Signal Sciences

Securing Modern Applications: Containers, Microservices and APIs

BENEFITS

- High performance and massive scale
- Immediate insight to abuse
- Full coverage

Containers, microservices, and APIs work together as the backbone for modern application development. Containers make applications portable to run across different infrastructures. Containers can wrap microservices, which break down applications into smaller, more performant pieces. Microservices can talk to each other and externally using APIs. You've heard the astounding results: companies like Walmart were able to use cheaper commodity infrastructure, experience no down-time on Black Friday, and increase mobile orders immediately by 98%.[1] So how do we secure all this?

There are a lot of great articles on best practices to secure these new architectures. Most advocate focusing on monitoring the most important services. Yet legacy web application firewalls (WAFs) as well as some newer runtime application self-protection (RASP) approaches can't provide visibility or coverage due to a few key problems.

Problems with Legacy WAFs and RASPs

Latency ruins everything

APIs and microservices need to be fast. Other approaches add significant overhead in processing requests, leading to a poor user experience.

Detection is limited

Other approaches can't monitor application misuse and abuse or attacks against APIs and microservices due to difficulties with creating and tuning individual rules.

They can't scale at speed

WAFs and RASPs that rely on updating rules or running in learning mode don't scale when making constant production changes to your APIs and microservices.

Fewer deployment options

Microservices are often composed of dozens or even hundreds of individual services that can be running in different languages and in many containers. WAFs don't show granular attacks at the microservices level, making them harder to stop. RASPs are limited by language support, leaving services exposed.

1 https://blog.risingstack.com/how-enterprises-benefit-from-microservices-architectures/

Why Signal Sciences

High performance and massive scale

Our patented architecture with a module and agent guarantees your site's uptime and performance work as designed. We're deployed in blocking mode on some of the highest trafficked production sites on the internet with no noticeable impact.

Immediate insight to abuse

Without tuning, network or code changes, see where and how your APIs and microservices are attacked, including OWASP top 10 and application misuse and abuse around business logic flaws.

Full coverage

Get visibility and protection wherever your APIs and microservices live. Signal Sciences runs natively in any cloud, data center, or container with the most deployment options in the market, including the application code or web server layer.

A Practical Example: Signal Sciences on Kubernetes

With Signal Sciences, you can easily build in application security to your build process. Just include the module and agent as part of a template in your container build process, whether it's running a web server (NGINX, Apache, IIS) or the application code (PHP, Java, Node.js, Python, Go, and others).

Learn more at labs.signalsciences.com/using-signal-sciences-with-kubernetes.

> More than 95% of customers deploy Signal Sciences in blocking mode in production.

Any App

Cloud, Containers, PaaS, and Serverless

Web Servers and Languages

Gateways and Proxies

Any Attack

OWASP Top 10

Application DoS

Brute force attacks

+ MORE

Any DevOps Toolchain

Slack

HipChat

Datadog

Webhooks

Splunk

SIEM/SOC tools via APIs

+ MORE