KubeCon | CloudNativeCon

North America 2019

Nov. 18 - 21, 2019
San Diego, CA

kubecon.io

DIAMANTI

CLOUD NATIVE
COMPUTING FOUNDATION

Challenges in Deploying Kubernetes on Hyperconverged Infrastructure (HCI)

Presenters

Naren Narendra
Director
Product Marketing

Naveen Seth
Founding Engineer

Hiral Patel
Founding Engineer

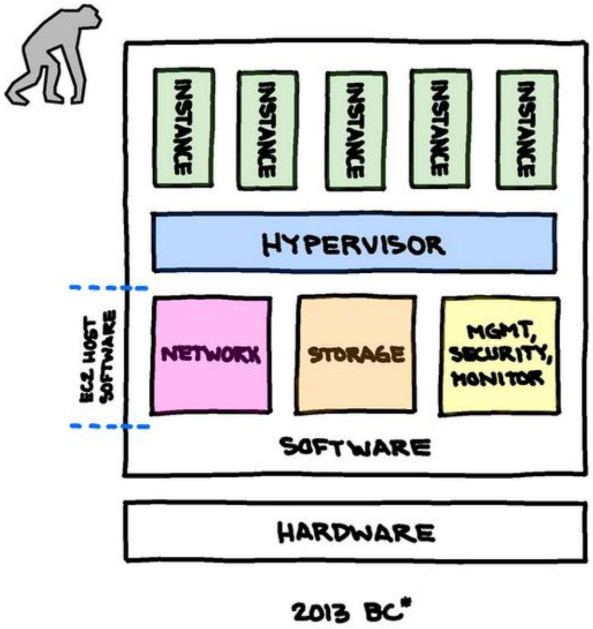
© 2019 DIAMANTI | CONFIDENTIAL | DO NOT DISTRIBUTE

Agenda

1. Background with AWS EC2 evolution - example
2. Requirements for Kubernetes on bare metal HCI
3. Demo
4. Q&A

© 2019 DIAMANTI | CONFIDENTIAL | DO NOT DISTRIBUTE

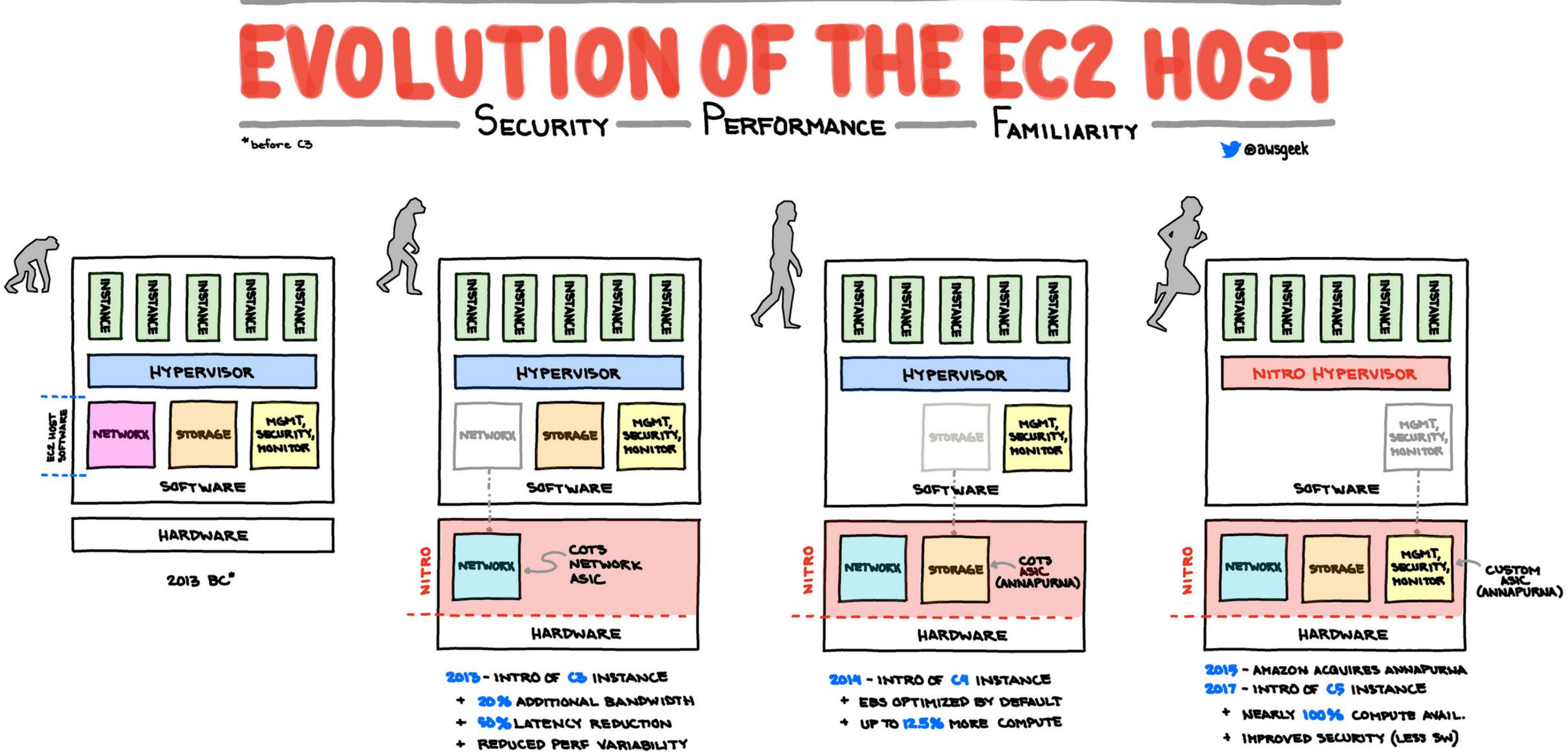
The Beginning



***Networking is**, in effect, **“in the way”** and blocking the efficient optimization of the most valuable resources in the data center.* [1]

Picture source: Evolution of the EC2 Host, https://bit.ly/2lV2WTk (@awsgeek, Jerry Hargrove, AWS)

[1] https://perspectives.mvdirona.com/2009/10/vl2-a-scalable-and-flexible-data-center-network/ (James Hamilton, DE, AWS, 2009)


EVOLUTION OF THE EC2 HOST
SECURITY
PERFORMANCE
FAMILIARITY
*before C3
@awsgeek
INSTANCE
HYPERVISOR
EC2 HOST SOFTWARE
NETWORK
STORAGE
MGMT, SECURITY, MONITOR
SOFTWARE
HARDWARE
2013 BC*
NITRO
COTS NETWORK ASIC
2013 - INTRO OF C3 INSTANCE
+ 20% ADDITIONAL BANDWIDTH
+ 50% LATENCY REDUCTION
+ REDUCED PERF VARIABILITY
COTS ASIC (ANNAPURNA)
2014 - INTRO OF C4 INSTANCE
+ EBS OPTIMIZED BY DEFAULT
+ UP TO 12.5% MORE COMPUTE
NITRO HYPERVISOR
CUSTOM ASIC (ANNAPURNA)
2015 - AMAZON ACQUIRES ANNAPURNA
2017 - INTRO OF C5 INSTANCE
+ NEARLY 100% COMPUTE AVAIL.
+ IMPROVED SECURITY (LESS SW)

1G
HDD

→

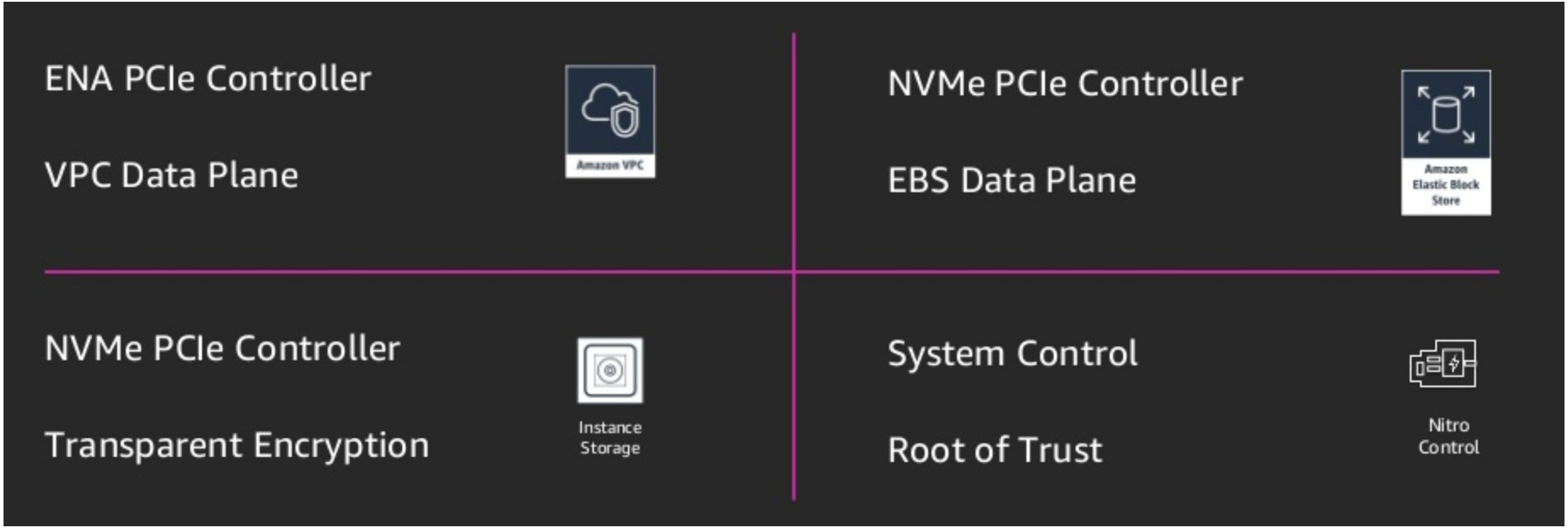
1G/10G
SSD

→

10G/100G
NVMe

Picture source: Evolution of the EC2 Host, https://bit.ly/2IV2WTk (@awsgeek, Jerry Hargrove, AWS)
Reference blog: https://perspectives.mvdirona.com/2019/02/aws-nitro-system/ (James Hamilton, DE, AWS, Feb 2019)

AWS Nitro Cards



ENA PCIe Controller

VPC Data Plane

NVMe PCIe Controller

EBS Data Plane

NVMe PCIe Controller

Transparent Encryption

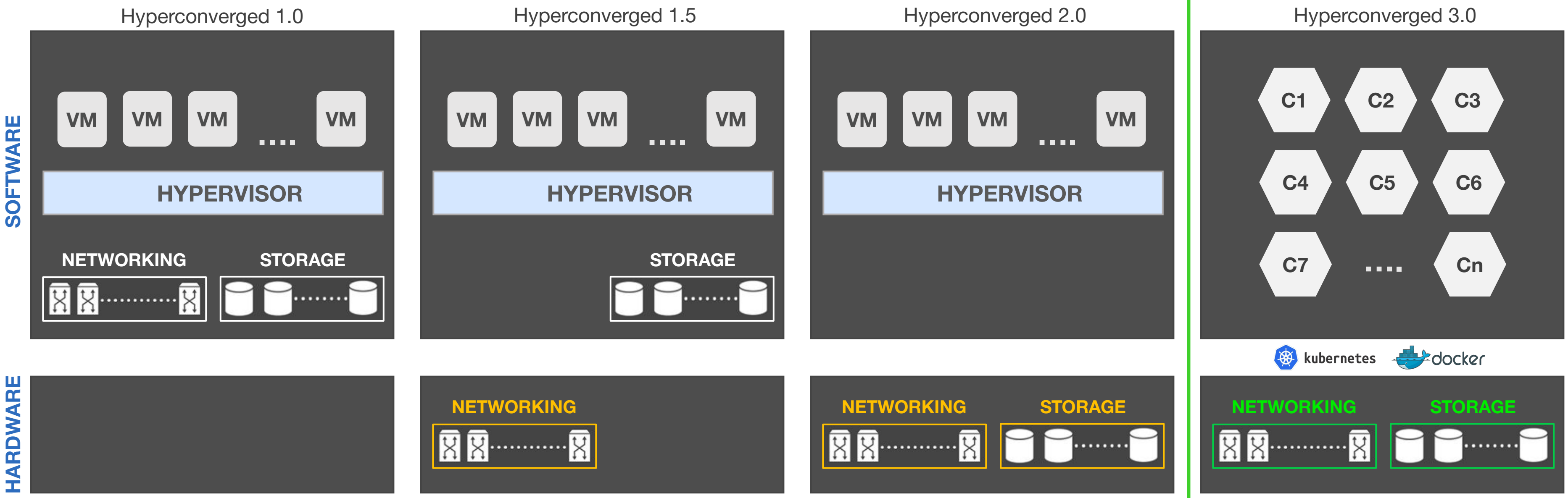
System Control

Root of Trust

Source: Powering Next-Gen EC2 Instances: Deep Dive into the Nitro System - AWS re:Invent 2018, https://bit.ly/2ltCPma
Reference blog: https://perspectives.mvdirona.com/2019/02/aws-nitro-system/ (James Hamilton, DE, AWS, Feb 2019)

Hyperconverged Infrastructure (HCI) Evolution

DIAMANTI



SOFTWARE

HARDWARE

Hyperconverged 1.0

VM VM VM .... VM

HYPERVISOR

NETWORKING STORAGE

Hyperconverged 1.5

VM VM VM .... VM

HYPERVISOR

STORAGE

NETWORKING

Hyperconverged 2.0

VM VM VM .... VM

HYPERVISOR

NETWORKING STORAGE

Hyperconverged 3.0

C1 C2 C3

C4 C5 C6

C7 .... Cn

kubernetes docker

NETWORKING STORAGE

Hyperconverged 1.0

- Host is heavily taxed on utilization and performance, applications starve
- Noisy neighbors
- No SLA guarantee
- Highest TCO

NUTANIX vmware

Hyperconverged 1.5

- Host is heavily taxed on utilization and performance, applications starve
- Noisy neighbors
- Some NW acceleration
- Very High TCO

aws Azure

Hyperconverged 2.0

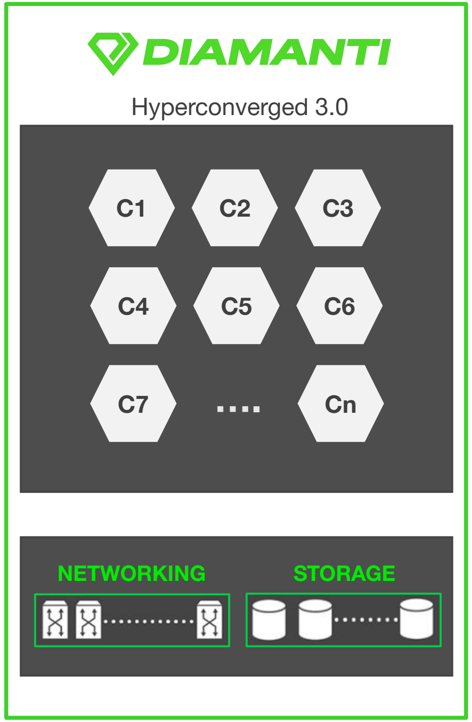
- Host is heavily taxed on utilization and performance, applications starve
- Noisy neighbors
- Storage offload, NW acceleration
- High TCO

aws (Outpost + Nitro)

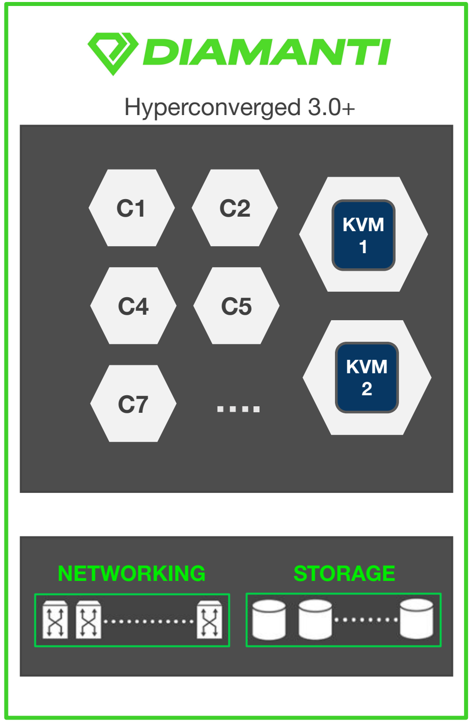
Hyperconverged 3.0

- Cloud native
- Applications get >95% host utilization, Hypervisor removed
- Storage and NW acceleration
- No noisy neighbors
- Guaranteed SLA (Latency, BW)
- Inherently secure
- Lowest TCO

“Container-Native Virtualization”



Bare-Metal Containers
+
Container-Native Virtualization



DIAMANTI

HCI Requirements

HCI Requirements for Containerized Applications

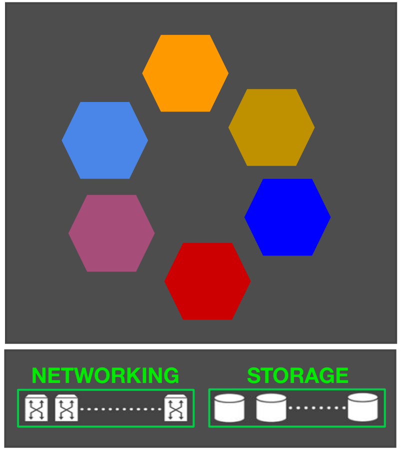
Multiple applications on a node ⇒ **Guaranteed SLAs (Jitter Free)**

Networking

- SR-IOV
  - Hardware queues
- Performance Tiers
  - Min guarantees
  - Max limits

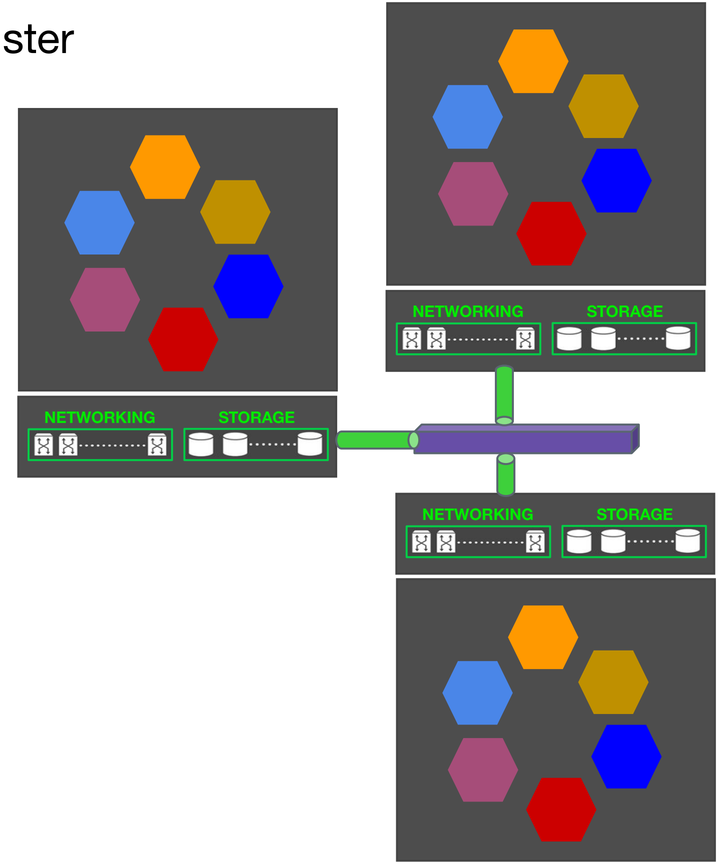
Storage

- SR-IOV
  - Hardware queues
- Performance Tiers
  - Min guarantees
  - Max limits



HCI Requirements for Kubernetes

Multiple application instances across a Kubernetes cluster



Networking (CNI)

- Static and Dynamic endpoint provisioning
- Multiple endpoints
- Endpoint visibility
- Separation of control and data planes
- Availability zones

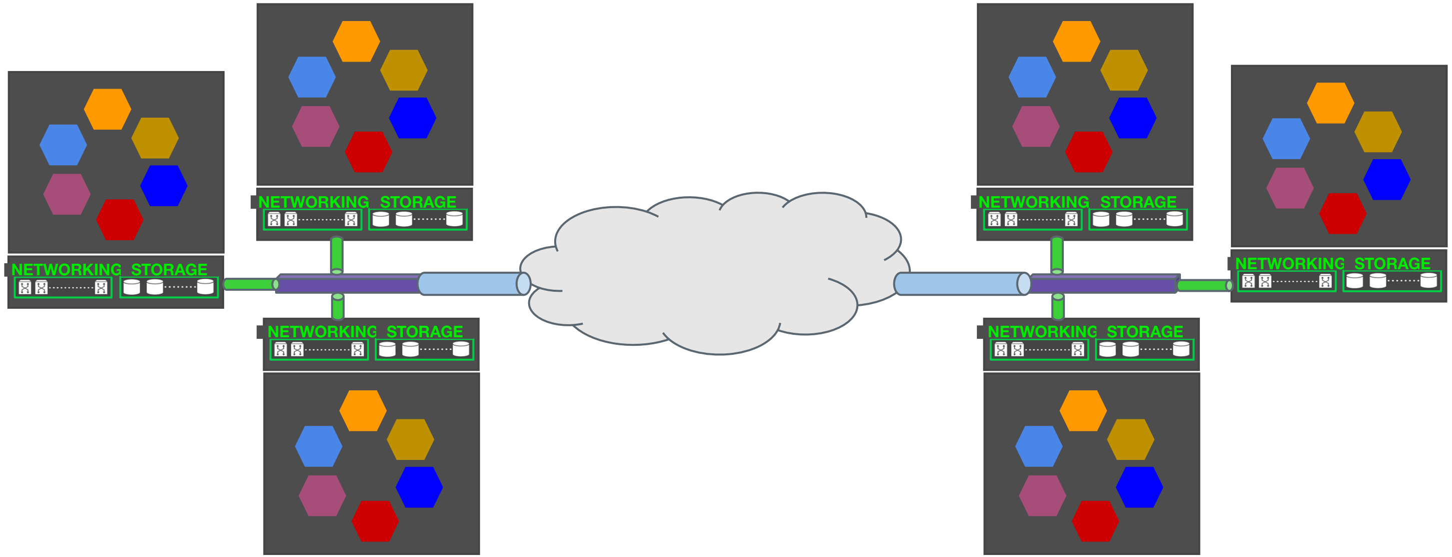
Storage (CSI)

- Static and Dynamic provisioning
- Synchronous mirroring
- Snapshots / Restore
- Backup / Restore
- Availability zones

© 2019 DIAMANTI | CONFIDENTIAL | DO NOT DISTRIBUTE

HCI Requirements for Kubernetes Clusters

High Availability Zones ⇒ Campus Clusters



Networking (CNI)

- Zone aware subnet management

Storage (CSI)

- Zone aware mirror placement

HCI Requirements for “*yet-to-be-containerized*” Legacy Applications

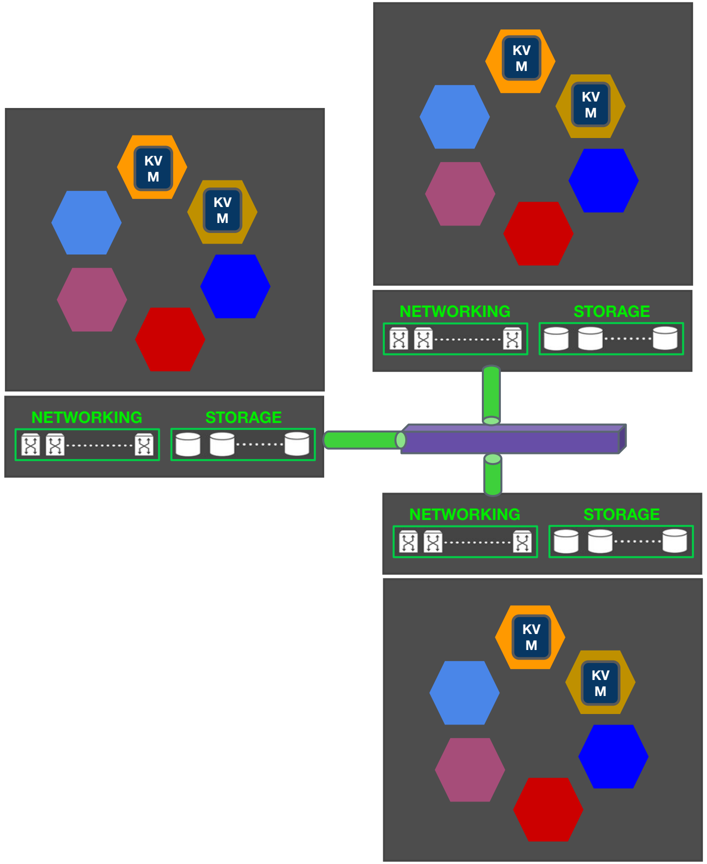
“Container-Native Virtualization”

Networking (CNI)

- Feature parity
- Performance parity
- Kubernetes managed

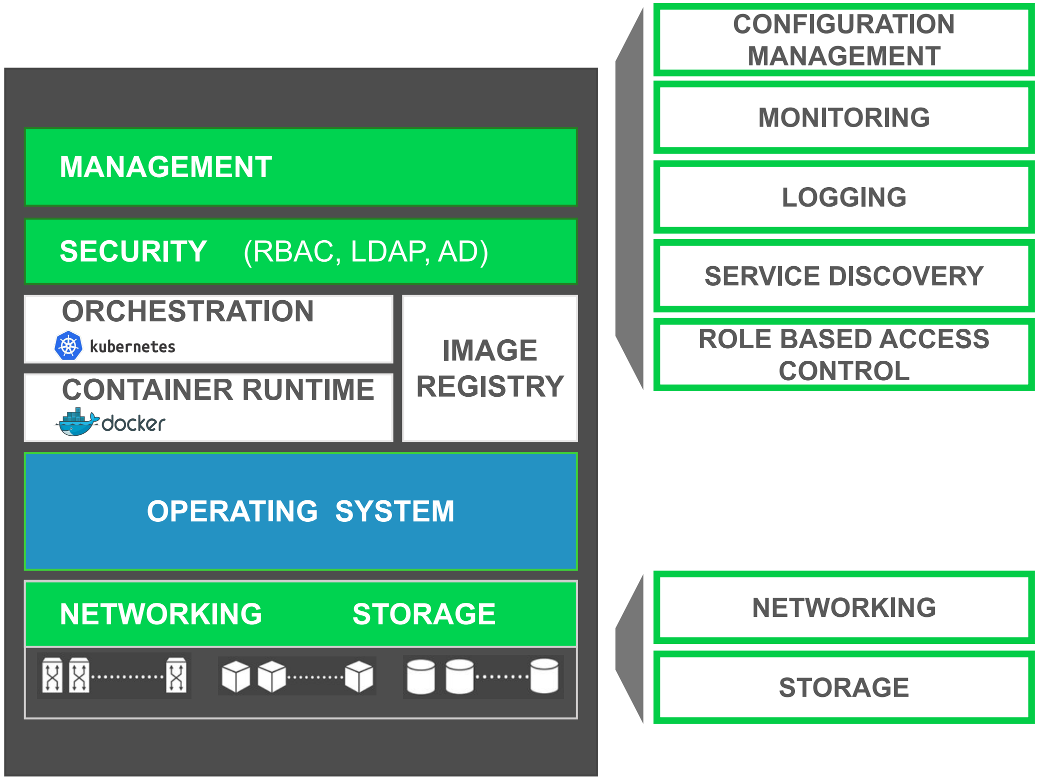
Storage (CSI)

- Feature parity
- Performance parity
- Kubernetes managed



© 2019 DIAMANTI | CONFIDENTIAL | DO NOT DISTRIBUTE

HCI Requirements for Kubernetes Beyond Networking and Storage



MANAGEMENT

SECURITY (RBAC, LDAP, AD)

ORCHESTRATION kubernetes

CONTAINER RUNTIME docker

IMAGE REGISTRY

OPERATING SYSTEM

NETWORKING STORAGE

CONFIGURATION MANAGEMENT

MONITORING

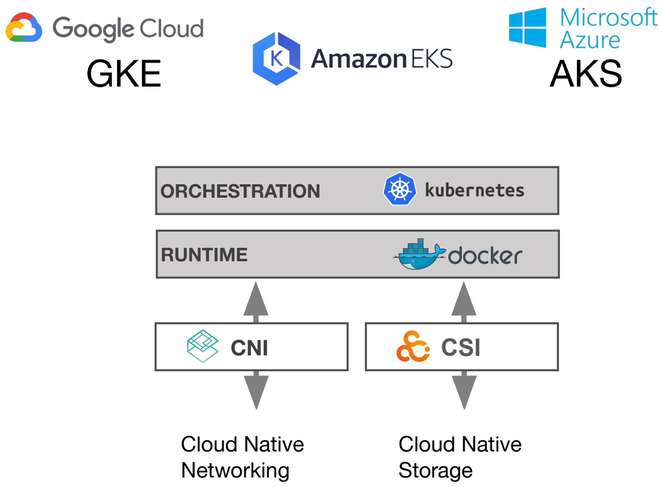
LOGGING

SERVICE DISCOVERY

ROLE BASED ACCESS CONTROL

NETWORKING

STORAGE



Google Cloud GKE

Amazon EKS

Microsoft Azure AKS

ORCHESTRATION kubernetes

RUNTIME docker

CNI

CSI

Cloud Native Networking

Cloud Native Storage

DIAMANTI

Demo

Demo Flow

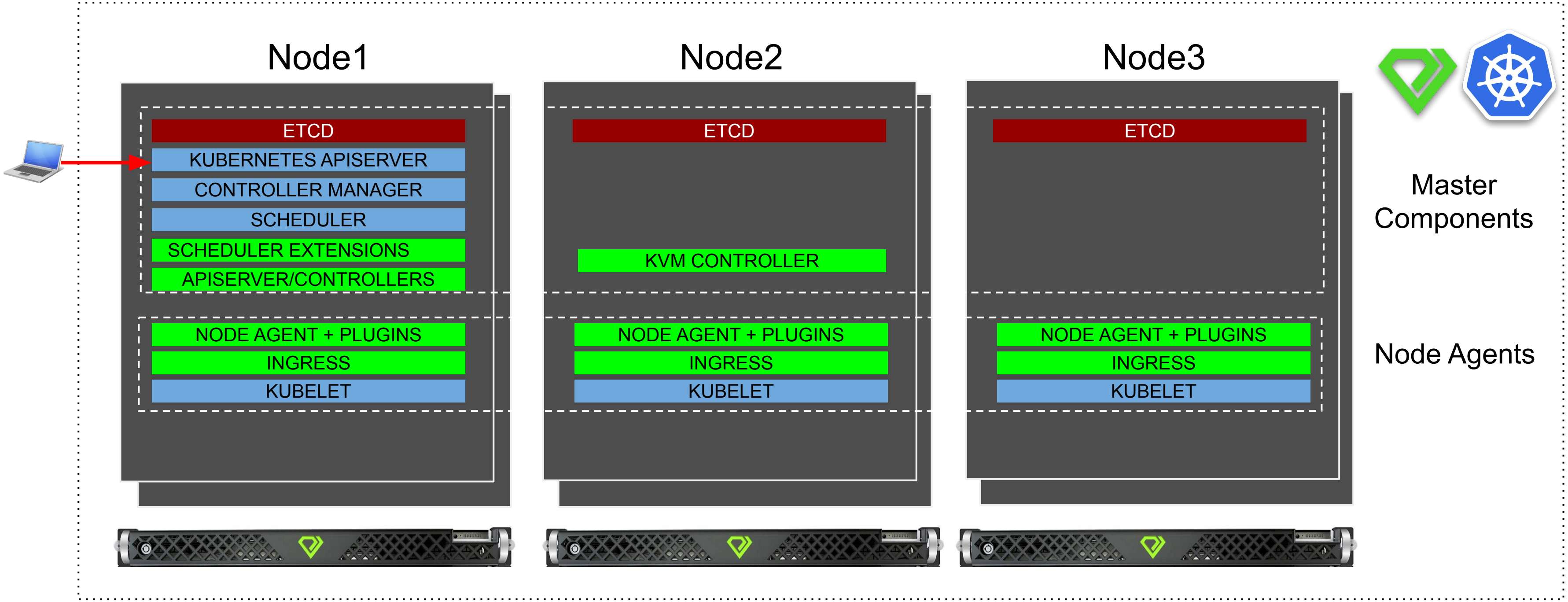
1. Demo setup
2. Deploy multi-instance WordPress application using Kubernetes
3. Deploy KVM for legacy application (yet-to-be-containerized) using Kubernetes
4. I/O isolation with QoS for performance

© 2019 DIAMANTI | CONFIDENTIAL | DO NOT DISTRIBUTE

Demo Flow

➡ 1. Demo setup
2. Deploy multi-instance WordPress application using Kubernetes
3. Deploy KVM for legacy application (yet-to-be-containerized) using Kubernetes
4. I/O isolation with QoS for performance

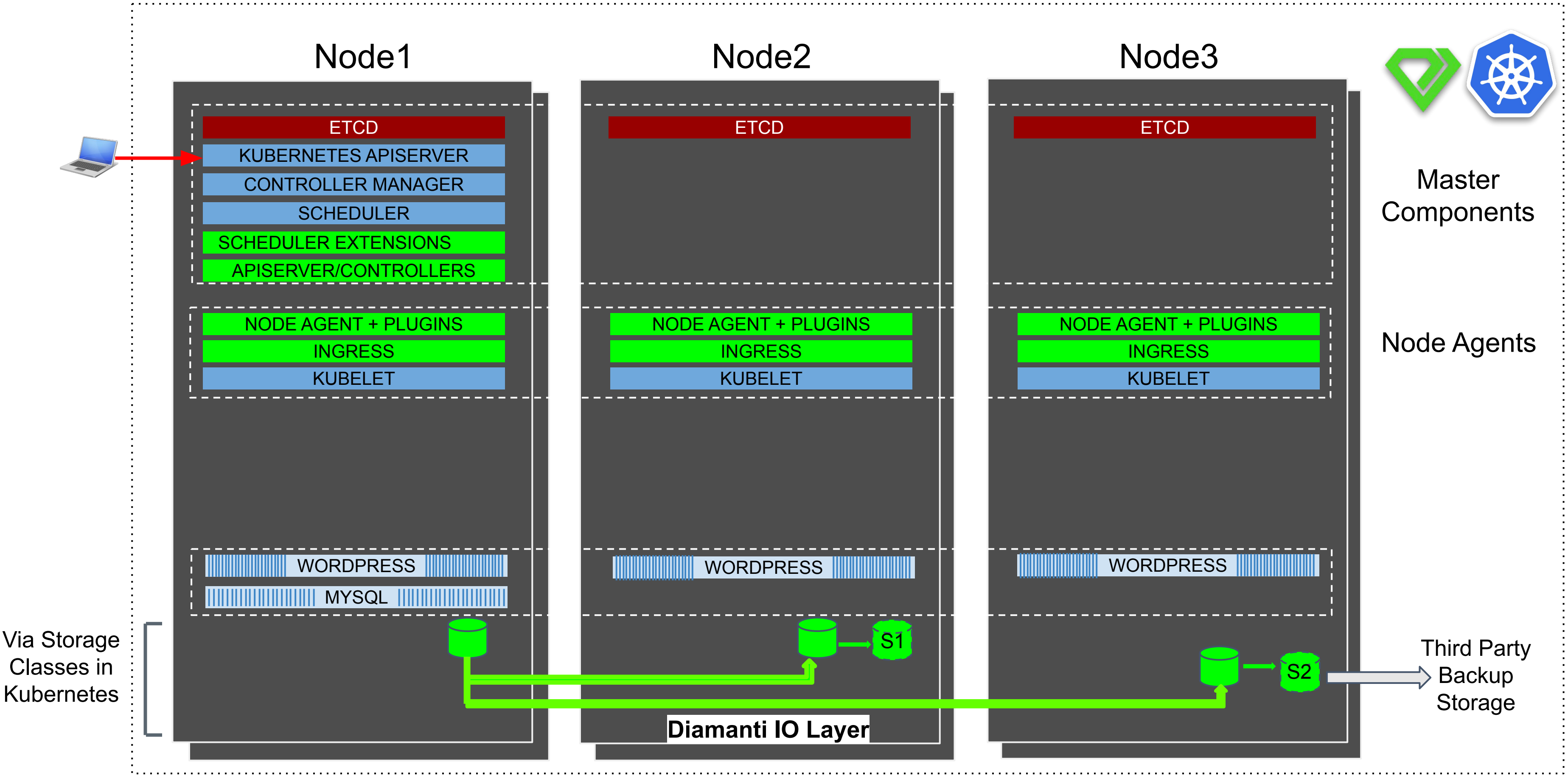
Diamanti Demo Cluster



Demo Flow

1. Demo setup
2. ➡ Deploy multi-instance WordPress application using Kubernetes
3. Deploy KVM for legacy application (yet-to-be-containerized) using Kubernetes
4. I/O isolation with QoS for performance

WordPress Application Deployment



Node1

ETCD

KUBERNETES APISERVER

CONTROLLER MANAGER

SCHEDULER

SCHEDULER EXTENSIONS

APISERVER/CONTROLLERS

NODE AGENT + PLUGINS

INGRESS

KUBELET

WORDPRESS

MYSQL

Node2

ETCD

NODE AGENT + PLUGINS

INGRESS

KUBELET

WORDPRESS

S1

Node3

ETCD

NODE AGENT + PLUGINS

INGRESS

KUBELET

WORDPRESS

S2

Master Components

Node Agents

Via Storage Classes in Kubernetes

Diamanti IO Layer

Third Party Backup Storage

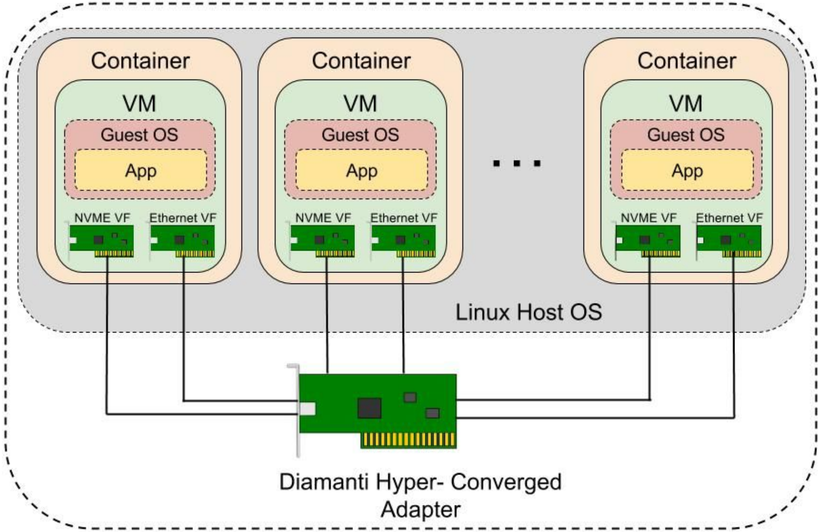
© 2019 DIAMANTI | CONFIDENTIAL | DO NOT DISTRIBUTE

Demo Flow

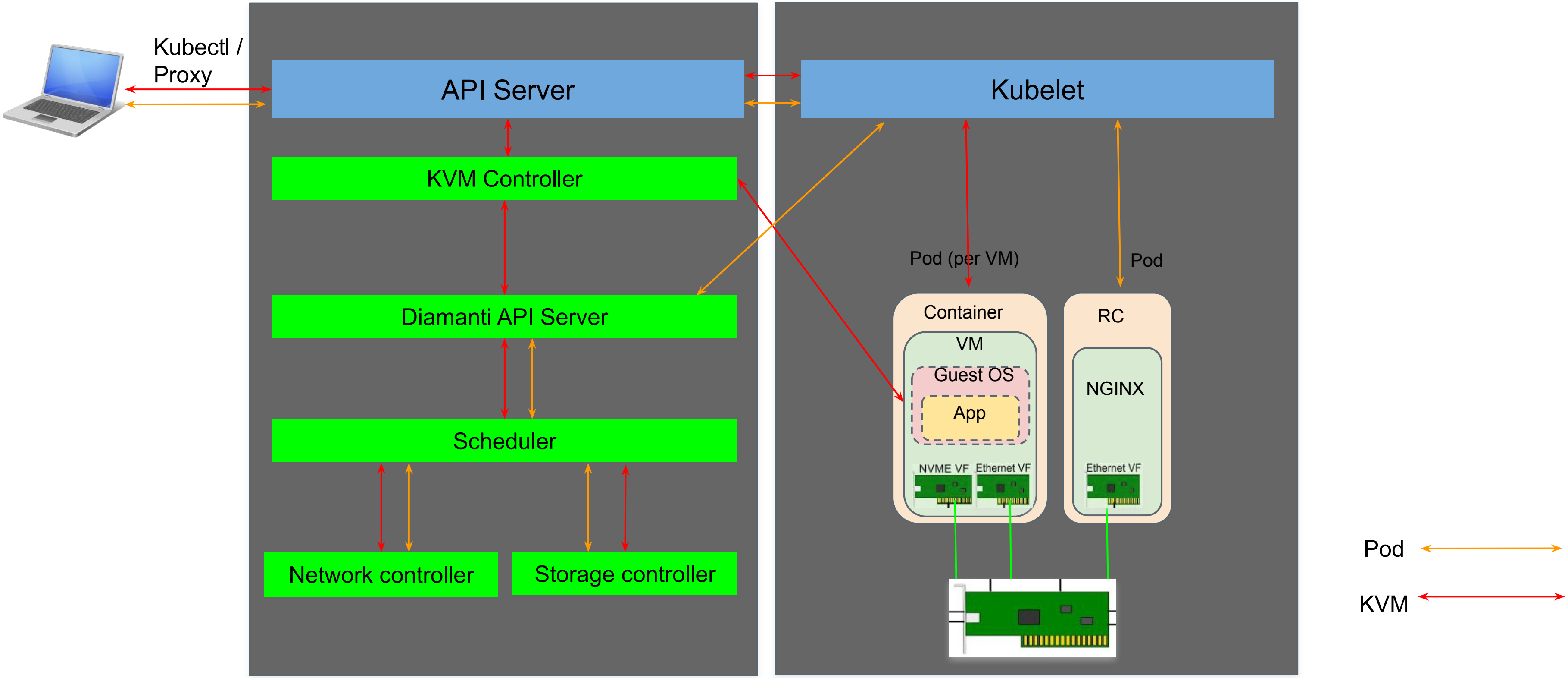
1. Demo setup
2. Deploy multi-instance WordPress application using Kubernetes
3. ➡ Deploy KVM for legacy application (yet-to-be-containerized) using Kubernetes
4. I/O isolation with QoS for performance

Container-Native Virtualization

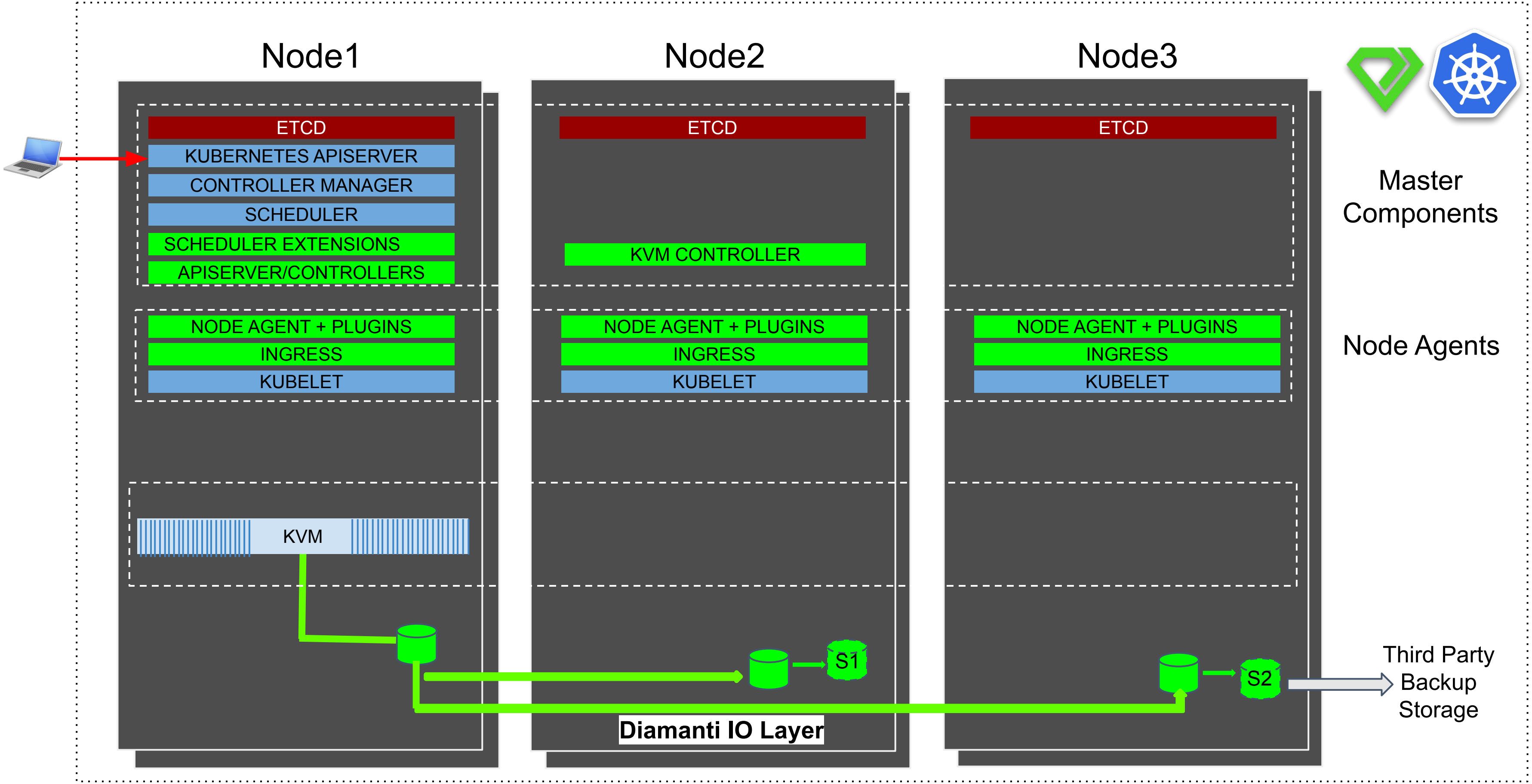
- A VM runs inside a container
- Can co-exist with container workloads/pods
- Based on KVM
- Uses Kubernetes as Orchestrator
- Consistent I/O isolation and quality of service for containers and VMs using PCI pass-through



Container-Native Virtualization(CRD) and Pod Deployment on Kubernetes



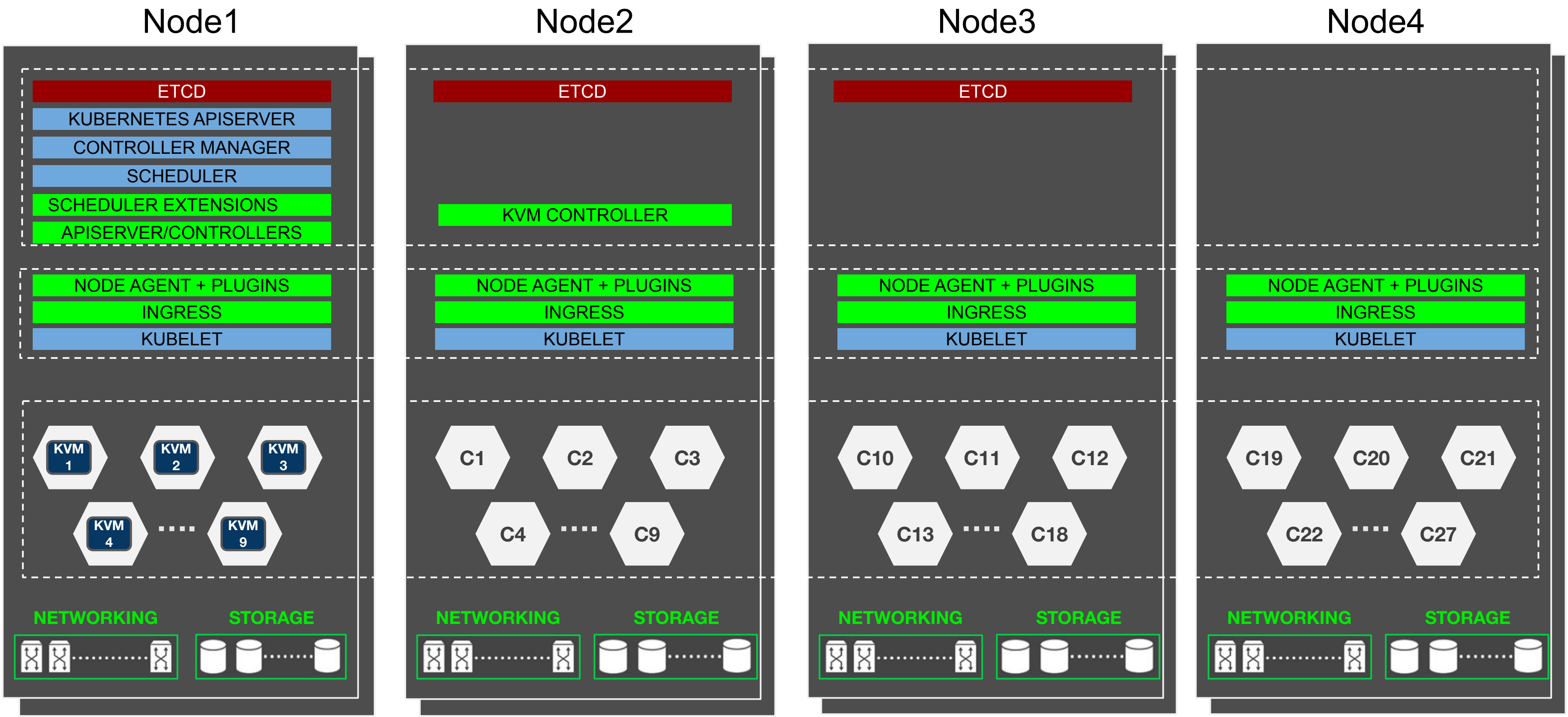
KVM Deployment



Demo Flow

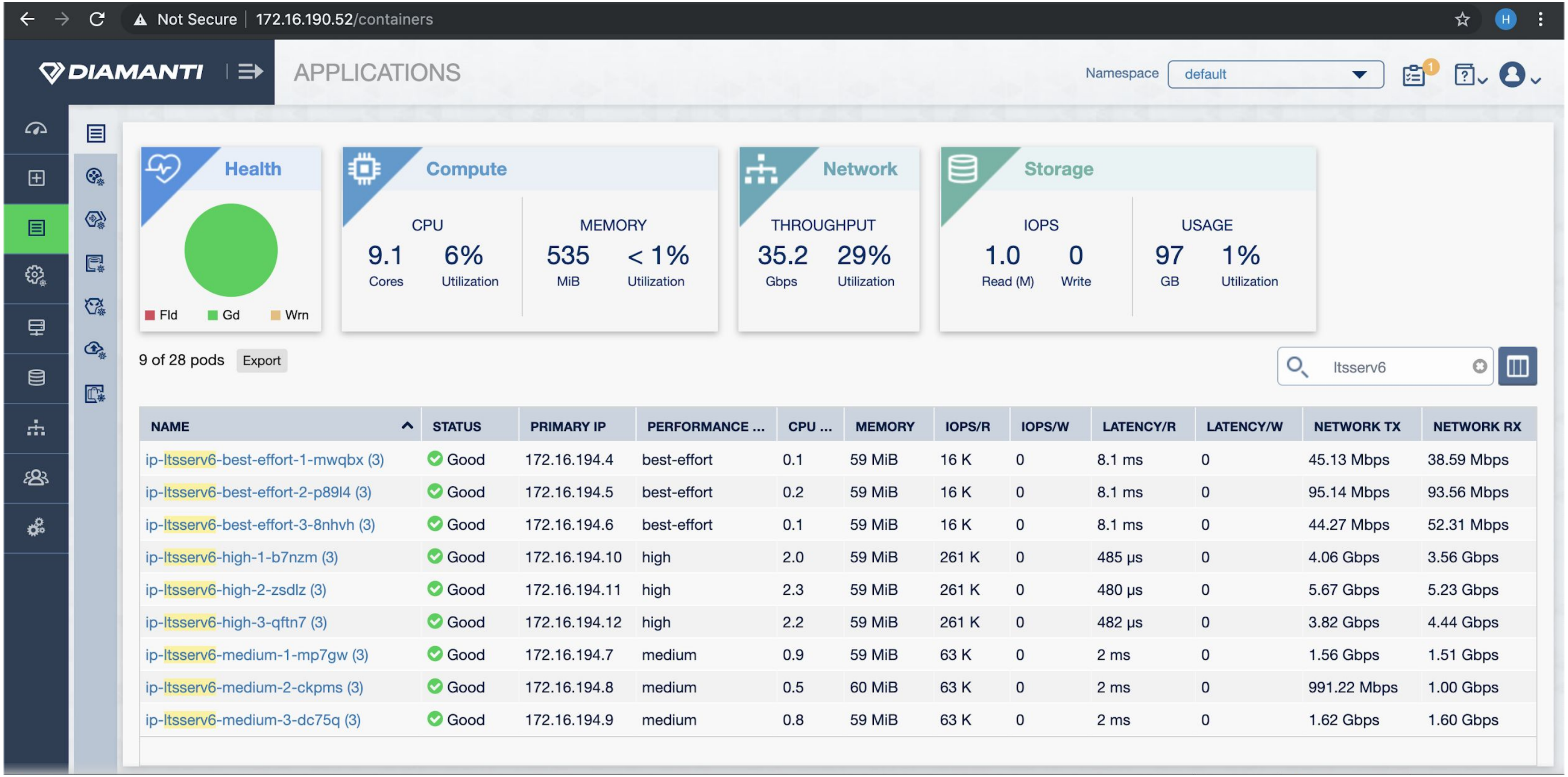
1. Demo setup
2. Deploy multi-instance WordPress application using Kubernetes
3. Deploy KVM for legacy application (yet-to-be-containerized) using Kubernetes
4. I/O isolation with QoS for performance

“Isolation” with “QoS” for Network and Storage



© 2019 DIAMANTI | CONFIDENTIAL | DO NOT DISTRIBUTE

“Isolation” with “QoS” for Network and Storage



Not Secure | 172.16.190.52/containers

DIAMANTI

APPLICATIONS

Namespace: default

Health

🟥 Fld 🟩 Gd 🟨 Wrn

Compute

CPU: 9.1 Cores, 6% Utilization

MEMORY: 535 MiB, < 1% Utilization

Network

THROUGHPUT: 35.2 Gbps, 29% Utilization

Storage

IOPS: 1.0 Read (M), 0 Write

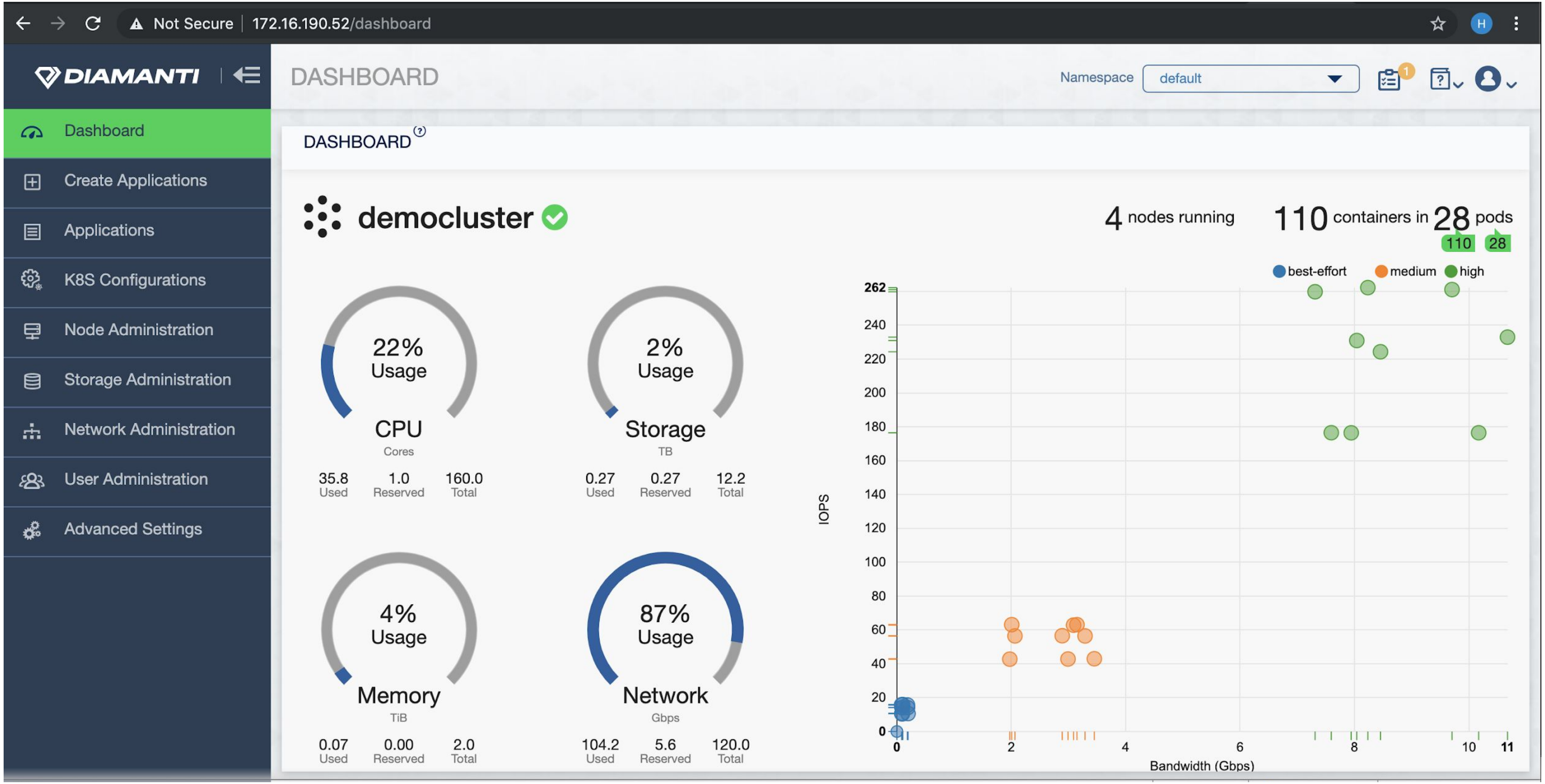
USAGE: 97 GB, 1% Utilization

9 of 28 pods Export

ltsserv6

NAME	STATUS	PRIMARY IP	PERFORMANCE ...	CPU ...	MEMORY	IOPS/R	IOPS/W	LATENCY/R	LATENCY/W	NETWORK TX	NETWORK RX
ip-ltsserv6-best-effort-1-mwqbx (3)	✓ Good	172.16.194.4	best-effort	0.1	59 MiB	16 K	0	8.1 ms	0	45.13 Mbps	38.59 Mbps
ip-ltsserv6-best-effort-2-p89l4 (3)	✓ Good	172.16.194.5	best-effort	0.2	59 MiB	16 K	0	8.1 ms	0	95.14 Mbps	93.56 Mbps
ip-ltsserv6-best-effort-3-8nhvh (3)	✓ Good	172.16.194.6	best-effort	0.1	59 MiB	16 K	0	8.1 ms	0	44.27 Mbps	52.31 Mbps
ip-ltsserv6-high-1-b7nzm (3)	✓ Good	172.16.194.10	high	2.0	59 MiB	261 K	0	485 µs	0	4.06 Gbps	3.56 Gbps
ip-ltsserv6-high-2-zsdlz (3)	✓ Good	172.16.194.11	high	2.3	59 MiB	261 K	0	480 µs	0	5.67 Gbps	5.23 Gbps
ip-ltsserv6-high-3-qftn7 (3)	✓ Good	172.16.194.12	high	2.2	59 MiB	261 K	0	482 µs	0	3.82 Gbps	4.44 Gbps
ip-ltsserv6-medium-1-mp7gw (3)	✓ Good	172.16.194.7	medium	0.9	59 MiB	63 K	0	2 ms	0	1.56 Gbps	1.51 Gbps
ip-ltsserv6-medium-2-ckpms (3)	✓ Good	172.16.194.8	medium	0.5	60 MiB	63 K	0	2 ms	0	991.22 Mbps	1.00 Gbps
ip-ltsserv6-medium-3-dc75q (3)	✓ Good	172.16.194.9	medium	0.8	59 MiB	63 K	0	2 ms	0	1.62 Gbps	1.60 Gbps

“Isolation” with “QoS” for Network and Storage



DASHBOARD

democluster

4 nodes running

110 containers in 28 pods

110 28

22% Usage
CPU
Cores
35.8 Used | 1.0 Reserved | 160.0 Total

2% Usage
Storage
TB
0.27 Used | 0.27 Reserved | 12.2 Total

4% Usage
Memory
TiB
0.07 Used | 0.00 Reserved | 2.0 Total

87% Usage
Network
Gbps
104.2 Used | 5.6 Reserved | 120.0 Total

DIAMANTI

Q&A

To Learn More...

- www.diamanti.com
- info@diamanti.com
- @diamanticom
- https://www.linkedin.com/company/diamanti

© 2019 DIAMANTI | CONFIDENTIAL | DO NOT DISTRIBUTE

DIAMANTI

Thank You!