IONATE Cloud-Native Security Solution

For innovative advantage, cloud-native is the name of the game. These containers-based environments are the epicenter of the modern scalable, real-time and data-driven applications that enable users to interact with network-connected devices and services. With containers, microservices and the cloud, businesses can develop new features that customers value and appreciate.

And yet, the shift to cloud-native also has an underbelly--multiple new attack vectors that pose a great risk to consumers and enterprises alike. Devices once equated with convenience and connection, including IoT and mobile phones, increasingly represent potential threats.

In its January 2020 “Cyber Crime Prevention: Principles for Internet Service Providers,” the World Economic Forum (WEF) describes the types of risks from the ever-increasing connections between the physical and the online worlds.

For example, as the number of devices connected to the Internet increases, so does the risk posed to the online ecosystem. Poorly secured devices can lead to high-impact attacks. But it’s not just devices. Cloud-native attacks can also prey upon poorly protected interfaces to find valuable data, which is then exfiltrated to another location.

Naming and routing protocols are another vulnerability. Techniques to undermine these protocols are often used in DoS attacks. According to the report, web-based attacks and DoS attacks are the main contributing factors to revenue loss, with an average downtime cost of $221,836.86 for each DoS attack in 2018.

What do these security issues mean for enterprises that are cloud-native to gain a competitive edge? Trouble, potentially, in the form of a damaged brand, consumer trust, as well as monetary loss. That’s because security has not evolved alongside cloud-native technologies. Reasons for this include the speed of adopting new infrastructure technology as well as the pace of change in said technology. Also, more enterprises are now multi-cloud, which means the environment that needs securing typically spans multiple CSPs. Most current security approaches are sub-par. Many are still reactive, only treating the symptoms of an attack. Or, they are incomplete, only protecting for some vulnerabilities, while leaving others wide open. But that doesn’t mean a solution doesn’t exist.

Zero Trust

The only way to ensure security in a distributed, cloud-native environment is through the Zero Trust concept, which is rooted in the idea that enterprises should not trust anything *inside* or *outside* its IT perimeters.

For an environment to be secure, then, the enterprise must verify anything that attempts to connect to its systems before granting access. Further, the security must be as distributed, flexible and responsive as the environment itself. For an organization's security to be this distributed, both micro-segmentation and granular perimeter enforcement are needed to decide whether a user, machine or application trying to access a particular part of the enterprise should be trusted. In addition to micro-segmentation, security that is specific to cloud-native features is also needed, especially given that containers are ephemeral. Implementing this concept requires a complete security solution that is able to:

- **Secure All Data Access Points in an Environment (Ingress and Egress),** including all protocols, networks, and interfaces that comprise a distributed environment's everchanging fabric
- **Constantly Reassess Applications' Interdependencies in an Environment,** as the parameters of activity constantly shift in a cloud-native environment

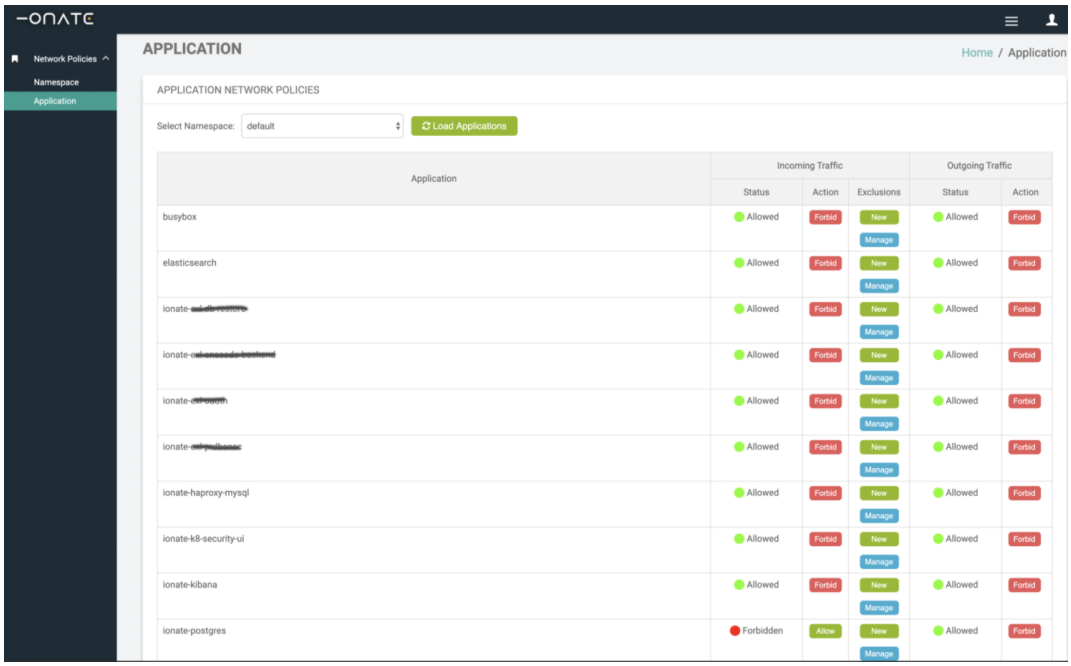
With the IONATE Cloud-Native Security Solution, we meet every challenge and task involved above. With our Zero-Trust model, security is distributed, flexible and responsive, capable of considering the quality and diversity of every data input, source, and type, the associated gateway or access point, as well as all the interrelationships among these multiple components.

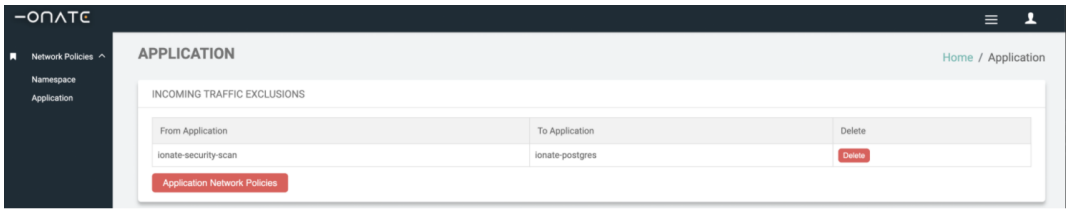
Securing All Data Inputs, Access Points, Networks and Interfaces

The IONATE Cloud-Native Security Solution uses several capabilities, described below, to ensure the security of all internal and external data access/entry points.

Container-Level Firewalls:
Containers do not provide the same isolation as Virtual Machines (VMs). Containers provide resource isolation, VMs provide process isolation. While it may have been a design choice to not build in security at the container level, that doesn't mean it's not needed. A key requirement of cloud-native security is the need to secure microservices dynamically, specifically, in response to changes in an application environment.

Static firewalls are not enough, as firewalls must be close to the asset that needs protection. We implement security at the Container level. Now consider that a microservice could have one container or 20 containers. Protecting each manually is not feasible (again, containers are ephemeral, which means the container IP changes with every deployment). Our dynamic firewalls protect each and every container deployment.



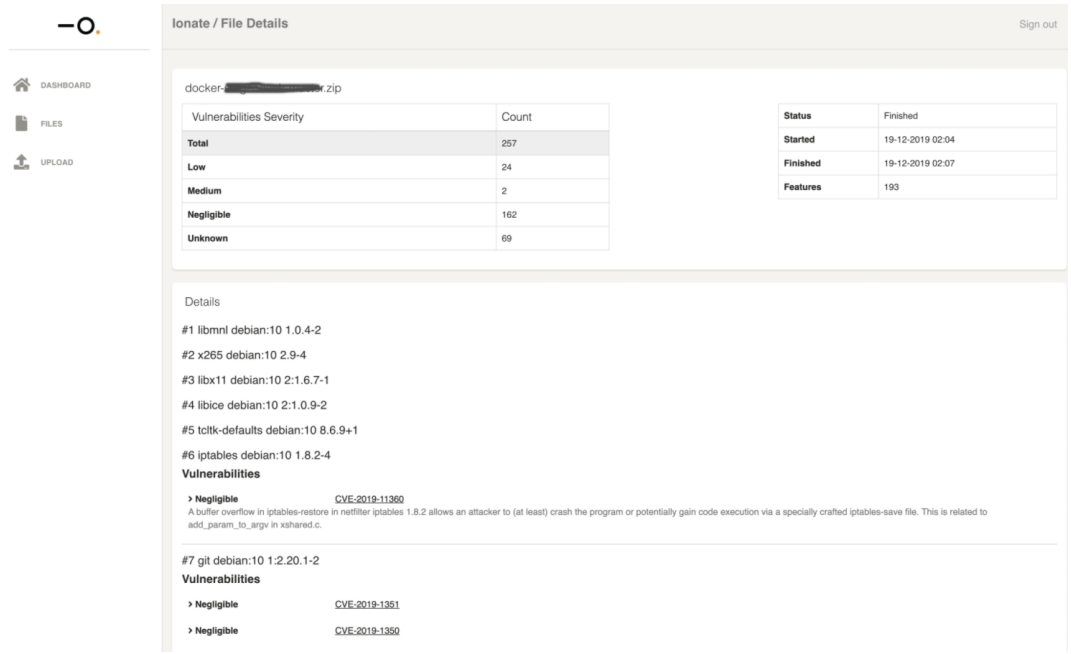


In the IONATE User Interface (UI), each microservice is an application running on top of Kubernetes. Through the Network Policy Manager, users define how containerized apps are allowed to communicate with each other as well as other network endpoints. Network traffic can be restricted in a friendly way. A policy can be set defining what is allowed to communicate with a microservice (incoming and outgoing).

Security Scanner, Proactive Patch Management & Container Vulnerability Management: Containers have multiple layers. Each language, library, and framework is a potential risk. For example, a container running a Java application, such as SpringBoot, can be exposed to threats, stemming from vulnerability coming from an 'underlayer' library that isn't a developer's purview. Our Security Scanner examines an application for all known security vulnerabilities through container vulnerability detection. Based on static analysis, it detects any security threats discovered in each underlying OS, frameworks, and libraries that an app depends on. This includes kernel, language (i.e., Java) or framework layer.

To detect vulnerabilities before or during deployment at each layer of the container, we utilize the Common Vulnerabilities and Exposures (CVE), an international cybersecurity community effort that tracks all known vulnerabilities. NIST's U.S. National Vulnerability Database (NVD) is synchronized with, and based on, the CVE List. Our solution provides a comprehensive description of each vulnerability and links to an upstream database where the issue was published, and where you can track the status and possible resolutions.

Further, we build immediate and Proactive Patch Management & Container Vulnerability Management into the CI/CD build pipeline. We automate policy governance throughout the infrastructure lifecycle to include best practices, organizational standards, and formal regulations. By allowing developers or DevOps to define network policies that, through automation, are always used at build time, security is implemented as part of the fundamental structure of the application. Each time a developer starts a build, this feature is incorporated and automated to detect a container vulnerability.



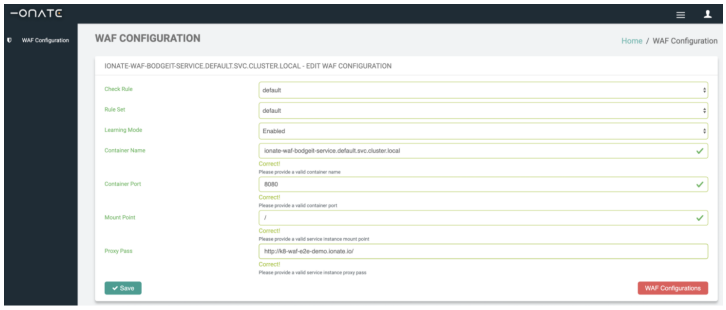
As shown, our Security Scanner detects vulnerabilities before or during deployment at each layer of the container by utilizing the Common Vulnerabilities and Exposures (CVE). When a fix is identified, our platform knows right away. As soon as the fix is available and fixed by a developer, the patch is released immediately through the IONATE CI/CD platform (instead of the usual 3- to 6-months).

MicroServices-to-Service Authorization & Security Vault:
It's not enough to build, deploy and manage microservices at scale if you can't secure all the connections among them at scale. Security Vault, Microservices registration and auditing, and service-to-service authorization are part of the IONATE Key Management infrastructure. Service authorization is performed via known OAuth2 endpoints in combination with OpenID Connect.

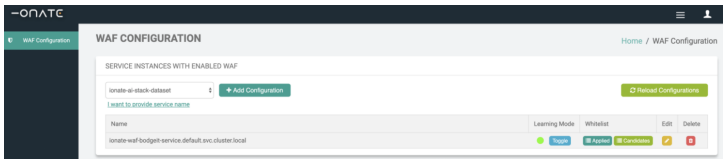
With cloud-native and microservices architecture, there is no state. Secrets management is a key aspect of cloud-native architecture as stateless microservices running as containers are incapable of storing secrets within the containers. As such, these containers must retrieve secrets securely, use them and thereafter dispose of them. Secret retrieval is typically achieved through a secured vault. Each microservice first registers to retrieve appropriate secrets. IONATE Key Management Infrastructure then audits the microservice to ensure it has not compromised. If it has been compromised, secrets are rotated as part of the key management.

Web Application Firewall (WAF) for Microservices:
The IONATE AI/ML-based WAF for Microservices operates through a set of rules called policies, which protect against potential threat vectors in the application by filtering malicious web traffic. While APIs require authentication and authorization to ensure that they aren't letting in risk, neither authentication and authorization prevent malware embedded in a payload. As the cloud is API-driven and fully programmable, all aspects can and should be automated. IONATE WAF mitigates this risk as well as that of poorly secured APIs.

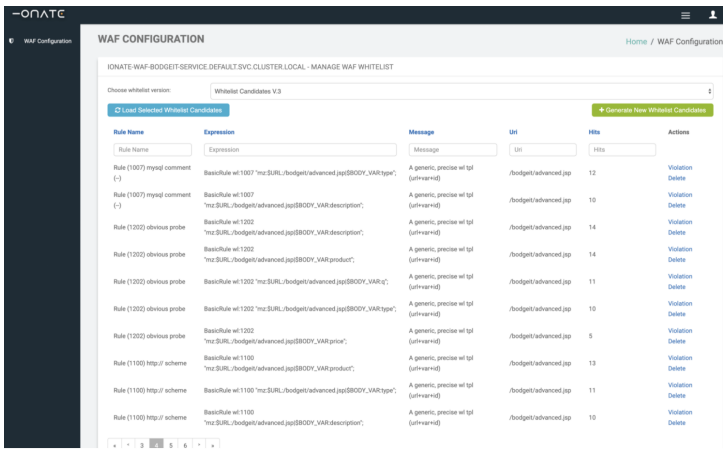
All newly created applications are by default set in Learning Mode, which allows our WAF to gather data and learn about web traffic to the application under a range of circumstances. Once learned, IONATE WAF can then detect good vs. bad traffic. Then, when active (the DENY Mode), IONATE WAF rejects and drops malicious traffic and potential attacks.



Configuring a new WAF protection for a microservice is very simple - no need to know complex regular expressions. Simply add the name of the microservice and IONATE AI/ML-based WAF will learn automatically.



All newly created applications are by default set in Learning Mode, which allows IONATE WAF to gather data and learn about web traffic to the application. Once learned, IONATE WAF can then detect good vs. bad traffic. Then, when active (the DENY Mode), IONATE WAF rejects and drops malicious traffic and potential attacks.



SecOps analyzes logs from WAF to retrain the system for false negatives and improve precision and recall for the WAF application.

Constantly Assessing Cloud-Native's Elastic, Abstracted Nature

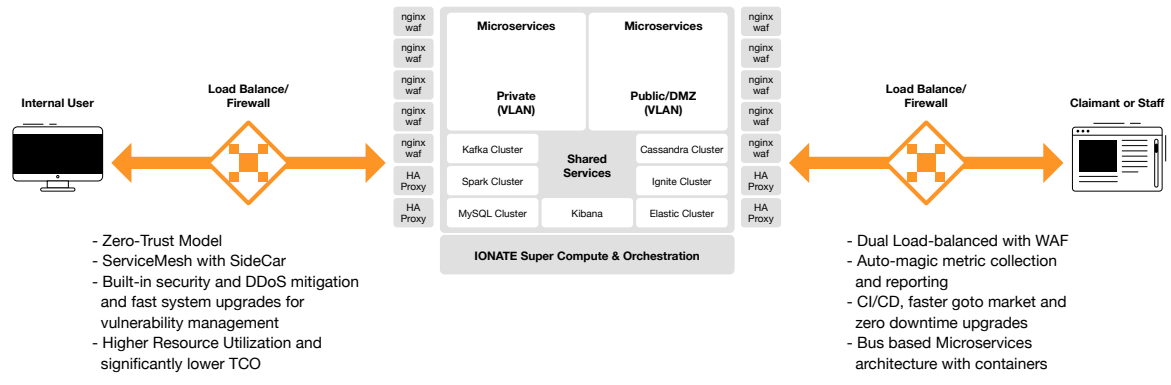
Visibility into complex large-scale distributed environments requires specialized capabilities and tools to provide insights. Measurement and context are critical. As cloud-native applications align with the underlying infrastructure (growing or shrinking), their attack surface also scales up and down. This makes it increasingly difficult to find and respond quickly to a security anomaly or incident.

Further, because containers split applications into interdependent services, the relationships among them must also be monitored. This level of monitoring can be an extreme challenge. Two surefire paths to failure are:

- **Manually correlating volumes of data to locate anomalies, especially as environment complexity and experiment velocity continues to increase**
- **The inability to monitor ALL DATA (as well as the interrelationships among and between them)**

True visibility requires tools that can provide insights into the interrelated nature of all components and services, understanding, for example, what a deviation discovered in one part of an environment means for another area. Below are details about how IONATE overcomes both issues.

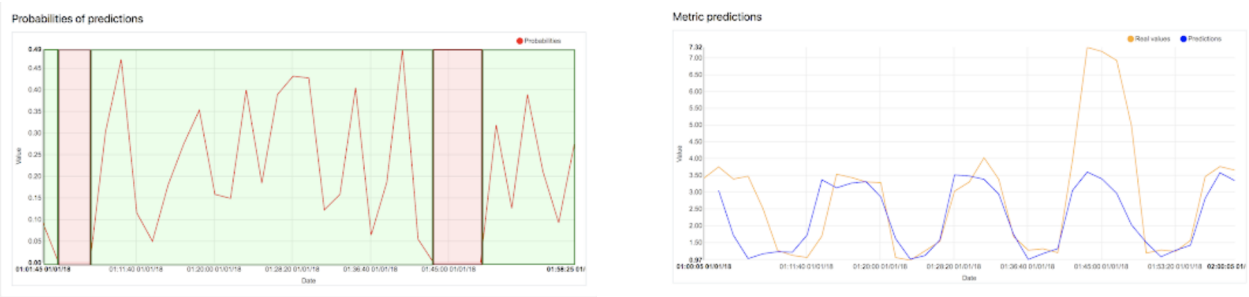
With IONATE, security is as elastic as the infrastructure it's protecting. We accomplish this through our Container Firewall, our Web Application Firewall (WAF), and our Policy Governance. Each microservice represents a certain number of containers, the exact amount depends on the size of the workload. During times where your infrastructure is scaling to meet a demand spike, the microservice could represent dozens of containers or hundreds. We can handle either scenario, because of how we implement our firewall and WAF. As an elastic cloud grows, the policies auto-adapt and are applied regardless of the number of elastic cloud instances. Whereas others have to attend to what is running and apply policies accordingly, with IONATE, policies are automatically applied given our platform's ability to auto-adapt.



The above figure demonstrates all aspects of our solution.

The IONATE AI/ML-Powered Cyber Security fully monitors systems and data, providing key insights into the interrelationships among both. With some attacks, the revealing anomaly will only appear in data analysis of the entire system, versus that of a singular device. We can monitor and raise alerts in any application or device.

Our #aiops monitoring approach can predict how one failure will affect other services, as well as gauge the criticality of a particular failure. Our solution continually reassesses applications and their environments, as the parameters of normal and abnormal constantly shift in a cloud-native environment.



Our solution has the tools needed to identify anomalies. We use application data(logs, metrics, etc) to create customer-specific KPIs of the anticipated normal range (baseline predictions) for any measure, behavior or performance. Any blip or deviation from the expected performance is identified and investigated. We can compare periods of performance for the batch anomaly (a specific time period) as well as live.

Summary

With our Zero-Trust security model, security is distributed, flexible and responsive. Our model considers each and every data input, source, and type, as well as the associated gateway or access point. And, our model monitors and learns from the interrelationships among these multiple components. This means our solution doesn't stop here. It adapts intelligently, even proactively. It's a given that cybercriminal activity shapeshifts. Now, with the IONATE Cloud-Native Security Solution, your cloud-native environment won't just keep up, it will stay one step ahead.