42Crunch for Microservices

Extend protection beyond the edge with 42Crunch's automated API Security Platform

MONOLITHIC

VS.

MICROSERVICES

Securing the New Perimeter

Organizations are constantly striving to find faster ways to innovate and scale. Such efforts have naturally led to the rapid adoption of microservices architecture and Kubernetes as the cloud platform to run them at scale. This modular approach allows developers to decompose complex applications into atomic services owned by smaller teams, integrate them, and scale on the fly – bringing applications to production faster than ever before.

The decentralized nature of microservices has led to a proliferation of APIs, which are used by microservices to communicate. Instead of tens, there are hundreds or even thousands of API endpoints that need to be protected. With each individual API becoming a potential target for hackers, attack surfaces are growing exponentially.

Enterprises need to build a zero-trust architecture: microservices must not trust any request, even if coming from the service mesh.

API security has traditionally been enforced at the edge through a gateway pattern with a set of manually configured static rules and policies, but modern applications have redefined the rules of the game.

The vast numbers of microservices and the frequency with which developers are rolling out changes require security to be handled in an automated way and protection to be deployed as a sidecar proxy, together with the microservice through a DevSecOps process that begins at design and extends throughout the entire lifecycle. Traditional processes and solutions no longer work for the new microservices perimeter.

According to Akamai, 83% of web traffic is API traffic.

Secure by Design

Security becomes more complex as microservices are introduced. Companies now have hundreds or even thousands of APIs, and attackers target them to circumvent the normal application security model because it's easier to exploit the systems by going directly against the backend APIs. That means that the overall system becomes only as secure as your weakest microservice behind it.

How do you make sure that no rogue calls get in? Static blacklisting rules, such as the ones looking for SQL injection patterns, do not work because API request and response payloads are specific to each and every API.

AI and machine learning solutions fail because they attempt to magically deconstruct developer intent from traffic that by definition includes both legitimate and rogue calls.

The 42Crunch solution starts with developers. Right within IDEs and CI/CD tooling, we analyze API contracts that developers create, locate all potential vulnerabilities and places where developers are not following the latest security best practices, and tell developers how these can be fixed.

Later, in production, that locked-down contract ensures that APIs are only getting intended calls and API responses contain only intended data. All edge-cases get handled by the API firewall and no rogue calls or data leaking responses can get through.

Automation is King

The microservices revolution is happening for a reason. Decomposing complex systems into smaller microservices that can get developed and maintained by small agile teams launches the speed of innovation to the whole new level.

Hundreds of microservices can push their individual changes and improvements every day helping businesses serve their customers better and out innovate competition. This flow of rapid changes makes manual security scrutiny and configuration impossible. Security teams attempting to do that are bound to become a major bottleneck and miss major vulnerabilities.

The only practical way to enable agile development while maintaining security is turning DevOps processes into DevSecOps.

Security is shifted left and starts with developers. Every time they introduce a new API or change an existing one, 42Crunch analyzes the change and guides developers into keeping the APIs inline with security standards and best practices, and when the new code is moved along the CI/CD pipeline to testing - security tests are automatically run.

When the new version of microservices gets pushed to production, so do the security definitions and policies. Any update to functionality is automatically updating the API firewall configuration so the system remains secure no matter how many changes get introduced at which frequency.

ABOUT 42CRUNCH

42Crunch bridges the gap between API development and security teams with a simple, automated platform that provides auditing, live endpoint scanning, and micro API firewall protection.

API Native
Addresses natively APIs' unique security requirements across data validation, authentication, authorization, confidentiality and integrity.

Positive Security Model
The API Contract is the core of the security configuration, allowing to automatically enforce traffic inbound and outbound.

Integrate into CI/CD
Push your OpenAPI definition to your CI/CD pipeline and automatically audit, scan and protect your API.

API Micro-firewall for Kubernetes
Thanks to its low footprint, 42Crunch API Firewall can be deployed at scale on Kubernetes as sidecar proxy. It has been tested on all major cloud platforms, including Azure, AWS, Google Cloud and Red Hat OpenShift.

Intuitive User Interface
The intuitive interface makes it easy to get started on day one, and provides real-time Security dashboards with actionable data.

Designed for DevSecOps
Enables a seamless DevSecOps experience from development to deployment through automation.

Protection Beyond the Edge

Whatever the intended use scenario you have in mind for your system, you can be sure that attackers will not feel bound by it. They will sniff traffic behind your applications, construct calls with unexpected paths and payloads, iterate over parameters, hack your routers and cloud platforms to get inside the network. Each and every microservice, each layer of its stack, each function it implements becomes an individual target in that vastly expanded surface of attack.

This means that there is no such thing as secure edge anymore and each individual component behind it needs to be protected.

The 42Crunch micro API firewall has been built for such cloud-native applications. Its tiny size of mere 20 MB, submillisecond overhead, and ability to run right within Kubernetes pods as sidecar proxy makes it a perfect way to ensure that each and every microservice gets protected.