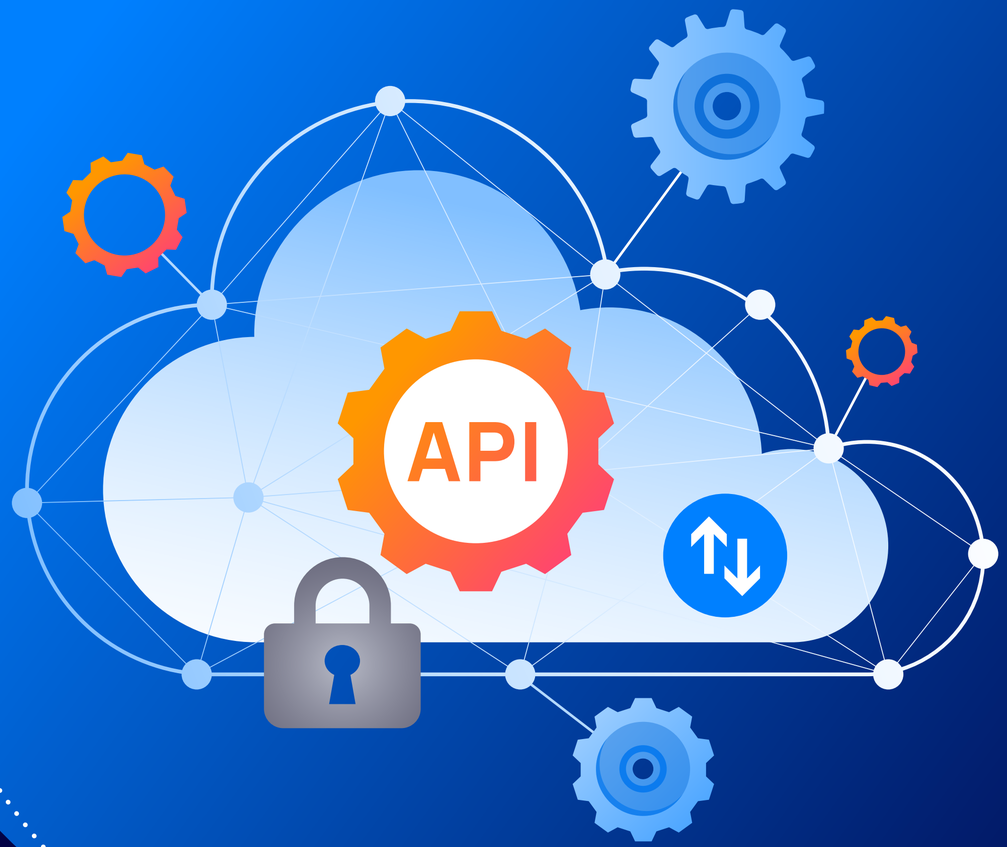
orca security

EBOOK

Addressing the Top Five API Security Challenges



What are APIs?

Application Programming Interfaces (APIs) serve as the connections between computer programs, web applications, and mobile applications and allow applications to send exchange requests for data and functions.

Without even knowing it, we use APIs many times a day through our GPS, social media, banking, and e-commerce apps. For example, if you open the weather app on your phone, you are in fact triggering an API to retrieve weather reports. If you pay your gardener's invoice using PayPal, you are using PayPal's embedded API. In short, APIs are everywhere and an integral part of the digital world.

Why is API usage growing so fast?

What is driving this enormous growth in API usage? There are a number of factors that are contributing to this development:

Growth of Microservices

Most organizations are moving towards the use of microservices architectures, where an application is composed of many independently deployable smaller components or services. Since each service can operate independently, it is much easier to scale and develop applications than if you have just one large monolithic application. APIs are used to enable the microservices to communicate and use each other's functions, which means that one application alone ends up deploying many APIs. Since microservices are quickly becoming the cloud native architectural approach of choice, this is causing an exponential growth of API sprawl.

Application Architecture Evolution

Back in the day, web applications used to rely on mega-frameworks that got requests from the clients' browsers and generated new HTML pages for each request with the new state. Today, the client-side has been decoupled from the server (with frameworks like React and Angular), and together with mobile applications they rely on the server-provided APIs to get data.

Unused APIs are Not Decommissioned

The CI/CD process enables APIs to be produced faster and more frequently. When newer APIs are produced, previous APIs are not always decommissioned. In this way, APIs can quickly multiply, and leave a large amount of unmaintained (zombie) APIs that are flying under the radar and not being actively secured.

Anything-as-a-Service (XaaS)

Anything-as-a-service, where products and tools can be consumed using an 'as a service' model, such as software as a service (SaaS), platform as a service (Paas), and infrastructure as a service (IaaS), is on the rise. The tremendous amount of managed products, tools, and technologies that are now delivered to users as a service over the Internet is driving the creation of more and more APIs since APIs allow the user to programmatically access the 'as a service' offering.

Why are APIs an attractive target?

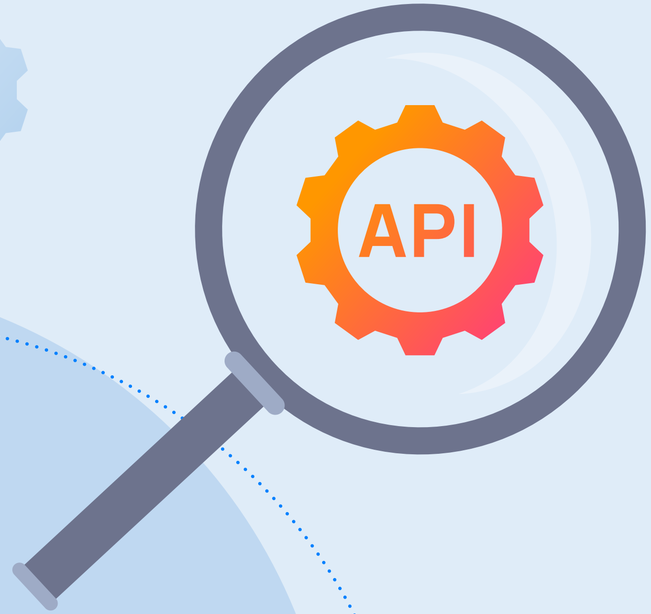
Because APIs provide access to sensitive application functions and data, they are an attractive target for attackers. By exploiting a weakness in an API, attackers can infiltrate sensitive data, inject malicious code, and perform unauthorized operations on the application. In addition, bad actors can use APIs to get privileged access within an application — and act on the Administrator's behalf.

API attacks can lead to application downtime, lost revenue, customer frustration, and leakage of sensitive data. According to the Gartner report 'Predicts 2022: APIs Demand Improved Security and Management': "API security challenges have emerged as a top concern for most software engineering leaders, as unmanaged and unsecured APIs create vulnerabilities that could accelerate multimillion dollar security incidents."

Source: Predicts 2022: APIs Demand Improved Security and Management

API Security

As API usage increases, so does the need for API security. With API threats increasing exponentially and more and more attackers seeking to exploit API vulnerabilities, the OWASP® Foundation has published a list specifically dedicated to API risks: the OWASP API Security Top 10. The list includes API risks ranging from authentication issues, excessive data exposure, and lack of resource limiting, to insufficient logging and monitoring.



OWASP API Security Top 10

1. Broken Object Level Authorization (BOLA)
2. Broken User Authentication
3. Excessive Data Exposure
4. Lack of Resources & Rate Limiting
5. Broken Function Level Authorization (BFLA)
6. Mass Assignment
7. Security Misconfiguration
8. Injection
9. Improper Assets Management
10. Insufficient Logging & Monitoring

Types of API Security Tools

So how can organizations avoid these API Security risks? Assuming there will always be shadow IT, human error, and software bugs, API Security solutions can help organizations detect and address API risks and reduce their attack surface.

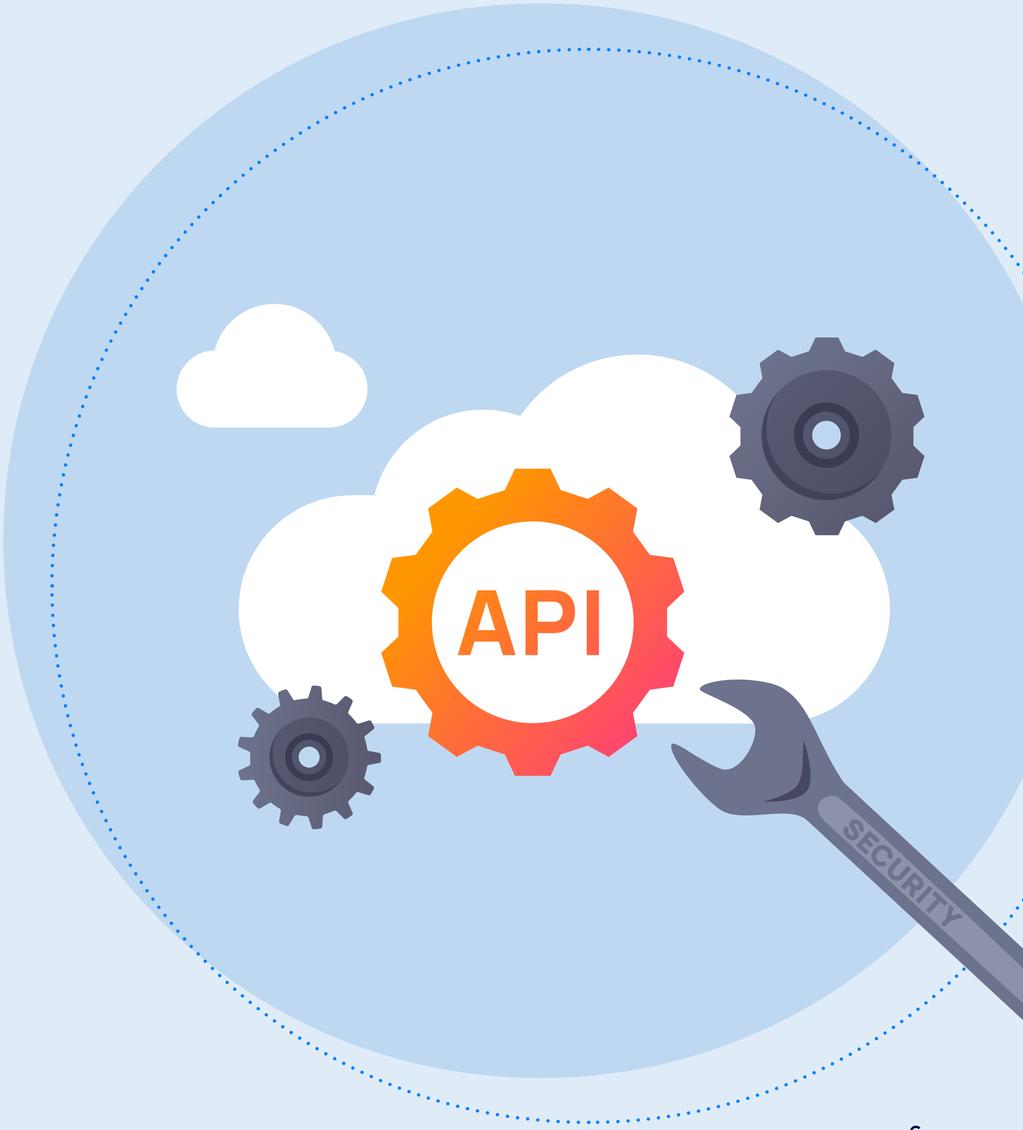
Most existing API Security tools use one of the following methods to scan, discover, and secure APIs:

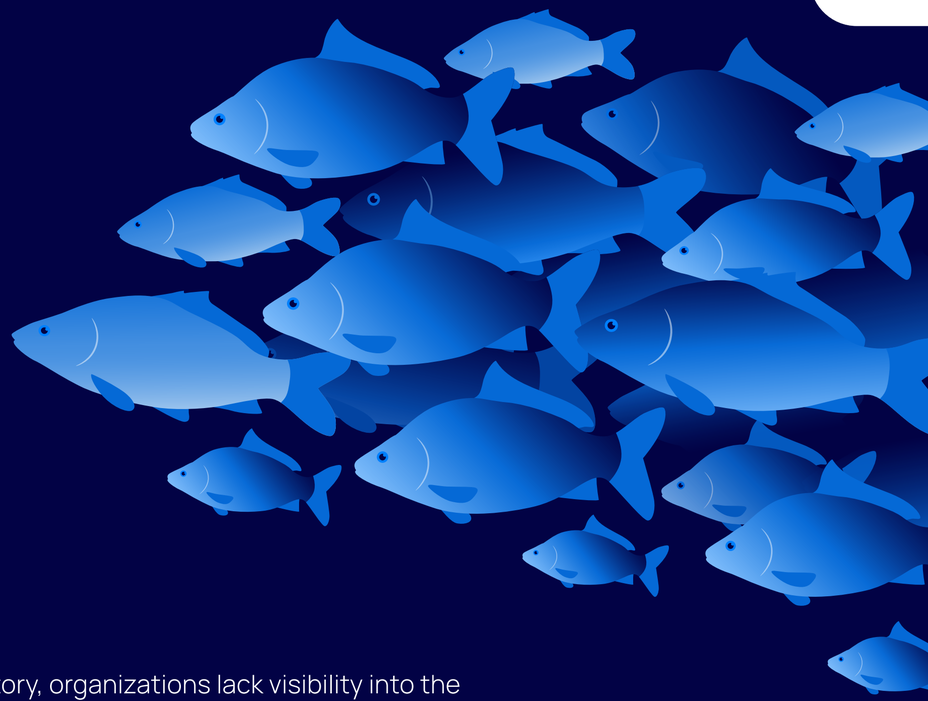
Firewalls and API Gateways

This includes API security products that utilize Web Application Firewalls (WAF), network firewalls, traffic mirroring, or API gateways to scan API traffic and perform signature-based threat detection, alerting to possible malware, SQL injections, and controlling input requests. The drawback of these solutions is that organizations need to ensure that their API traffic is directed through these firewalls and gateways. If APIs are not configured to do this, they will go unmonitored. Moreover, these approaches lack the visibility from within the workload, which is needed to provide context on risk findings.

Agents

This category includes API security products that, in order to discover APIs and detect API risks, require an agent to be installed on the workload, or running a worker on the CDN edge. The disadvantage of this type of solution is the high TCO of installing and maintaining agents and even more importantly, the API blind spots that are created when not all workloads have agents installed.





Top Five API Security Challenges

Despite there being a number of API Security tools available, there are certain API Security challenges that make it harder for organizations to ensure they are fully protected against API risks. We list the top five challenges below.

Challenge #1: API Sprawl

Just knowing which APIs are in your cloud workloads is tremendously complicated. Why? APIs are scattered all over the place. According to a Forrester report, rogue endpoint discovery is often cited as customers' most urgent API challenge. "Many have only a rough idea of how many APIs they have but no accurate inventory — especially since APIs can be buried inside a mobile app or web app, and they can masquerade as quasi-API endpoints like AJAX requests, webhooks, and more."

The Forrester analysts compare an application with APIs to a school of fish — "the group together looks like a unified coalition, but a number of fish may be hidden or flit in and out of the group. Without some clear definitions and measurements, it's difficult to know exactly how many and which fish are in the school — or how many API endpoints are buried in an application."

Without a proper API inventory, organizations lack visibility into the APIs they have, what their functions are, and whether they contain risks.

Traditional tools like API gateways and agent-based API Security products lack the ability to offer a complete inventory of all APIs. Since they need to be manually integrated for each API, they end up missing many APIs and cannot be relied on to offer a complete API inventory.

In addition, if you are using multiple cloud platforms and rely on cloud platform native API security tools, each tool will only discover APIs used on that particular cloud platform. This results in the need to deploy an API Security solution for each platform, which quickly becomes unmanageable.

Challenge #2: Lack of Visibility into Unmanaged APIs

You can have the most sophisticated security system, but if you don't secure all the doors, attackers can still get in. Unmanaged APIs can constitute open doors waiting for attackers to be found. But the problem is, if they are unmanaged, how will you even know about them?

Unmanaged APIs create vulnerabilities that could potentially lead to security incidents. Gartner predicts that by 2025, more than half of APIs will be unmanaged, as explosive growth in APIs surpasses the capabilities of traditional API management tools.

Why are some APIs unmanaged?

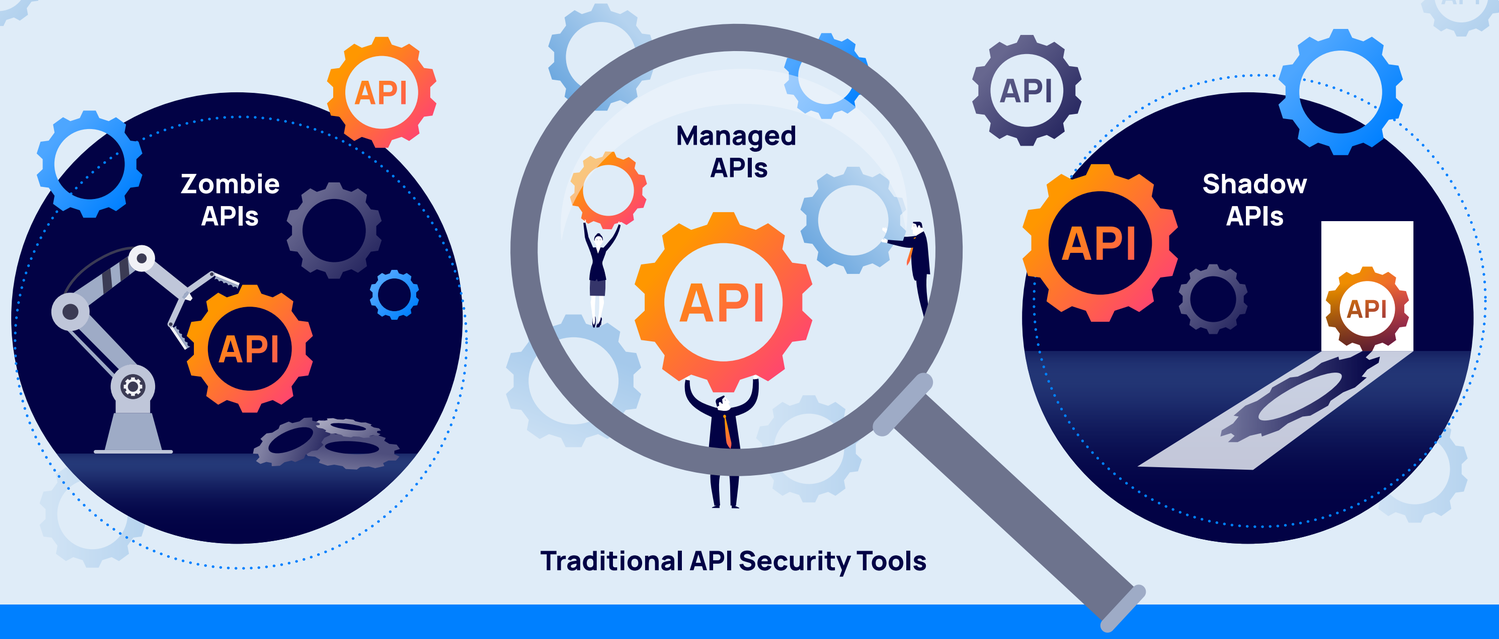
The two main reasons for APIs escaping management and security policies are the following:

Zombie APIs

When newer APIs are produced, previous APIs are almost never decommissioned. Even though these APIs might have been managed previously, they can quickly become neglected and forgotten.

Shadow APIs

Shadow IT is the use of information technology systems, devices, software, applications, and services without IT department approval and visibility. For instance, if developers implement APIs without notifying IT, this can lead to APIs remaining 'hidden' from security teams.



Traditional API Security tools only see managed APIs

Many API Security products require an agent or edgeworker to be installed on the resource, or require API traffic to be routed through a gateway or firewall before they can get insight into APIs. Since unmanaged APIs are not seen by API Security Tools, this results in exposure to possibly dangerous API risks without the organization even knowing it.

Challenge #3: Understanding High vs. Low Priority API Risks

Many API security products are point solutions and take a narrow view of API Security, without accessing cloud control plane and workload data to assess risk context. This means that they have limited ability to prioritize API risks.

For instance a vulnerability in an API may not even seem that dangerous to an API security tool, but a cloud security solution that has full contextual insight can see that the API vulnerability will allow an attacker to get access to keys that enable them to move to another resource containing Personal Identifiable Information (PII). The API security tool will flag the issue with normal priority, but the contextually aware cloud security solution will mark this API issue as high priority.

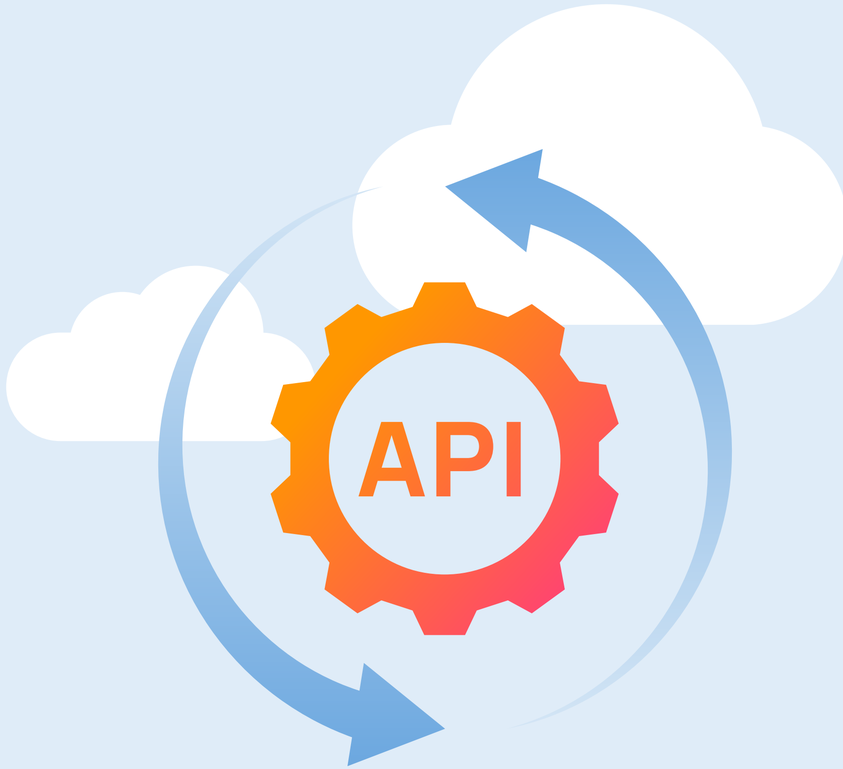
If, instead of hundreds of alerts, security teams can be alerted to a far smaller number of high priority alerts that need immediate attention, this avoids alert fatigue, burnout, turnover, and ultimately will prevent critical alerts being missed due to desensitization.

Challenge #4: APIs Change Fast and Often

Given the pace of cloud-based application development and the growing adoption of CI/CD processes, it's almost impossible to manually keep track of changes in APIs. Each time an API changes, traditional security tools need manual reconfiguring or fine tuning to still have insight into the API.

If this is not done rigorously, APIs can become unmanaged and potentially expose the organization to new risks. In addition, the continuous manual intervention that is required to run API security creates a tremendous burden on security and DevOps teams, taking up valuable time that could be spent on higher-value tasks.

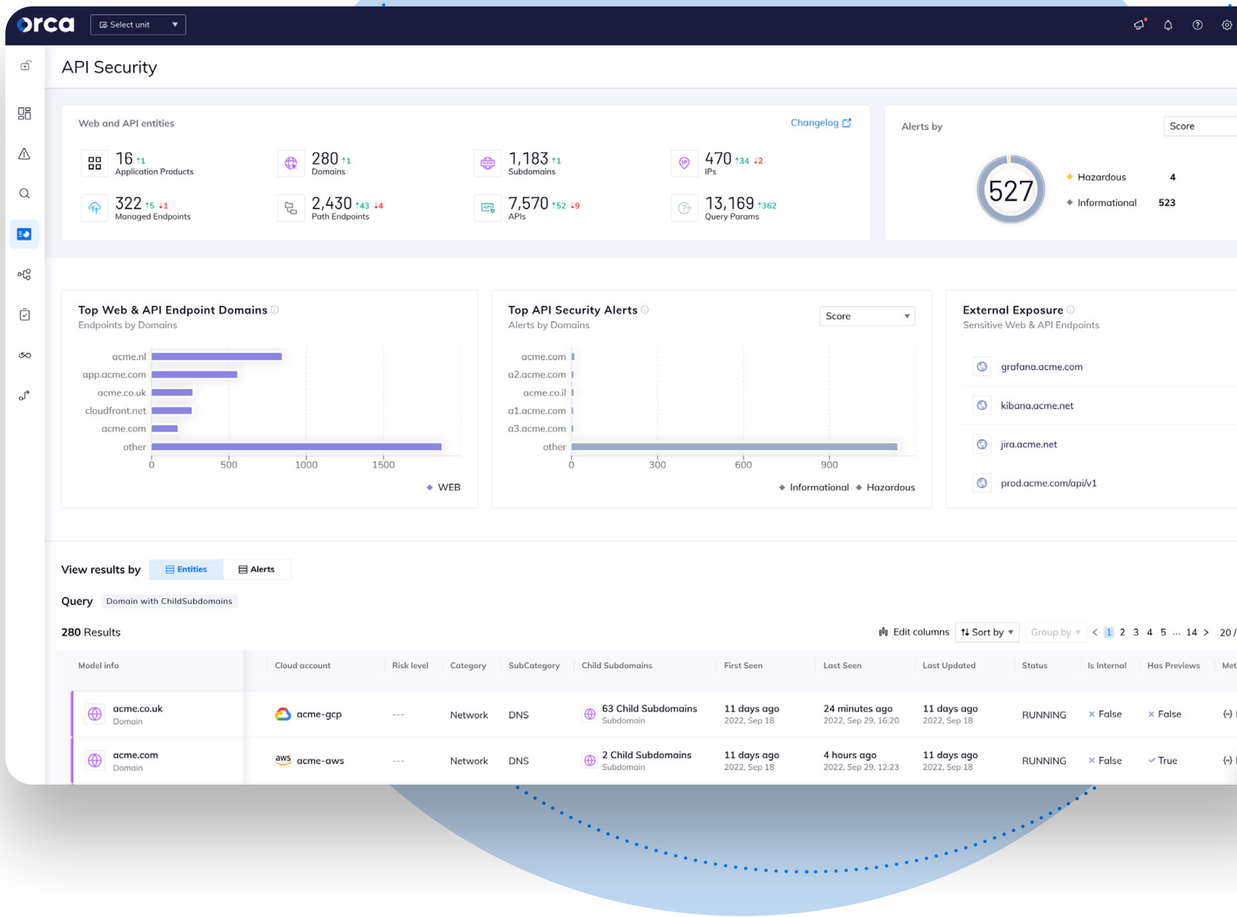
In addition, this can lead to significant tension between security, development and DevOps teams since responsibilities may be unclear, and development and DevOps teams may feel unfairly burdened, and undermined in their ability to quickly deliver applications.



Challenge #5: Understanding API Risk Exposure is Difficult

As a result of incomplete API inventories, not keeping track of API changes, and incomplete insight into context, organizations cannot easily get answers to their API exposure questions. For instance in a zero-day situation, organizations should be able to get answers to questions such as:

- Which EC2 instances have an exposed API?
- Which APIs does a particular cloud asset serve?
- Do we have publicly exposed APIs from Kubernetes?
- Which APIs have insufficient logging configured?
- Which APIs expose PII?
- Which APIs are served from workloads vulnerable to a certain CVE?
- Which APIs is the cloud provider exposing on our behalf?



A New Approach to API Security is Needed

Recognizing these challenges, Orca has expanded its agentless Cloud Security Platform to include complete coverage of managed as well as unmanaged APIs in cloud environments, without the need for agents, firewalls or gateways. Leveraging its patent-pending SideScanning™ technology as well as information retrieved from cloud provider APIs, Orca delivers a complete and continuously updated multi-cloud inventory of APIs, detects API-related misconfigurations and vulnerabilities, and alerts to API drift and changes.

By combining detected weaknesses in APIs with other risks found in the cloud environment, such as vulnerabilities, malware, asset or identity misconfigurations, and potentially exposed PII, the Orca Cloud Security Platform offers the necessary context to understand which API risks are the most critical so that security teams can focus on what matters most. With Orca, application security and cloud security teams have complete observability of API endpoints in their environment, giving them the ability to manage API risks and adhere to API security compliance requirements.

About the Orca Cloud Security Platform

Orca Security is the industry-leading agentless Cloud Security Platform that identifies, prioritizes, and remediates risks and compliance issues across your cloud estate spanning AWS, Azure, Google Cloud and Kubernetes. Instead of layering multiple siloed tools together or deploying cumbersome agents, Orca delivers complete cloud security in a single platform by combining two revolutionary approaches: SideScanning, which enables frictionless and complete coverage without the need to maintain agents, and a Unified Data Model, which allows for centralized contextual analysis of your entire cloud estate.

Orca's agentless platform connects to your environment in minutes and provides 100% visibility of all your assets, automatically including new assets as they are added. Orca detects and prioritizes cloud risks across every layer of your cloud estate, including vulnerabilities, malware, misconfigurations, lateral movement risk, API risk, weak and leaked passwords, and overly permissive identities.

Would you like to find out how many of these risks are in your environment? Take our free, no obligation risk assessment to find out.