Verizon 2020, All Rights Reserved. Information contained herein is provided AS IS and subject to change without notice. All trademarks used herein are property of their respective owners.

verizon

K8S platform tunings – Logging and Monitoring

Log shipping tunings

- Fluent Bit for shipper
- Fortify Elasticsearch cluster
- Tune log indexes and archiving policies
- UltraWarm for Amazon Elasticsearch Service
- Docker log size to 1024M



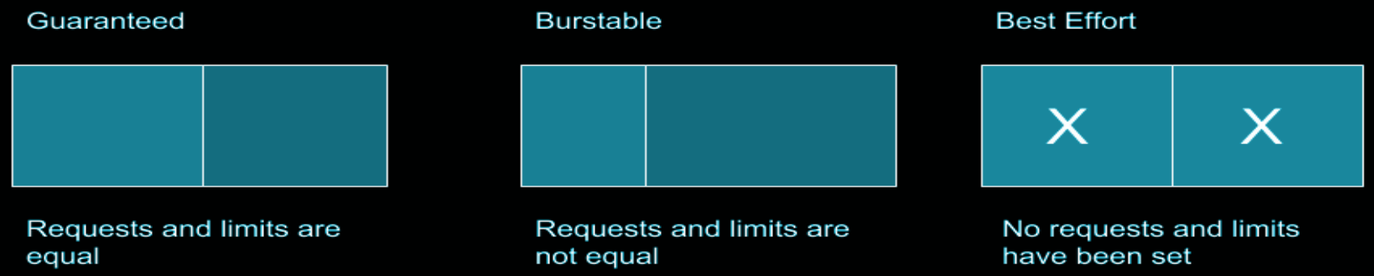
Monitoring

- Control plane and core component monitoring
- Services monitoring and alerting
- Centralized dashboard for cluster and services

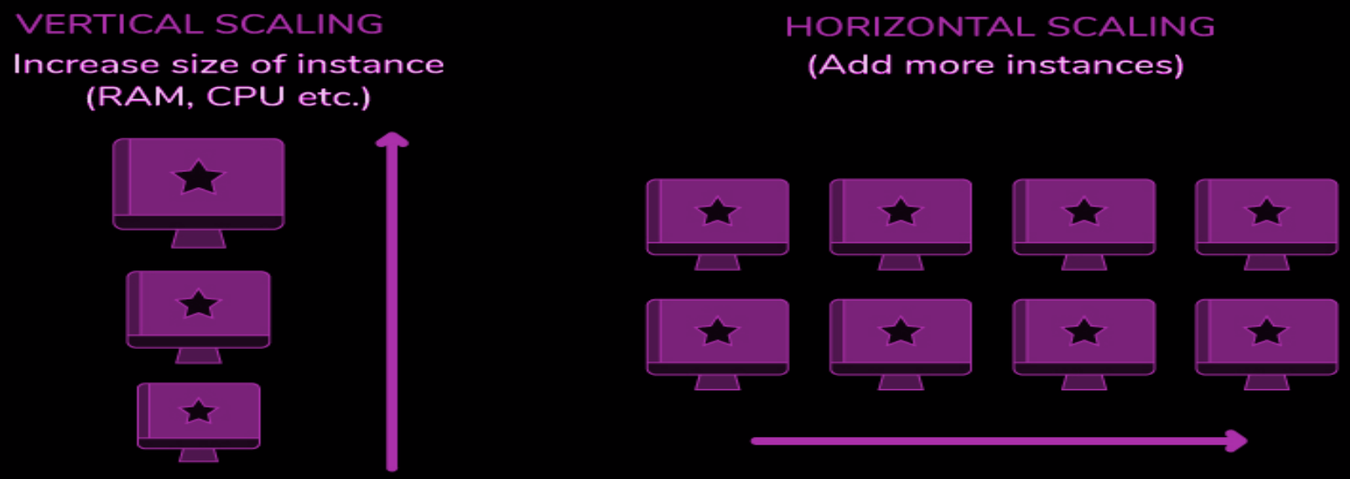
© Verizon 2020, All Rights Reserved. Information contained herein is provided AS IS and subject to change without notice. All trademarks used herein are property of their respective owners.

verizon

K8S platform tunings – Pods and Nodes



- Guaranteed instead of burstable
- Increase pod size and decrease count
- Pre-provision capacity for flash traffic
- Pod memory tunings and HPA



© Verizon 2020, All Rights Reserved. Information contained herein is provided AS IS and subject to change without notice. All trademarks used herein are property of their respective owners.

verizon

Conclusion

Upskill team on containers and K8S

Optimize application for containers

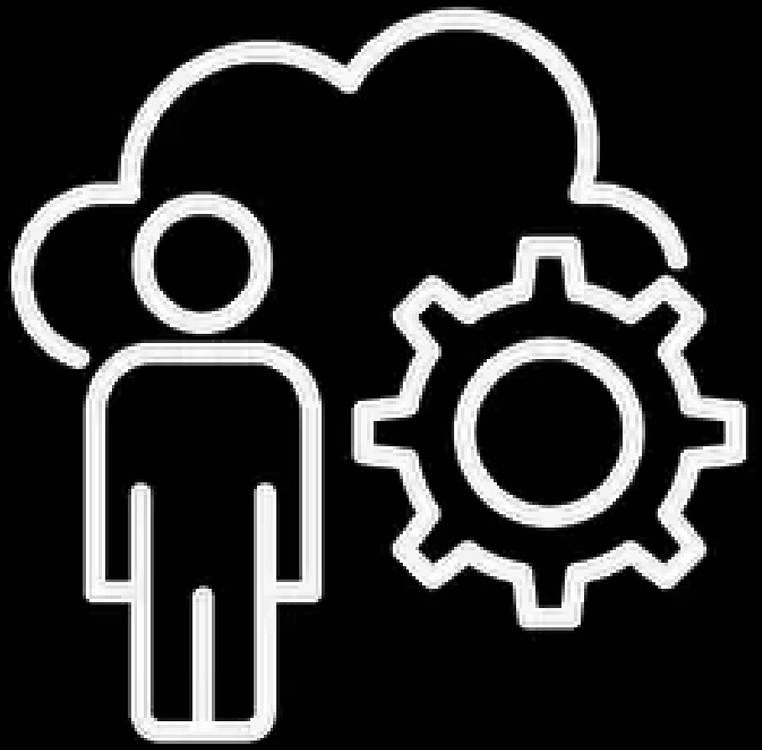
Security for container build and runtime

Automate app and infra deployments

Tune K8S cluster to fit in ecosystem

Comprehensive monitoring for platform

Exhaustive logging and dashboards for platform



© Verizon 2020, All Rights Reserved. Information contained herein is provided AS IS and subject to change without notice. All trademarks used herein are property of their respective owners.

verizon

Thank you!



AWS re:Invent

© 2020, Amazon Web Services, Inc. or its affiliates. All rights reserved.



Please complete the session survey

AWS re:Invent

© 2020, Amazon Web Services, Inc. or its affiliates. All rights reserved.