Dell Technologies 5G Core Solution with Affirmed and Red Hat OpenShift Container Platform

July 2021

H18831

Reference Architecture Guide

Abstract

This reference architecture guide describes how to design and specify an Affirmed 5G converged core solution using Dell Technologies servers and switch infrastructure required to run telecom-specific workloads with validated hardware configurations. It also facilitates the deployment of Red Hat OpenShift Container Platform 4.6 following a Dell Technologies infrastructure deployment.

Dell Technologies Solutions

Copyright

The information in this publication is provided as is. Dell Inc. makes no representations or warranties of any kind with respect to the information in this publication, and specifically disclaims implied warranties of merchantability or fitness for a particular purpose.

Use, copying, and distribution of any software described in this publication requires an applicable software license.

Copyright © 2021 Dell Inc. or its subsidiaries. All Rights Reserved. Dell Technologies, Dell, EMC, Dell EMC and other trademarks are trademarks of Dell Inc. or its subsidiaries. Intel, the Intel logo, the Intel Inside logo and Xeon are trademarks of Intel Corporation in the U.S. and/or other countries. Other trademarks may be trademarks of their respective owners. Published in the USA 04/21 Reference Architecture Guide H18727.

Dell Inc. believes the information in this document is accurate as of its publication date. The information is subject to change without notice.

Contents

Chapter 1 Introduction

This chapter presents the following topics:

Solution overview and key benefits

Introduction

Dell Technologies 5G Core Solution Reference Architecture with Affirmed and Red Hat OpenShift Container Platform is designed to help communications service providers (CSPs) quickly deploy 5G services. This reference architecture features a 5G core platform, container management and orchestration platform, and telco-grade infrastructure in a fully validated solution design.

The Reference Architecture documentation set consists of the following:

- Dell Technologies 5G Core Solution with Affirmed and Red Hat OpenShift Container Platform Reference Architecture (this document)
- Dell Technologies 5G Core Solution Reference Architecture with Affirmed Solution Brief

Both documents are available at the Dell Technologies Solutions Info Hub for Communication Service Providers.

The Reference Architecture provides:

- A solution approach developed by Affirmed, Dell Technologies, and Red Hat to help CSPs accelerate 5G network deployments while reducing the cost and risk associated with network transformation efforts.
- A trusted, validated, industry-leading foundation that provides CSPs with the flexibility and reliability that is required to move forward with confidence.
- An approach to delivering leading-edge technology on a proven platform for a true competitive advantage.

The Reference Architecture includes the following components:

- Affirmed UnityCloud 5G Core solution offers microservices-based 5G network functions that are fully virtualized and containerized for simple deployment in private, public, or hybrid cloud environments.
- Red Hat OpenShift Container Platform 4.6 provides the management and orchestration platform for the containerized 5G core network functions and services.
- Dell EMC PowerEdge R640 and R740 servers deliver high availability in both mobile core and harsher edge environments with customizable hardware acceleration options to provide optimal server performance characteristics for core/edge workloads.
- Dell EMC PowerSwitch S3048-ON, S5248-ON, and S5232-ON switches are optimized for high-performance data center environments, deliver low latency, offer superb performance, and manage high density with hardware and software redundancy.

Affirmed UnityCloud cloud-native solution

Affirmed UnityCloud is a 5G core (5GC), cloud native solution built on an open, scalable, web-based architecture. This architecture enables mobile operators to build the most innovative 5G core network while dramatically reducing the network operating costs up to 90 percent by simplifying and automating network functions. UnityCloud converges “Any G” core, including 2G, 3G, 4G, 5G networks, and wireline core, onto one unified platform,

which simplifies the overall network architecture. For more information, see Chapter 3 Affirmed 5G Core Logical Design on page 13.

OpenShift Container Platform and Kubernetes

Red Hat OpenShift Container Platform 4.6 consists of many open-source components that have been carefully integrated to provide a consistently dependable platform on which you can develop and deploy scalable containerized applications. OpenShift Container Platform provides great flexibility for accommodating platform deployment preferences. For more information, see OpenShift Container Platform 4.6 Documentation.

At the heart of OpenShift Container Platform is Kubernetes container orchestration software. For more information, see What Is Kubernetes?

Document purpose

This reference architecture guide describes the component options for building a 5GC system using the Affirmed 5GC solution platform to support high-density, high-performance 5GC and edge workloads. The components and features include:

- Full set of cloud native network functions
- Advanced analytics
- Service automation
- Robust network slice management
- Red Hat OpenShift Container Platform
- Kubernetes-based container management and orchestration for 5G container-based network functions
- Dell Technologies telco-grade PowerEdge server infrastructure

The specific deployment methodology depends on identified business outcomes and other variables that may require consulting and services from Dell Technologies, Affirmed Networks, and Red Hat. This guide focuses on a specific deployment model as defined in this reference architecture.

Audience

This reference architecture guide is for network and system architects and system administrators who want to deploy a 5G core Affirmed solution running on Red Hat OpenShift. Some experience with containers, Kubernetes, and the OpenShift Container Platform is recommended.

We value your feedback

Dell Technologies and the authors of this document welcome your feedback on the solution and the solution documentation. Contact the Dell Technologies Solutions team by email or provide your comments by completing our documentation survey.

Note: For links to additional documentation for this solution and other Telecom solutions, see the Dell Technologies Info Hub for Communications Service Provider Solutions.

Chapter 2 Technology Overview

This chapter presents the following topics:

Affirmed 5G UnityCloud

Overview

Affirmed UnityCloud is a cloud native solution built on an open, web-scale architecture that enables mobile operators to build an innovative 5G core (5GC) network while reducing the operating cost significantly, by simplifying and automating network functions. UnityCloud converges multiple networks and the wireline core into one unified platform, streamlining the overall network architecture.

Affirmed UnityCloud includes a unique Platform as a Service (PaaS) layer that provides a robust, open-source architecture featuring the industry-leading cloud ecosystem of applications to address everything from container life cycle management (Kubernetes, Helm) and database services (MongoDB) to network monitoring (Prometheus, Grafana). It also consists of a cloud native Operations and Policy Manager (OPM) that automates the deployment of applications. UnityCloud supports a seamless transition to 5G by integrating legacy network services into an advanced, open 5GC architecture that leverages leading cloud-based technologies.

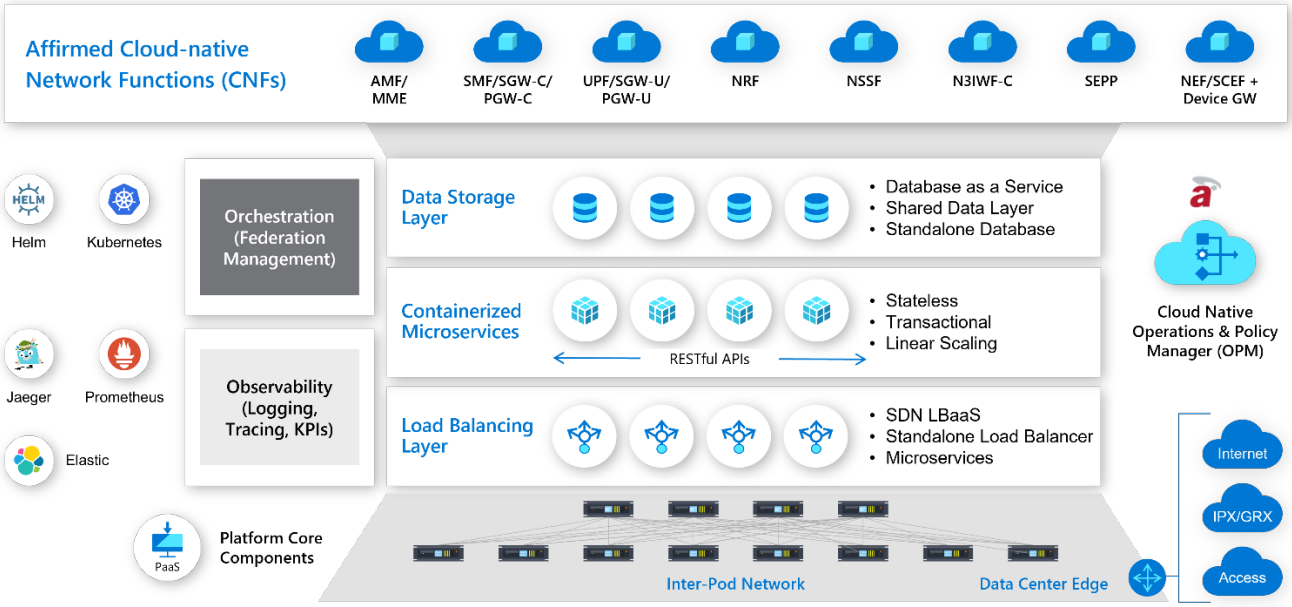

Figure 1. Affirmed Networks Cloud-Native Architecture

Red Hat OpenShift Container Platform

Overview

The OpenShift Container Platform is an enterprise-grade, declarative-state machine that has been designed to automate application workload operations based on the upstream Kubernetes project. In a Kubernetes context, “declarative” means that developers can specify, in code, a configuration for an application or workload without knowing how that application is going to be deployed. OpenShift Container Platform uses the enterprise-grade Kubernetes distribution, called the OpenShift Kubernetes Engine, to provide production-oriented container and workload automation. OpenShift Container Platform 4.6 is based on Kubernetes version 1.19, which includes native support for cluster snapshots, enabling cluster backup and recovery. In addition, OpenShift Container Platform 4.6 is the

first Extended User Support release of OpenShift Container Platform, providing 18 months of support. Built on top of Kubernetes, the OpenShift Container Platform provides administrators and developers with the tools required to deploy and manage applications and services at scale.

Note: OpenShift Container Platform is a certified Kubernetes distribution. Certification for Kubernetes distributions is provided by the Cloud Native Computing Foundation.

The following figure shows the OpenShift Container Platform architecture:

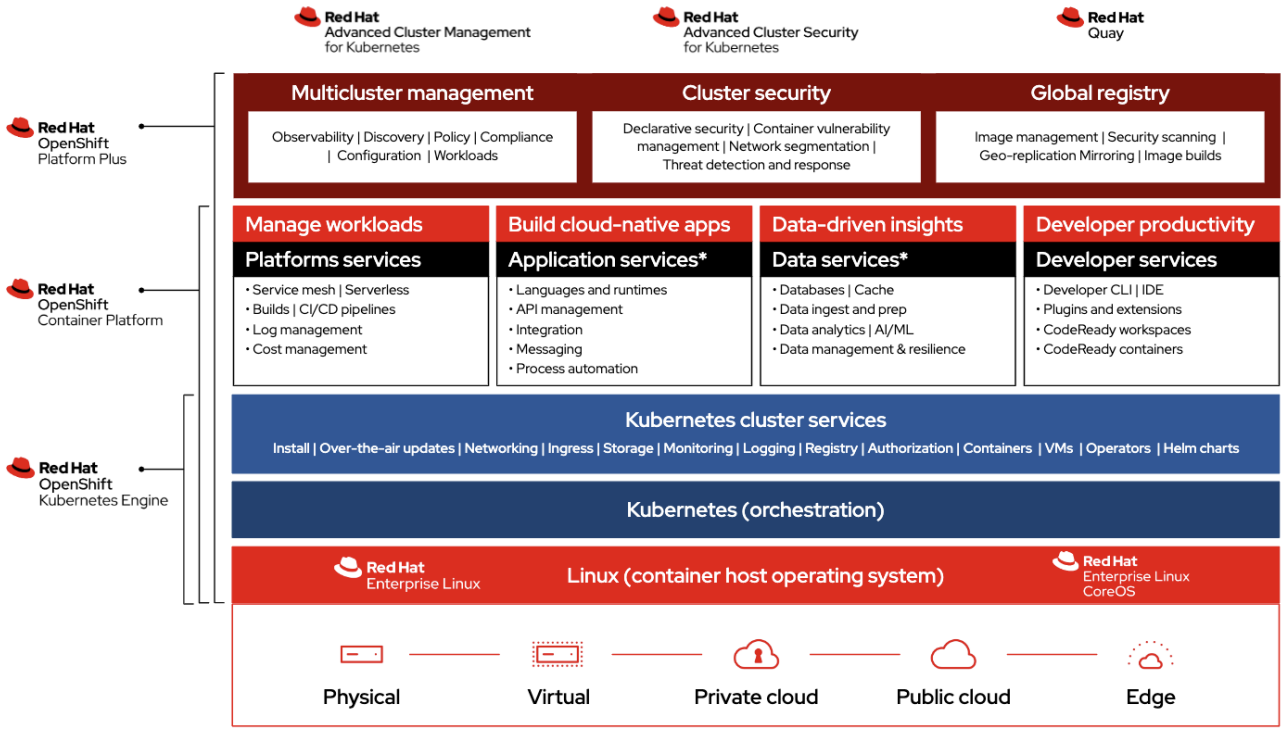

Figure 2. OpenShift Container Platform architecture

The foundational layer is the hosting platform, which uses Dell EMC PowerEdge servers and Dell EMC Networking switches in this reference architecture. Dell EMC storage products may be integrated into this layer to create a comprehensive platform for hosting the Red Hat OpenShift Container Platform.

Infrastructure requirements

Cluster sizing

In OpenShift Container Platform 4.6, two different types of cluster deployments are available: a three-node cluster and a standard cluster (5+ nodes). In a three-node cluster, the control plane and cluster workloads run on the same nodes, enabling small-footprint deployments of OpenShift for testing, development, and production environments. While the three-node cluster can be expanded with additional compute nodes, an initial expansion of a three-node cluster requires the addition of at least two compute nodes at the same time. This step is mandatory because the ingress networking controller deploys two router pods on compute nodes for full functionality. If compute nodes are added to a

three-node cluster, deploy two compute nodes for full ingress functionality. You can add more compute nodes subsequently as needed. A standard cluster deployment has three control-plane nodes and at least two compute nodes. With this deployment type, control-plane nodes are marked as “unschedulable,” which prevents cluster workloads from being scheduled on those nodes. Both cluster deployment types require two CSAH nodes for cluster management and resilient load-balancing.

Basic guidance

A container cluster can be deployed quickly and reliably when each node is within the validated design guidelines. The following table provides basic cluster infrastructure guidance.

Table 1. Hardware infrastructure for OpenShift Container Platform 4.6 cluster deployment

Type	Description	Number	Notes
CSAH node	Dell EMC PowerEdge R640 server	2	Creates a bootstrap VM. CSAH runs a single instance of HAProxy. For an enterprise high availability (HA) deployment of OpenShift Container Platform 4.6, Dell Technologies recommends using a commercially supported L4 load-balancer or proxy service, or an additional PowerEdge R640 CSAH node running HAProxy and Keepalived alongside the primary CSAH node. Options include commercial HAProxy, Nginx, and F5.
Control-plane nodes	Dell EMC PowerEdge R640 server	3	Deployed using the bootstrap VM.
Compute nodes	Dell EMC PowerEdge R640 or R740xd server	Minimum 2* per rack, maximum 30	No compute nodes are required for a three-node cluster. A standard deployment requires a minimum of two compute nodes (and three controller nodes). To expand a three-node cluster, you must add two compute nodes at the same time. After the cluster is operational, you can add more compute nodes to the cluster through the Cluster Management Service.
Data switches	Either of the following switches: Dell EMC PowerSwitch S5248-ON Dell EMC PowerSwitch S5232-ON	2 per rack	Configured at installation time. **Note:** HA network configuration requires two data path switches per rack. Multirack clusters require network topology planning. Leaf-spine network switch configuration may be necessary.
iDRAC network	Dell EMC PowerSwitch S3048-ON	1 per rack	Used for OOB management.
Rack	Selected according to site standards	1–3 racks	For multirack configurations, consult your Dell Technologies or Red Hat representative regarding custom engineering design.

*A three-node cluster does not require any compute nodes. To expand a three-node cluster with additional compute machines, first expand the cluster to a five-node cluster using two additional compute nodes.

Dell Servers Overview

PowerEdge R640

The PowerEdge R640 is a general-purpose platform that is expandable up to 7.68TB of memory, up to twelve 2.5 inch drives, and flexible I/O options. The PowerEdge R640 can handle demanding workloads such as virtualization, dense private cloud, High Performance Computing (HPC), and software-defined storage. The PowerEdge R640 is the ideal dual-socket, 1U platform for dense scale-out data center computing. The PowerEdge R640 combines density, performance, and scalability to optimize application performance and data center density.

The PowerEdge R640 features:

- 2nd Generation Intel® Xeon® Scalable Processor product family (with up to 28 cores and two threads per core)
- Up to six DDR4 memory channels with two DIMMs per channel per CPU and 24 DIMMs (supports DDR4 RDIMM/LRDIMM/NVDIMM-N/DCPMM)
- PCI Express® (PCIe) 3.0 enabled expansion slots (with up to 48 lanes per CPU)
- Networking technologies, such as Ethernet, InfiniBand, OCP, OPA

PowerEdge R740

The Dell EMC PowerEdge R740 and R740xd are two socket, 2U rack servers designed to run complex workloads using highly scalable memory, I/O capacity and network options. The R740 and R740xd features the 2nd Generation Intel® Xeon® Scalable processor family, up to 24 DIMMs, PCI Express® (PCIe) 3.0 enabled expansion slots, and a choice of network interface technologies to cover NIC and rNDC.

The PowerEdge R740 is a general-purpose platform capable of handling demanding workloads and applications, such as data warehouses, eCommerce, databases, and high-performance computing (HPC).

The PowerEdge R740xd adds extraordinary storage capacity options, making it well suited for data-intensive applications that require greater storage, without sacrificing I/O performance.

Chapter 3 Affirmed 5G Core Logical Design

This chapter presents the following topics:

Affirmed 5GC Network Architecture

Affirmed Networks delivers the following 5G NG-Core functions:

- AMF—Access and Mobility Management Function
- SMF—Session Management Function
- UPF—User Plane Function
- NRF—Network Repository Function
- NSSF—Network Slice Selection Function
- NEF—Network Exposure Function
- N3IWF—Non-3GPP Interworking Function

In addition, Affirmed Networks also supports legacy 3GPP (2G, 3G, 4G) and non-3GPP functions in the same architecture. Affirmed Networks delivers these 5G functions through its partners:

- AUSF—Authentication Server Function
- UDM—Unified Data Management
- PCF—Policy Control Function
- BSF—Binding Support Function
- SEPP—Security Edge Protection Proxy
- SMSF—Short Message Service Function

The UnityCloud network architecture fully embraces cloud native software design patterns, network slicing, and 4G-backward compatibility. From a cloud native perspective, the UnityCloud architecture embraces the three pillars of cloud native design (microservices, containers, and orchestration) as established by the Cloud Native Computing Forum (cncf.io).

Additionally, the UnityCloud architecture adds a fourth pillar – Stateless. (An example of Stateless would be the durable state that is stored in an external in-memory NoSQL database and not directly within the application.)

The benefits of the Affirmed Networks' Cloud Native Architecture include:

- **Increased service and feature velocity**: Autonomously developed and released composable service components
- **Third-party software integration promotes fast innovation**: HTTP-API's for both external and internal usage, which is published by OpenAPI/Swagger 3.0
- **Redundant solution is simpler and easier to test**: Rely on NoSQL DB redundancy instead of application-specific mechanisms
- **Geo-redundant solution is simpler and easier to test**: Rely on NoSQL DB redundancy instead of application-specific mechanisms

- **Highly scalability**: Microservices-based architecture allows for scaling and growth
- **Life cycle management simplicity**: Stateless microservices, intent-driven container orchestration, and reduced anti-affinity rules
- **Smaller computational units**: Most microservice instances have 1-2 vCPUs which means that there will be less fragmentation of server resources

Affirmed UnityCloud is consists of the following network functions in a 5GCe network:

- AMF
- SMF
- NRF
- NSSF
- UPF

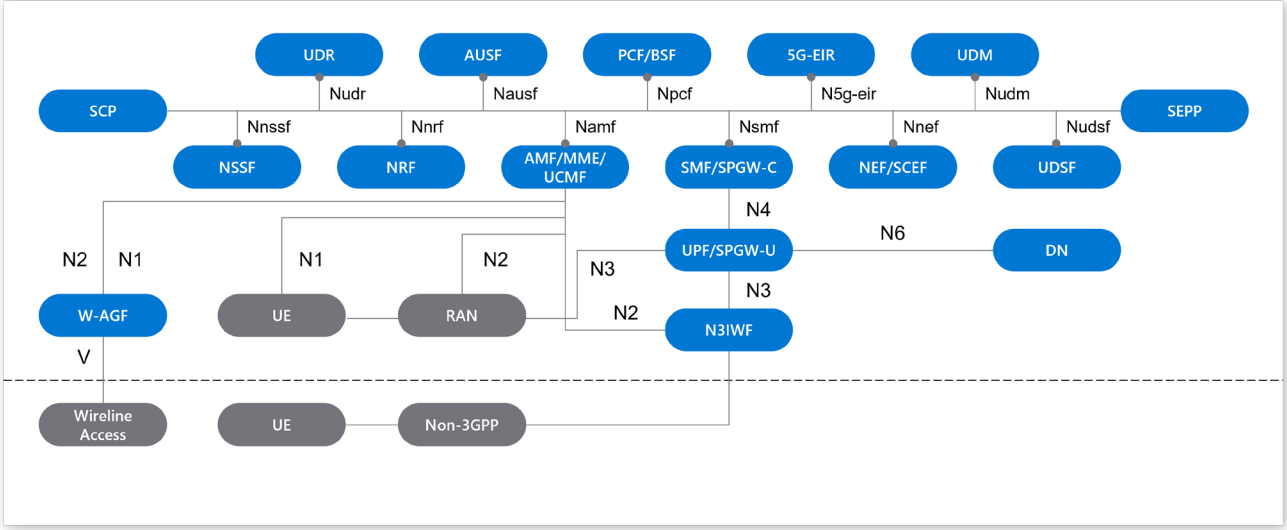

Figure 3. Affirmed Networks UnityCloud™ CNF Components Diagram

AMF

In the 5G Core Network, the Access and Mobility Function (AMF) is responsible for the access and mobility management of the mobile subscribers. It is the point of contact for all mobile users in the core network. It maintains connections with the Radio Access Network (RAN) to transport signaling messages to and from the users.

Affirmed AMF not only provides the required functionalities of the AMF but also a rich set of value-added capabilities to meet many possible use cases in the wireless networks. Affirmed AMF fulfills the functionalities in a pure, cloud native design that leads to unparalleled reliability and operability.

SMF

The Session Management Function (SMF) provides session management within a 5G Standalone Architecture (5G SA) core network and, at the highest level, controls creating, modifying, and deleting Protocol Data Units (PDU) sessions. This provides data access from the user equipment (UE) to one or more data networks.

The SMF also works with the 4G Packet Data Gateway Control Plane (PGW-C) function to provide seamless handovers between 4G and 5G network technology. This allows for a migration from 3G and 4G networks to 5G technology.

The Affirmed Networks SMF is standardized as part of the 5G cloud native architecture (CNA). As such, it is implemented as a collection of microservices that interacts natively with each other within a Kubernetes ecosystem.

NRF

The Network Repository Function (NRF) in 5GC supports managing different network function (NF) instances and their respective profiles. This allows different NFs to register and de-register their services/profile with the NRF. NRF supports the discovery of different NF instances based on their state and local policies, similar to how 2G and 3G Domain Name System (DNS) services supported looking up different network elements. In contrast to the DNS that uses static information for the Network Element (NE) selection, the NRF uses dynamic state information gathered from periodic heartbeat procedure (with NFs) for NF selection. In essence, the NRF can be viewed as an evolution of the DNS in legacy wireless network with enhanced capabilities. Additionally, the NRF also allows NFs to subscribe to status updates of peer NFs/NF types. This capability enables NFs to take appropriate action based on peer NF status.

NSSF

In today's networks, operators plan to provide differentiated services to their end users. Network slicing allows operators to effectively use their network resources while still meeting the varying needs and demands of different consumers. For example, an operator may need to support enhanced mobile broadband (eMBB) and IoT traffic. Given the differing traffic characteristics of these systems, providing different slices for these use cases enables an operator to better utilize their network resources.

In the context of a 5G system, a network slice is defined in the scope of a Public Land Mobile Network (PLMN) and consists of the following network functions which contribute to the network:

- Control plane functions (AMF, SMF)
- User plane functions (UPF)
- Radio access network (gNB, N3WIF)

UPF

The User Plane Function (UPF) is a fundamental component of the 5G core infrastructure system architecture. It allows packet processing, traffic aggregation, and management functions to move to the edge of the network.

The UPF provides an IP anchor point for Intra/Inter Radio Access Technology (RAT) mobility. It also implements the user plane portion of policy enforcement, as well as traffic usage reporting and lawful intercept functionality.

The UPF is standardized as part of the 5G CNA. It is implemented as a federation of microservices that interact with each other to provide user plane functionality in 5G networks.

Deployed in a dynamic-cloud native compute infrastructure, the Affirmed Networks UPF also contains the user plane functions of Serving Gateway (SGW-U), Packet Data Network Gateway (PGW-U) and System Architecture Evolution Gateway (SAEGW-U).

This infrastructure can be programmed appropriately by the control plane entities such as SMF, SGW-C, and PGW-C.

UnityCloud Architecture

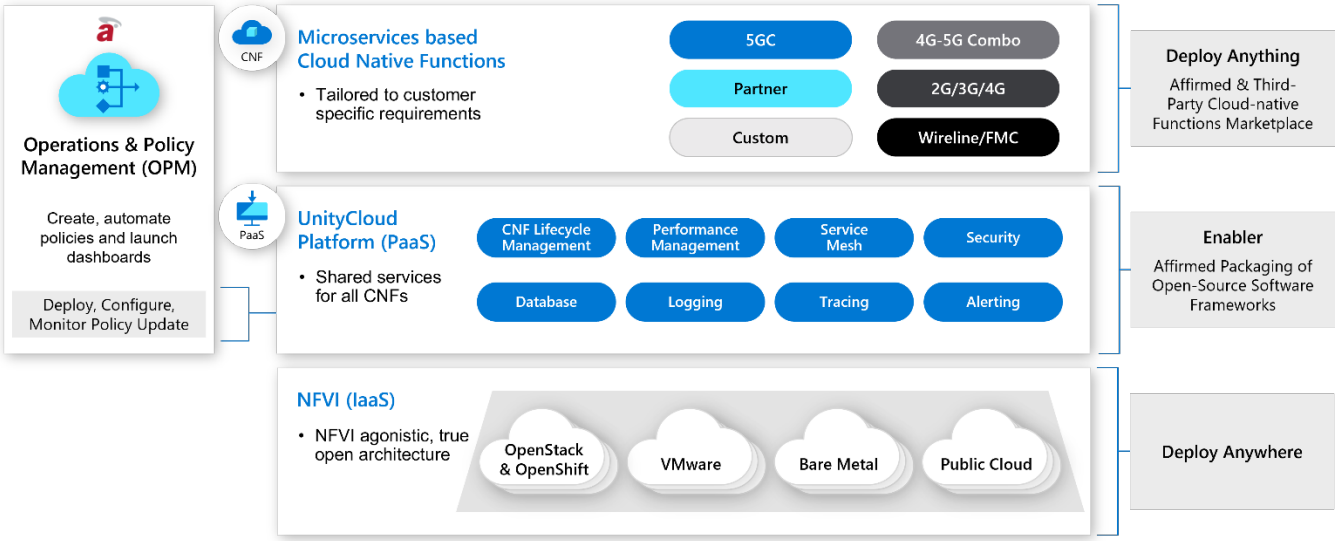

Figure 4. Affirmed Networks UnityCloud™ Architecture

Affirmed UnityCloud is the industry's first cloud native solution that is built on an open, web-scale architecture. This characteristic allows Mobile Network Operators (MNOs) to build an innovative 5GC network, converge multiple networks into one unified network without being locked into a proprietary platform, and monetize revenue generating services. Affirmed UnityCloud also includes a unique PaaS that provides a robust, open-source architecture. This architecture features an industry-leading cloud ecosystem of applications to address everything from container life cycle management (Kubernetes, Helm) and database services (MongoDB) to network monitoring (Envoy, Jaeger, Prometheus). It also consists of a cloud native operations and policy manager (OPM) that automates and creates policies and launches dashboards. These functions provide operators with complete control of their network. UnityCloud supports a seamless transition to 5G by integrating legacy network services—including wireline services—to create an advanced, open 5GC architecture that leverages leading cloud-based technologies alongside Affirmed UnityCloud's industry-leading NFVi platform and microservices-based, cloud native functions (CNFs).

UnityCloud network operations

By default, UnityCloud worker nodes will run control plane CNFs. These nodes can also be configured to run data plane CNFs utilizing SR-IOV ports on a separate network. The worker nodes can have the same or different network for OAM and Kubernetes. The HAProxy and CSAH servers are added for illustrative purposes. Unity Cloud worker nodes do not have a specific requirement on how OpenShift API load balancing is implemented. CSAH servers may or may not be made to host DNS, DHCP, HA Proxy services. The following figure provides a network diagram for a sample Unity Cloud deployment.

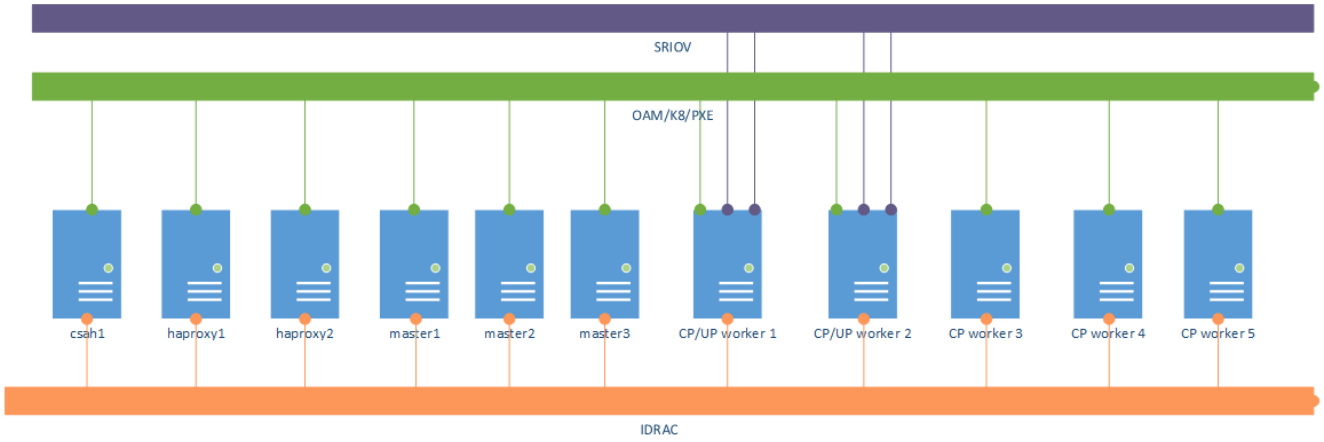

Figure 5. UnityCloud deployment network diagram

Chapter 4 Networking Infrastructure and Configuration

This chapter presents the following topics:

Introduction

The components and operations that make up the container ecosystem each require network connectivity plus the ability to communicate with all other components and respond to incoming network requests. This reference design uses Dell EMC PowerSwitch networking infrastructure.

OpenShift network operations

Operating components

In Kubernetes, containers separate applications from underlying host infrastructure. Each container is assigned resources including CPU, memory, and network interfaces. Kubernetes provides a mechanism to enable the orchestration of network resources through the Container Network Interface (CNI) API.

The CNI API uses the Multus CNI plug-in to enable attaching multiple adapter interfaces on each pod. Container Resource Definition (CRD) objects are responsible for configuring Multus CNI plug-ins.

Container communications

A pod, a basic unit of application deployment, consists of one or more containers that are deployed together on the same compute node. A pod shares the compute node network infrastructure with the other network resources that make up the cluster. As service demand expands, more identical pods are often deployed to the same or other compute nodes.

Networking is critical to the operation of an OpenShift Container cluster. Four basic network communication flows occur within every cluster:

- Container-to-container connections (also called highly coupled communication)
- Pod communication over the local host network (127.0.0.1)
- Pod-to-pod connections, as described in this guide
- Pod-to-service and ingress-to-service connections, which are handled by services

Containers that communicate within their pod use the local host network address. Containers that communicate with any external pod originate their traffic based on the IP address of the pod.

Application containers use shared storage volumes (configured as part of the pod resource) that are mounted as part of the shared storage for each pod. Network traffic that might be associated with non-local storage must be able to route across node network infrastructure.

Services networking

Services are used to abstract access to Kubernetes pods. Every node in a Kubernetes cluster runs a kube-proxy and is responsible for implementing virtual IP (VIP) for services. Kubernetes supports two primary modes of finding (or resolving) a service:

- **Using environment variables**—This method requires a reboot of the pods when the IP address of the service changes.

- **Using DNS**—OpenShift Container Platform 4.6 uses CoreDNS to resolve service IP addresses.

Part of the application (such as front-ends) might require exposure to a service outside the application. If the service uses HTTP, HTTPS, or any other TLS-encrypted protocol, use an ingress controller; otherwise, use a load balancer, external service IP address, or node port.

A node port exposes the service on a static port on the node IP address. A service with `NodePort-type` as a resource exposes the resource on a specific port on all nodes in the cluster. Ensure that external IP addresses are routed to the nodes.

Ingress controller

The OpenShift Container Platform uses an ingress controller to provide external access. The ingress controller defaults to running on two compute nodes, but it can be scaled up as required. Dell Technologies recommends creating a wildcard DNS entry and then setting up an ingress controller. This method enables you to work only within the context of an ingress controller. An ingress controller accepts external HTTP, HTTPS, and TLS requests using SNI, and then proxies them based on the routes that are provisioned.

You can expose a service by creating a route and using the cluster IP. Cluster IP routes are created in the OpenShift Container Platform project, and a set of routes is admitted into ingress controllers.

You can perform sharding (horizontal partitioning of data) on route labels or name spaces. Sharding enables you to:

- Load-balance the incoming traffic.
- “Hive off” the required traffic to a single ingress controller.

Networking operators

The following operators are available for network administration:

- **Cluster Network Operator (CNO):** Deploys the OpenShift SDN plug-in during cluster installation and manages kube-proxy on each node.
- **DNS operator:** Deploys and manages CoreDNS and instructs the pods to use the CoreDNS IP address for name resolution.
- **Ingress operator:** Enables external access to OpenShift Cluster Platform cluster services and deploys and manages one or more HAProxy-based ingress controllers to handle routing.

Container Networking Interface

Affirmed 5GC with Red Hat OpenShift solution is primarily using the OVN-Kubernetes CNI plug-in. The CNI specification makes the networking layer of containerized applications pluggable and extensible across container run-times. The CNI specification is used in both upstream Kubernetes and OpenShift in the pod network. This use is not implemented by Kubernetes, but by various CNI plug-ins. The most commonly used CNI plug-ins are:

- **Multus:** CNI plug-in that supports the multinetwork function in Kubernetes. Typically, Kubernetes pods have only one networking interface, but the use of Multus means that pods can be configured to support multiple interfaces. Multus acts as a “meta plug-in”: a plug-in that calls other CNI plug-ins. In addition to other CNI plug-ins, Multus supports SR-IOV and DPDK workloads.

- **DANM:** Developed by Nokia, DANM is a CNI plug-in for telecom-oriented workloads. DANM supports the provisioning of advanced IPVLAN interfaces, acts like Multus (as a meta plug-in), can control VxLAN and VLAN interfaces for all Kubernetes hosts, and more. The DANM CNI plug-in creates a network management API to give administrators greater control of the physical networking stack through the standard Kubernetes API.

OpenShift SDN

The OpenShift Software Defined Network (SDN) creates an overlay network that is based on Open Virtual Switch (OVS). The overlay network enables communication between pods across the cluster. OVS operates in one of the following modes:

- Network policy mode (the default), which allows custom isolation policies
- Multitenant mode, which provides project-level isolation for pods and services
- Subnet mode, which provides a flat network

OpenShift Container Platform 4.6 also supports using Open Virtual Network Kubernetes (OVN Kubernetes) as the CNI network provider. OVN-Kubernetes will become the default CNI network provider in a future release of OpenShift. OpenShift Container Platform 4.6 supports additional SDN orchestration and management plug-ins that comply with the CNI specification.

Service Mesh

Distributed microservices work together to make up an application. Service Mesh provides a uniform method to connect, manage, and observe microservices-based applications. Service Mesh is not installed automatically as part of a default installation. You must use operators from the OperatorHub to install Service Mesh.

Service Mesh has key functional components that belong to either the data plane or the control plane:

- **Envoy proxy:** Intercepts traffic for all services in Service Mesh. Envoy proxy is deployed as a sidecar.
- **Mixer:** Enforces access control and collects telemetry data.
- **Pilot:** Provides service discovery for the envoy sidecars.
- **Citadel:** Provides strong service-to-service and end user authentication with integrated identity and credential management.

Users define the granularity of the Service Mesh deployment, enabling them to meet their specific deployment and application needs. Service Mesh can be deployed at the cluster level or project level. For more information, see the OpenShift Service Mesh documentation.

SR-IOV and multiple networks

Single Root Input/Output Virtualization (SR-IOV) enables the creation of multiple virtual functions from one physical function for a PCIe device such as NICs. In the network, you can use this capability to create many virtual functions from a single NIC, where you can attach each virtual function to a pod. Latency is reduced because of the reduced I/O overhead from the software switching layer. You can also use SR-IOV to configure multiple networks by attaching multiple virtual functions with different networks to a single pod. You can configure SR-IOV in OpenShift by using the SR-IOV operator, which can create virtual functions and provision additional networks. Dell Technologies has validated

Intel XXV710 25G and Mellanox CX-4 NIC cards with OpenShift Container Platform 4.6 and supports using the cards with the platform. For more information, see the Dell Technologies – Red Hat OpenShift Container Platform Reference Architecture for Telecom Deployment Guide at the Dell Technologies Solutions Info Hub for Communication Service Providers.

Chapter 5 Cluster Scaling

This chapter presents the following topics:

Introduction

This chapter describes node design options that enable you to build a cluster for a wide range of workload handling capabilities. This expands on information in the Technology Overview chapter. Usually, the platform design process ensures that the OpenShift Container Platform 4.6 cluster can meet initial workloads. The cluster must also be able to be scalable as the demand for workload handling grows. With a clear understanding of your workloads, it is easier to approach CPU sizing, memory configuration, network bandwidth capacity specification, and storage needs. Many operational factors can affect how the complexity of a container ecosystem influences operational latencies. Dell Technologies recommends adding a safety margin to all physical resource estimates. Our goal for providing this information is to help you get Day 2 operations running as smoothly as possible.

Cluster scaling

The design and architecture of OpenShift Container Platform places resource hosting limits on an OpenShift cluster. The following table shows the limits:

Table 2. OpenShift cluster hosting limits

Resources	Limit
Nodes per cluster	500
Pods per cluster	62,500
Pods per node	500
Pods per core	Not specified limited by maximum pods per node
Namespaces per cluster	10,000
Number of builds	10,000 (based on 512 MB RAM per image)
Pods per namespace	25,000
Services per cluster	10,000
Services per namespace	5,000
Back ends per service	5,000
Deployments per namespace	2,000

Red Hat offers support for OpenShift Container Platform 4.6 up to these limits. For more information see, Planning your environment according to object maximums.

Requirements planning

Workload resource requirements

This section describes how to size an OpenShift-based container ecosystem cluster by using a sample cloud native application. Memory, CPU core, I/O bandwidth, and storage requirement estimates are indicative of resource requirements at peak load times. The following two tables show the estimated and overall resource requirements for a cloud

native inventory management application with a customized quotation generation system workload.

Table 3. Estimated workload resource requirements by application type per pod

Application type	Number of pods	Maximum memory (GB)	CPU cores	Typical IOPS: Kb/s @ block size (KB)	Persistent storage (GB)
Apache web application	150	0.5	0.5	10 @ 0.5	1
Python-based application	50	0.4	0.5	55 @ 0.5	1
JavaScript runtime	220	1	1	80 @ 2.0	1
Database	100	16	2	60 @ 8.0	15
Java-based tools	110	1.2	1	25 @ 1.0	1.5
Totals per pod	630	19.1	5	N/A	19.5

Table 4. Overall resource requirements

Pods	CPU cores	RAM	Storage	Aggregate network bandwidth
630	630	2,047 GB	1.9 TB	130 Gbps

Our calculations using the workload information from Table 3 account for the following considerations:

- We recommend reserving four physical CPU cores per node for infrastructure I/O handling systems for each compute node configuration.
- Memory configuration is constrained to six DIMM modules per CPU socket (a total of 12 DIMM modules per node).
- DIMM module choices based on current trends for 3,200 MHz memory are 16 GB, 32 GB, and 64 GB. The use of 16 GB DIMM modules results in a minimum node memory configuration of 192 GB.
- NIC options are: 2 x 25 GbE, 4 x 25 GbE, and 2 x 100 GbE.
- Compute node configuration options considers the increased overall node workload handling capacity with processor and memory configuration. The configuration assumes that the compute nodes might be used over time for higher-performance workloads and that additional nodes will be installed to meet future growth in compute, storage, and network areas.

Control-plane node requirements

At a minimum, we recommend the control-plane node configuration is a PowerEdge R640 server with dual Intel® Xeon® Gold 6238 scalable processors and 192 GB RAM. As the Red Hat resource requirements show, this node is large enough for a 250-node cluster and higher. Dell Technologies recommends that you do not scale beyond 200 nodes, which means that the proposed reference design is sufficient for nearly all deployments. The following table shows the sizing recommendations:

Table 5. Control-plane node sizing guide

Number of compute nodes	CPU cores*	Memory (GB)
25	4	16
100	8	32

*Does not include provisioning of at least four cores per node for infrastructure I/O handling.

Cluster hardware planning

Design limits

The Red Hat OpenShift Container Platform 4.6 Reference Architecture design requires a minimum of four servers for a three-node cluster, with each node running as both a control-plane node and a compute node. If required, a three-node cluster can be expanded to a standard cluster. You can also expand the standard cluster with more compute nodes at any time. The maximum configuration that the customized Dell Technologies deployment tools support is 210 servers.

Appendix A Bill of Materials

This appendix presents the following topics:

Software Bill of Materials

Tested BIOS and firmware

The following table lists the PowerEdge R640 BIOS and firmware versions that were tested for this reference architecture guide.

Table 6. Tested BIOS and firmware versions

Product	Version
BIOS	2.10.0
iDRAC with Lifecycle Controller	4.40.00.00
Intel Ethernet 25G 2P XXV710 Adapter	19.5.12
Intel Gigabit 4P I350-t rNDC	19.5.12
PERC H730P Mini	25.5.8.0001
System CPLD	1.0.6

Hardware Bill of Materials

Dell EMC PowerEdge R640 node BOM

The following table lists the key recommended parts per node. Memory, CPU, NIC, and drive configurations are preferred but not mandated.

Table 7. PowerEdge R640 baseline server BOM

Qty	Description
1	PowerEdge R640 server
2	Intel® Xeon® Gold 6238 Scalable processor 2.1 GHz, 22C/44 T, 10.4 GT/s, 30.25 M Cache, Turbo, HT (140 W) DDR4-2933
2	Intel® Ethernet 25G 2P XXV710 Adapter
1	Intel® Gigabit 4P I350-t rNDC
1	PERC H730 Mini controller
12	32 GB RDIMM, 3200 MT/s, Dual Rank
1	Dual, hot-plug, redundant power supply (1+1), 750 W
2	800 GB SSD SAS Mix Use 12 Gbps 512e 2.5 in Hot-plug AG Drive, 3 DWPD, 4380 TBW
1	iDRAC9, Enterprise

Dell EMC PowerEdge R740xd node BOM

The following table lists the key recommended parts per node. Memory, CPU, NIC, and drive configurations are preferred but not mandated.

Table 8. PowerEdge R640 baseline server BOM

Qty	Description
1	PowerEdge R740xd server
2	Intel® Xeon® Gold 6238 Scalable processor 2.1 GHz, 22C/44 T, 10.4 GT/s, 30.25 M Cache, Turbo, HT (140 W) DDR4-2933

Qty	Description
2	Intel® Ethernet 25G 2P XXV710 Adapter
1	Intel® Gigabit 4P I350-t rNDC
1	PERC H730 Mini controller
12	32 GB RDIMM, 3200 MT/s, Dual Rank
1	Dual, hot-plug, redundant power supply (1+1), 750 W
2	800 GB SSD SAS Mix Use 12 Gbps 512e 2.5 in Hot-plug AG Drive, 3 DWPD, 4380 TBW
1	iDRAC9, Enterprise

Dell EMC PowerSwitch BOM

The following tables list the switches used in this reference architecture.

Table 9. Dell EMC PowerSwitch S5248-ON or Dell EMC PowerSwitch S5232-ON BOM

Qty	Description
2	Two data switches per rack, used for HA network configuration

Table 10. Dell EMC PowerSwitch S3048-ON BOM

Qty	Description
1	Used for OOB management