WHITE PAPER–NOVEMBER 2020

Containerized Network Functions on Virtual Machines or Bare Metal?

Securing, Managing, and Optimizing CNFs and 5G Services at Scale

vmware®

Table of Contents

Executive Summary

CSPs are turning to containers to streamline and scale the deployment of network functions and 5G services. A container wraps a network function in a consistent, portable package that can be independently distributed and modified with little effort and few dependencies. Containers then run on a host operating system and share its kernel. The host operating system resides on either a virtual machine or a physical server.

Cost-effectively putting containerized network functions (CNFs) into production hinges on your ability to secure, manage, and automate them at scale in an efficient and integral way. This paper explains how running containers on VMs establishes the perfect catalyst for efficiently and securely operating CNFs at scale. Combining containers and VMs produces a powerful synergy that taps the benefits of both technologies.

Virtual machines let you securely and efficiently run containerized functions and 5G services on software-defined infrastructure that you can easily manage, monitor, scale, automate, and optimize. Bare metal servers, in contrast, can root existing monolithic stacks in place and, in a multi-vendor environment, create silos, making management, automation, and maintenance difficult. Adding CNFs and an orchestrator like Kubernetes to a multi-vendor bare metal environment can compound complexity and further complicate management.

Hardware virtualization was originally developed to address the pain of working with physical hardware, pain that ranges from time-consuming management problems and cash-consuming underutilization to the difficulty of scaling hardware for an elastic workload. By optimizing utilization and simplifying management, virtualization reduces physical hardware costs while improving scalability. The ease of scalability that comes with virtualization is one of the reasons why major public cloud providers use hypervisors and VMs to run containers.

For CSPs, performance, security, and management are key factors. Many of the studies that compare container performance on virtual machines with bare metal overlook the integral requirements of securing and managing containers in a real-world environment.

- Studies show that optimizations in the vSphere CPU scheduler for NUMA architectures quashes the belief that running containers on VMs comes with a performance tax.
- Noisy neighbor situations can cause interference for co-located containers on physical hardware, and cross-container interference can result from containers sharing the same kernel resources or components.
- Kubernetes on bare metal is unlikely to outperform Kubernetes on VMware vSphere, which uses advanced scheduling algorithms to optimize all workloads. A recent test of vSphere 7 with Kubernetes shows better performance compared with a bare-metal Kubernetes node because the VMware hypervisor does a better job at scheduling pods on the right CPUs, thereby reducing random memory accesses.
- Containers alone are inadequate security boundaries; containers do not establish security boundaries and strong isolation as VMs do.
- Running CNFs on bare metal would create a complex patchwork of bolted-on security controls and tools. In contrast, running CNFs on virtual machines lets you impose security by using built-in mechanisms that can be managed at scale without silos.
- Running containers on physical hardware would resurrect difficult infrastructure management and operational problems that hardware virtualization solved years ago.
- Operating containers in production requires lifecycle management, high availability, resource management, data persistence, networking, and automation.

Using VMware Telco Cloud Platform to run and automate containers on virtual machines instead bare metal satisfies the complete set of operational, management, and security requirements for deploying CNFs in production.

"Although containers are sometimes thought of as the next phase of virtualization, surpassing hardware virtualization, the reality for most organizations is less about revolution than evolution. Containers and hardware virtualization not only can, but very frequently do, coexist well and actually enhance each other's capabilities. VMs provide many benefits, such as strong isolation, OS automation, and a wide and deep ecosystem of solutions. Organizations do not need to make a choice between containers and VMs. Instead, organizations can continue to use VMs to deploy, partition, and manage their hardware, while using containers to package their apps and utilize each VM more efficiently."

APPLICATION CONTAINER SECURITY GUIDE, NIST SPECIAL PUBLICATION 800-190

Introduction

Communications service providers are increasingly turning toward containers to accelerate the development and deployment of network functions and 5G services.

Containerization is a form a operating system virtualization. A container holds a self-described application and the software components the application requires. The container runs on a container host operating system like Linux, which provides the container with the components of an operating system, such as the kernel, hardware scheduler, memory page abstraction, and the user space. With more than one container, the containers share the same underlying operating system. The container host in turn resides on either a virtual machine (VM) or a physical server (often referred to as bare metal).

Because each container is self-describing, specifying the computing and networking resources that it needs, it packages an application in a consistent, reproducible way: It can be distributed, reused, and managed with minimal effort and few or no dependencies.

Embodied in the term *cloud-native technologies*, this trend is advanced by using a microservices architecture and a container orchestration system—typically Kubernetes. Microservices break up the functions of an application into a set of small, discrete processes, each of which can be independently developed, deployed, modified, and scaled. Kubernetes automates the deployment and management of containerized applications at scale.

Running containerized network functions (CNFs) in production in a telecommunications network comes with an established set of operational requirements: security, compliance, resource management, scalability, availability, data persistence, networking, and monitoring. CNFs carry an additional requirement: orchestration.

For CSPs, performance is another typical requirement, but although the performance of containers on virtual machines and bare metal is comparable, putting containers into production in a cost-effective and operationally efficient way hinges on your ability to secure and manage containers at scale in an integral way.

You can, at significant risk and expense, build a custom stack on physical hardware to try to fulfill your containerized functions' requirements, or you can use proven, cost-effective, low-risk virtualization solutions as the underlying infrastructure for managing, securing, and orchestrating containerized functions.

But there is more: Combining containers and VMs taps the benefits of each technology, creating an organized whole that is greater than the sum of its parts—which is one reason why the major cloud providers, such as Google and Amazon, use VMs to run containers.[1]

Virtual machines let you securely and efficiently run containerized functions and 5G services in production on software-defined infrastructure that you can easily manage, monitor, scale, automate, and optimize. Containers, meanwhile, empower you to make developers more agile, functions more portable, and deployments more automatable. The combination of the two streamlines the development, deployment, and management of CNFs.

This paper explains how running containers on VMs establishes the perfect catalyst for reliably and robustly operating containerized functions at scale. VMware Telco Cloud Platform™, which uses Kubernetes to orchestrate containers on virtual machines in a software-defined data center and a telco cloud, stands at the center of this combination.

1 Combining containers and virtual machines to enhance isolation and extend functionality on cloud computing, Ilias Mavridis, Helen Karatza, Future Generation Computer Systems, Volume 94, 2019, Pages 674-696, ISSN 0167-739X, https://doi.org/10.1016/j.future.2018.12.035.

"Containers promise bare metal performance, but as we have shown, they may suffer from performance interference in multi-tenant scenarios. Containers share the underlying OS kernel, and this contributes to the lack of isolation. Unlike VMs, which have strict resource limits, the containers also allow soft limits, which are helpful in overcommitment scenarios, since they may use underutilized resources allocated to other containers. The lack of isolation and more efficient resource sharing due to soft-limits makes running containers inside VMs a viable architecture."

CONTAINERS AND VIRTUAL MACHINES AT SCALE: A COMPARATIVE STUDY

Virtual Machines, Bare Metal, and the Transition to 5G

Amid a backdrop of fierce competition and digital transformation, communications service providers seek to develop new business models, simplify operations, and launch new services, all in a quest to increase revenue and expand profit margins. Although 5G opens up new business opportunities, the complex, siloed architecture of CSPs' existing networks stands in the way of rapid innovation and operational agility, hampering the digital transformation.

These existing networks, which tend to be founded on vertically integrated monolithic stacks designed to run vendor-specific virtual network functions (VNFs), make automating deployment and management difficult. Bare metal servers root these monolithic stacks in place and can, especially in a multi-vendor environment, create difficult-to-manage silos. In this environment, maintenance updates can spiral into a complex cycle.

If one of the silos needs an update, for example, you must also check whether the hardware is still supported. Likewise: Have the north-bound APIs of the management system like VNFM changed? Are the VNMFs using any old APIs, or will the VNFM now need to be updated? If the VNFM is updated, will the VNF also need updating? Is there an automation layer using the old VNFM APIs, or will the automation layer also need to be upgraded? If there are hardware differences among the servers, additional components, such as drivers, will likely also need attention. The more silos there are, the greater the challenge.

When CSPs turn to cloud-native technology to run network functions in containers on bare metal alongside VNFs in multi-vendor environments, the complexity spirals further out of control. CNFs require additional interfaces and tools beyond those used by VNFs—such as Kubernetes clusters, container networking interfaces, container image registries, minimalist Linux container hosts, and tools like Helm and Docker—that would make the stack even more difficult to visualize, secure, operate, and maintain.

In this way, infrastructure that relies too heavily on physical hardware without exploiting the abstraction that virtual machines provide makes it difficult to automate multi-tenant, distributed containerized network functions and to deliver the resiliency and reliability that's required in a highly regulated industry with strict service-level agreements and demanding consumers. Several emerging telecommunications regulations, for example, promote security and resiliency through supplier diversity.

To achieve web-scale speed and agility while maintaining carrier-grade performance and quality, CSPs need a platform that combines telco-specific cloud-native solutions and cloud-first automation with consistent infrastructure. CSPs must be able to automate and orchestrate their functions and services across systems from multiple vendors.

The following elements are critical to establishing a modern holistic multi-vendor platform with the power to innovate quickly, scale with elasticity, adopt a multi-cloud strategy, and manage functions and services efficiently:

- Hybrid infrastructure that spans multiple clouds and sites, from the core and the edge to private and public clouds, so you can run hybrid network services that combine functions in different formats.
- Cloud-native technology such as containers and Kubernetes that lets you build, manage, and run containerized network functions (CNFs) across distributed sites.
- Multi-layer, cloud-first automation that unites your infrastructure and multi-cloud resources, including containers and VMs, in a centralized orchestration system.

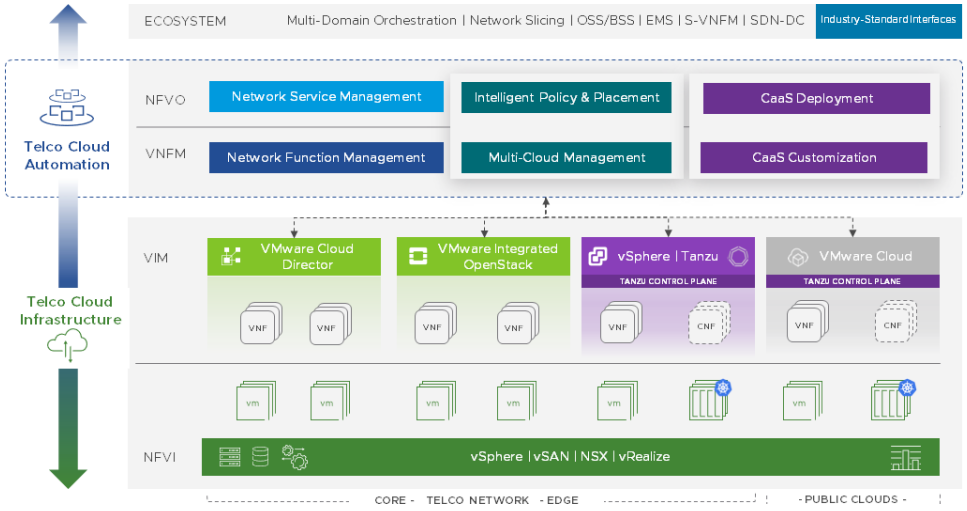

Figure 1: VMware Telco Cloud Automation and VMware Telco Cloud Infrastructure work together to run and manage CNFs and containerized 5G applications on consistent horizontal infrastructure.

VMware Telco Cloud Platform

This paper explains how VMware Telco Cloud Platform and its components radically simplify security, operations, and management of 5G networks and functions by running CNFs on virtual machines. A quick overview of VMware Telco Cloud Platform helps solidify the concepts that appear later in the paper.

By solving the problems that undermine the architecture of existing telecommunications networks—monolithic stacks marred by complexity, silos, vendor lock-in, and an over-reliance on old physical hardware—VMware Telco Cloud Platform empowers CSPs to reduce operational complexity and launch innovative services on consistent infrastructure.

The two fundamental elements of VMware Telco Cloud Platform are VMware Telco Cloud Infrastructure™ and VMware Telco Cloud Automation™.

VMware Telco Cloud Automation orchestrates network functions, services, and resources from a centralized location. The platform integrates with multiple virtual infrastructure managers (VIMs) and Kubernetes clusters to form a powerful multi-tenant environment to securely manage the service and application layer. The platform uses VMware Tanzu Standard for Telco to orchestrate containers, and VMware Telco Cloud Automation centralizes the provisioning and management of the Kubernetes clusters.

VMware Telco Cloud Infrastructure supplies infrastructure as a service (IaaS) and containers as a service (CaaS) with the following virtualization technology: VMware vSphere, VMware NSX-T™ Data Center, and VMware vSAN™. The deployment and management of virtual machines on vSphere furnishes the foundation for running CNFs and VNFs. Tanzu Standard for Telco provides a carrier-grade Kubernetes distribution with telco-grade extensions to run and manage CNFs at scale.

VMware Telco Cloud Platform can be deployed across 5G networks to meet target design and scalability objectives. The VMware telco cloud *reference architecture* simplifies the platform's implementation and describes how the platform gives you a flexible foundation to fulfill various 5G use cases.

By enabling you to deploy virtual network functions (VNFs) and containerized network functions (CNFs) on a consistent horizontal infrastructure, your CSP can evolve from infrastructure as a service to containers as a service.

"Applications can benefit from the security and performance isolation provided by the VM, and still take advantage of the provisioning and deployment aspects of containers. This approach is quite popular, and is used to run containers in public clouds where isolation and security are important concerns."

CONTAINERS AND VIRTUAL MACHINES AT SCALE: A COMPARATIVE STUDY

Cloud-native technology and cloud-first automation

Capitalizing on the opportunities of 5G in a multi-cloud world hinges on two key ingredients of VMware Telco Cloud Platform: CaaS and cloud-first automation.

Containers and Kubernetes decouple network functions from the infrastructure so they can be deployed quickly, shared among services, updated easily, and managed independently. Orchestration and automation dynamically scale network functions to meet changes in demand. By implementing containers as a service (CaaS), CSPs can use the same technology to meet different requirements across their 5G networks. As a result, CSPs can design more efficient 5G networks.

Cloud-first automation unites multi-cloud resources in a centralized orchestration system and then uses intent-based placement for optimization. With cloud-first automation, which continuously synchronizes with registered clouds, CSPs obtain context-aware information about their diverse set of sites, the state of these sites, the applications running there, the embedded technologies available to foster service delivery, and the cloud resources available for allocation. When the orchestrator can access this information, it can recommend placement of network services and functions in a way that aligns requirements with available cloud resources and capabilities. In this way, cloud-first automation further simplifies and optimizes the deployment and management of CNFs.

Performance

The CPU scheduler of VMware ESXi™ enables the hypervisor to provide equivalent or better overall workload performance for containers than multi-purpose Linux operating systems running on physical hardware.

A comparative study by VMware shows that an enterprise web application can run in Docker containers on vSphere 6.5 with better performance than Docker containers on bare metal, largely because of optimizations in the vSphere CPU scheduler for nonuniform memory access (NUMA) architectures, quashing the belief that running containers on VMs comes with a performance tax.[2] vSphere is better at scheduling VMs on NUMA nodes where their memory resides. Linux, on the other hand, tries to maximize processor utilization, meaning processes may be scheduled on different NUMA nodes from their memory, slowing memory access and degrading performance. A *performance analysis* of big data workloads on vSphere shows the same results.

Virtualization can offer better performance isolation than running containers in Linux, especially in noisy neighbor situations. The results of an academic comparative study of containers and VMs at scale show that "co-located applications can cause performance interference, and the degree of interference is higher in the case of containers for certain types of workloads."[3]

Because of how the Linux kernel works, you can also get cross-container interference from containers sharing the same kernel resources or components. It's a mistake to assume that "the kernel is fully isolating every underlying resource at a container granularity."[4]

Kubernetes clusters also benefit from running on vSphere. Kubernetes on bare metal is unlikely to outperform running Kubernetes on vSphere, which uses advanced scheduling algorithms to optimize all workloads, including those using containers. Companies that run Kubernetes on physical hardware can find it complex and difficult to scale and efficiently

2 VMware. "Performance of Enterprise Web Applications in Docker Containers on VMware vSphere 6.5." September 2017. https://www.vmware.com/content/dam/digitalmarketing/vmware/en/pdf/techpaper/performance/dockervsphere65-weathervane-perf.pdf

3 ACM. "Containers and Virtual Machines at Scale: A Comparative Study." Prateek Sharma, Lucas Chaufournier, Prashant Shenoy, Y.C. Tay. November 28, 2016. http://dx.doi.org/10.1145/2988336.2988337

4 Sysdig. "Container isolation gone wrong." Gianluca Borello. May 22, 2017. https://sysdig.com/blog/container-isolation-gone-wrong/

operate the infrastructure. Running Kubernetes clusters on vSphere, in contrast to physical hardware, benefits from years of fine-tuning to make it adept at optimizing the performance of large clusters and mixed workloads.

Moreover, a recent test of vSphere 7 with Kubernetes shows better performance compared with a bare-metal Linux Kubernetes node. A recent blog post titled "How Does vSphere 7 with Kubernetes Deliver 8 Percent Better Performance Than Bare Metal?" shows that vSphere native pods running on vSphere, isolated by the hypervisor, can achieve up to 8 percent better performance than pods on a bare-metal Linux Kubernetes node.[5]

This benefit primarily comes from ESXi doing a better job at scheduling the natively run pods on the right CPUs, thus providing better localization to dramatically reduce the number of remote memory accesses. The ESXi CPU scheduler knows that these pods are independent entities and ensures that their memory accesses are within their respective local non-uniform memory access (NUMA) domain. This results in better performance for the workloads running inside these pods and higher overall CPU efficiency. On the other hand, the process scheduler in Linux does not provide the same level of isolation by default across NUMA domains.

Boosting performance by selecting a Linux kernel version

With containers, an important factor is the version of the container host's kernel and its performance characteristics. Another performance advantage of running containers on virtual machines is that you can not only easily select the container host that you want to use but also maximize the CPU resources of the underlying hardware by running multiple virtual machines, each with its own choice of container host.

In other words, each virtual CPU allocated to each virtual machine on vSphere can run a different kernel; a bare metal server, however, can run only one kernel.

By using virtual machines, you can easily select the container host operating system that works best for the performance demands of your CNF. On vSphere, one example is Photon OS, a security-hardened minimalist host operating system for containers. Its Linux kernel is optimized for performance on vSphere, and it supports new devices such as ARM64 (Raspberry Pi 3) to, for instance, help enable Internet of things applications at edge sites.

Separating the container host operating system from bare metal by using VMs also eases the lifecycle management of those operating systems.

Performance in production environments

Many of the studies that compare the performance of containers on virtual machines and bare metal overlook the integral requirements of securing and managing containers in a real-world environment.[6] Studies that consider performance in a real-world context take into account such integral requirements as security and lifecycle management. One such study evaluating the performance of containers and virtual machines when running a database workload, for instance, includes the following acknowledgment:

"Even though the Docker container solution is showing very low overhead and system resource consumption, it suffers from security problems when storing data, which is crucial for database protection. Comparing containers with VMs, containers cannot be secure candidates for databases because all containers share the same kernel and are

5 How Does vSphere 7 with Kubernetes Deliver 8 Percent Better Performance Than Bare Metal? By Karthik Ganesan and Jared Rosoff, October 31, 2019, at https://blogs.vmware.com/performance/2019/10/how-does-project-pacific-deliver-8-better-performance-than-bare-metal.html. vSphere 7 is planned to be supported in a future release of VMware Telco Cloud Platform.

6 See the hundreds of studies and papers comparing the performance of containers running on virtual machines and physical servers through the *search results on Google Scholar*; many of these studies compare the performance of containers on virtual machines with containers on bare metal without considering the integral requirements of securing, managing, and automating containers and their related tools in a real-world production environment.

"While network virtualization presents risk, it also allows advanced and flexible network protections. For this reason, a well-built virtualised network can be more secure and resilient than an equivalent network built on dedicated hardware."

SECURITY ANALYSIS FOR THE UK TELECOM SECTOR: SUMMARY OF FINDINGS, NATIONAL CYBER SECURITY CENTRE, JANUARY 2020

therefore less isolated than VMs. A bug in the kernel affects every container and results in significant data loss. On the other hand, hypervisor-based virtualization is a mature and (relatively) secure technology."[7]

Security

The transition from 4G networks to 5G, coupled with pressure to protect customer information, increases the complexity of the security landscape for CSPs. The combinatorial nature of 5G, in which service providers can mix elements of 4G and 5G networks, can result in an uneven application of network security measures, which are likely to evolve and shift as the network combines various 4G and 5G elements. The use of public clouds will likely intensify the importance of centralized management and monitoring. The use of CNFs requires special attention to secure them, their containers, and their orchestration system.

In contrast to using bare metal, running containers on VMs improves your ability to secure them and manage their security at scale; the following sections explain why. Security for CNFs involves container images, the containers themselves, their orchestration system, and their interfaces with the network and other services.

Three governmental bodies have produced or are preparing publications documenting security standards for containers or containerized 5G services and functions:

1. National Institute of Standards and Technology (NIST)
2. National Cyber Security Centre (NCSC) of the U.K.
3. European Commission

NIST guidelines for securing containers

In September 2017, the National Institute of Standards and Technology (NIST) published its Application Container Security Guide, also known as NIST Special Publication 800-190. It explains the security concerns with containers and recommends how to address them. The guide exposes several fundamental areas of concern with containers:

- Degree of isolation
- Operating system management and configuration
- Orchestration systems without adequate protection

Containers alone are inadequate security boundaries

Containers are not miniature VMs, and containers do not establish security boundaries as VMs do. An important implication of the Application Container Security Guide is to run containerized functions on virtual machines: Containers, the guide says, "do not offer as clear and concrete of a security boundary as a VM. Because containers share the same kernel and can be run with varying capabilities and privileges on a host, the degree of segmentation between them is far less than that provided to VMs by a hypervisor."[8]

Deploying containers with VMs encases an application with a layer of strong isolation, an approach that is well-suited to cloud-style environments with multitenancy and multiple workloads. "Docker containers pair well with virtualization technologies by protecting the virtual machine itself and providing defense in-depth for the host," a Docker security white paper says.[9]

To run containers that properly isolate tenants on physical Linux hosts, you would need to run different tenants on separate physical machines. The likely outcome is either low

7 Performance evaluation of containers and virtual machines when running Cassandra workload concurrently; Shirinbab, S, Lundberg, L, Casalicchio, E.; Concurrency Computat Pract Exper. 2020; 32:e5693. https://doi.org/10.1002/cpe.5693
8 NIST. "NIST Special Publication 800-190, Application Container Security Guide." Murugiah Souppaya, John Morello, Karen Scarfone. September 2017. https://doi.org/10.6028/NIST.SP.800-190
9 Docker. "Introduction to Container Security." August 2016. https://www.docker.com/sites/default/files/WP_IntrotoContainerSecurity_08.19.2016.pdf

MICRO-SEGMENTATION

Micro-segmentation divides a virtual data center and its workloads into logical segments, each of which contain a single workload. You can then apply security controls to each segment, restricting an attacker's ability to move to another segment or workload. This approach reduces the risk of attack, limits the possible damage from an attack, and improves the overall security posture. When you use virtual machines instead of bare metal in a VMware telco cloud, you can use VMware NSX to establish and manage micro-segmentation for the virtual machines.

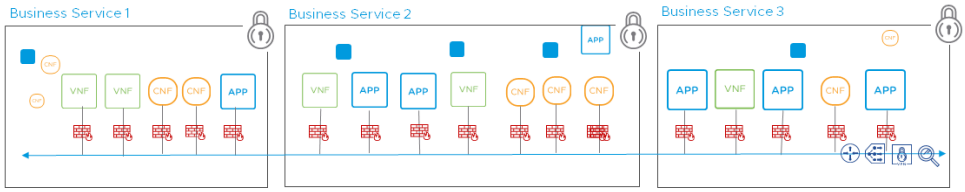

Figure 2: Distributed firewalls and micros-segmentation can isolate network functions on consistent infrastructure.

resource utilization stemming from fragmentation or overly high utilization leading to long wait times for using new hardware. The major cloud providers, such as Google and Amazon Web Services (AWS), isolate the container workloads of tenants by using separate VMs. Because containers are inadequate security boundaries, only highly trusted code should be run in containers on the same VM or physical host.

The same holds true for pods in Kubernetes when you deploy a CNF with Kubernetes. "Ultimately, in the case of applications running in both VMs and containers, the VM provides the final security barrier. Just like you wouldn't run programs with mixed security levels on the same VM, you shouldn't run pods with mixed security levels on the same node due to the lack of guaranteed security boundaries between pods," writes Jianing Guo on the Google Cloud Platform Blog.[10]

Risks of misconfiguration on a physical host

Containers or the operating system of a physical host can easily be misconfigured, increasing the attack surface and the level of risk, the NIST Application Container Security Guide says. "Carelessly configured environments can result in containers having the ability to interact with each other and the host far more easily and directly than multiple VMs on the same host."

In contrast, the abstraction, automation, and isolation of an operating system running on a VM in a hypervisor reduces the attack surface and decreases the risk of a security breach.

Securing the orchestration system

Another concern of the Application Container Security Guide is recommending countermeasures to secure the orchestration system managing containers. The suggested countermeasures in the NIST guide include the following:

- The use of enterprise-grade authentication services using strong credentials and directory services
- Granular access control for administrative actions based on hosts, containers, and images
- Isolating containers to separate hosts based on the sensitivity level of the applications running in them

A second NIST document, Security Assurance Requirements for Linux Application Container Deployments, sets forth security requirements and countermeasures to help meet the recommendations of the Application Container Security Guide when containerized services are deployed in production environments. The orchestration system or its components and tools should have the following capabilities:

- Logging and monitoring of resource consumption of containers to ensure availability of critical resources
- The orchestration system must work with many container hosts, not just one, to be able to provide a global summary of resource usage for all running containers

Running containers on physical hardware and managing the containers with an

10 Google. "Demystifying container vs VM-based security: Security in plaintext." Jianing Guo. August 9, 2017. https://cloudplatform.googleblog.com/2017/08/demystifying-container-vs-VM-based-security-security-in-plaintext.html

orchestration system would require you to connect each physical machine to an authentication and access control system.

To isolate containers by sensitivity level, you would have to use an inefficient number of physical machines. As a result, resource utilization would suffer while management overhead would increase.

Taking advantage of advanced trends

Running containers on VMs lets you take advantage of security innovations in virtualization technology. AMD SEV-ES provides an example. Secure Encrypted Virtualization (SEV) technology integrates memory encryption with AMD-V virtualization to support encrypted VMs, which are ideal for multitenant environments.

SEV with Encrypted State (SEV-ES) builds upon SEV to provide an even smaller attack surface and additional protection for a guest VM from the hypervisor even if the hypervisor is compromised. SEV-ES blocks attacks by encrypting and protecting all CPU register contents when a VM stops running to prevent the leakage of information in CPU registers to the hypervisor. SEV-ES can detect and prevent malicious modifications to the CPU register state.[11]

Securing microservices with VMs

Microservices add another dimension to container security. According to a Docker white paper on security, "Deploying Docker containers in conjunction with VMs allows an entire group of services to be isolated from each other and then grouped inside of a virtual machine host."[12]

NCSC requirements for telecom security

In a landmark analysis at the dawn of 5G, the United Kingdom's National Cyber Security Centre published a January 2020 paper that recommends the establishment of a new set of telecommunications security requirements, or TSRs, that are intended to drive communications service providers (CSPs) to operate secure networks.

Security risks and requirements are shifting as telecommunications providers transition to 5G networks and increasingly rely on virtualization and cloud computing, including containers and Kubernetes. The service-oriented architecture of the 5G core network introduces a broader range of data and services than 4G, increasing the attack surface. The common web protocols and APIs of 5G networks open up more attack vectors.

Containers, in particular, shift the security burden to either virtual machines or bare metal. Kubernetes and cloud-native architectures require security enhancements to lock down API interfaces, manage microservices, and protect network end points. The risks and attack vectors that accompany 5G give rise to key security imperatives for protecting the CNFs and their infrastructure, including Kubernetes as well as the virtual machines or bare metal that the CNFs run on. Reducing risk relies on your ability to do at least the following:

- Keep container images and their underlying machines and operating systems up to date and patched.
- Maintain the deployment fabric en masse and at scale, including the orchestration system, container images, containers' underlying operating systems, and the machines, whether virtual or bare metal, on which CNFs run.
- Implement mitigations that neutralize known attack vectors.
- Isolate machines with security domains and pools to prevent movement.
- Protect sensitive data through segmentation of workloads and storage.

INTRINSIC SECURITY

With the VMware Telco Cloud, security is intrinsic—integrated with the software and built into the infrastructure so that security is programmable, automated, adaptive, and context-aware. Intrinsic security improves visibility, reduces complexity, and focuses your defenses by enabling you to apply and automate adaptive security measures like micro-segmentation in the right place.

11 AMD. "Protecting VM Register State with SEV-ES." David Kaplan. February 2017. https://www.amd.com/system/files/TechDocs/Protecting%20VM%20Register%20State%20with%20SEV-ES.pdf

12 Docker. "Introduction to Container Security." August 2016. https://www.docker.com/sites/default/files/WP_IntrotoContainerSecurity_08.19.2016.pdf

VMWARE TELCO CLOUD PLATFORM AT A GLANCE

Powered by field-proven virtual infrastructure and cloud-first automation, VMware Telco Cloud Platform is a cloud-native platform that enables CSPs to rapidly deploy and efficiently operate multi-vendor CNFs and VNFs with agility and scalability across 5G networks.

KEY BENEFITS

- Deploy virtual network functions (VNFs) and containerized network functions (CNFs) on consistent horizontal infrastructure
- Quickly provision and optimize Kubernetes clusters on demand
- Manage CNFs at scale
- Optimize the placement of network functions
- Automate lifecycle management of Kubernetes clusters, network functions, and services
- Follow a reference architecture from VMware to implement a solution that works best for your requirements
- Add optional components to meet your requirements, including VMware vRealize® Suite, VMware vSAN, VMware Telco Cloud Operations, and a VIM — either VMware Cloud Director or VMware® Integrated OpenStack

LEARN MORE

For more information about VMware Telco Cloud Platform, call 1-877-VMWARE (outside North America, dial +1-650-427-5000) or visit *https://telco.vmware.com/*

- Encrypt data in transit and at rest.
- Architect the infrastructure by using automated provisioning, automated management, secure administration, and micro-segmentation.
- Strictly control access to and use of the CNFs' management layer by using the principles of least privilege and separation of duties.
- Monitor and audit the machines that run containers and Kubernetes.

A higher-level risk stems from the perceived cost of implementing security controls for some of these attack vectors.

Built-in security for virtual machines

Mitigating security risks in a complex multi-cloud environment that mixes 4G and 5G as well as VNFs and CNFs requires security to be built into the networking fabric, the stack, and the management systems. Otherwise, managing security will spiral out of control with too much complexity, too many silos, and an intricate patchwork of security tools.

Security that is built into the software and infrastructure improves visibility, reduces complexity, and focuses your defenses by enabling you to apply and automate adaptive security measures like micro-segmentation in the right place.

Running CNFs on bare metal would create a complex web of bolted-on security controls and tools.

In contrast, running CNFs on virtual machines lets you impose security by using built-in, proven mechanisms that can be managed en masse, at scale, and without silos.

For more information on protecting telecommunications infrastructure with build-in security measures, see *Intrinsic Security for Telco Clouds* and *Intrinsic Security for Telco Clouds at the Dawn of 5G*.

European Union toolkit for cybersecurity of 5G networks

The European Commission has published a paper listing risk-mitigating measures for 5G networks. *Cybersecurity of 5G networks EU Toolbox of risk mitigating measures* identifies a common set of measures that help mitigate the main cybersecurity risks of 5G networks identified in an EU coordinated risk assessment report.[13]

Among other things, the EU toolbox aims to ensure that "each operator has an appropriate multi-vendor strategy to avoid or limit any major dependency on a single supplier (or suppliers with a similar risk profile)."

The measures in the toolbox, which are similar to the high-level requirements espoused by the NCSC, include the following:

- Ensuring the application of baseline security requirements (secure network design and architecture)
- Evaluating and ensuring the implementation of security measures in existing 5G standards
- Ensuring strict access controls
- Increasing the security of virtualized network functions
- Ensuring secure 5G network management, operation, and monitoring
- Reinforcing software integrity, updates, and patch management

13 Cybersecurity of 5G networks EU Toolbox of risk mitigating measures, CG Publication, January 2020, NIS Cooperation Group, https://ec.europa.eu/newsroom/dae/document.cfm?doc_id=64468; European Commission paper overview at https://ec.europa.eu/digital-single-market/en/news/cybersecurity-5g-networks-eu-toolbox-risk-mitigating-measures.

Infrastructure Management, IT Operations, and Lifecycle Management

Running containers on physical hardware would resurrect difficult infrastructure management and operational problems. Kubernetes can manage containerized functions and services, not the underlying infrastructure on which they are running. If you were to choose to use physical hardware as the underlying infrastructure, you must address a number of requirements while avoiding the creation of difficult-to-manage silos:

- Infrastructure deployment and configuration
- Patching, updating, and upgrading
- Backup and disaster recovery
- Logging and monitoring

Multiple infrastructure silos can duplicate teams, tooling, and processes. Rather than driving focused innovation on a single platform, IT ends up repeatedly performing the same tasks.

By running containers on physical hosts, many of the old problems that virtualization solved would come back to plague IT at the same time that IT is under pressure to increase agility, help accelerate time to market for new 5G services, improve security, and minimize costs—all without increasing complexity and risk. These are now core IT requirements, or rapidly becoming new requirements. But as heterogeneous cloud services enter the telecommunications landscape, IT is finding it more and more difficult to fulfill these requirements.

A VMware software-defined data center (SDDC) with VMware Telco Cloud Platform solves these problems with a comprehensive, flexible solution that uses the power of automation to deploy and manage multiple Kubernetes clusters as well as to patch and upgrade the container host operating system.

Because VMware Telco Cloud Platform provides containers as a service through Tanzu Standard for Telco, you can quickly fire up Kubernetes clusters by using your existing virtualized networking and storage—an approach that is likely to be much faster than provisioning containers on bare metal.

VMware Telco Cloud Platform includes multi-layer lifecycle management to automate the provisioning and management of all the layers of the telco cloud, including virtual machines, to reduce costs.

TELCO-GRADE KUBERNETES

The CaaS functionality of VMware Telco Cloud Platform simplifies the operation of Kubernetes for multi-cloud deployments, centralizing management and governance for clusters. The platform provides telco-grade CaaS enhancements, such as the following:

- Multus to attach multiple container networking interfaces to Kubernetes pods through its plugins
- Topology Manager to optimally allocate CPU memory and device resources on the same NUMA node to support performance-sensitive applications
- Kubernetes cluster automation to simplify deployments and management of Kubernetes master and worker nodes.

With these enhancements, CSPs can take advantage of a telco-grade Kubernetes platform to address emerging 5G use cases.

Availability

Containerized functions that are perfectly architected with microservices can rely on a container orchestration system like Kubernetes to manage availability. But most containerized applications are not perfectly architected: They may be replatformed monolithic functions separated into a few macro components, or they may be partially refactored with some microservices and some remaining n-tier patterns. Such functions continue to rely, at least in part, on the underlying infrastructure for availability. Proven technologies from a VMware software-defined data center—such as VMware vSphere vMotion®, VMware vSphere High Availability, and VMware vSphere Distributed Resource Scheduler™ (DRS)—are essential to maintaining availability.

To meet the operational policies and SLAs for workloads, closed-loop automation is necessary across shared cloud infrastructure. VMware Telco Cloud Infrastructure uses vSphere DRS to optimize the initial and runtime placement of workloads to ensure health and performance for cloud infrastructure. Organization VDCs and workloads are monitored to ensure that the resources are being tuned and balanced dynamically. Capacity and demand planning, SLA violations, performance degradations, and issue isolation capabilities can be augmented with the analytics-enabled reference architecture.

The analytics-enabled architecture provisions a workflow automation framework to provide closed-loop integrations with NFVO and VNFM for just-in-time optimizations.

Resource Management

Kubernetes provides powerful quality-of-service (QoS) mechanisms to share a cluster between teams running different workloads. Running Kubernetes clusters on vSphere complements the QoS mechanisms of Kubernetes, especially when the workloads require strong workload isolation. The advanced scheduling and dynamic resource management of vSphere help reclaim and share unused resources between teams or across clusters.

The resource management capabilities of vSphere include dynamic rebalancing, resource pools, shares, reservations, limits, and safe overcommitment. All these capabilities empower you to run traditional and containerized workloads on common infrastructure while ensuring optimal performance and preventing interference between workloads.

For Kubernetes clusters running on vSphere, such VMware technologies as vMotion and DRS maximize hardware utilization by dynamically rebalancing clusters without disrupting workloads.

VMware Telco Cloud Platform extends resource management to telco-specific networks through powerful intent-based placement and dynamic resource allocation with late binding to optimize a Kubernetes cluster for a given CNF.

Intent-based placement through service-aware infrastructure

VMware Telco Cloud Platform can optimize resource utilization when you analyze infrastructure usage and service requirements across clouds. Based on holistic information gleaned from continuously synchronizing with registered clouds, VMware Telco Cloud Platform recommends where containerized network functions should be deployed. This capability improves resource utilization and operational efficiency by dynamically adjusting the deployment schema. As a result, you can architect your 5G systems for optimal application response, scale, and service availability.

Dynamic resource allocation and late binding for optimization

A CNF is placed using late binding in Kubernetes clusters that were fine-tuned during instantiation to meet the CNF's requirements. The container network interface (CNI) and the operating system for the container host are configured to fulfill the requirements of the CNF. This automation improves the resource utilization of clusters. More specifically, during the workload instantiation process, if none of the available Kubernetes clusters is suitable, the system will optimize an existing cluster or create new ones that match its network function requirements.

Data Persistence

Although many containerized applications are stateless, the drive to port network functions to containers is generating requirements to host stateful apps on containers and to furnish them with host-local, shared-nothing storage for data persistence across hosts.

Managing physical storage devices, however, is a painful, manual process often worsened by service-specific workflows. Adding new SSDs to expand capacity is inefficient. Different technologies and processes across infrastructure silos can complicate the situation, and dedicating physical hardware to an individual application is uneconomical.

A single software-driven model that works for network functions and services, whether containerized or not, radically simplifies storage management, operations, troubleshooting, capacity expansion, and storage operations like backup and disaster recovery. By providing a distributed, shared-nothing storage abstraction, VMware vSAN

HIGH PERFORMANCE IAAS AND CAAS INFRASTRUCTURE

VMware Telco Cloud Platform enables CSPs to deploy both CNFs and VNFs on consistent horizontal infrastructure. The platform includes a Kubernetes distribution—Tanzu Standard for Telco—that is designed to support telecommunications use cases. Tanzu implements containers as a service for deploying and managing CNFs. With NSX-T providing enhanced networking between network functions, the platform offers high performance and scaling, with the following functionality providing examples:

- VMware NSX managed Virtual Distributed Switch in Enhanced Data Path mode (N-VDS (E)) that leverages Data Plane Development Kit (DPDK) techniques to provide a fast virtual switching fabric on VMware vSphere
- Low-latency data plane through CPU pinning, fine-grained non-uniform memory access (NUMA) placement, and vertical NUMA alignment
- Improved performance through multi-tiered routing, bare-metal NSX Edge nodes, and huge pages with the access efficiency of translation lookaside buffers

simplifies storage operations and consolidates workloads, both traditional and cloud native, on the same storage infrastructure. For both virtual machines and containers, virtualized storage not only hides the complexity of adding storage capacity but also makes the task much easier.

Scalability

Hypervisors were originally developed to address the pain of working with physical hardware, pain that ranges from time-consuming management problems and cash-consuming underutilization to the difficulty of scaling hardware for a developer environment or an application's expanding workload. By optimizing utilization, virtualization lets you reduce physical hardware costs while improving scalability.

If IT operations deploys a bare-metal server with a container runtime to which developers can push their containers, scaling the system is difficult and time consuming: You would have to add another bare-metal server, install a container runtime on it, manually hook the server up to the network, and connect the runtime to a container orchestration engine.

In contrast, with a hypervisor, you can connect a new bare-metal server to the container domain in minutes. The ease of scalability that comes with virtualization is one of the reasons that major public cloud providers use hypervisors to run their container services. For example, when you create a Kubernetes cluster on Google Container Engine or Amazon Elastic Container Service, the respective cloud provider fires up one or more VMs. The turn-around time is much faster with VMs than bare metal.

The same formula holds true on a global scale. If you were to build Kubernetes on bare metal in a data center, how would you scale it to 1,000 sites around the world while maintaining security, networking, monitoring, and centralized management?

Networking

Containers rely on three levels of abstraction within networking:

- Underlay network
- Overlay network
- Service mesh

The underlay network connects machines, whether virtual or physical, by using either a traditional hardware-based approach or a combination of hardware and software. The overlay rides on top of the underlay to provide networking, such as IP addresses and ports, for the lifecycle of containers and hosts. The service mesh moves above IP addresses and ports to focus on connecting services for containerized applications.

In this context, how would you securely and efficiently connect a large number of bare-metal container hosts running Linux? If you use a flat L2 network that pushes everything to the overlay, you would have to group containerized functions by sets of physical hosts (because containers alone are inadequate security boundaries).

If the networking for the container is not isolated and not tied to the container's lifecycle, however, an attacker could gain connectivity to all the other physical hosts in the environment. To ensure network security, the environment would require hardware-level network isolation with VLANs, east-west firewalls, or other techniques—all of which call for manual, time-consuming management that is not tied to the containerized function's lifecycle. And when the lifecycle changes, more manual, error-prone network changes would be required.

NSX-T Data Center also supports a single underlay network on VMs to provide end-to-end connectivity and management for both CNFs and VNFs. A single underlay network comes with several advantages:

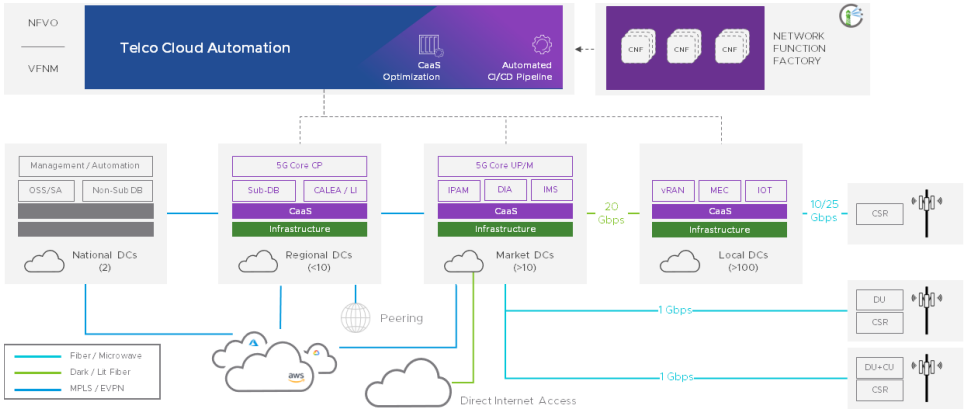

Figure 3: VMware Telco Cloud Automation provides CaaS for running and managing containerized network functions.

- Connect containerized functions to traditional, non-containerized components like databases with ease.
- Radically simplify network management with centralized policies and advanced security, such as micro-segmentation.[14]
- Select the overlay network and the service mesh that works best for your containerized function.

NSX Data Center integrates with the container network interface to furnish an overlay network. When a new containerized function is deployed, NSX Data Center can automatically create a new virtual network that fully isolates the function from all other functions in the environment.

Container networking for Kubernetes clusters

NSX-T Data Center supplies Kubernetes clusters with advanced container networking, security policies, and micro-segmentation. It furnishes networking between CNFs and VNFs. You can quickly deploy networks with micro-segmentation and on-demand network virtualization, including load balancing and ingress services. NSX also furnishes security policies. Multus lets you establish multiple network interfaces for Kubernetes pods to address 5G telco use cases with containers.

NSX-T Data Center delivers an immediate, far-reaching impact on network operations for containers on VMs in a telco cloud:

- NSX-T Data Center load balancers provide highly reliable, high-performance distribution of traffic to applications running on a Kubernetes cluster.
- Policies for micro-segmentation go beyond the standard security policies of Kubernetes.
- Network polices can help secure traffic across Kubernetes namespaces and between pods in the same namespace.
- Operational tools and troubleshooting utilities can debug inter-pod communication.

CSPs can capitalize on the advantages of a microservices architecture with micro-segmentation. You can use microservices with a resource-optimized Kubernetes for device attachment, NUMA alignment, resource reservation, and placement. This cloud-native architecture establishes the foundation to roll out 5G networks with Multus for container networking, DPDK modules, an SR-IOV plugin, a CPU topology manager for NUMA alignment, and Kubernetes cluster automation tailored for telco use cases.

A tenant's external network can carry network traffic to NSX-T Data Center switches that

THE SYNERGY OF CONTAINERS AND VIRTUAL MACHINES

VMs solve infrastructure-related problems by better utilizing servers, improving infrastructure management, and streamlining IT operations.

Containers solve application-related problems by streamlining DevOps, fostering a microservices architecture, improving portability, and further improving resource utilization.

Running containers on virtual machines produces a synergy that helps CSPs transition from 4G to 5G networks with ease.

14 John Wiley & Sons, Inc. "Micro-segmentation for Dummies." Lawrence Miller and Joshua Soto. 2015.

MULTI-LAYER LIFECYCLE MANAGEMENT AUTOMATION

- VMware Telco Cloud Platform lets CSPs centrally manage and automate the virtualized architecture, from CaaS to network services.
- Application management (G-xNFM) unifies and standardizes network function management across the VM and container-based infrastructure.
- Domain orchestration (NFVO) simplifies the design and management of centralized or distributed multi-vendor network services. CSPs can onboard VNFs and CNFs using standard-compliant TOSCA templates.
- The multi-cloud infrastructure and CaaS automation ease multi-cloud registration (VIM/Kubernetes), enable centralized CaaS management, synchronize multi-cloud inventories and resources, and collect faults and performance from infrastructure up to network functions. Kubernetes clusters can be created and optimized automatically to align with the requirements of network functions and services.

can be either N-VDS in Standard mode, Enhanced mode, or both. This dual-mode N-VDS switching fabric design can be used for accelerated and non-accelerated workloads within the same or separated compute clusters.

In addition, VMware Container Networking with Antrea can enhance Kubernetes network policies to help address custom telco use cases—such as boosting service load-balancing performance—for workloads across multiple clouds.

Accelerating workloads and application-response times

To architect the network for optimum application response times and scalability, VMware Telco Cloud Platform extends the following performance features to CNFs:

- Data Plane Development Kit, or DPDK, an Intel-led packet processing acceleration technology, accelerates data-intensive workloads. Capabilities include optimizations through poll mode drivers, CPU affinity and optimization, and buffer management.
- VMware NSX managed Virtual Distributed Switch in Enhanced Data Path mode (N-VDS (E)) that leverages DPDK techniques to provide fast virtual switching fabric on VMware vSphere.
- N-VDS to offload data plane traffic onto a physical NIC to further improve the Enhanced Data Path performance of VMware NSX-T. Both control and user plane workloads can take advantage of this capability and achieve higher throughput, reduced latency, and scale in performance.
- Improved performance through multi-tiered routing and huge pages with the increased access efficiency of translation lookaside buffers.
- A bare-metal NSX Edge cluster acts as an edge router that connects to a physical router. A bare-metal NSX Edge delivers improved performance, higher throughput, sub-second Bidirectional Forwarding Detection (BFD) convergence, and faster failover.

Workload acceleration with SR-IOV

SR-IOV is a specification that allows a single Peripheral Component Interconnect Express (PCIe) physical device under a single root port to appear as multiple separate physical devices to the hypervisor or the guest operating system. SR-IOV uses physical functions (PFs) and virtual functions (VFs) to manage global functions for the SR-IOV devices. PFs are full PCIe functions that can configure and manage the SR-IOV functionality. VFs are lightweight PCIe functions that support data flow but have a restricted set of configuration resources. The number of virtual functions provided to the hypervisor or the guest operating system depends on the device. SR-IOV enabled PCIe devices require appropriate BIOS and hardware support, and SR-IOV support in the guest operating system driver or hypervisor instance.

Automation

VMware Telco Cloud Platform automates the discovery, registration, and creation of Kubernetes clusters while enabling continuous synchronization between the CaaS layer and VMware Telco Cloud Automation. This synchronization creates constant Kubernetes cluster resource awareness, centralizes fault and performance monitoring, and optimizes workload placements.

Furthermore, during the workload instantiation process, if none of the available Kubernetes cluster profiles is suitable, the system will on-demand or automatically optimize an existing cluster or create a new one to match the cloud-native network function's requirements through late binding configurations such as operating system, PaaS, and networking. Achieving this kind of automation and dynamic optimizations using bare metal is likely to be difficult and expensive.

THE VMWARE TELCO CLOUD

We help communications service providers build, run, manage, and protect telco cloud infrastructure to transform their networks, accelerate the delivery of modern services, and thrive in a multi-cloud world.

The VMware Telco Cloud puts in place consistent infrastructure for operating all generations of cellular and fixed-line technology while leading the way to 5G adoption with solutions for orchestration, automation, optimization, and intrinsic security.

With the VMware Telco Cloud, the imposition of security is adaptive, automated, and context-aware so that CSPs can quickly, effectively, and economically capitalize on new market opportunities without undermining the security of their virtualized network or its management.

At the dawn of 5G, the VMware Telco Cloud combines consistent infrastructure and operations with intrinsic security to give CSPs a strong foundation for digital transformation and rapid innovation.

LEARN MORE

For more information about the VMware telco cloud, call 1-877-VMWARE (outside North America, dial +1-650-427-5000) or visit *https://telco.vmware.com/*

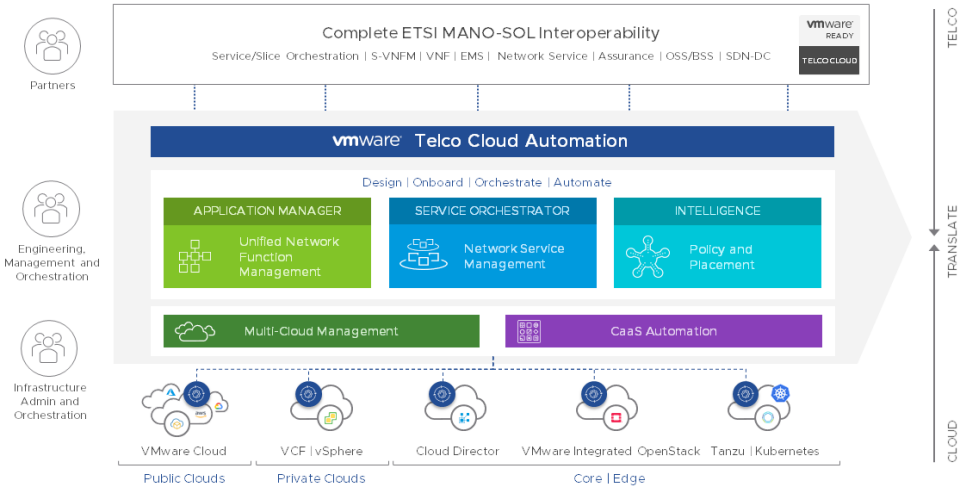

Figure 4: VMware Telco Cloud Automation manages CNFs and VNFs across clouds.

The platform also enables CSPs to automate the onboarding and upgrading of network functions and infrastructure components with zero-touch provisioning. Full lifecycle management can define and apply policies using a decisioning engine to automate deployments, operations, and maintenance.

As investments in infrastructure continue and 5G proliferates, it is increasingly important for operators to match their resources with their operational needs. If you can effectively use your fast-evolving cloud resources to fulfill your operational requirements, it can reduce complexity. Using bare metal as the basis for a multi-cloud strategy that puts a premium on the deployment, management, and automation of CNFs is a step in the wrong direction.

Conclusion: Management, Security, and Automation

Using a VMware software-defined data center with VMware Telco Cloud Platform to run and orchestrate containers on virtual machines instead of physical hardware satisfies the complete set of operational, management, and security requirements for containerized functions while extending their portability to the cloud. The multi-cloud flexibility of using VMware Telco Cloud Platform to run, manage, and automate containers on vSphere yields a range of benefits:

- Secure containers and the orchestration system with isolation, strong security boundaries, authentication, access control, micro-segmentation, and other measures.
- Optimize the performance of large clusters and mixed workloads.
- Scale containerized functions without the pain of adding, configuring, and managing physical hardware.
- Streamline network management and improve network security by using NSX-T Data Center.
- Minimize operational complexity and simplify management while maximizing hardware utilization and economies of scale.
- Automate the deployment and management of CNFs across clouds.

Using VMware Telco Cloud Platform to run containers on virtual machines fuses the benefits of proven virtualization technology with the emerging benefits of cloud-native technology. The combination produces a sustainable multi-cloud solution with the power and flexibility to drive your 5G strategy to fruition.

VMware, Inc. 3401 Hillview Avenue Palo Alto CA 94304 USA Tel 877-486-9273 Fax 650-427-5001 www.vmware.com