WHITE PAPER

Rethinking Kubernetes Ingress

Avi's All-in-one Approach for Modern Enterprises

By Jim Frey, Principal Analyst
Enterprise Strategy Group

January 2026

Contents

Introduction

Running Kubernetes at scale brings challenges, among them legacy ingress optimization solutions that lack automation, elasticity, and deep visibility. The situation has become more urgent with the discontinuation of Ingress NGINX as a primary ingress option. These types of operational gaps complicate traffic management, security, and lifecycle maintenance, creating obstacles for teams aiming to efficiently deliver cloud-native applications. As the industry advances toward the Gateway API standard, organizations need ingress solutions that cannot only address today's needs but also position them for the future. The VMware Avi Load Balancer (Avi) delivers this modernization through a unified, built-in platform that integrates load balancing, application analytics, and security into a single, automated fabric. Designed as future-proof and enterprise-grade, Avi simplifies operations across virtualized and containerized environments while providing consistent policies and observability. Further, Avi's existing, established integration with VMware Cloud Foundation (VCF) enables rapid deployment, scalability, and lifecycle automation for resilient, secure, high-performing application services across private cloud instances.

The Evolution of Application Delivery for Kubernetes

The shift to containers and microservices, now well-established for over a decade, has made container infrastructures mainstream, with Kubernetes dominating as the production platform of choice. Its capacity to support dynamic, resource-intensive workloads and CI/CD pipelines makes it the preferred platform for modern application development, including AI. As workloads scale and demand fluctuates rapidly, application delivery services, especially ingress load balancing, must keep pace, transforming them from simple components into critical design points.

Many traditional and open source solutions for ingress and application services are inadequate for modern containerized environments, leading to significant challenges:

- **Increased operational complexity and risk:** Individual ingress solutions often complicate management and security of Kubernetes environments.
- **Incompatibility with modern architectures:** Appliance-based load balancing is obsolete. Traditional or open source tools frequently lack support for application autoscaling and native integration with peripheral services (like DNS, IPAM, and WAF).
- **Fragmented tooling:** To cover all requirements, organizations resort to using multiple, disparate, multi-vendor products, which increases IT operational complexity and makes troubleshooting difficult.
- **Lack of end-to-end observability:** Multi-product approaches often prevent the comprehensive visibility crucial for container-based applications, hindering application developers and operations teams from identifying errors, security violations, and latencies across applications.
- **Limited flexibility and automation:** Reliance on multi-vendor solutions restricts flexibility and portability. Application and networking services often lack full API-driven programmability, limiting seamless alignment with DevOps, restricting integration into automated CI/CD pipelines, and making consistent, policy-driven multi-tenancy nearly impossible.
- **The Ingress NGINX retirement:** The Kubernetes ingress ecosystem faces immediate risk, as the Ingress NGINX project is retired, leaving only a few months for transition. This leaves thousands of organizations reliant on it without new features or major improvements, receiving only essential patches. Recent security flaws, such as the May 2025 "IngressNightmare" vulnerability, further escalate long-term risk concerns.

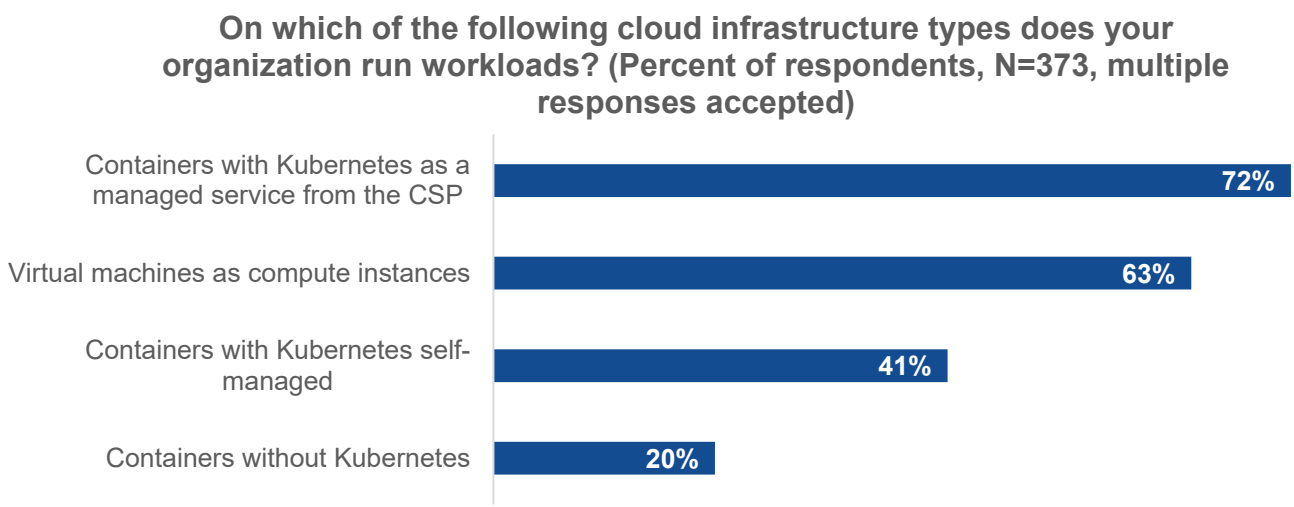
These challenges highlight the strong motivations to transition to an enterprise-grade Kubernetes ingress solution to maintain robust, secure, scalable operation for applications, now and into the future. Avi, with its software-defined architecture, offers a unified, enterprise-grade, and future-ready solution capable of delivering advanced, resilient,

and consistent ingress load balancing across both containerized and virtualized infrastructures. Integrated seamlessly with VMware Cloud Foundation (VCF), including VMware Kubernetes Service (VKS), Avi serves as a native component rather than an add-on, establishing itself as a preferred plug-and-play solution for deploying production-ready Kubernetes applications on VCF. This integration ensures enterprise reliability and enables customers to future-proof their microservices infrastructure investments.

Application Platforms in Transition

The rapid evolution of application delivery in container environments, and Kubernetes in particular, has transformed how software is developed, deployed, and managed in modern IT landscapes. Kubernetes has distinct advantages over older environments, including better scalability and resource management, high availability and reliability, portability and vendor independence, operational efficiency advantages, and more. Today, Kubernetes has become the platform of choice for new development, including most AI applications, particularly within cloud environments (see Figure 1).[1]

Figure 1. Kubernetes Dominates Container Deployments

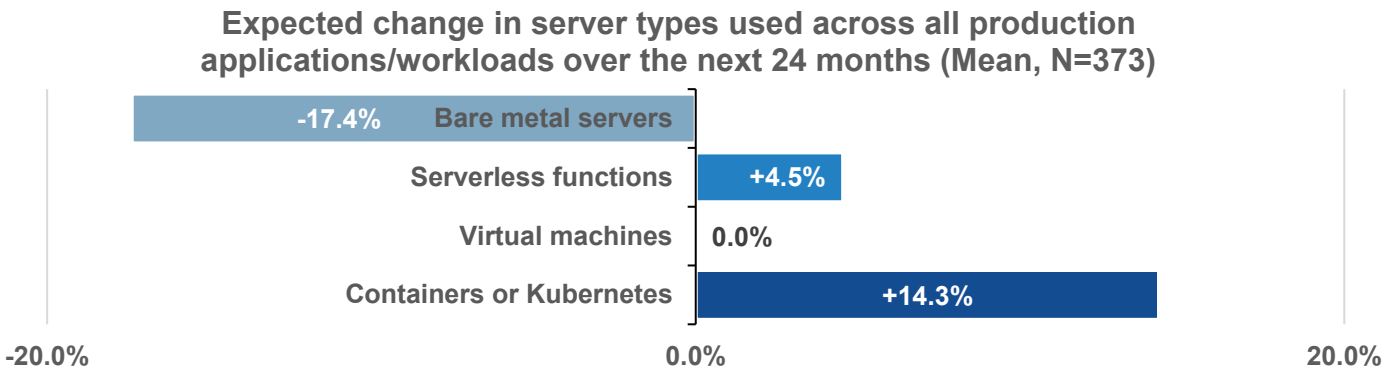


Source: Enterprise Strategy Group, now part of Omdia

Despite its momentum, Kubernetes does not live alone when it comes to the production IT landscape. While the current focus is being placed on containers and Kubernetes as the most modern approach, most organizations still have a mixed operating environment (see Figure 2). Use of containers and Kubernetes is expected to grow, fueled by the transition of newly developed applications into production, but multiple platforms remain in use today, and this is expected to continue to be the norm for at least the next several years. This mixed scenario brings challenges, opportunities, and the necessity to balance specialized or best-of-breed infrastructure components with the need for consistency across environments.

[1] Source: Enterprise Strategy Group Complete Survey Results, *State of DevSecOps and Cloud Security Platforms: Scaling Security Practices to Accommodate Cloud-native Application Development*, May 2025. All Enterprise Strategy Group research references and charts in this White Paper have been taken from this survey results set.

Figure 2. Mixed Application Platforms Exist and Persist



Source: Enterprise Strategy Group, now part of Omdia

A unified experience for these mixed environments would be ideal. However, common application services typically available in traditional application environments, such as load balancing, network performance monitoring, and application security, need to be implemented or approached differently in container-based applications. Typical ingress design objectives include:

- **Local traffic management**, including path-based or host-based routing, protocol support, and load balancing. The goal here is to make sure that inbound traffic is received, processed, and distributed across the cluster in the most optimal fashion.
- **Global load balancing**, which directs clients to the appropriate site/region-based on a range of criteria, including availability, locality of the user to the site, site persistence, and site load
- **Security and authentication**, including authorization policies, rate limiting, TLS/SSL termination, and web application firewall (WAF) capabilities. Objectives here are to minimize threats and prevent unwanted traffic from entering clusters or otherwise interfering with normal activities.
- **Scaling and performance**, including connection pooling, caching, and session affinity. These are optimizations that are designed to accommodate load variability and maximize the speed and efficiency of sessions.
- **High availability and resilience**, including multi-zone deployments, health checking, and graceful degradation. The purpose of these capabilities is to ensure minimal disruption to cluster microservices across a range of potential failure or degradation scenarios.
- **Observability, monitoring, and configuration**, including comprehensive logging and metrics, alerting, and API-driven programmable configurability. These components ensure constant visibility into the health and proficiency of critical ingress functions and alignment with the highly automated practices typically employed for setting up and maintaining Kubernetes-based infrastructure.

Fortunately, there are several options for addressing the needs for production deployment of Kubernetes clusters, but not all options can deliver the same breadth or depth of capabilities nor do they all have the same degree of proven enterprise-grade readiness. And few present an opportunity to standardize ingress protection and optimization across all production environments, including but not limited to Kubernetes.

Rethinking Kubernetes Ingress Controls

Given the criticality of cluster ingress and the need to reduce both security and operational risk at every point in a production Kubernetes architecture, it is worth considering a modern approach by examining enterprise-grade ingress load balancing products to fill the gap. The impending Ingress NGINX end of life makes this need more urgent. Such solutions bring several key benefits to the table:

- **Feature depth and breadth:** One of the great advantages of looking toward proven, enterprise-grade load balancing and application delivery solutions is that they have been tried and tested across multiple deployment scenarios and multiple types of organizations and businesses. These solutions have a broad range of fully integrated features and capabilities and were built to support the wide range of needs of production application environments. As a result, they already solve many of the ingress challenges that will be faced by those moving Kubernetes-based applications into production at scale.
- **Native to DevOps:** The hallmark of Kubernetes is programmability and automation. Load balancing solutions that have been designed from the ground up to be software-defined (versus hardware appliance-based) and offering well-formed, mature APIs can be fit into highly automated and virtualized Kubernetes environments without requiring re-platforming. Such solutions align perfectly with preferred "built-in" design strategies, versus having to be "bolted on" around (and outside of) Kubernetes clusters.
- **Proven reliability:** With years of fielded deployments, commercial load balancers have tuned and hardened solutions to meet the capacity, performance, and reliability needs of the most demanding production settings. This type of operational integrity is highly valuable in reducing availability and performance risks.
- **Enterprise support:** Enterprise applications require enterprise-grade support. This includes applications running on Kubernetes. Having access to 24/7/365 support and expertise from trusted vendors can significantly reduce both operational and security risks while also avoiding the need to build deep internal expertise.

Collectively, such approaches can and should be considered both technically viable and operationally advantageous. The necessity of high-performing, secure production ingress requires rethinking and redefining the path to solution and is a perfect example of where rigorous analysis of all viable options should be employed.

Making Kubernetes Apps Production-ready With Avi

Avi's Consolidated Container Ingress solution helps make Kubernetes applications production-ready by seamlessly integrating key capabilities such as application resiliency, enterprise-grade security, DevOps automation, and comprehensive analytics into a single, unified platform. The Avi solution ensures high availability through dynamic scaling and rapid failover, while protecting applications with built-in Web Application Firewall (WAF), access controls, and automated security policies. It accelerates DevOps workflows by providing full lifecycle automation and self-service capabilities, reducing manual effort and operational overhead. Finally, Avi delivers real-time telemetry and actionable insights into application performance, user experience, and security events, enabling faster troubleshooting and optimization.

Designed to simplify management across multi-cluster Kubernetes environments, Avi empowers enterprises to securely deliver scalable, resilient, and highly available Kubernetes applications with confidence and efficiency in the following ways:

- **Truly software-defined (elastic, automated, active-active and self-healing) capabilities**: Avi is the only hybrid cloud load balancer that is fully software-defined, with a decoupled centralized control plane and distributed data plane, enabling full automation and programmability for dynamic Kubernetes environments. This makes it an agile and efficient choice that is also future-proof, supporting the growing demands of generative AI and other advanced technologies that require highly performant, scalable, intelligent load balancing solutions. Key Avi capabilities include elastic scale-out and scale-in based on real traffic patterns,

self-healing service engines that automatically redeploy when unhealthy, dynamic resource optimization, N-way active-active high availability, and centralized management for multi-cluster deployments.

- **Comprehensive Gateway API support:** Support for the Kubernetes Gateway API is essential for organizations seeking a modern, flexible, and scalable approach to traffic management. The Gateway API offers advanced routing, improved security, and better role-based control, ensuring teams can manage complex, multi-cloud environments efficiently. Embracing Gateway API support helps future-proof infrastructure by providing a standardized, extensible foundation that meets the demands of evolving technologies and dynamic application needs. Avi is Gateway API-ready today, offering advanced L4–L7 traffic management, guaranteed long-term support including patch cycles, and production-validated integration with VMware Cloud Foundation (VCF), VMware vSphere Kubernetes Service (VKS), and on-prem environments.
- **Post Quantum Cryptography (PQC) support:** Anticipating the ever-evolving cryptographic threats exponentially expanding in the future, now is a crucial time to invest in solutions that include advanced security. Moving to a platform with built-in PQC capabilities ensures future-proof protection against emerging quantum threats. This proactive step strengthens an organization's security posture and aligns with the evolving standards in Kubernetes environments. Avi is among the first load-balancing platforms to offer PQC-ready cipher support, hybrid cryptographic mode options, and transition pathways aligned with NIST guidance.
- **Application resiliency:** Avi's elastic, software-defined architecture and advanced global server load balancing (GSLB) capabilities significantly enhance application reachability, performance, and resilience. The software-defined architecture supports elastic auto-scaling and healing, delivering high performance and low latency even throughout large-scale deployments or widely fluctuating application load levels. GSLB intelligently distributes traffic across geographically diverse locations, ensuring high availability and disaster recovery by dynamically directing users to optimal cluster locations.
- **Enterprise-grade security:** Embedded security is a top priority at all levels of infrastructure, and Avi's integrated web application firewall (WAF) delivers enterprise-grade container security for ingress via a comprehensive, intelligent security stack that has been purpose-built for modern, distributed applications, including those running in Kubernetes environments. The Avi WAF protects against a wide range of threats, including OWASP Top 10 vulnerabilities such as SQL injection and cross-site scripting, as well as DDoS attacks, brute force attempts, and malicious bots. Avi WAF leverages a distributed application security fabric, enforcing security through closed-loop analytics and application learning mode and enabling adaptive policy creation based on observed traffic patterns.
- **Application insights with real-time monitoring and analytics:** Maintaining optimal application performance and diagnosing service issues within Kubernetes environments requires deep, continuous, real-time visibility into how workloads are behaving within and between clusters. Traditional monitoring tools often fall short, offering only fragmented metrics or delayed insights. Avi bridges this gap with a centralized analytics engine designed specifically for Kubernetes-native observability, simplifying how operators, SREs, and developers monitor, analyze, and respond to real-time traffic, errors, and application health—all without deploying separate observability stacks.
- **Native DevOps experience:** The Avi solution was built from the ground up to enable DevOps teams to efficiently consume and automate load balancing services, leveraging Kubernetes Gateway API and seamless CI/CD pipeline integration. Included are support for additional custom resource definitions (CRDs) for tighter alignment with automated deployment pipelines, enhanced security with end-to-end mutual TLS (mTLS) authentication for Kubernetes, and Gateway API HTTPRoute Enhancements, enabling easy application of Avi web application security.
- **Plug-and-play integration with VCF:** Avi offers true plug-and-play integration with VMware Cloud Foundation (VCF), delivering automated lifecycle management and load balancing for VCF workloads. Deep native integrations with VCF compute (vCenter), VCF networking (NSX), and VKS enable Avi to automatically discover infrastructure objects, configure virtual IPs, DNS, and route without manual intervention. This integration enhances application availability, scalability, and operational simplicity by providing unified local and global load balancing, Kubernetes ingress, and web application security across VMs, containers, and bare-metal workloads.

Conclusion

As Kubernetes adoption continues to surge, and AI development drives ever more strategic demands around this environment, IT teams must find and secure production-quality infrastructure solutions that meet immediate operational needs while also eliminating future barriers. Avi stands out as exactly the type of enterprise-grade solution that enterprises should consider. Avi's proven software-defined architecture delivers the flexibility and automation required, along with native integration with VCF, ensuring seamless support for emerging capabilities like inference routing and load balancing for large language models (LLMs). With features such as the GenAI Assistant—a conversational bot enabling simplified operations and self-service—Avi puts IT teams in a ready position to manage complex AI workloads with ease and efficiency across Kubernetes-based deployments. Avi represents not only a powerful load balancing solution today but also a forward-looking platform built to scale and evolve alongside advancing AI and Kubernetes technologies. Choosing Avi means investing in a solution that is resilient, scalable, and aligned with the future of AI-powered application services, making it an ideal foundation for faster deployment of production-ready applications.

About Enterprise Strategy Group
Enterprise Strategy Group, now part of Omdia, provides focused and actionable market intelligence, demand-side research, analyst advisory services, GTM strategy guidance, solution validations, and custom content supporting enterprise technology buying and selling.

contact@esg-global.com
www.esg-global.com