KuppingerCole Report

EXECUTIVE VIEW

By Richard Hill
August 27, 2020

Radware Kubernetes WAF

Containerized microservices are gaining momentum in IT organizations today, requiring tools such as Kubernetes for automating the orchestration and management of those containers. The Radware Kubernetes WAF meets the unique requirements of the Kubernetes environment to protect its containerized applications and data.

By **Richard Hill**
rh@kuppingercole.com

Content

1 Introduction

The landscape of software architecture is continuing to be refined and is still evolving. Traditionally, software products and its architecture were primarily monolithic — software in which everything from the UI to the data layer and everything in between was developed into a single platform. Next came Service Oriented Architectures (SOA), in which the SOA's architectural style can be thought of as taking a monolithic system and breaking it down into smaller services that work together over a network. SOA is a collection of loosely coupled services that often rely on middleware or an Enterprise Service Bus (ESB) when deployed to an enterprise infrastructure. Although SOA is considered a refinement in software architecture services, it is still regarded as coarse-grained and heavyweight.

Microservices is a software architectural style that is gaining momentum in IT organizations. Microservices can be considered a variant of the SOA architectural style, where the coarse-grained SOA units can be decomposed into even smaller services. Each microservice is characteristically small and autonomous by using a separation-of-concerns and single-responsibility type of design principles, making microservices fine-grained. Microservices also use lightweight protocols and utilize APIs extensively.

Most microservices application architectures use containers (e.g., Docker) to implement their solution. Containers are units of software that encapsulate a lightweight runtime environment for the application code, making them ideal when using the one microservice per container design principle. Tools are used for automating the deployment, orchestration and management of the many containers used in microservices, by utilizing software tools like Apache Mesos, Kubernetes (a.k.a K8s), or Swarm Mode as examples, although Kubernetes has quickly become the de facto tool of choice in this area.

Security solutions for monolithic and SOA architected products and services are well understood, and Identity & Access Management (IAM) and Web Application Firewalls (WAF) products are well-founded. In contrast, security solutions for containerized environments and their orchestration management tools are less established. These environments have new requirements in which traditional security solutions are no longer adequately suited.

New security solutions must be built for this next generation of software architectures to support their unique environmental needs. The security solution should protect the applications and their data, but it also must provide seamless integration and interoperability with the container management and orchestration platforms where the microservices applications reside. These security solutions must also integrate well into the Continuous Integration/Continuous Deployment (CI/CD) pipelines that facilitate the automation of the software delivery process and accelerate product iterations.

Radware has created a WAF designed to meet the requirements of the Kubernetes environment and the containerized microservices applications it manages. Radware is headquartered in North America with offices globally and specializes in application delivery and cybersecurity solutions. Their Application Protection portfolio includes Web Application Firewalls, Bot Management, API & Mobile Security, and, more recently, Microservices protection.

2 Product Description

When it comes to automating the deployment, orchestration, and management of containers, Kubernetes is one of the fastest-growing open-source projects making it the de-facto platform of choice. Kubernetes is used by organizations, both on-premises, and the cloud. Kubernetes services are available on all major cloud platforms such as Azure, IBM, Google, and AWS, to name a few.

Radware Kubernetes WAF offers web application security before each container in a Kubernetes pod where TLS is already terminated rather than at some intermediating gateway, for example, in which it either requires multiple encryptions and decryption phases or exposes the clear traffic over the wire. Also, Radware Kubernetes WAF not only provides integration into the DevOps CI/CD pipeline but further provides insights through analytics via its dashboards.

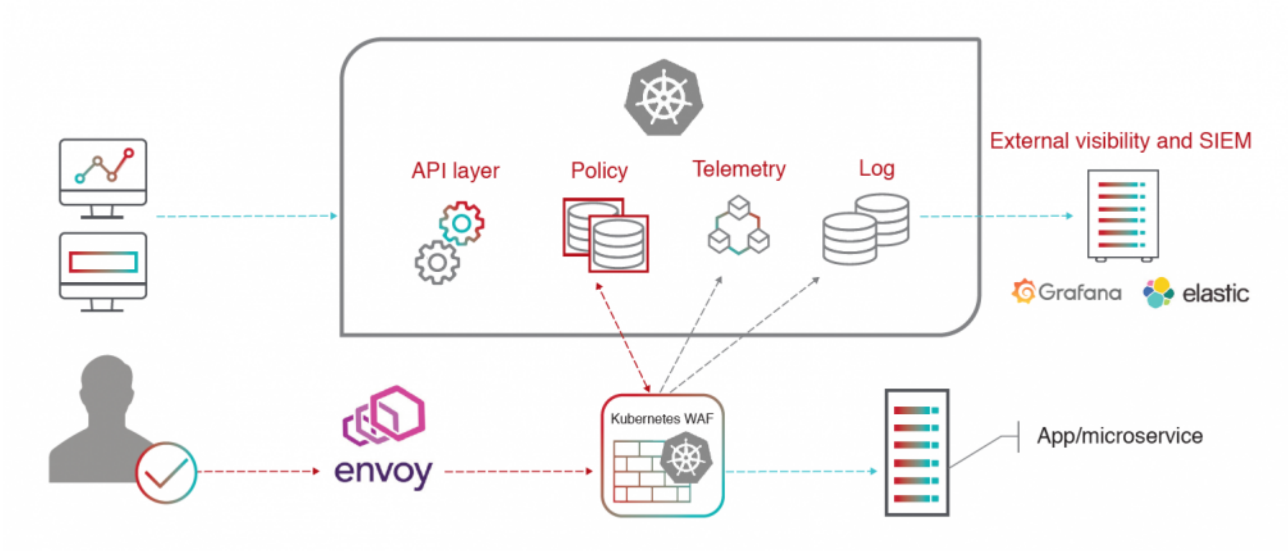

Figure 1: Radware Kubernetes WAF (source: Radware)

Deployment & Integration

Radware Kubernetes WAF is cloud-agnostic and runs on all public infrastructures that support Kubernetes. Popular cloud managed Kubernetes services include Amazon EKS, Azure Kubernetes Service (AKS), and Google Kubernetes Engine.

Radware Kubernetes WAF also supports platforms within on-premises data centers where Kubernetes microservices environments are deployed.

Radware Kubernetes WAF is agnostic to the reverse proxy terminating the TLS at the application level. It can be deployed behind NginX ingress controllers or run in an Envoy-Istio service mesh setup. Radware Kubernetes WAF is set up in a Reverse Proxy mode, as a sidecar, inside the Kubernetes pod in front of the microservices containers, performing web traffic inspection and enforcement.

The Radware Kubernetes WAF is controlled by Kubernetes, allowing for a more lightweight WAF footprint and making it easier to manage. All deployments of instances and resource allocation can be done through Kubernetes. Policies, telemetry, and log collection reside within the Kubernetes management control plane. Kubernetes allows syncing policies across dozens, hundreds, or even thousands of WAF policy enforcement sidecars in the different provisioned pods, allowing management of a single policy across the deployment.

Security Models

There are some limitations to existing Kubernetes solutions in the market today. For example, some focus primarily on container security that protects the container images from known vulnerabilities. Others fail to scan the HTTP payloads, which can lead to data leakage, encoded attacks, or other types of API manipulations.

Radware Kubernetes WAF gives coverage of OWASP Top 10 vulnerabilities of web applications. At the fundamental level, Radware Kubernetes WAF performs the following functions. First, it can terminate the TCP connection to prevent attacks like packet replay attacks, packet reordering, or other WAF evasion techniques. Next, it parses the HTTP to look for header injection or HTTP request splitting, as examples. It will then decode the traffic for an encoded type of attack, as well as prevent against zero-day attacks for example.

Radware Kubernetes WAF combines positive and negative security models against known and zero-day attacks. The negative security model uses the Radware signature and expression engine that provides the rules for known attacks, which is based on Radware’s core WAF technology (named AppWall). Radware Kubernetes WAF also prevents leakage of sensitive information through web applications and APIs by masking sensitive data that comes from a given microservice. Layer 7 block-listing is also given. Radware’s positive security model includes layer 7 allow-listing based on AppWall AllowList module, JSON key-value inspection, IP-based access controls, and validation checks.

In addition to negative and positive security models, Radware offers behavior-based protection. The Radware Kubernetes WAF utilizes both the correlation engine, based on their AppWall source blocking, and behavioral analysis based on the AppWall activity tracking. This behavior-based type of security monitors the traffic to Kubernetes pod's across an application over time to protect against persistent or repetitive sources of attack.

The security enforcement and learning modules also protect RESTful services that meet the OpenAPI specification and prevent access violations, injections, data theft, and denial of service, just like in a web application.

Adaptive Security Policies

Some WAFs rely on behavioral learning of baseline traffic to the traditional type of application. However, there may not be the same application context in a microservices environment, which can lead to incorrect policies. This can happen when a WAF processes all the traffic to the various containerized microservices that may differ from each other. This can lead not only to inaccurate policies and rules, but it would also require substantial policy maintenance (e.g., from exception handling, false-positive alerts, and manual intervention) to adjust to the frequent changes that occur in a microservices environment.

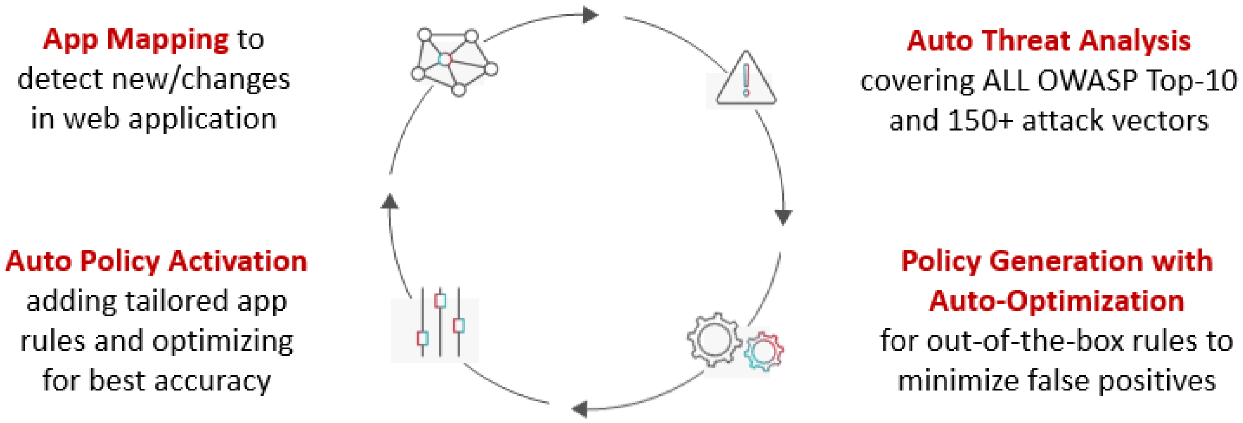
App Mapping to detect new/changes in web application
Auto Threat Analysis covering ALL OWASP Top-10 and 150+ attack vectors
Policy Generation with Auto-Optimization for out-of-the-box rules to minimize false positives
Auto Policy Activation adding tailored app rules and optimizing for best accuracy

Figure 1: Machine-learning Algorithms to Automatically Generate and Refine Policies (source: Radware)

Radware leverages their expertise and machine learning (ML) capabilities with their WAF, such that when it is deployed, it starts to learn the traffic specific to the application. It begins by mapping the threats to the application in multiple phases by inspecting the folders, files, and other sensitive data.

Next, it assesses what types of attacks to expect against those resources and how to protect against those threats. With the assessed attack vector knowledge, Radware Kubernetes WAF generates positive security and the negative security model rules. This process is continuous in that it adapts to newly detected threats and creates the rules to protect against them.

DevSecOps Support

There is sometimes a struggle in organizations in which the DevOps team tends to favor smooth product integrations into the production environments at the expense of security. On the other hand, the security team's primary goal is to protect the organizations' systems and information without impeding the business. Both DevOps and Security teams need to work together in a DevSecOps manner to ensure CI/CD, while at the same time ensuring the security of the applications and its data. To facilitate this, both teams need visibility into the Kubernetes environment.

Radware Kubernetes WAF gives a modern and well laid out dashboard to provide visibility to both DevOps and Security teams. Dashboard widgets display analytical insights of security data and events with the ability to drill down to a granular level. Access to application telemetry, network stats, policies, and latency and performance metrics is also available.

Also given is standard integration with external visibility tools such as Grafana for a DevOps dashboard, or Kibana for a security dashboard. This is accomplished natively through the Kubernetes APIs to the external visibility tools. External SIEM platforms can use log data in the same manner.

3 Strengths and Challenges

Given the increased need to protect microservices at the container management level, the Radware Kubernetes WAF offers the capabilities to meet the needs of web application security in a Kubernetes environment. Radware leverages their Web Application Firewall, and API Security capabilities and experience to bring the WAF technology into the Kubernetes environment.

Radware Kubernetes WAF offers deployment models that run within Kubernetes environments both on the major cloud platforms and on-premises platforms that support Kubernetes microservices environments. Working with Kubernetes requires a level of specialized expertise, but for organizations that have that experience, installing the Radware Kubernetes WAF and defining the relevant rules are straightforward procedures.

Radware provides a WAF defense-in-depth strategy by layering security models from a fundamental level to more advanced levels, incorporating positive and negative security models, API security, and learning the behavioral aspects of security. Bot management against malicious bots is limited, although Radware Kubernetes WAF data leakage prevention is provided, which is a unique capability at the WAF level.

The Radware Kubernetes WAF takes a proactive approach to preventing attacks and maintaining policies. Having the ability to automate the WAF policy generation process is a definite benefit. It reduces the overhead maintenance of updating the policy rules to defend against attacks targeted at the application and reduces gaps or delays in updating rules as new types of attack vectors appear.

A well laid out and modern DevOps and Security dashboard UI is provided. The dashboard provides valuable insights into the WAF security incidents as well as performance. Also, offered is the ability to export log data to Grafana and Elastic Kibana visibility platforms as well as SIEMs.

The benefits of having a dedicated WAF in a microservices container environment are manifold and therefore, beneficial in other microservice container management and orchestration platforms too. Tight integration with Kubernetes prevents use with other container management platforms but has the benefits of being a lightweight solution that leverages the power of the Kubernetes platform.

Radware leverages their expertise with Web Application Firewall, and API Security technologies to bring the WAF into the Kubernetes environment. Radware Kubernetes WAF is worth evaluating for organizations making a move into the containerized microservices development paradigm using Kubernetes.

radware

Strengths

- Strong at basic-to-advanced WAF security models
- Adaptive security policies
- Data leakage prevention
- API Security
- Allows Single TLS termination
- Flexible deployment models
- Modern UI with analytics insights
- Integrates with external visibility tools and SIEM platforms
- Good DevSecOps support

Challenges

- Tight integration with Kubernetes has limitations but many benefits as well
- Requires Kubernetes expertise
- Limited bot management
- Small but well-selected partner ecosystem

4 Related Research

Advisory Note: Software Defined Infrastructures - 71111
Market Compass: Web Application Firewalls – 70324
Whitepaper: The Dark Side of the API Economy - 80019

Content of Figures

Copyright

KuppingerCole Analysts support IT professionals with outstanding expertise in defining IT strategies and in relevant decision-making processes. As a leading analyst ompany, KuppingerCole provides first-hand vendor-neutral information. Our services allow you to feel comfortable and secure in taking decisions essential to your business.

KuppingerCole, founded back in 2004, is a global, independent analyst organization headquartered in Europe. We specialize in providing vendor-neutral advice, expertise, thought leadership, and practical relevance in Cybersecurity, Digital Identity & IAM (Identity and Access Management), Cloud Risk and Security, and Artificial Intelligence, as well as for all technologies fostering Digital Transformation. We support companies, corporate users, integrators and software manufacturers in meeting both tactical and strategic challenges and make better decisions for the success of their business. Maintaining a balance between immediate implementation and long-term viability is at the heart of our philosophy.

For further information, please contact clients@kuppingercole.com.