12/2023 DOD Weapons System Software Summit

AWS, Kubernetes, and the Disconnected Edge.

Designing DDIL-Resilient Systems with Kubernetes and AWS Managed Edge Devices

RAFT

Vice President of Platform

www.teamraft.com

Dagan Henderson

- Software Engineer for 25+ years
- Developed commercial/government software in 10+ languages (Go, Java, Swift, Objective-C, C#)
- 10+ years AWS experience
- Occasional Security Researcher (4 CVEs)
- 2 software patents

Mission Statement

Develop DDIL-Resilient *Everywhere* Platform

- Hybrid cloud extends regional systems to users on-premises and at the edge.
- Centralized DevSecOps to deploy apps *from* anywhere *to* anywhere.
- Increased application uptime and shorter feedback-to-feature lifecycle.

Bottom Line Up Front

Reliable On-Premises and Edge Operations During DDIL

- Operational stability during **200+ hours of fully disconnected operations** with both AWS Outpost and Snowball Edge.
- DDIL scenarios:
  - Degraded network connectivity (moderate bandwidth, high latency)
  - Severely degraded network connectivity (low bandwidth, very high latency, packet loss)
  - Fully disconnected network link
- Tested Architectures:
  - Outpost with in-region services (EKS and AutoScaling Groups)
  - Outpost without in-region services (EC2 + EBS only)
  - Snowball Edge / Outpost + Snowball Edge

Bottom Line Up Front

Reliable On-Premises and Edge Operations During DDIL

	Degraded	Severely Degraded	Disconnected
In-Region EKS + Outpost	❌	❌	❌
EKS-D + Outpost	✅	✅	✅
EKS-D + Snowball Edge	✅	✅	✅
EKS-D + Outpost + Snowball Edge	✅	✅	✅

- Much of the benefit of “cloud” comes from the managed services, which are generally lost if DDIL-resiliency is required.
- Even so, the operational efficiencies of hybrid cloud and the inherent benefits of managed hardware make the devices attractive replacements for traditional on-premises and edge hardware.

Hybrid Cloud for Contested Environments

Operational Efficiencies

- DevSecOps from the cloud to the edge delivers new features and bug fixes faster
- A uniform application platform simplifies deployments and enables edge-to-cloud monitoring
- Vendor-owned, vendor-managed hardware offers flexibility

Benefits to the Warfighter

- Faster feedback-to-feature lifecycle
- Problems can be detected and addressed before they cause outages
- Increased application up times

Hybrid Cloud with AWS Managed Devices

“One accurate measurement is worth a thousand expert opinions.”

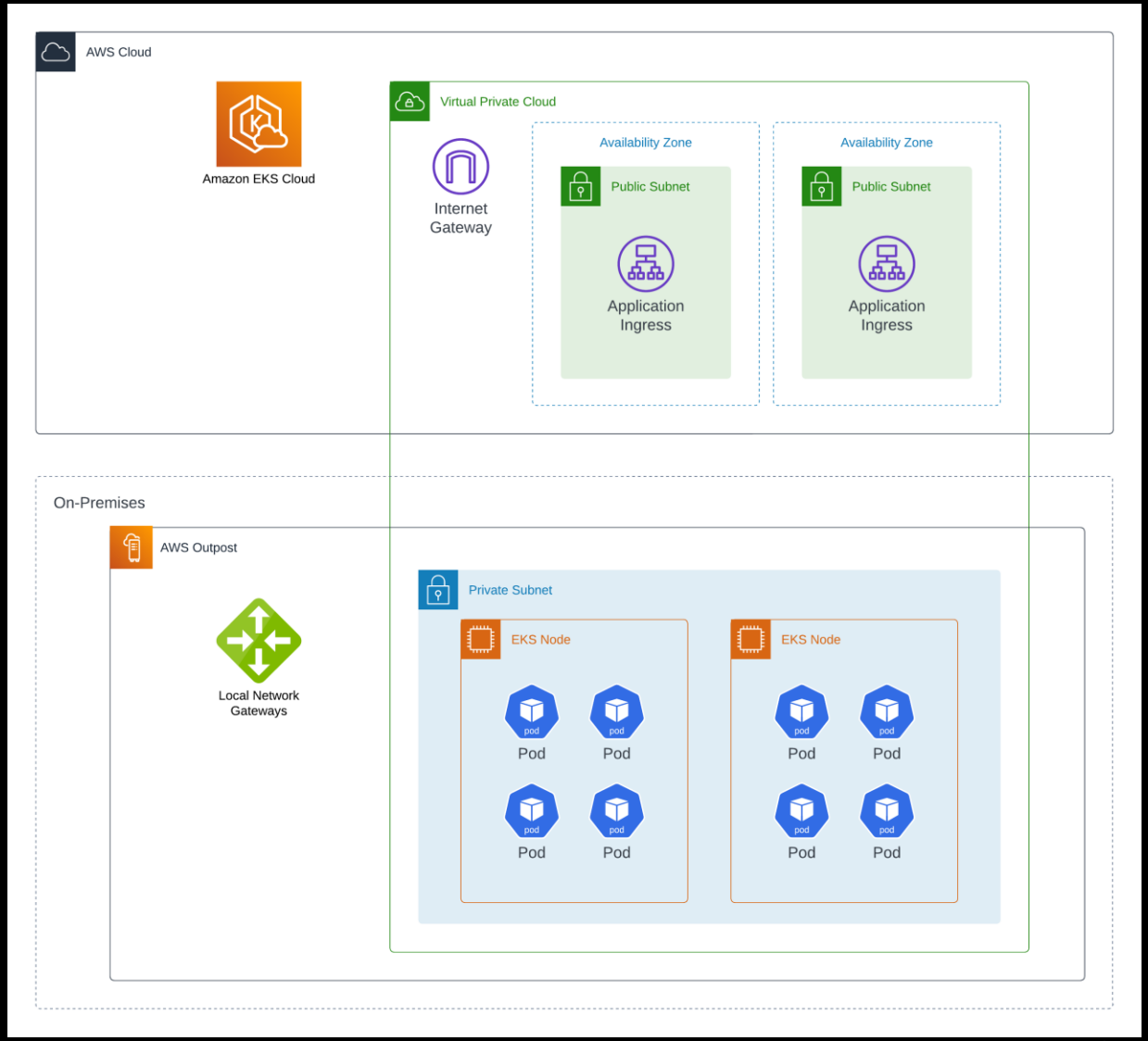
—Rear Adm. Grace Hopper, United States Navy

Scenario 1: EKS on AWS Outpost

Scenario 1

EKS on Outpost

- Simple architecture extends in-region EKS onto Outpost
- Expected all pods to be rescheduled following disconnect > 5 min
- Lack of control plane availability during a full disconnect was a concern



Scenario 1

EKS on Outpost: Degraded Network

- Control plane reconciliation loops continued
- Kubelet maintained communication with control plane
- KubeProxy continued to function
- Pods and PVs remained operational
- New EC2 instances could not be provisioned due to AMI constraints
- CloudWatch Metrics and IAM fail under high latency
- IAM failures eventually lead to loss of EKS nodes at the control plane, as well as CSI and CNI failures

Scenario 1

EKS on Outpost: Disconnected Network

- Loss of control plane caused issues with Istio and any workload relying on the API server (e.g., Leases)
- Pods and bound/attached PVs remained operational
- KubeProxy continued to function locally
- CoreDNS continued to work
- Self-healing was unavailable
- Failing pods were not removed from Service Endpoints

Scenario 1

EKS on Outpost: Disconnected Network (Cont.)

- Following short disconnects (5 mins–1 hour), pods were rescheduled due to node eviction
- Following longer disconnects (> 1 hour), catastrophic failures were observed:
  - Nodes were not able to immediately rejoin the cluster
  - EBS volumes could not be reliably detached/reattached as pods were rescheduled
  - New EC2 instances could not be provisioned for approx. 60 min
  - Autoscaling Groups observed terminating functioning instances
  - AWS services can take up to 120 min to fully recover

Lessons Learned

Do NOT Lose the Kubernetes Control Plane

- Self-healing and Service Discovery are immediately impacted
- Lease resources will expire and coordinated Services will fail after ~7.5 sec
- Operators will fail to function, including CoreDNS
  - The version of CoreDNS we tested with (1.8.7-eksbuild.1) does not register an error handler with the Kubernetes client, so the ListAndWatch loop fails silently. Cached records go stale but remain accessible.
  - According to CoreDNS documentation, stale records should result in NX domain errors after 30 seconds.
- Istio Proxy intermittently introduces ~4 seconds of latency
  - By default, Istio certificates are valid for 24 hours, and Istio Proxy attempts to renew certificates when they are halfway between issuance and expiration
  - A bug in Istio Proxy introduces ~4 seconds of latency in some requests while training to renew certificates
  - The added latency was observed within the first 12 hours of disconnect and persisted until new certificates were issued

Lessons Learned

Do NOT Lose ~~the Kubernetes~~ *Any* Control Planes

- During full disconnect and severe latency (>400ms) IAM and CloudWatch Metrics fail
- IAM and CloudWatch failures cascade into failures in other AWS services (e.g., Autoscaling Groups, EKS, EBS, VPC, SSM, etc.)
- AWS service failures begin affect workloads, leading to catastophic operational impacts

A New Architecture

From Our Lessons Learned

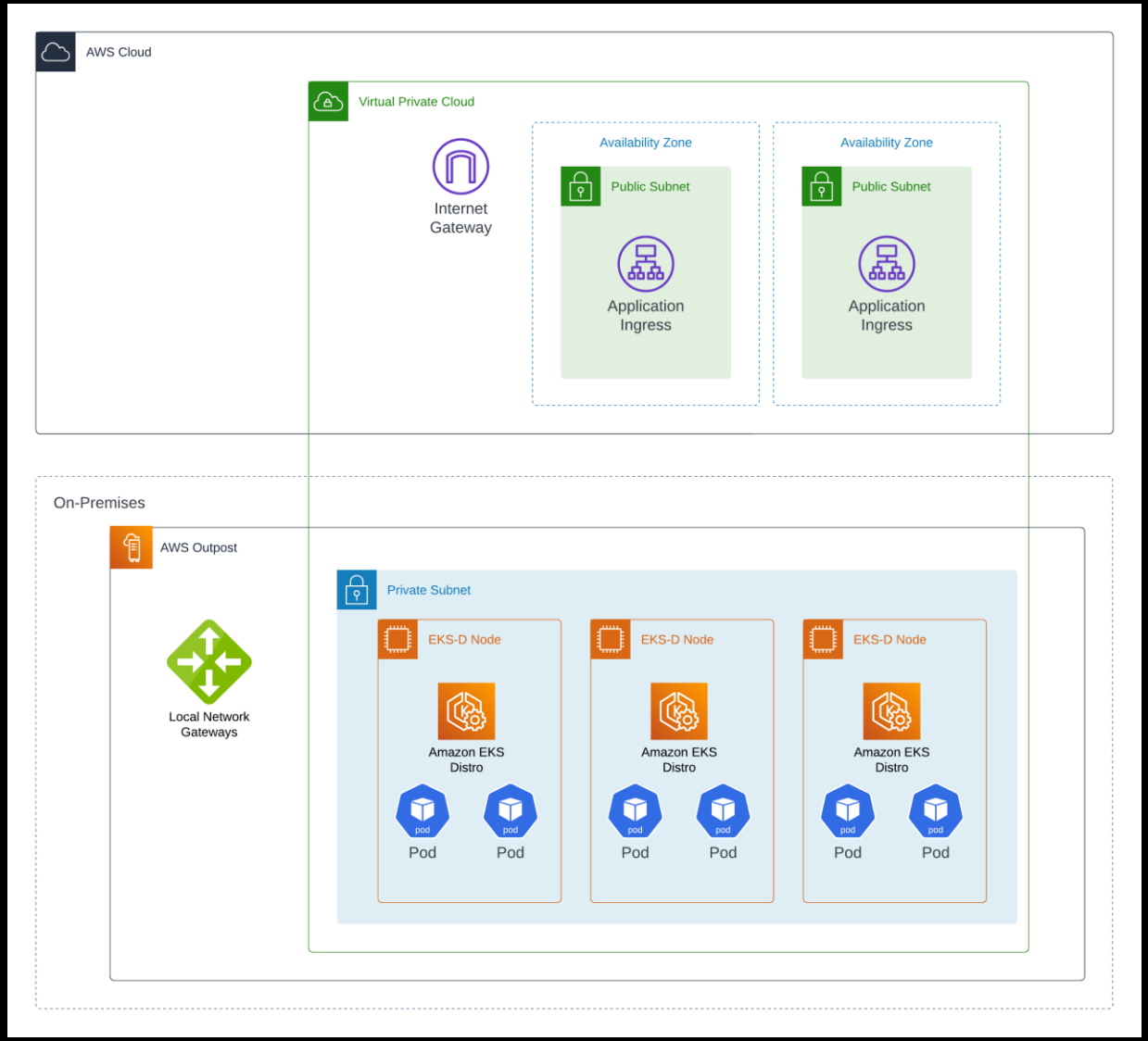
- Local Kubernetes control plane (EKS-D)
- Cilium for overlay CNI
- Large per-node EBS volumes managed by Rook Ceph:
  - Block storage with failure zones managed via CSI
  - Object storage managed via Rook Ceph resources
- Per-cluster image registry
- Inwardly cascading observability
- Fully automated cloud-based deployment and monitoring

Scenario 2: EKS-D on AWS Outpost

Scenario 2

EKS-D on Outpost

- EC2 instances provisioned *without* Autoscaling Groups
- Secondary EBS volumes added to each node for Rook Ceph-managed PVs
- EKS-D built into AMIs for automated-provisioning via GitLab pipelines

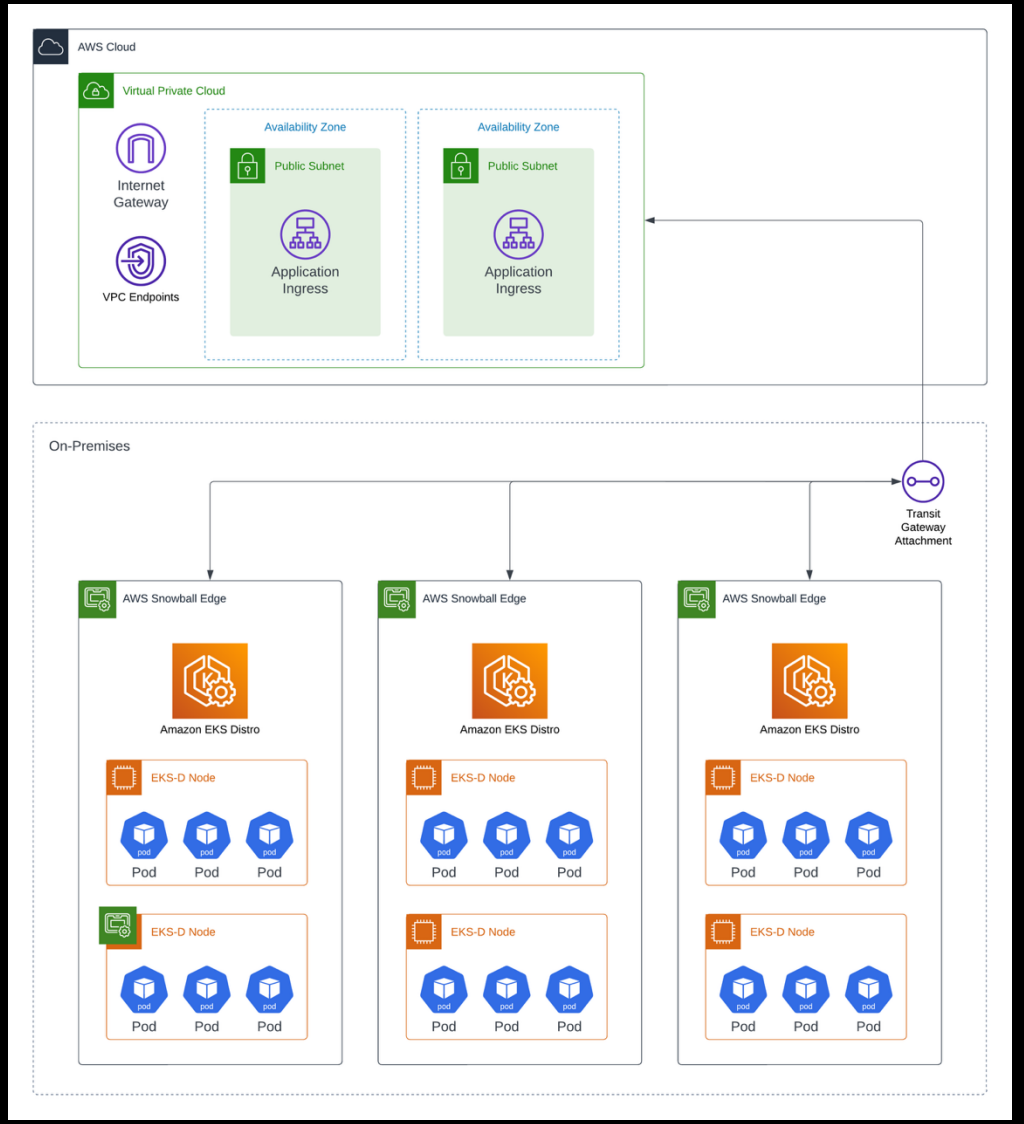


Scenario 3: EKS-D on AWS Snowball Edge

Scenario 3

EKS-D on Snowball Edge

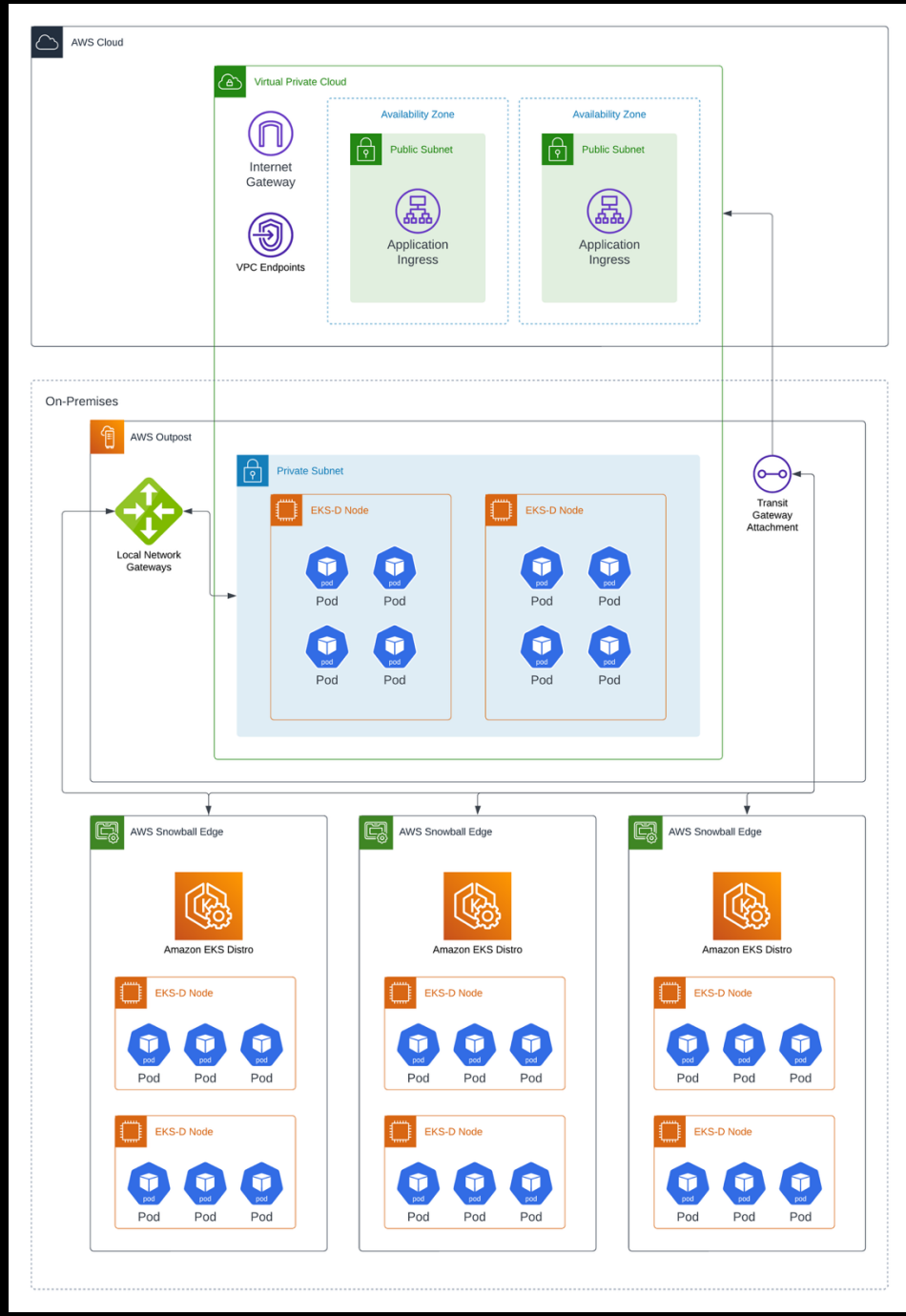
- Transit Gateway connects on-premises network to VPC
- EC2 instances provisioned on Snowball Edge
- Secondary EBS volumes added to each node for Rook Ceph-managed PVs
- EKS-D built into AMIs for automated-provisioning via GitLab pipelines



Scenario 4

EKS-D on Snowball Edge + Outpost

- EC2 instances provisioned on Outpost without AutoScaling Groups, joined to SBE-cluster
- EKS-D control plane hosted on Snowball Edge



Scenarios 2-4

EKS-D on Outpost + SBE: Degraded Network

- Control plane reconciliation loops continued
- KubeProxy continued to function
- Kubelet maintained communication with control plane
- Pods and PVs remained operational
- New EC2 instances could not be provisioned due to AMI constraints
- CloudWatch Metrics and IAM fail under high latency
- No observed **operational** impact

Scenarios 2-4

EKS-D on Outpost + SBE: Disconnected Network

- Control plane reconciliation loops continued
- KubeProxy continued to function
- Kubelet maintained communication with control plane
- Pods and PVs remained operational
- EC2 instances and attached EBS volumes unaffected
- No observed operational impact

Planning *Your* Edge

Deciding What Is Right for You

- If you must mitigate prolonged network uplink degradation and/or disconnects of any duration, do not rely on in-region services (EKS, RDS, SSM, etc.).
- If your site and network uplink(s) permit, deploy *either* (3) logical Outposts across (3) AWS Availability Zones *or* (2) logical Outposts across (2) AWS Availability Zones and (1) Snowball Edge.
- If your site or network uplink(s) cannot accommodate Outpost or you do not have sufficient resource demands, use (3) or more Snowball Edges with hardware failure mitigations.

Mitigating Snowball Edge Hardware Failures

- Snowball Edge is rugged and tamper resistant.

  It is not designed for datacenter deployments.

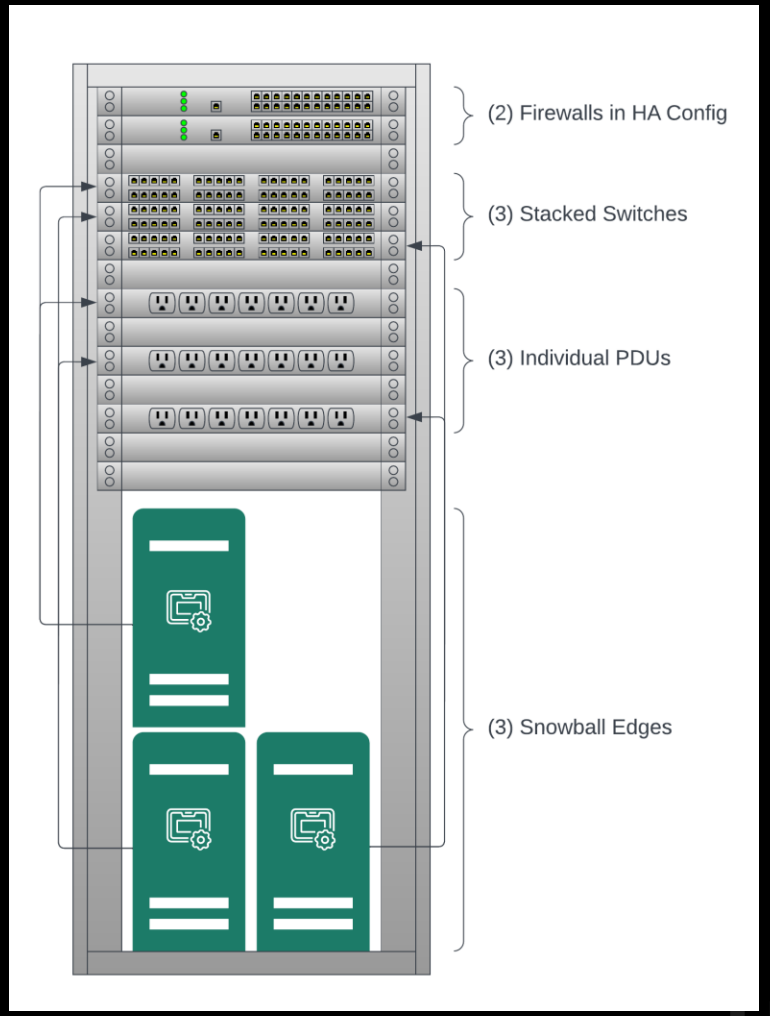
- No redundant power supply.

- NICs do not support link aggregation.

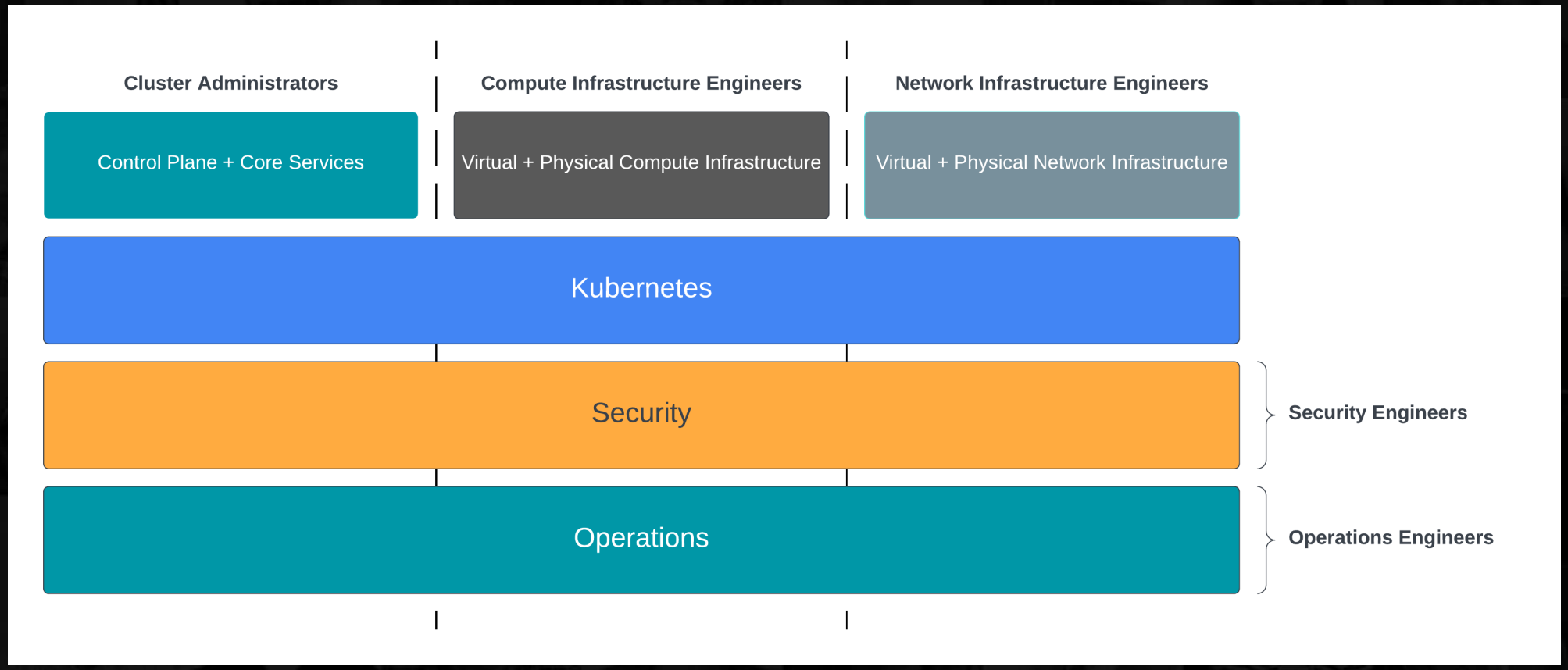
- No internal hardware redundancy (i.e., no

  RAID, etc.)

- (3) Stacked switches and (3) ATS PDUs protect against upstream hardware failures
- Span Rook-Ceph failure zones across SBEs

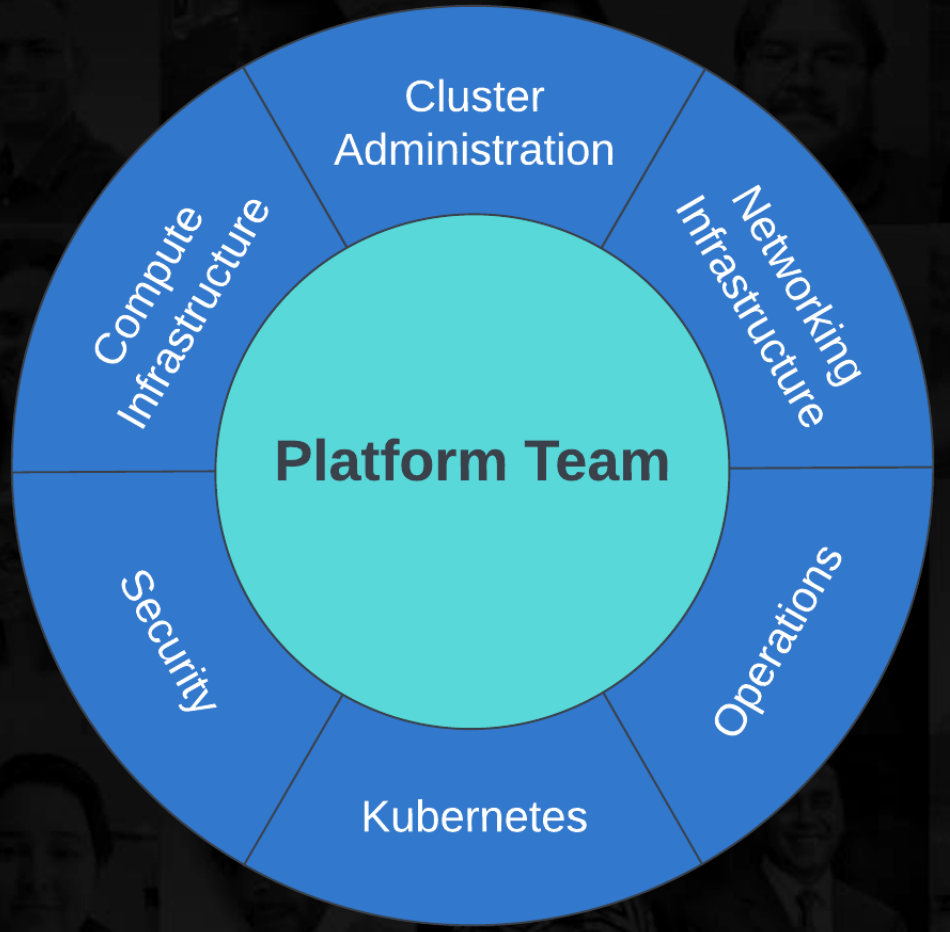


Platform Team Skillset



Platform Team Composition

- Centralized team of highly skilled, multi-disciplined engineers
- Each discipline includes Kubernetes, Security, and Operations
- The team scales through large/multiple Security and Operations teams

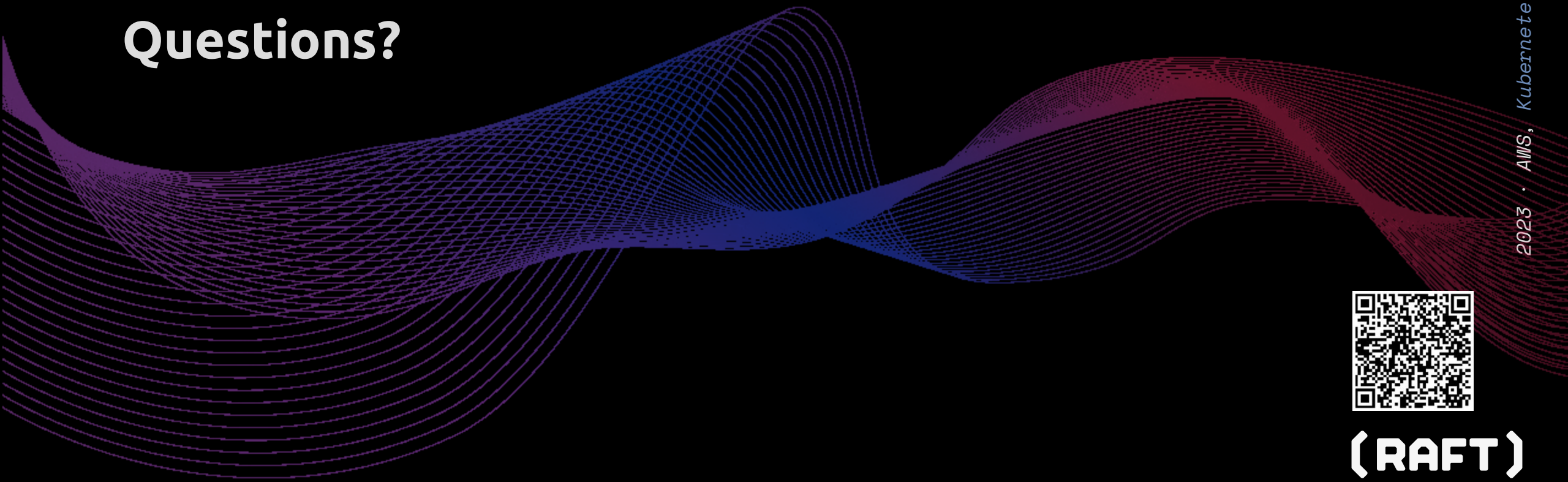


Designing Hybrid Cloud for Contested Environments

- Bias for vendor-owned, vendor-managed infrastructure inside the datacenter
- Bias for disposable units of compute outside the datacenter
- Push mission-critical control planes to the edge
- Restrict dependency on in-region control planes
- Scale outward from a centralized platform team through operations and security teams

Thanks for your time

Questions?



RAFT

2023 · AWS, Kubernetes, and the Disconnected Edge

www.teamraft.com