

8 Do's and Don'ts of API Security

Critical factors for a robust API security posture

What's so complicated about protecting APIs?

API security is topping the priority list for many IT executives – and with good reason. Consider the following:

“The explosion of APIs provides an attractive attack surface, and API security continues to flummox security leaders.”

– The Eight Components Of API Security, Forrester Research, Inc., September 28, 2023

Factors in API risk growth

More APIs

More automation

More connected devices




More partner integrations

In response to these risks, organizations must understand the following before they begin to implement effective API security:

APIs are a moving target	
Internal API awareness	External API exposure
Fast-moving DevOps processes create and decommission APIs continuously, leading to an incomplete API inventory	Immature API practices lead to unintended exposure of sensitive APIs to external parties, including many shadow APIs

APIs are vulnerable to two different types of threats	
Technical vulnerabilities	Misuse and abuse
Attackers can exploit software vulnerabilities and misconfigurations, including the OWASP API Security Top 10	Business logic abuse and other behaviors, like aggressive data scraping, can happen regardless of a technical vulnerability

Addressing the complex challenge of API security requires a well-considered approach that includes:

 <p>Incorporating the latest technology advances</p>	 <p>Breaking down organizational barriers</p>	 <p>Addressing the complete API threat landscape</p>
--	---	--

The following are some essential strategies to implement – and pitfalls to avoid – as you develop a more sophisticated API security strategy for your organization.



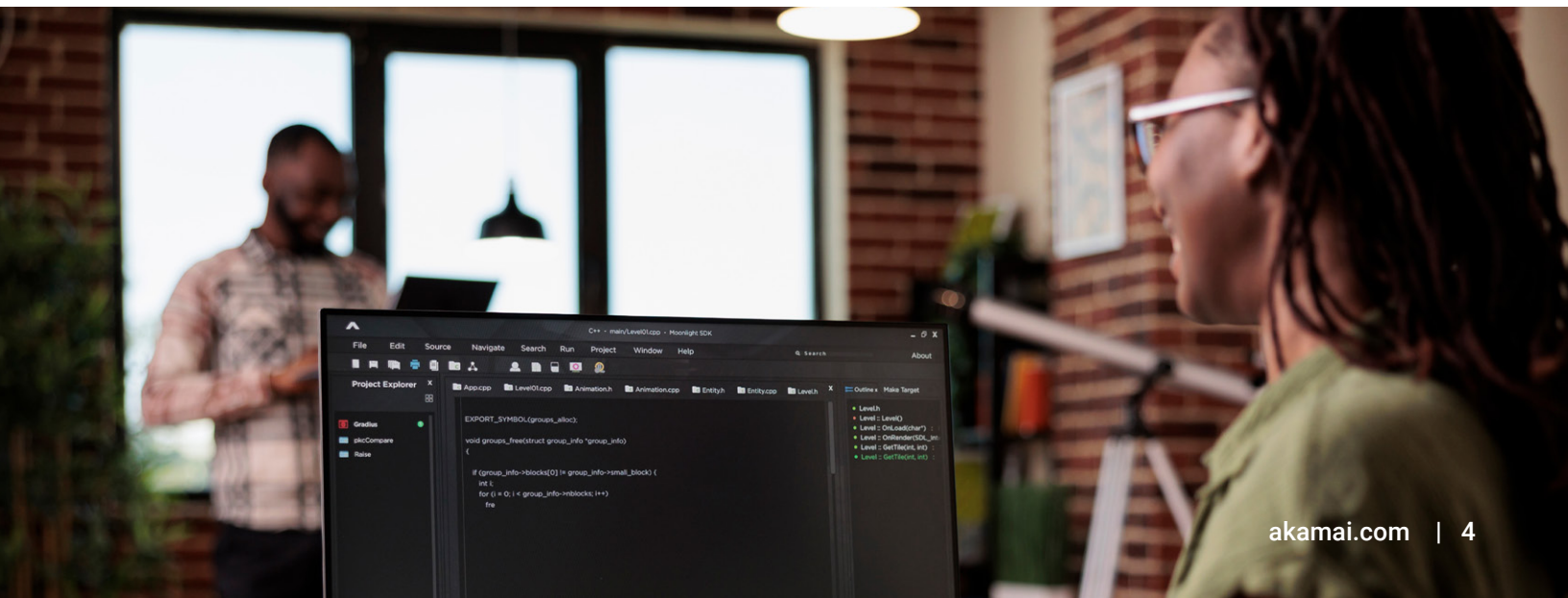
The 8 do's and don'ts of effective API security

1 Do strive for complete API visibility

It's worth repeating: You cannot protect APIs you do not know you have. The longer an API goes unidentified and unmonitored, the more likely it is to become a target for an attacker. The best way to achieve complete visibility is by ensuring that your API security platform can ingest information from the broadest possible range of data sources, including API gateways, network devices, microservices orchestration solutions, cloud providers, and more. Specifically, your API security solution should be able to do the following:

Time	Location
<ul style="list-style-type: none">• Discover APIs continuously• Monitor individual API calls• Record short-term session activity• Analyze APIs behavior over time	<ul style="list-style-type: none">• Discover APIs throughout the enterprise• Discover legacy APIs• Uncover shadow APIs

Complete API visibility will help save you from API data breaches, especially because the latest data breach technique involves attackers using low-and-slow attacks to scrape data from APIs. Knowing where all your APIs are is the first step to preventing this emerging type of attack.



2 **Don't be afraid of the cloud**

Web application firewalls (WAFs) use signature-based techniques to prevent unauthorized APIs from entering your organization. As API attacks have evolved, you need an extra layer to fully defend APIs from the complete array of possible risks using behavioral analytics. It is now critical to monitor the behavior of your APIs within your organization, not just those that are exposed externally.

To use behavioral analytics effectively, API traffic needs to be analyzed in the cloud. Security teams are sometimes reluctant to send sensitive information about their organization's activity to the cloud. However, performing true behavioral analytics using extended detection and response techniques on the volume of API data that most enterprises generate is highly impractical without the scale and elasticity that the cloud provides.

In addition, as security teams have their limited resources stretched thin, long and complex product deployments are a major obstacle to progress. Given the growing risk posed by broader API usage, security teams can't afford to fall further behind. Therefore, it's essential to take the leap to the cloud as part of your API security strategy.

3 **Do make business context central to your strategy**

Discovering APIs and identifying security risks are just the beginning of the journey to a smaller API attack surface. Consider the following three questions:

1. How would you know if the API credentials of a specific partner have been compromised?
2. How would you know if corporate espionage is happening in the form of data scraping on an API?
3. How would you know if your invoicing API is being abused by a user enumerating through invoice numbers to steal account data?

In the first scenario, activity would appear to be originating from a legitimate user. Therefore, the only way to detect malicious intent is by noticing a change from expected behavior on the API in question. The second and third scenarios are also examples of unsanctioned behavior that exploits legitimate API access models. These are other cases in which it is critical to understand the business context, in addition to what is occurring technically.

4 **Don't** make data a one-way street

One of the fundamental capabilities of an effective API security approach is the ability to send alerts and events to preferred security monitoring and IT workflow tools. A common mistake made by security vendors – and the teams that implement the alerts – is to view security alerts and automated responses as a one-way communication flow.

Just like many legitimate business processes, attacks can occur over a long time. To be effective, behavioral analytics for API usage must be performed over a period of at least 30 days. This provides a more complete and accurate picture of baseline expected behavior. It also makes it possible to detect attacks that are executed slowly across multiple days or weeks – and numerous API sessions. Consider a low-and-slow data scraping attack that falls below a defined rate limit: Such behavior would only be found by examining historical behavior versus any changes.

An alert without supporting details arguably does more harm than good. An alert with rich context about the cause and impact, however, is much more actionable. But the real win comes from providing a context-rich, actionable alert and giving the receiver the ability to query a more extensive dataset to analyze the incident. Then, you can leverage your WAF protections to immediately block traffic that presents a potential threat to your business.

5 **Do** prioritize cross-departmental collaboration

Some of the biggest API security gains can come from proactive avoidance of vulnerabilities during the design, development, and deployment phases. To accomplish this effectively, you need collaboration across your teams.

Start this collaborative process by giving API teams visibility into how APIs are being used (and abused) under real-world conditions. Over time, this exposure will foster a culture of thinking about security earlier in the API development and deployment processes. Also, make sure:

- There are nonsecurity benefits that help API teams work more effectively in addition to the core security features of your approach
- It is easy for nonsecurity users like developers to view and query API inventory and activity information
- To use contextual responses, such as integrations into development tools, like Jira, that proactively open tickets for security fixes that developers need to make

Thinking about API security as everyone's job and making it easy for stakeholders outside the security team to get involved eliminates finger-pointing and makes it possible for development, operations, and security teams to work together in mutually beneficial ways.

6 **Don't overlook third-party APIs**

Another common API security strategy pitfall to avoid is assuming that you only have to worry about your own APIs. As desirable as it is to believe that the WAF or API gateway you bought standardizes your entire API security strategy, that is not always the case.

For example, just because a centralized API gateway strategy is implemented, do not assume that shadow APIs won't circumvent the core API governance approach. If your business is relying on any third-party APIs, your gateway would see those APIs as authenticated, even if they were compromised before they connected with your ecosystem.

Your API protection strategy must tie in with your primary API technologies, like API gateways, while also collecting as much information as possible from other sources, such as network devices, cloud platforms, and microservices orchestration tools. This is the only way to create a complete picture of your API attack surface and to future-proof your security strategy as technology and infrastructure transitions inevitably occur.

7 **Don't respond and move on**

Although responding quickly and effectively to alerts is great, if you are only focused on mitigating alerts once they happen, you've missed an opportunity to avoid alerts altogether. Instead, consider proactive threat hunting. If your API security partner empowers you to perform data queries, you'll be able to test your own hypotheses, understand relationships, and identify potential threats before they escalate into a security incident. For example, if you identify a bad API usage behavior by a specific partner, you can look for similar behavior by other partners or suppliers with a few clicks.

Any API security partner must store historical data in a data lake and provide access to this data to allow for investigations and threat hunting.

Ideally, these type of rich query capabilities should be made available to you in two ways:

1. As a simple and intuitive user web interface
2. As a set of API interfaces into the API security providers themselves for use in developing more sophisticated workflows

8 Do approach API security as a continuous lifecycle

The best way to build API security directly into your business is through API testing. By adding this tool to the API lifecycle, you can limit the chances that a misconfigured or vulnerable API will be put into production. This testing and fixing earlier in the development cycle reduces headaches, saves time, and decreases expenses.

Next, security teams should begin their API protection efforts by creating an inventory of APIs in use by their organization. Because APIs are added and decommissioned continuously, it's critical for security teams to keep a living inventory of API interfaces in their sensitive applications and data repositories. When continuous API discovery is performed effectively, shadow, rogue, forgotten, zombie, orphaned, and deprecated APIs all become problems of the past.

Security teams should have the visibility they need to detect and mitigate a wide range of emerging API security threats. But detecting threats must also happen during runtime. Business logic abuse is found only on APIs in production. Comparing runtime behavior with baseline normal usage patterns helps reveal abusive behavior.

Finally, it's important to actually stop the threats that might take advantage of your APIs at any point during runtime. Automatic blocking by the WAF is critical to this step because simply having alerts for everything won't be enough to protect your business at the macro level. Other automated responses can be varied and customizable, such as lowering a rate limit on the API gateway, opening a Jira ticket for a developer to investigate, or sending an email to the security team. The ability to respond appropriately for every detected threat is only possible when context is understood and the response mechanism is customizable.





Summary

Do's	Don'ts
✓ Do strive for complete API visibility	✗ Don't be afraid of the cloud
✓ Do make business context central to your strategy	✗ Don't make data a one-way street
✓ Do prioritize cross-departmental collaboration	✗ Don't overlook third-party APIs
✓ Do approach API security as a continuous lifecycle	✗ Don't respond and move on

Get started today

Are you ready to take the first step to a modern, systematic approach to API security?

Learn more about [Akamai API Security](#).

Akamai's cloud-based approach makes it easy to get started in minutes. Within hours, you'll have a complete picture of API usage across your organization, including a detailed understanding of the relationships between your business logic and your APIs.



Akamai protects your customer experience, workforce, systems, and data by helping to embed security into everything you create – anywhere you build it and everywhere you deliver it. Our platform's visibility into global threats helps us adapt and evolve your security posture – to enable Zero Trust, stop ransomware, secure apps and APIs, or fight off DDoS attacks – giving you the confidence to continually innovate, expand, and transform what's possible. Learn more about Akamai's cloud computing, security, and content delivery solutions at akamai.com and akamai.com/blog, or follow Akamai Technologies on [X](#), formerly known as Twitter, and [LinkedIn](#). Published 11/23.