SALT SECURITY

State of API Security

Q1 2022

Report by SALT LABS

State of API Security

Q1 2022

Executive Summary

The State of API Security Report from Salt Labs is the industry's only report on API security risks, challenges, and strategies. This third edition of our pioneering research report offers security leaders, DevOps teams, risk management teams, and developers a deeper perspective into the dozens of factors that impact API security. It also provides insights on building plans and strategies to reduce the growing API attack surface.

As always, the report draws on a mix of survey results and empirical data from the Salt SaaS platform hosting our customers' API metadata. The most dramatic finding in this edition of the report comes from our customers. **Over the past year, Salt customers experienced a 681% increase in API attack traffic while their overall API traffic grew 321%.**

Our survey data reflects the input of more than 250 respondents. In that data set, we see that **26% of respondents have doubled the number of APIs in use from a year ago, and 5% have more than tripled the count.**

Unfortunately, 2021 saw a significant rise in API security incidents. As organizations continued to transform their ways of working, and as developers built more applications and APIs for services we all love and use, hackers also changed their tactics, making APIs their prime target. From the Experian API breach to the Log4j vulnerability, companies scrambled to find ways to prevent these security incidents from happening to them.

Our latest research reveals that **95% of respondents suffered an API security incident in the past year**, and nearly two-thirds of respondents delayed the rollout of an application as a result of API security concerns. Organizations remain woefully unprepared for API attacks – **34% lack any API security strategy at all, and only 11% have an advanced strategy, with dedicated API testing and protection**.

"Shift left" approaches continue to fall short, given the high rate of security incidents (95%) and the fact that **only 27% of respondents are deploying runtime protection**. WIth 83% lacking confidence that their API inventory is complete, even basic visibility is missing for most companies. And a higher rate (86%) lack confidence that they know which APIs expose sensitive data.

Traditional application security and API management tools aren't providing sufficient protection – only 15% of respondents state that their existing tools are "very effective" in preventing API attacks. Among Salt customers, 19% are enduring fewer than 10 attacks a month, more than half (51%) are suffering 11 to 100 attacks every month, and **12% are victims of more than 500 attacks every single month**.

As the industry takes another giant leap forward along the API security journey, one thing is certain – **APIs have emerged to be the broadest and most risky attack surface in the enterprise**. Perhaps the biggest lesson we can take from our latest research is that 2022 must be the year that organizations get serious about securing APIs.

The time is now.

Research Methodology

To understand the state of API security today, Salt Labs – the API threat research arm of Salt Security – initiated and compiled this API security industry report. Our in-depth research combines survey responses and empirical data from Salt Security customers. The findings reflect the input of more than 250 security, DevOps, and app development professionals across companies big and small, in a variety of industries across the globe (page 14). Salt Labs also pulls aggregated and anonymized data from the SaaS component of the Salt Security API Protection Platform – this empirical data gives more context to the survey response findings.

Contents

API attack traffic grew 681% in the last 12 months, compared to a 321% increase in overall API traffic

Average API counts more than doubled, growing 221%, indicating increased usage per API

APIs are driving our customers' critical development, efficiency, platform integration, and digital innovation projects (Page 4). In the midst of these crucial initiatives, Salt Security customers are battling a persistent stream of attacks.

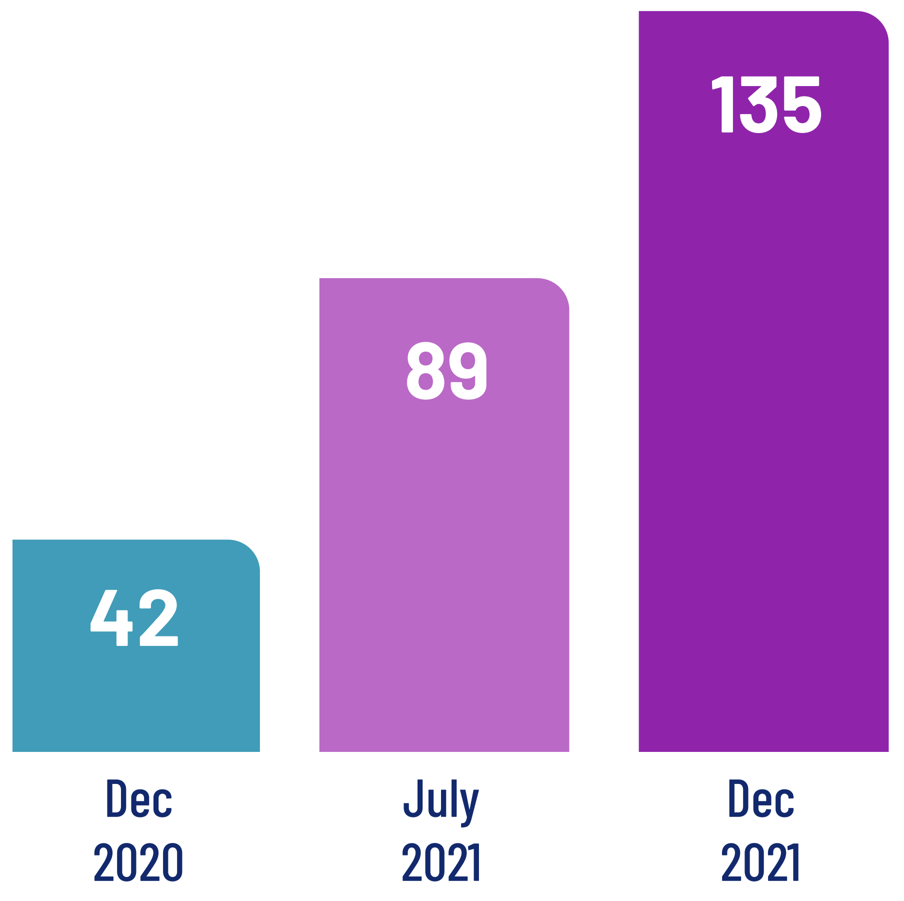
In the last 12 months, API attack traffic increased a staggering 681%, growing from a per-customer average of 2.73 million malicious calls in December 2020 to 21.32 malicious calls in December 2021. **Overall API traffic, in contrast, grew at a rate of 321% in the same time period**.

The average number of APIs per customer increased 221% over the last 12 months, growing from 42 in December 2020 to 135 in December 2021. It's interesting to compare that 221% increase to the 321% growth rate for API call volume. Taken together, those two data points illustrate that **our customers' APIs are being exercised far more frequently**.

Against the backdrop of such a steep rise in API attacks, it's understandable why companies are struggling to innovate, as evidenced by delays in production rollouts (Page 4) and lack confidence in their API security strategies (Page 5).

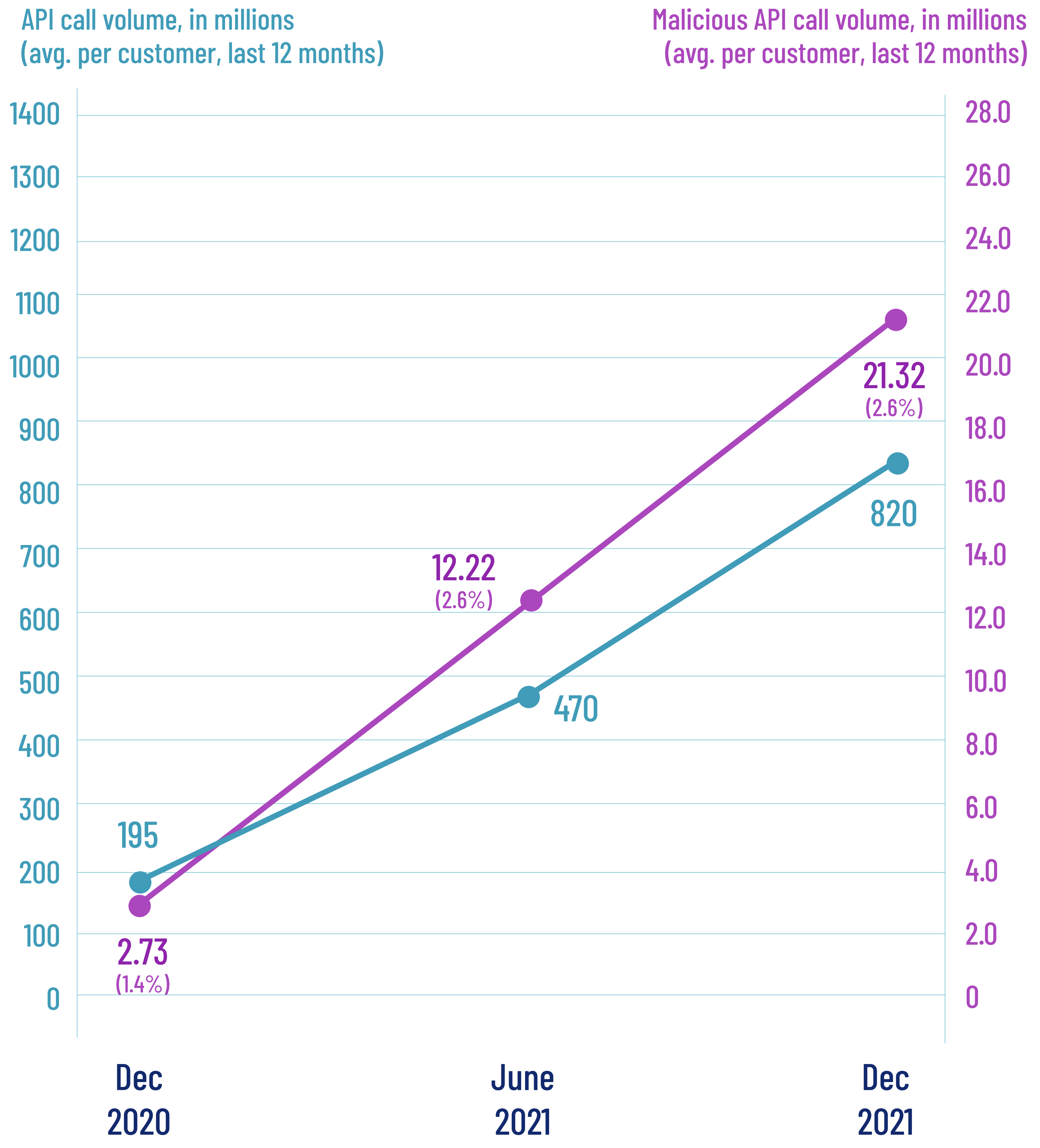
Salt customer data

Growth in average number of APIs per customer



Salt customer data

Growth in API call volume vs. malicious traffic



Despite companies' increased reliance on APIs, API security concerns continue to impede innovation

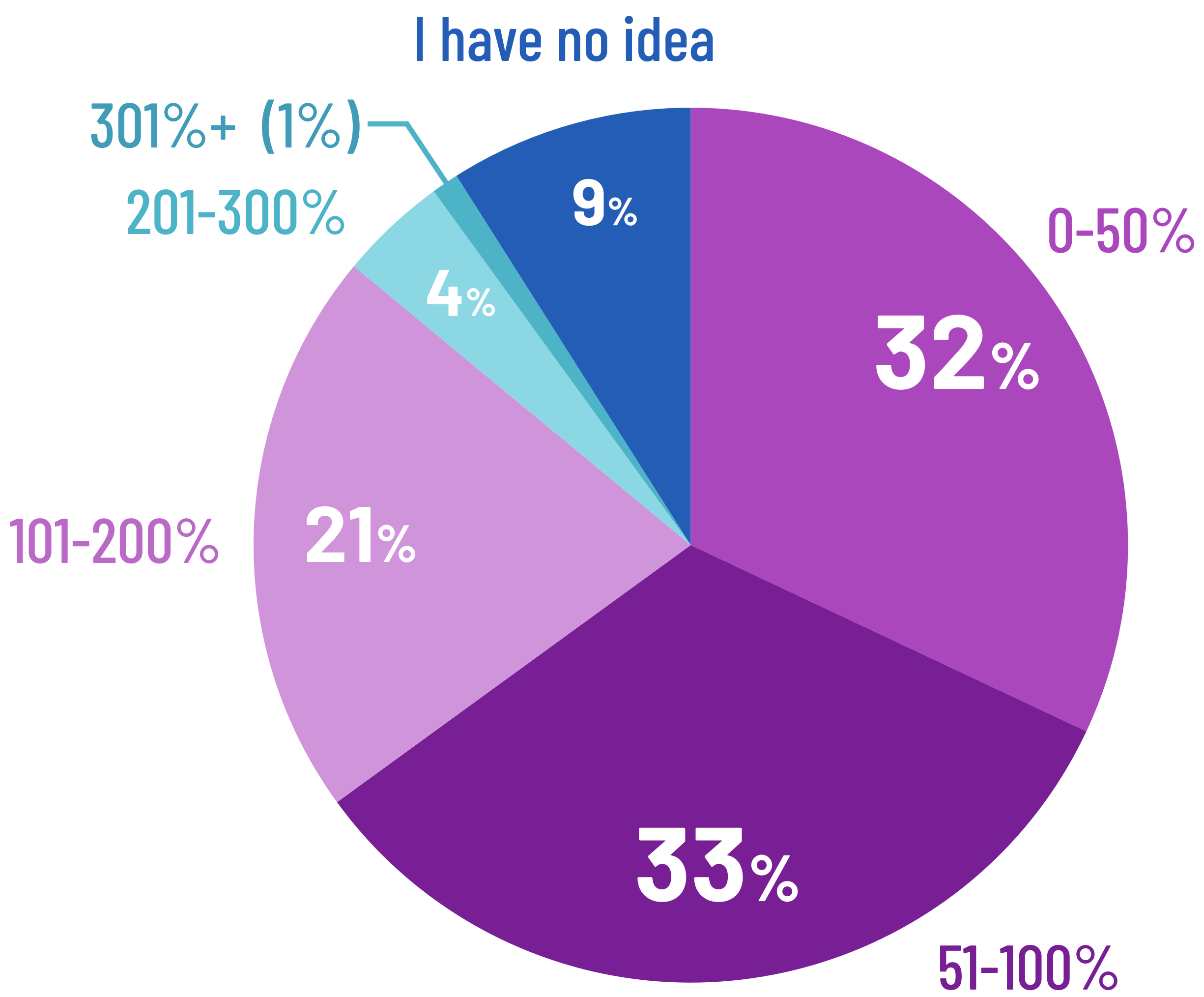
62% of respondents have delayed deploying applications into production because of API security concerns

APIs are fundamental to business innovation today. Our respondents rely on APIs to enable more efficient development (58%), platform or system integrations (44%), cloud migration (40%), and digital transformation (37%).

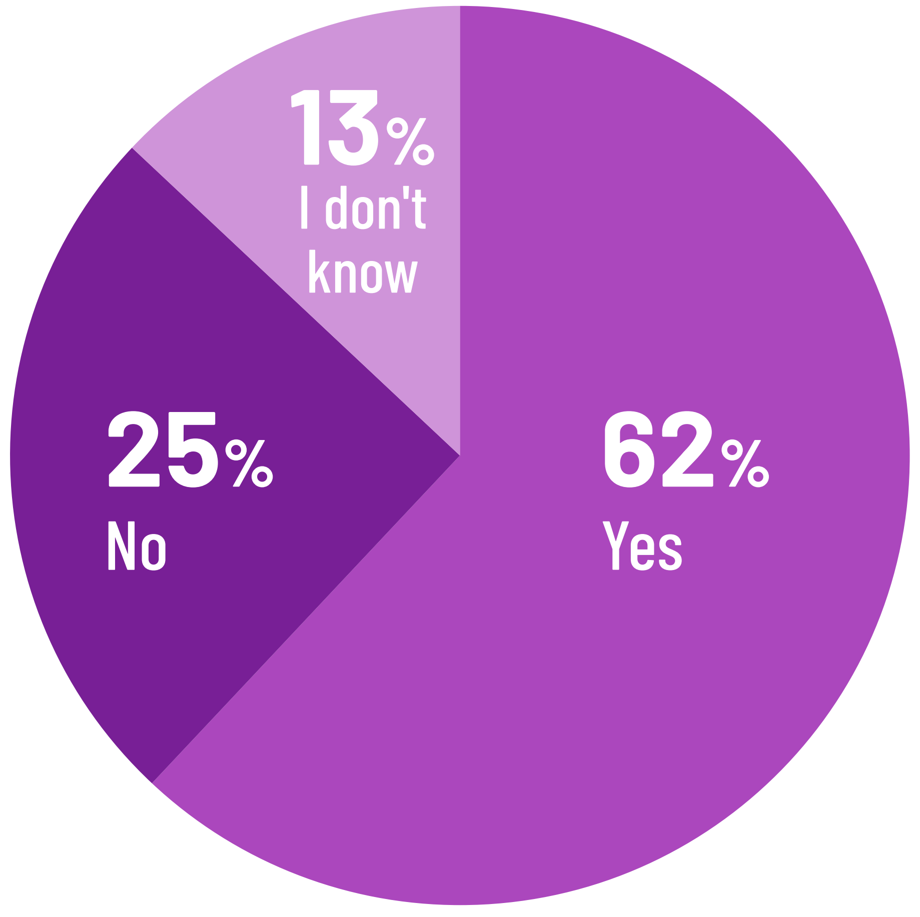
Use of APIs is increasing in all industries, with **26% of our survey respondents reporting they use at least twice the number of APIs as a year ago and 5% using more than triple the APIs**.

Organizations increasingly recognize the risk that APIs present, however. With 62% of respondents delaying application rollouts because of API security concerns, it's clear organizations face an urgent need to reduce the risk around APIs for them to continue to innovate quickly and their businesses to flourish.

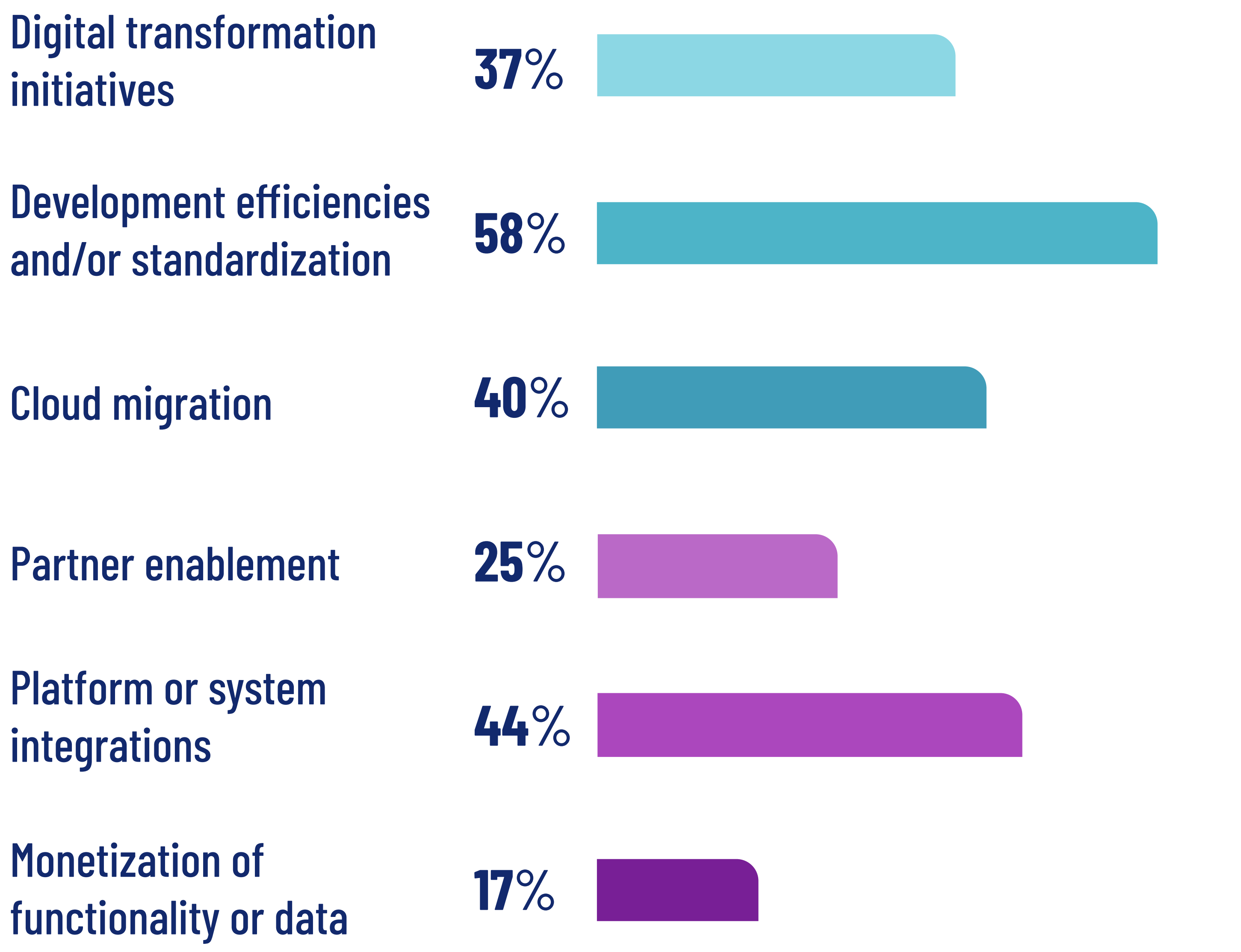
By how much has the number of APIs increased over the past 12 months?



Have you ever slowed the rollout of a new application into production because of API security concerns?



What are the main drivers behind the use of APIs in your organization? (*Select all that apply*)



The risk is real – 95% of respondents have suffered an API security incident in the past 12 months

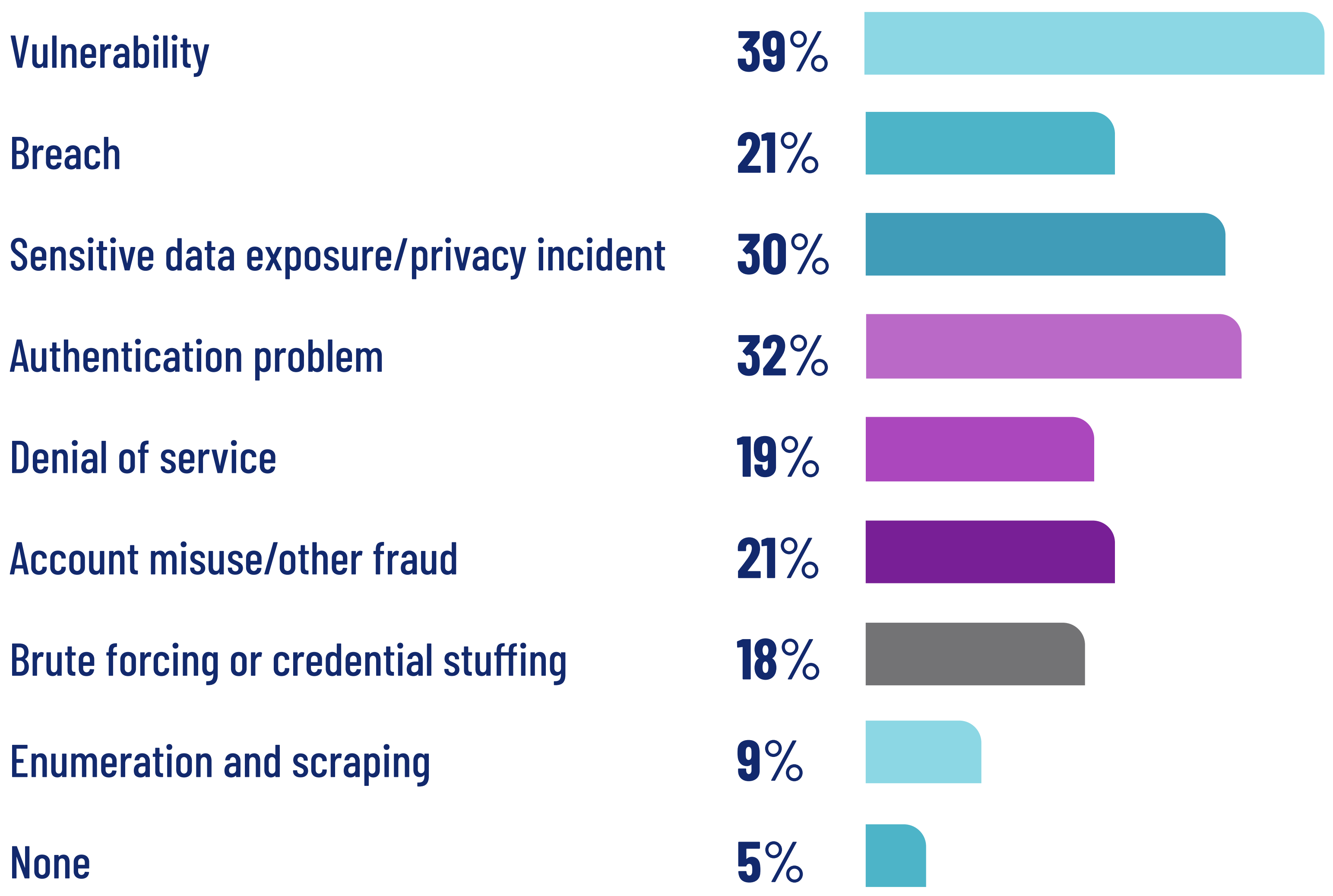
21% admit they've been the victim of an API breach

Despite robust efforts to validate APIs before deploying them into production, **nearly every company is finding security problems in their production APIs**. Vulnerabilities are the leading challenge, with 39% of respondents identifying them in their production APIs. Authentication problems are the next most common issue, at 32%, followed closely by sensitive data exposure at 30%.

These gaps put companies at serious risk of data exfiltration, account takeover, and exposed personally identifiable information (PII). Such incidents bring obvious consequences to customer confidence and brand loyalty, but more tangibly they can lead to sizable financial costs with compliance-related fines.

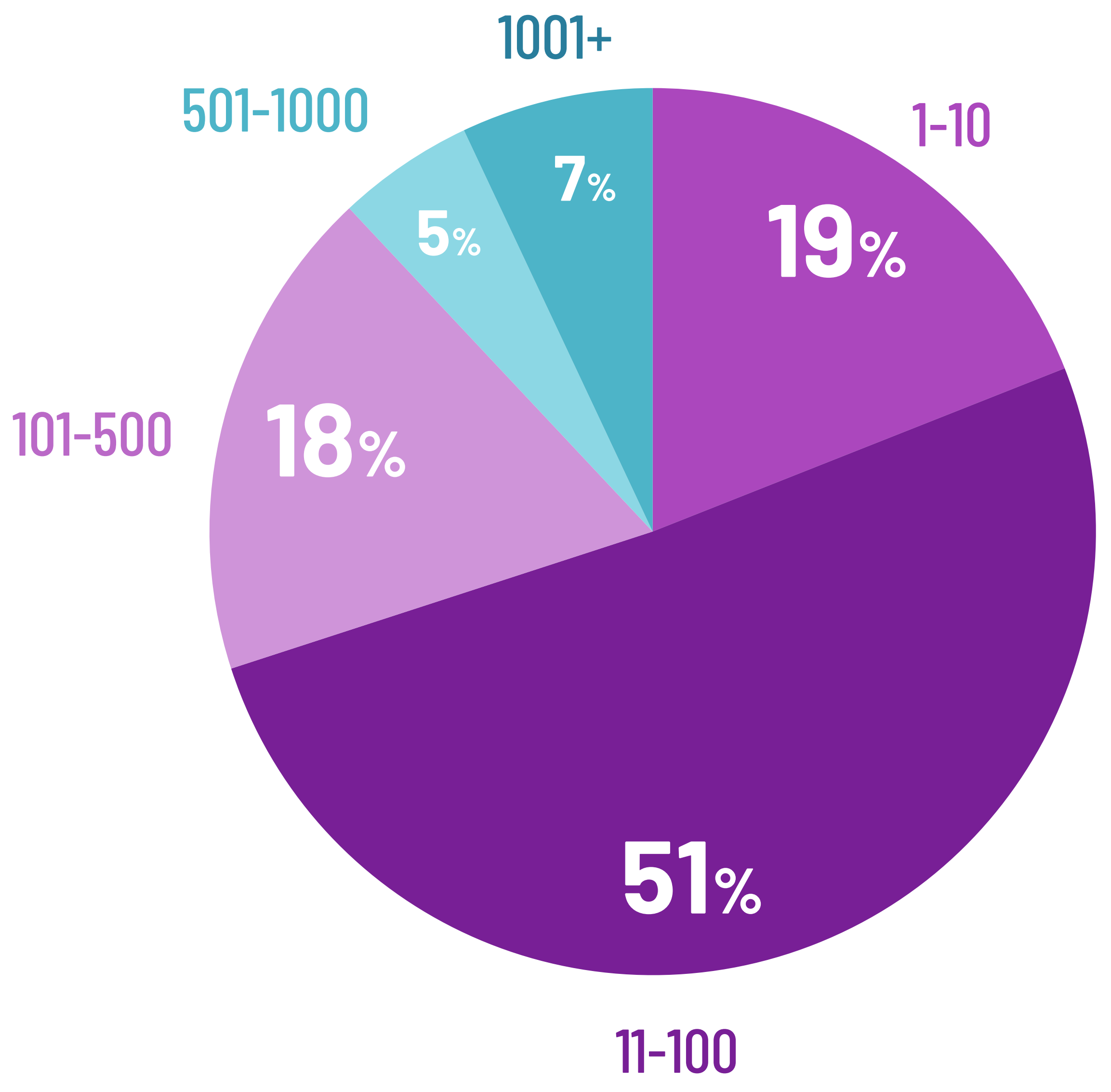
Empirical data from the Salt Security customer base confirms that **the pace of attacks has risen in the last six months**. The percentage of customers enduring fewer than 10 attacks a month fell from 22% to 19%, and more than half (51%) of our customers are suffering between 11 and 100 attacks every month. **A surprisingly high number of customers (12%) are suffering more than 500 attacks every single month**. The ability to leverage automation to find and stop those attacks is crucial to the business being able to continue operations.

In the past 12 months, what security problems have you found in production APIs? (*Select all that apply*)



Salt customer data

Average number of attacks per month per customer



Most companies remain unprepared for the onslaught of API attacks

34% lack any API security strategy at all, despite security topping the list of concerns (40%) about their companies' API programs

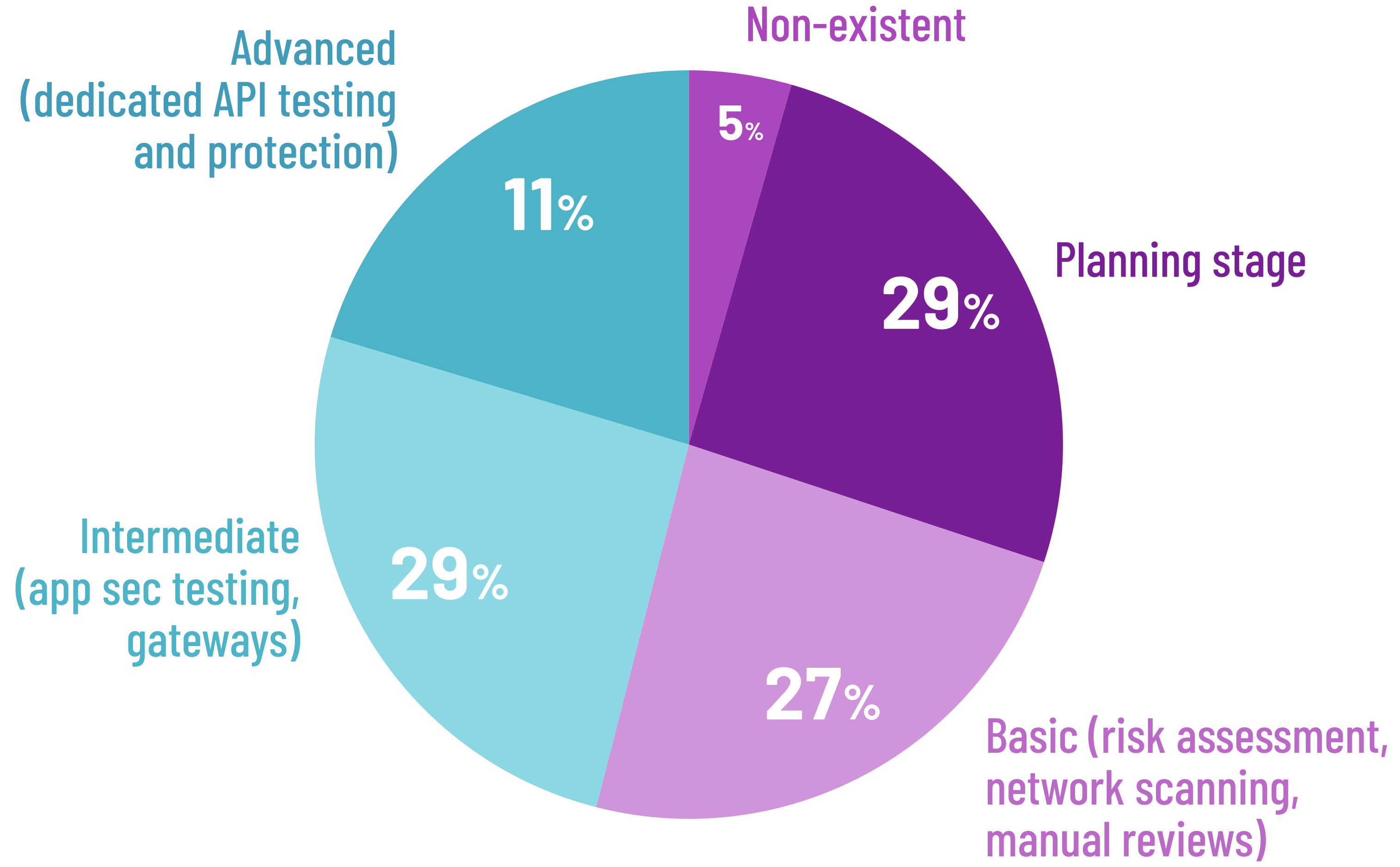
Despite increased use of APIs and a raft of highly publicized API security incidents, far too few companies using production APIs have implemented an effective API security strategy. **More than a third (34%) of respondents have no strategy in place**, and a quarter (27%) have just a basic strategy. **Only 11% have an advanced strategy, with dedicated API testing and protection**.

The security industry is all too familiar with the pain of a shortage in qualified security experts. When it comes to API security, those numbers become even more discouraging. **More than a third (35%) of respondents identify that a lack of expertise or resources/people inhibit their ability to implement an optimal API security strategy**. Budget is the second biggest inhibitor, with 20% of organizations citing budget constraints as an obstacle to implementing their API security strategy.

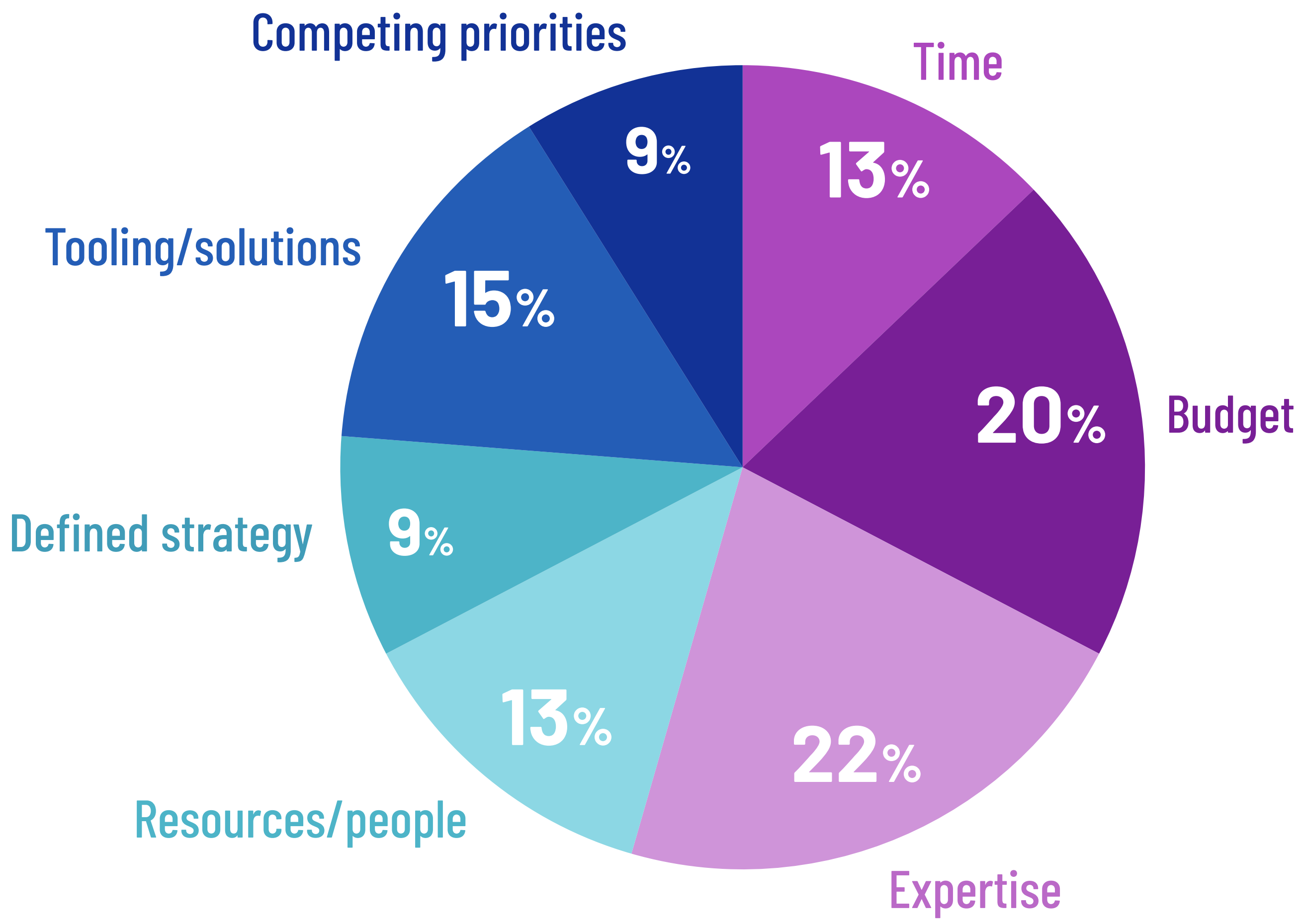
Anxiety over security gaps continues to dominate the list of concerns about companies' API programs. Among survey respondents, **22% cite a lack of investment in pre-production security, and another 18% worry about a lack of runtime or production security**. Beyond those issues, concerns are more evenly distributed, with 19% worried about a lack of clearly defined requirements and documentation, 18% concerned about a reliance on manual processes that slow down delivery, and 11% troubled over the lack of testing.

Companies are facing an urgent need to define and implement a robust API security program to shield their business from the risks associated with API attacks.

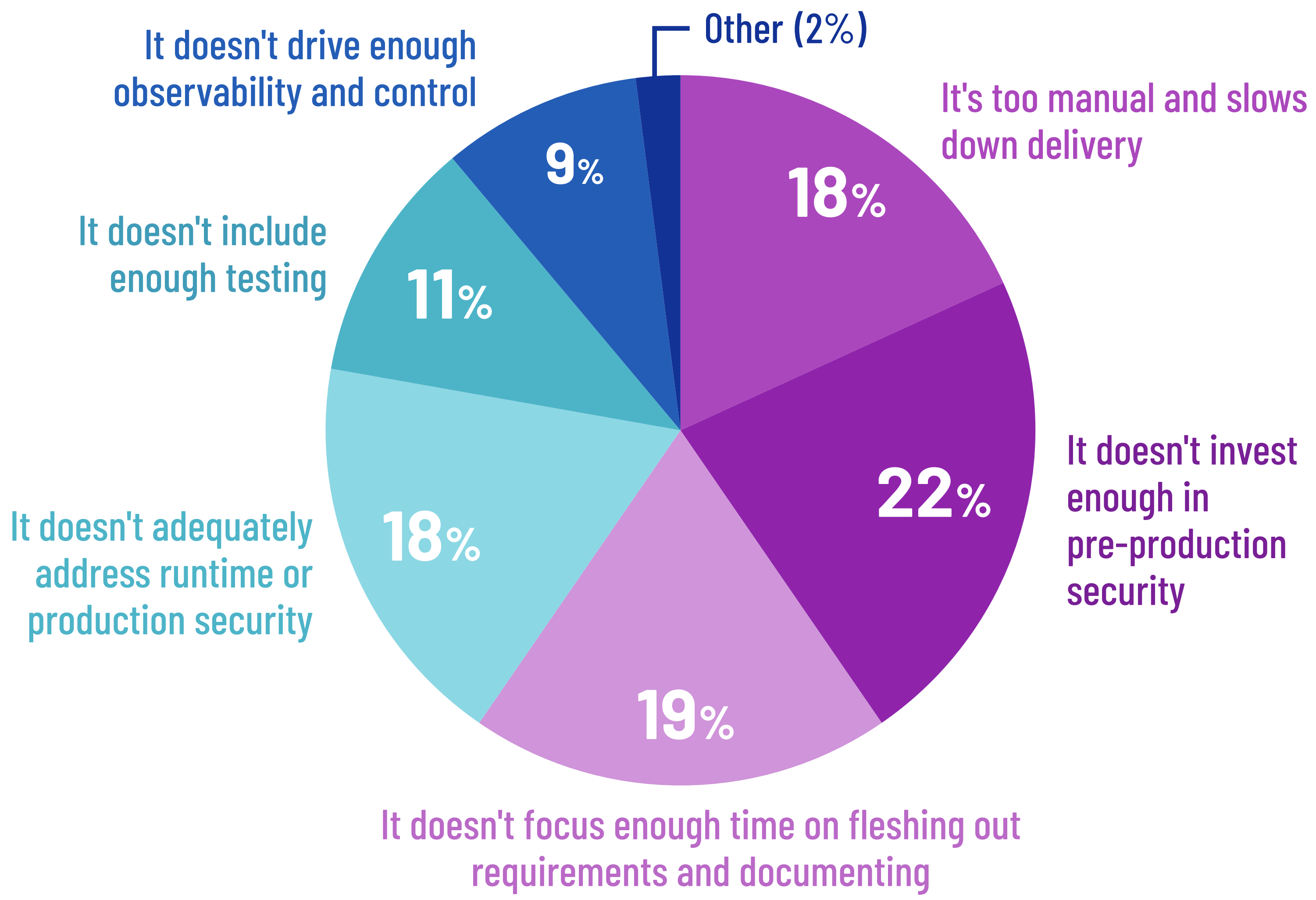
How would you describe the security strategy for your API development program?



What is the biggest obstacle keeping you from implementing an optimal API security strategy?



What is your biggest concern about your company's API program?



WAFs and gateways cannot protect organizations from API attacks

With 95% experiencing an API security incident last year, relying on gateways (55%) or WAFs (37%) is keeping organizations vulnerable

Traditional tools for managing APIs and protecting against application attacks continue to serve their original purpose, but **hoping that tools such as API gateways or WAFs can defend against today's sophisticated API attacks is a fool's errand**. The heavy reliance on analyzing log files (45%) reinforces how slow and reactive today's approaches are – the bad actors will be long gone with valuable data by the time a security analyst can parse log files.

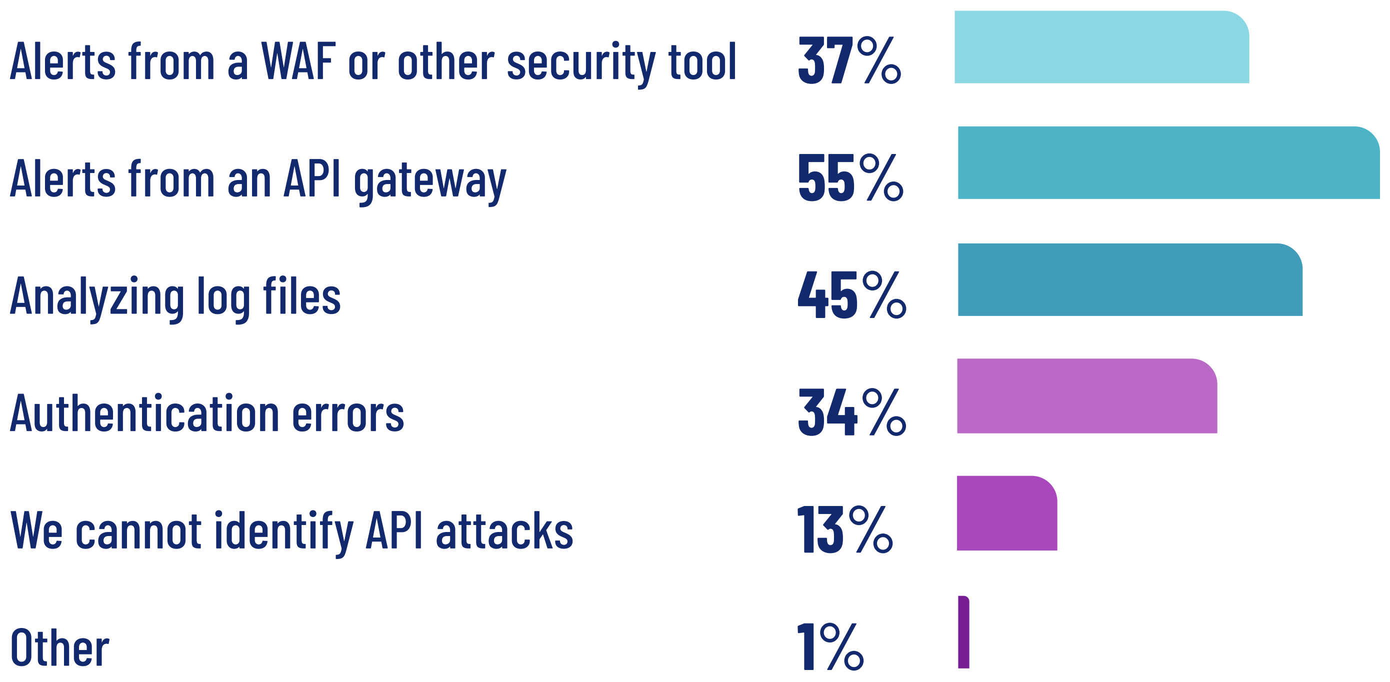
WAFs use proxy architectures to apply signatures that detect well-known attacks such as cross-site scripting (XSS), SQL injection (SQLi), and JSON injection. Next-gen WAFs have evolved to support more modern deployment options, often in the form of containerized architectures. What they will never be able to do, however, is hold enough data to stitch together context of API use over time, across millions of users, so they cannot identify the reconnaissance activity of a bad actor trying to learn and then exploit weaknesses in your APIs.

Many API gateways employ traditional protections such as authentication, authorization, encryption, and rate-limiting. While they are essential for protecting parts of an application, they're not nearly enough to protect against the threats in the OWASP API Security Top 10.

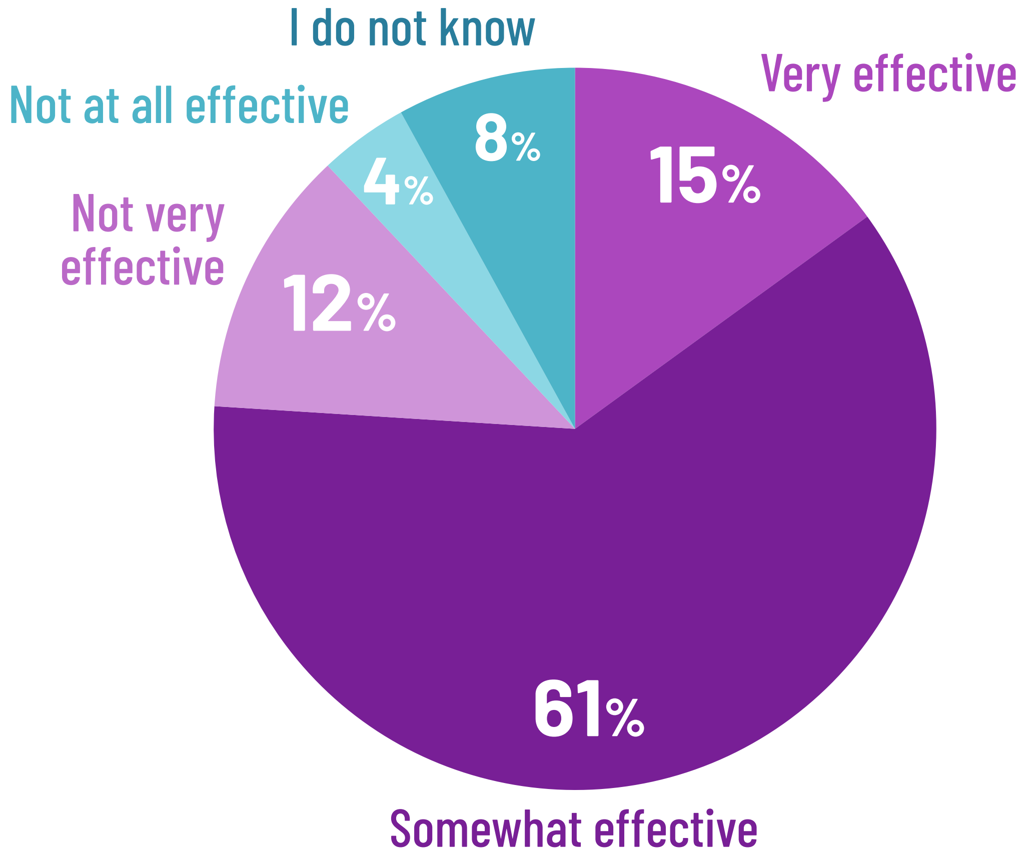
Authentication provides little protection for customer-facing applications. For many of these applications, setting up an account takes a matter of minutes, and an attacker then has access to view and probe the API. Given that **94% of exploits within the Salt customer base happen against authenticated APIs**, authentication alone is clearly insufficient.

Most survey respondents acknowledge the gaps in their existing tools' efficacy in stopping bad actors, with **85% noting their tools are not very effective in stopping API attacks**.

How do you identify an attack or attacker targeting your APIs? (*Select all that apply*)

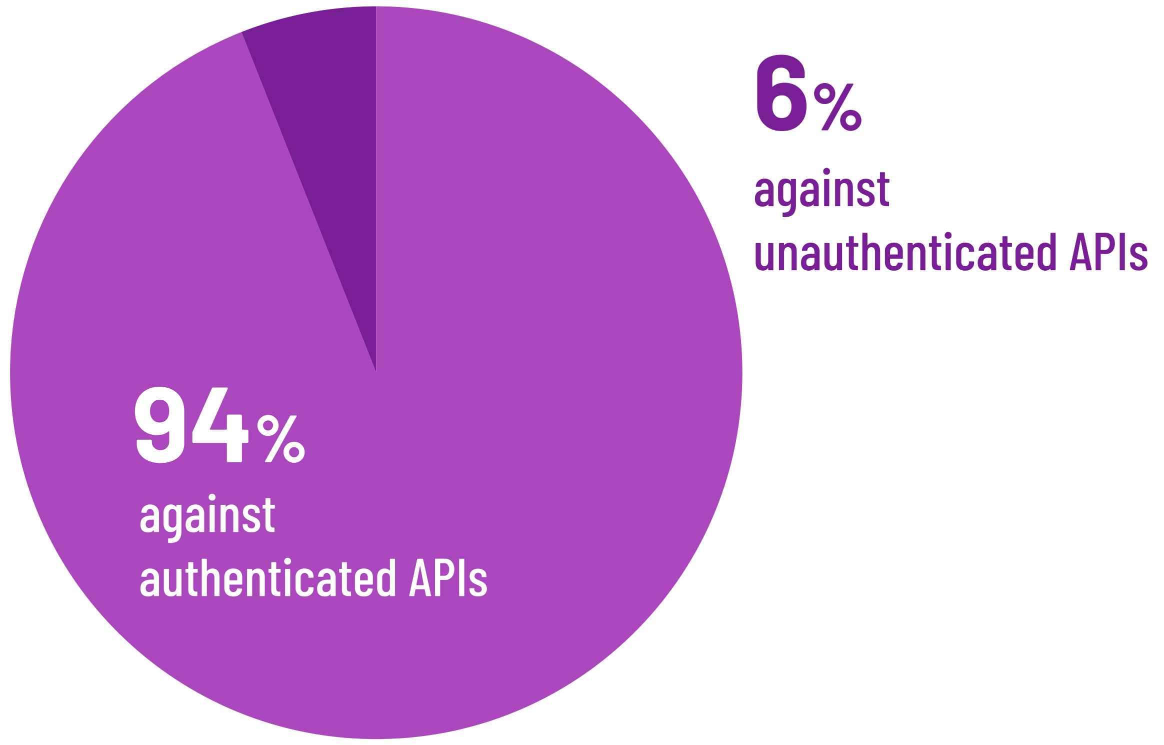


How effective are your existing tools in preventing API attacks?



Salt customer data

% API exploits against authenticated vs. unauthenticated APIs



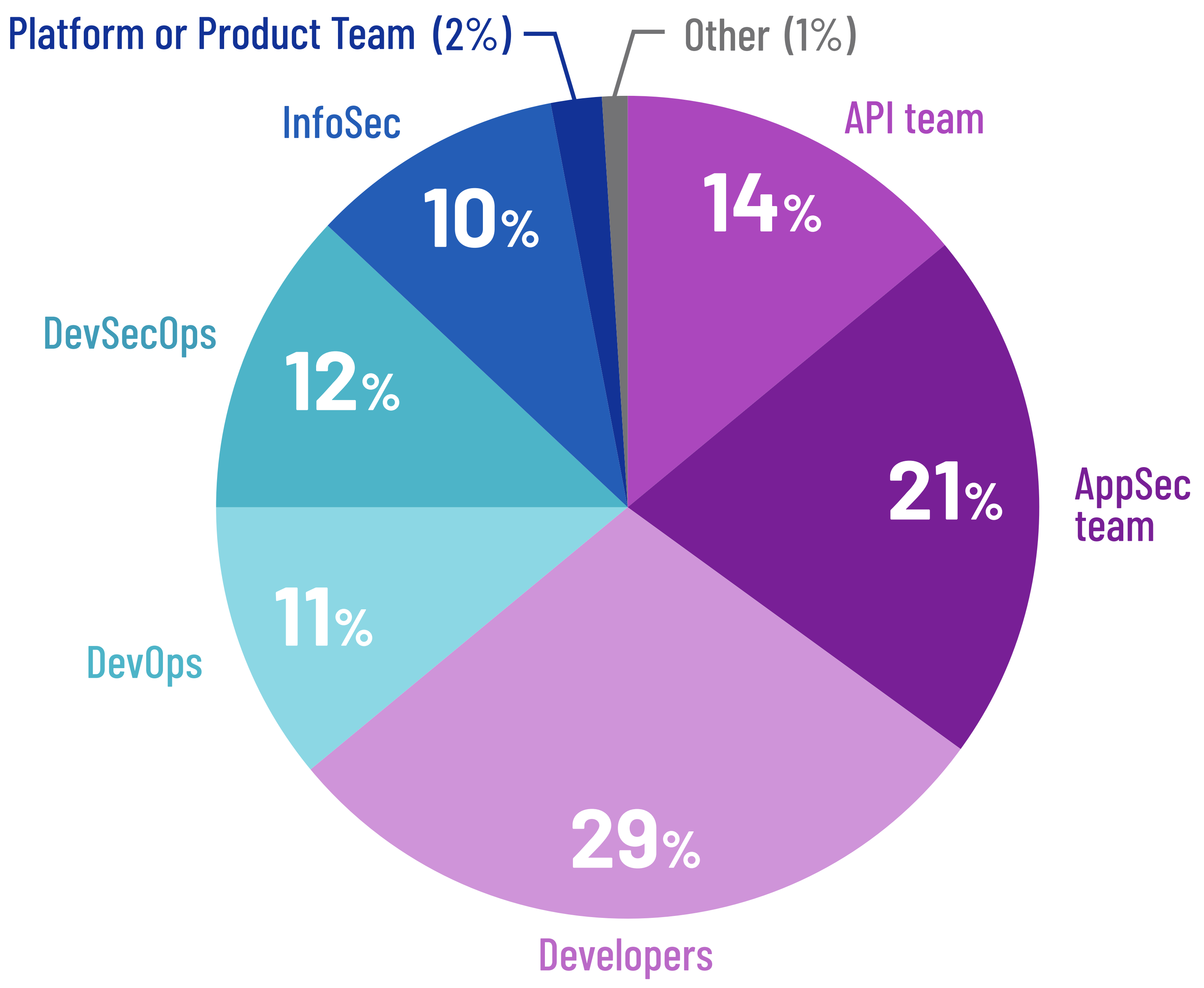
An overreliance on “shift left” tactics in API security is failing the modern enterprise

Only 27% of respondents are identifying API security gaps in runtime, despite the fact that pre-prod validation cannot detect gaps in logic flow

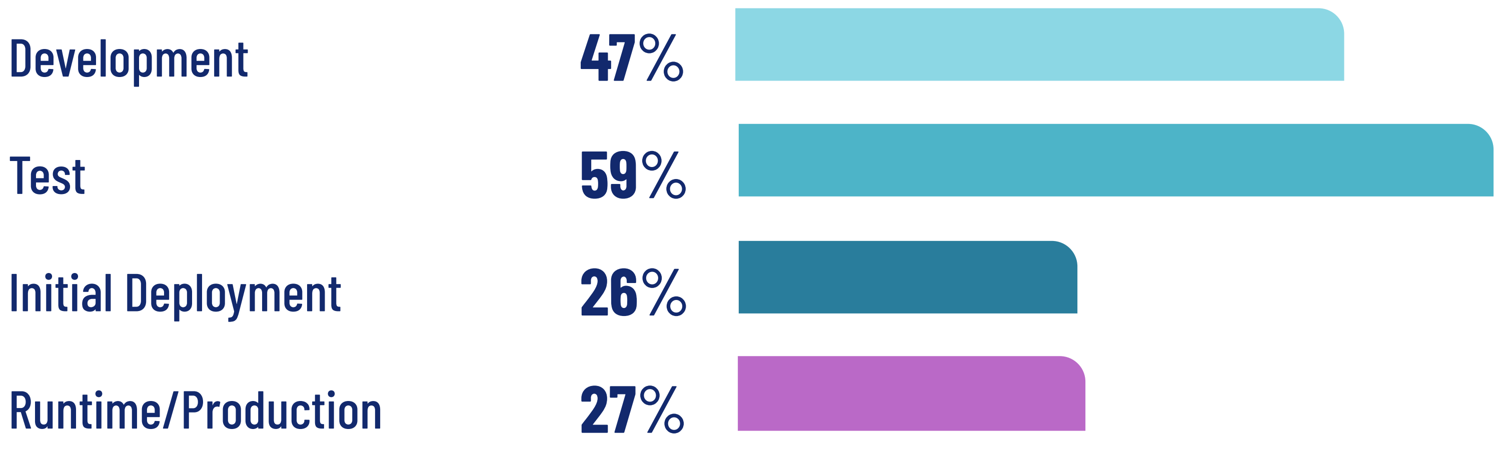
Organizations are struggling to find the right owner for API security. More than half (52%) say primary responsibility sits with developers, DevOps, or DevSecOps. **Only 31% of respondents put the responsibility of API security onto AppSec or InfoSec teams**. Given runtime protection is so fundamental to effectively protecting APIs against attack, and 95% have faced an API security incident, this “shift left” focus may not be serving companies well.

Of course organizations should be looking to identify and remediate API vulnerabilities in pre-production. However, the challenge comes in thinking that “shift left” principles applied to API security will yield sufficient protection. The reality is that API attacks target gaps in logic flow, and no amount of API testing and scanning will find those gaps. They can be surfaced only with production traffic running. Finding security problems before code is launched into production is important, but that tactic has fundamental limitations when it comes to robust API security.

Who's primarily responsible for securing APIs?



At what point(s) in the development life cycle does your company identify and remediate API security gaps? *(Select all that apply)*



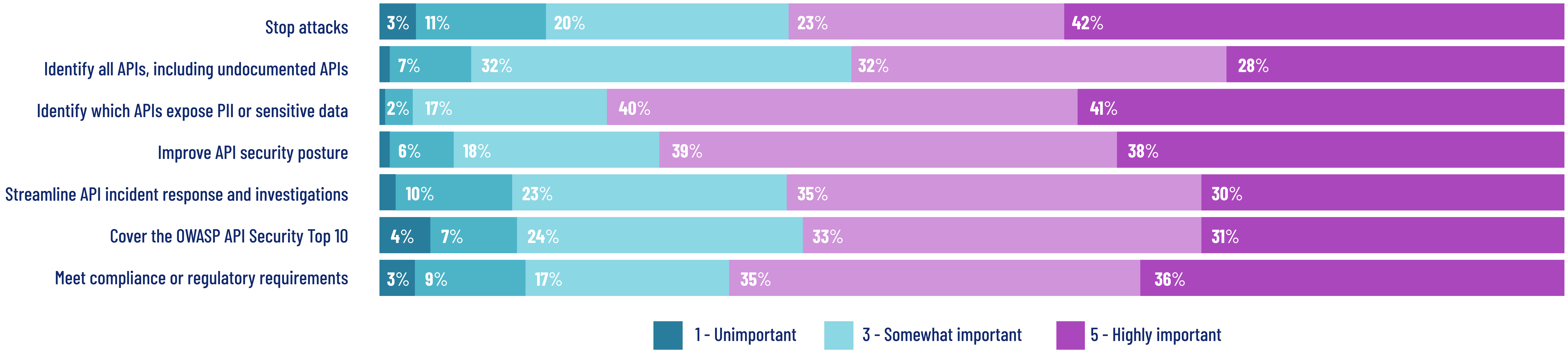
Stopping API attacks remains the #1 security priority for enterprises

42% of respondents cite that ability as the most important attribute in an API security platform

For the third time in a row, stopping API attacks remains the most valued capability in an API security platform. **Identifying which APIs expose PII and sensitive data follows as a close second (41%), another aspect of API security tied to risk at runtime**.

Respondents value the ability to harden APIs over time for improved security posture as the third most critical capability of an API security platform, with 38% rating it "highly important." Meeting compliance or regulatory requirements comes in at number 4, with 36% identifying that attribute as "highly important."

On a scale of 1-5, how would you rate the value of each of these attributes of an API security platform? (*1 is unimportant and 5 is highly important*)



You can’t protect what you can't see – 83% of respondents lack confidence that their API inventory is complete

Frequent API updates complicate the challenge – 40% of respondents are updating API at least weekly

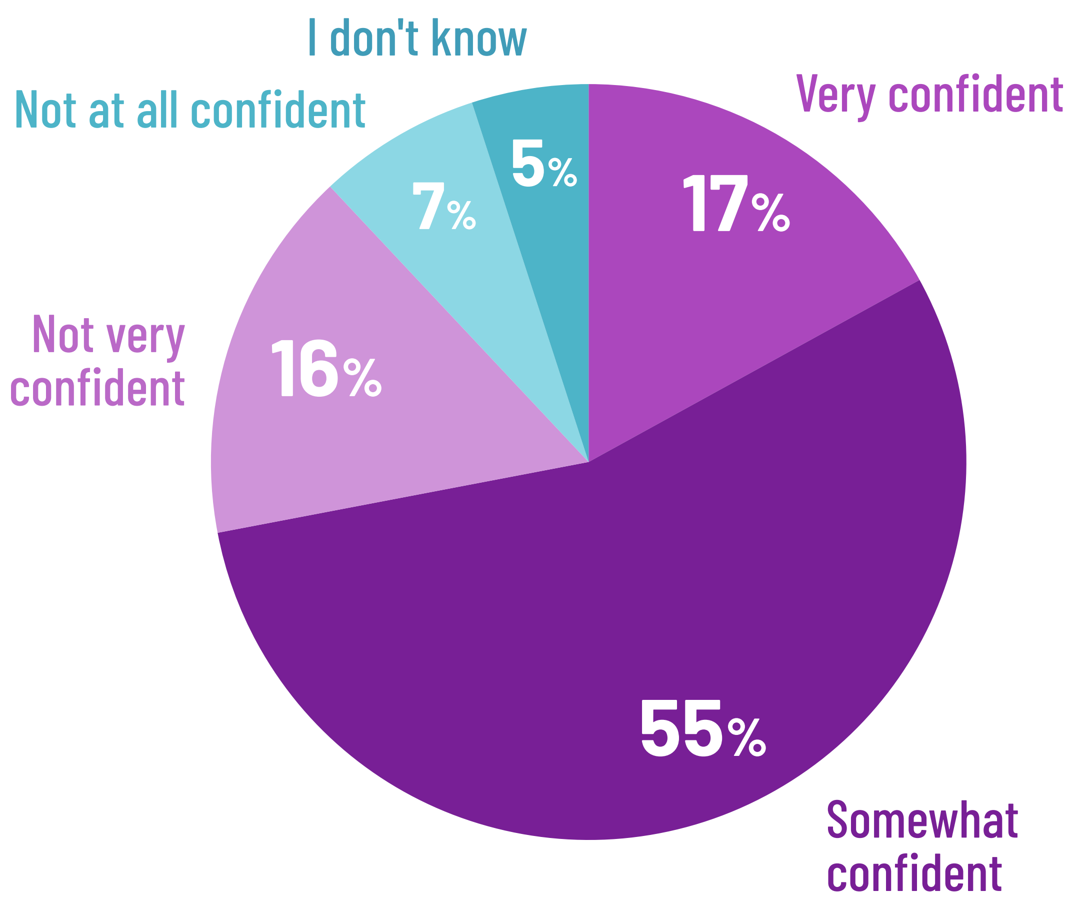
Most organizations lack a full accounting of their APIs. Our survey results show 83% of respondents lacking confidence that their API inventory is complete, and **nearly a quarter (23%) admit they’re not very or not at all confident in their inventory**.

Anecdotally, most customers cite incomplete documentation and APIs released outside of API management platforms as the prime reasons. If they do embark on a project to fully document their APIs, it’s a manual and time-consuming exercise that takes developers away from their primary task of building new apps and functionality.

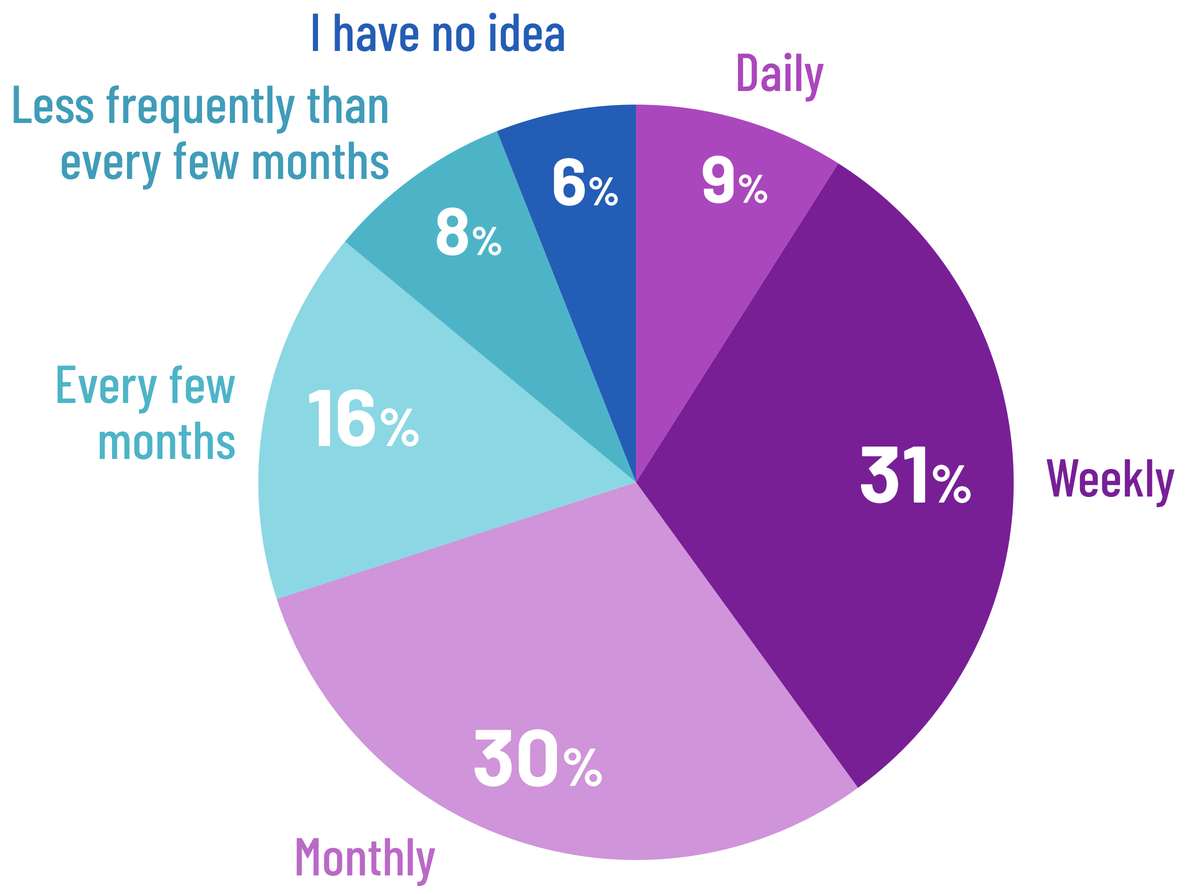
Complicating the process is the reality that such documentation efforts quickly get out of date because organizations are updating their API so frequently. In our survey, **9% of respondents are updating their APIs every day, 31% are doing so weekly, and only 24% of respondents are updating them less often than every month**.

In our customer engagements, we routinely find 40% to 80% more APIs in the environment than the organization was aware of, and one customer recently admitted their estimate was short by a factor of 10. Such “shadow” APIs introduce additional risk in that many of them expose sensitive data.
Bottom line – API discovery needs to be automated and continuous to keep up with the speed of development.

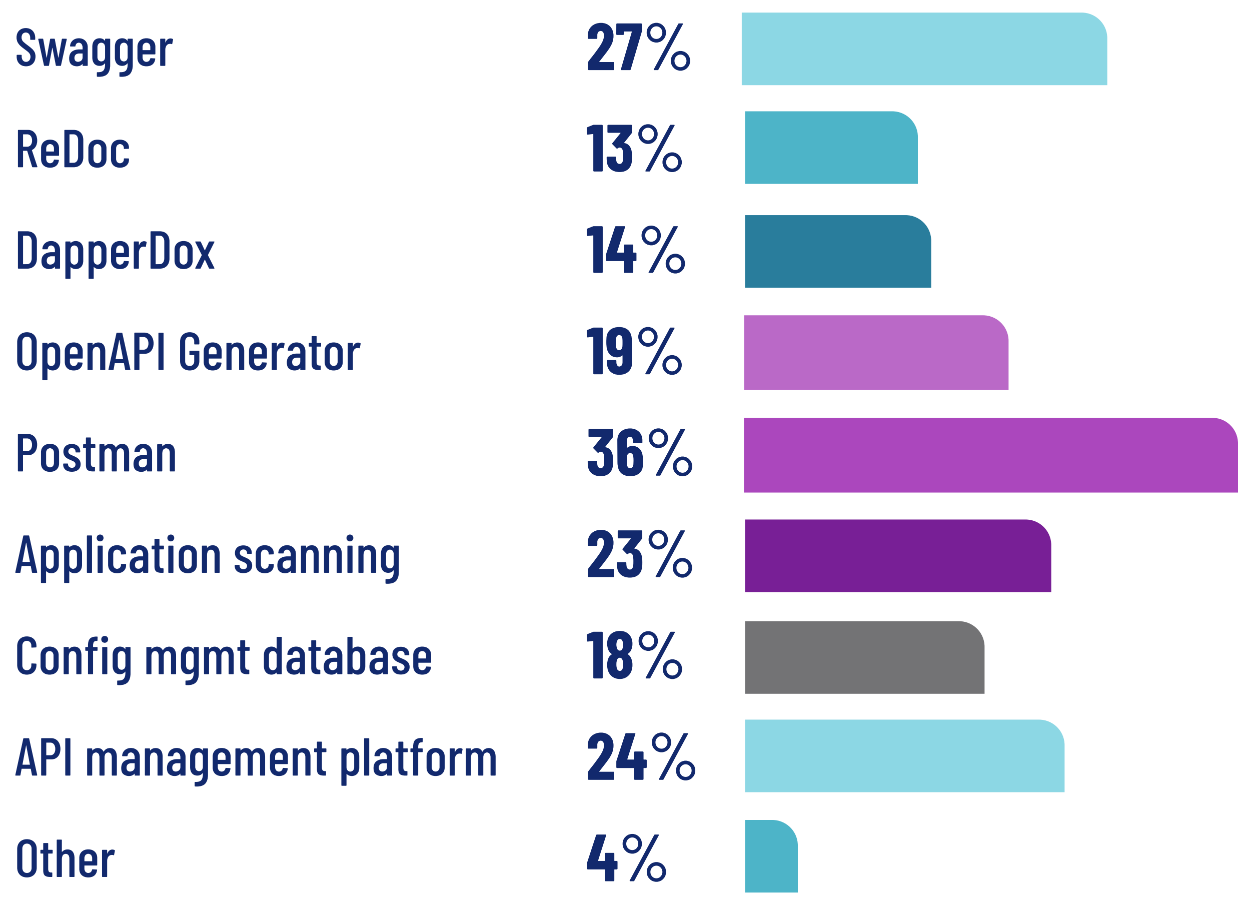
How confident are you that your API inventory is complete?



On average, how often do your primary APIs get updated?



What mechanism(s) do you use to document and inventory your APIs? *(Select all that apply)*



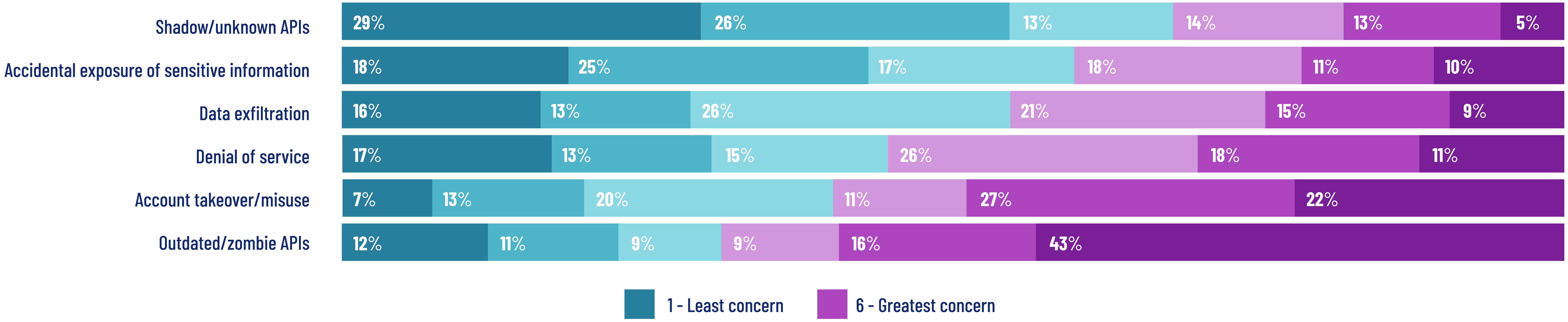
Nearly half of respondents (43%) cite “zombie” or outdated APIs as their top API security concern

Account takeover was the leading worry for 22% of respondents

Concerns over “zombie” or outdated APIs dominate the list of top worries for the third time in a row, dwarfing other concerns by more than twice the margin. **For 43% of respondents, the biggest API security worry focused on “zombie” or outdated APIs**. As companies release new APIs, teams often forget to deprecate older versions, which can leave companies vulnerable as no one is patching or validating these APIs any longer.

Account takeover ranked as the top concern for less than a quarter (22%) of respondents. For the other 45% of respondents, concerns were fairly evenly split across denial of service, accidental exposure of sensitive data, and data exfiltration. “Shadow” or unknown APIs was the top worry for just 5% of respondents, which begs the question of whether they realize the scope of these unknown APIs.

Please rank the following risks, with 1 being your least concern and 6 your greatest concern, related to API security



86% of respondents lack confidence that they know which APIs expose sensitive data

11% of respondents admit to having zero confidence in or knowledge about sensitive data being exposed via APIs

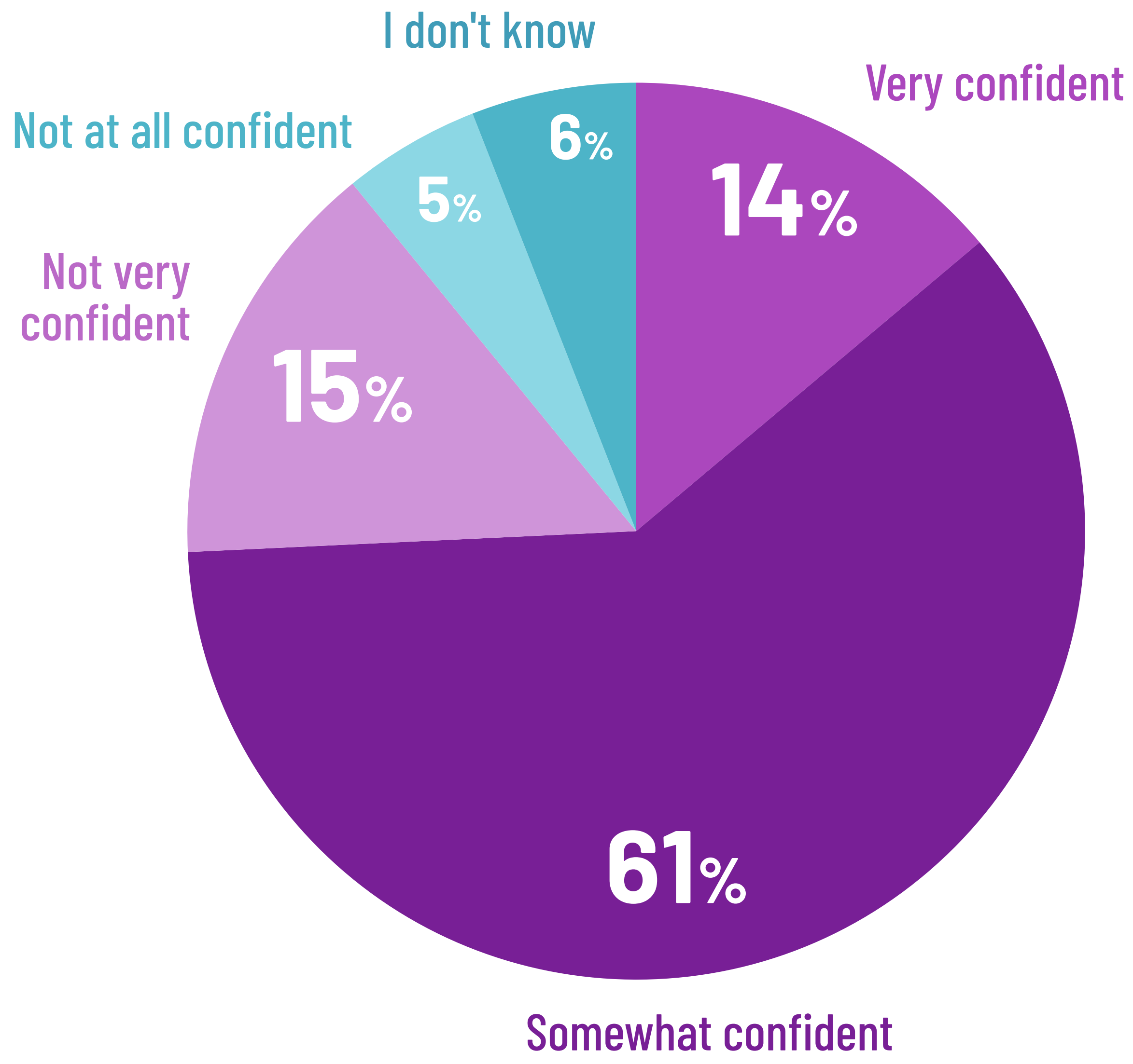
While it's crucial to know about the existence of every API in your inventory, it's just as important to know which of those APIs is exposing sensitive data such as PHI or PII. With **11% of organizations admitting they have no confidence in or knowledge about sensitive data being exposed via APIs**, this gap is clearly keeping companies at risk.

For Salt customers, 91% of APIs expose sensitive data. Accidental exposure of PII, PHI, or other sensitive information leaves organizations susceptible to significant business risk. Data exposure, privacy impacts, and regulatory penalties frequently follow such API security gaps.

Even APIs that work as designed but can be abused to pull excessive amounts of data can lead to these business risks. These so-called "leaky APIs" are designed to enable data sharing that's critical to the business, but bad actors can manipulate them to extract far more data than they should be able to access.

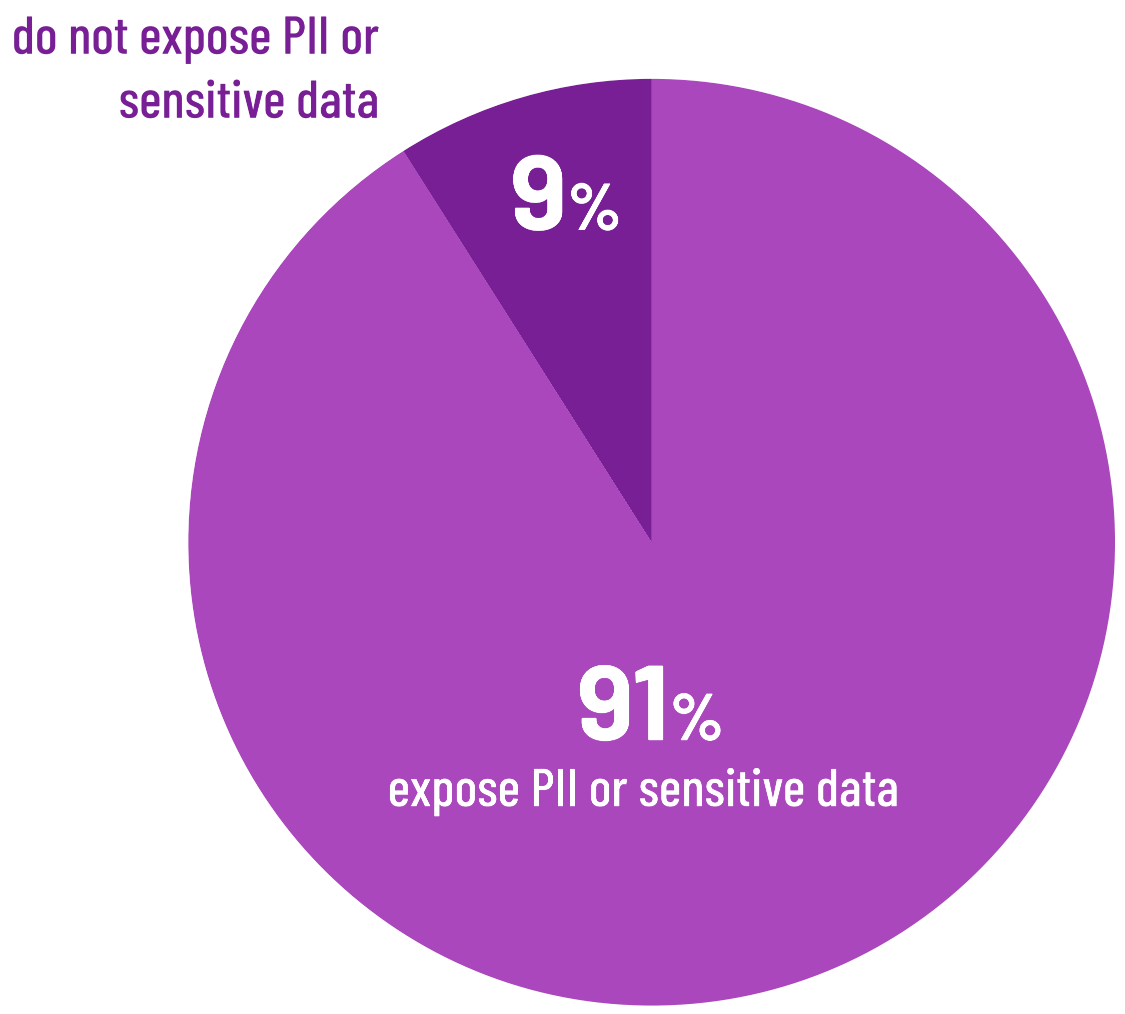
Organizations need automation to detect these API abuses and stay out of the headlines.

How confident are you that your API inventory provides enough detail about your APIs, including exposure of sensitive data or PII?



Salt customer data

Number of APIs that expose PII or sensitive data



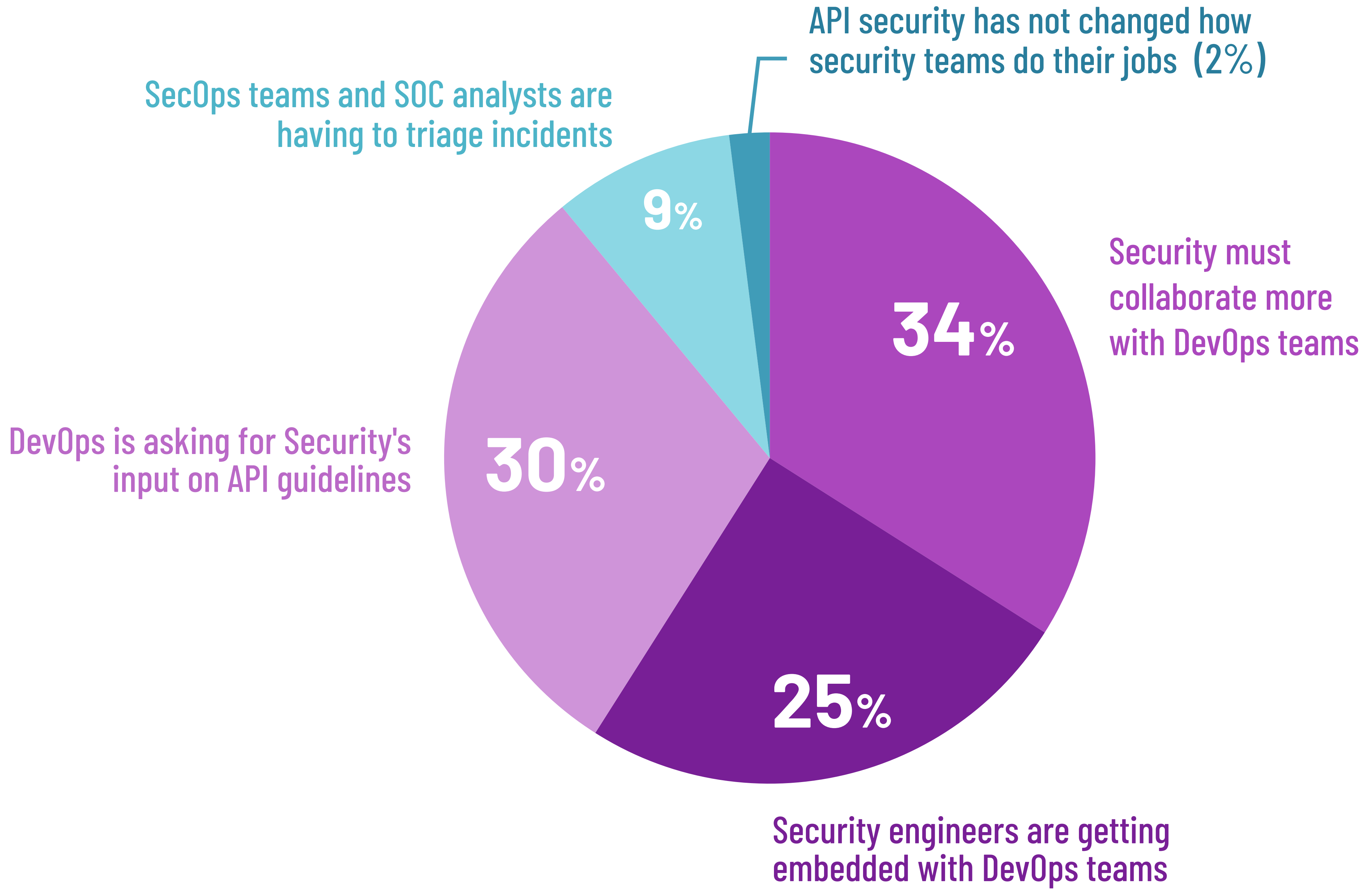
Cause for hope – API security is universally changing how security teams work, for the better

Collaboration and shared input between Security and DevOps teams are on the rise

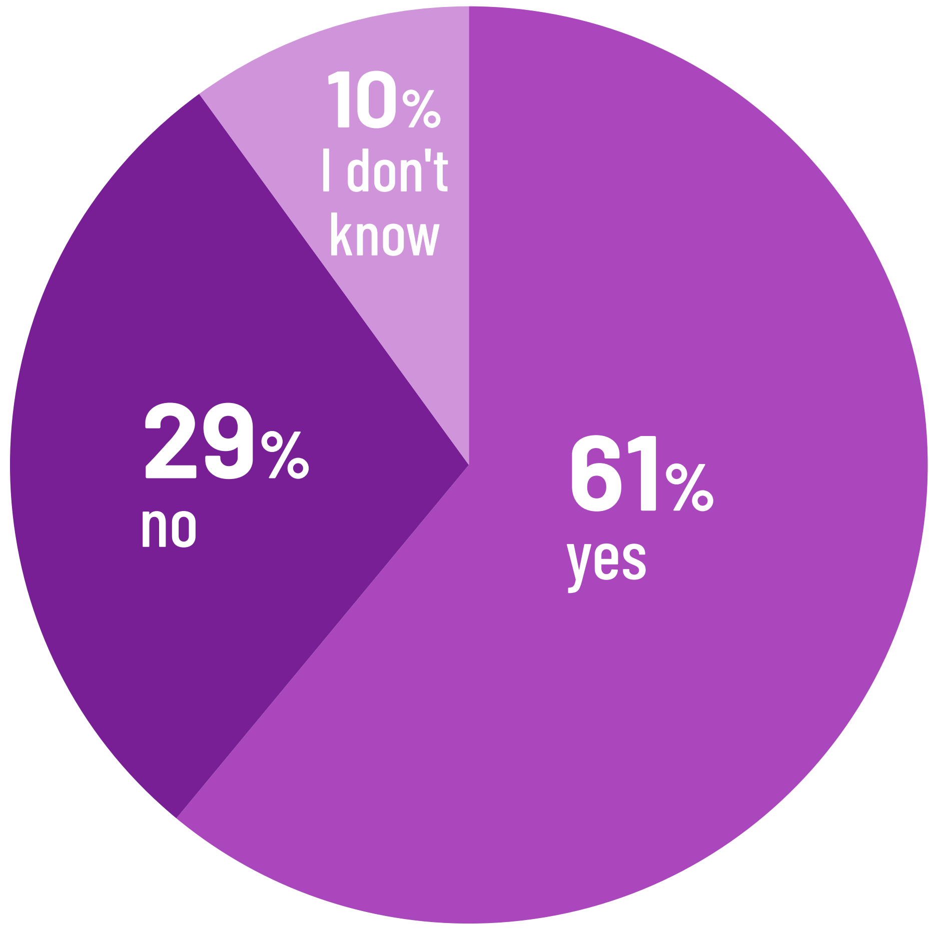
More than a third of respondents **(34%) say Security is collaborating more with DevOps**, and another 30% cite that DevOps seeks input from Security teams to shape API guidelines. A quarter of respondents **(25%) have security engineers getting embedded with DevOps teams**, driving real progress toward DevSecOps. The respondents who say API security hasn't changed the approach of Security teams fell from 9% six months ago to just 2%.

In another promising sign, the percentage of respondents who noted that **Security teams are highlighting the OWASP API Top 10 list of threats grew from 50% six months ago to 61%**. This list provides an excellent starting point for creating a framework for better API security. **Better knowledge and better collaboration are foundational to an organization's ability to effectively combat the rising threat of API attacks**.

How do you feel API security is creating changes in how security professionals do their jobs?



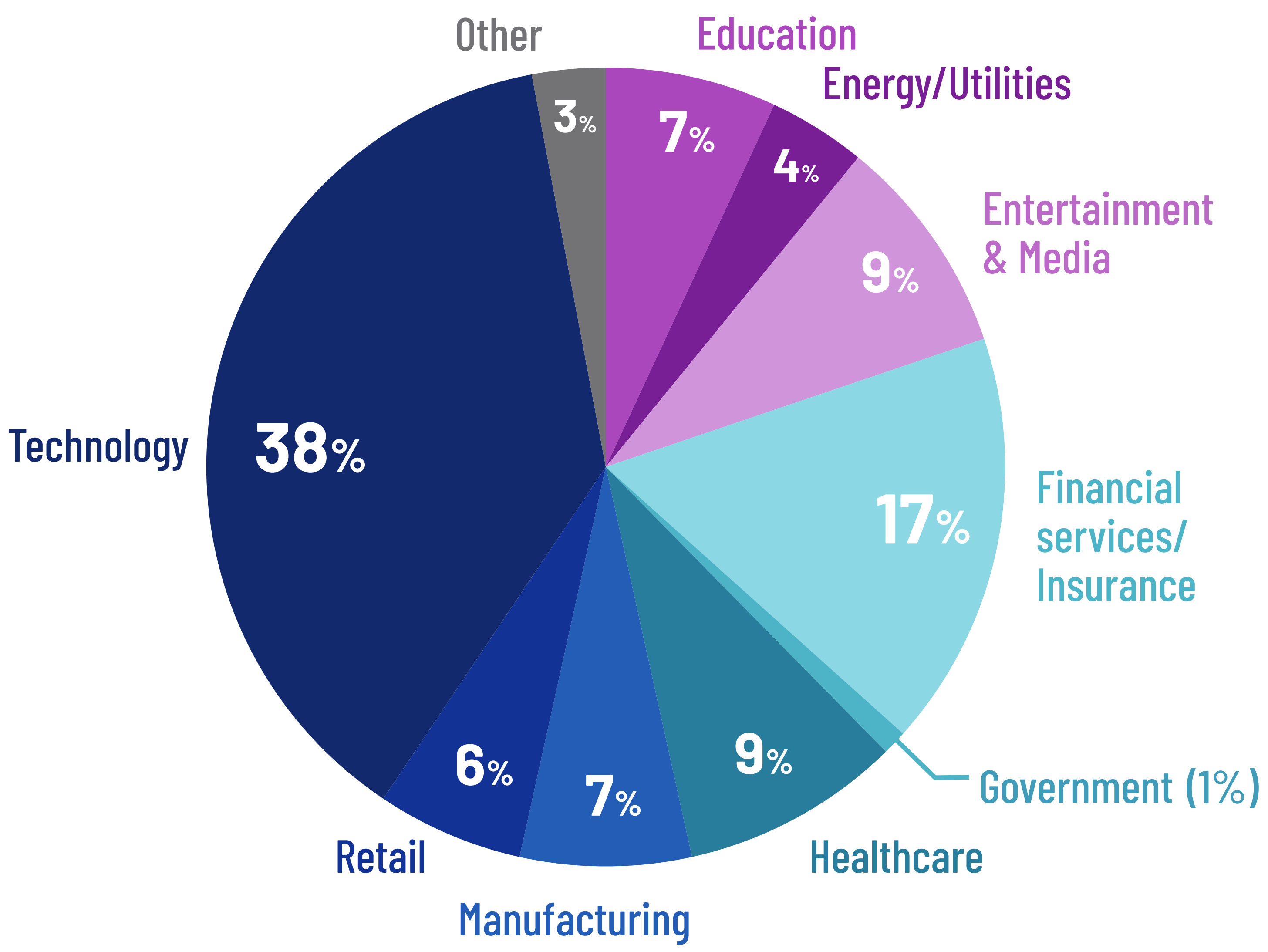
Has your security team highlighted the OWASP API Security Top 10 threats as a focus area for your security program?



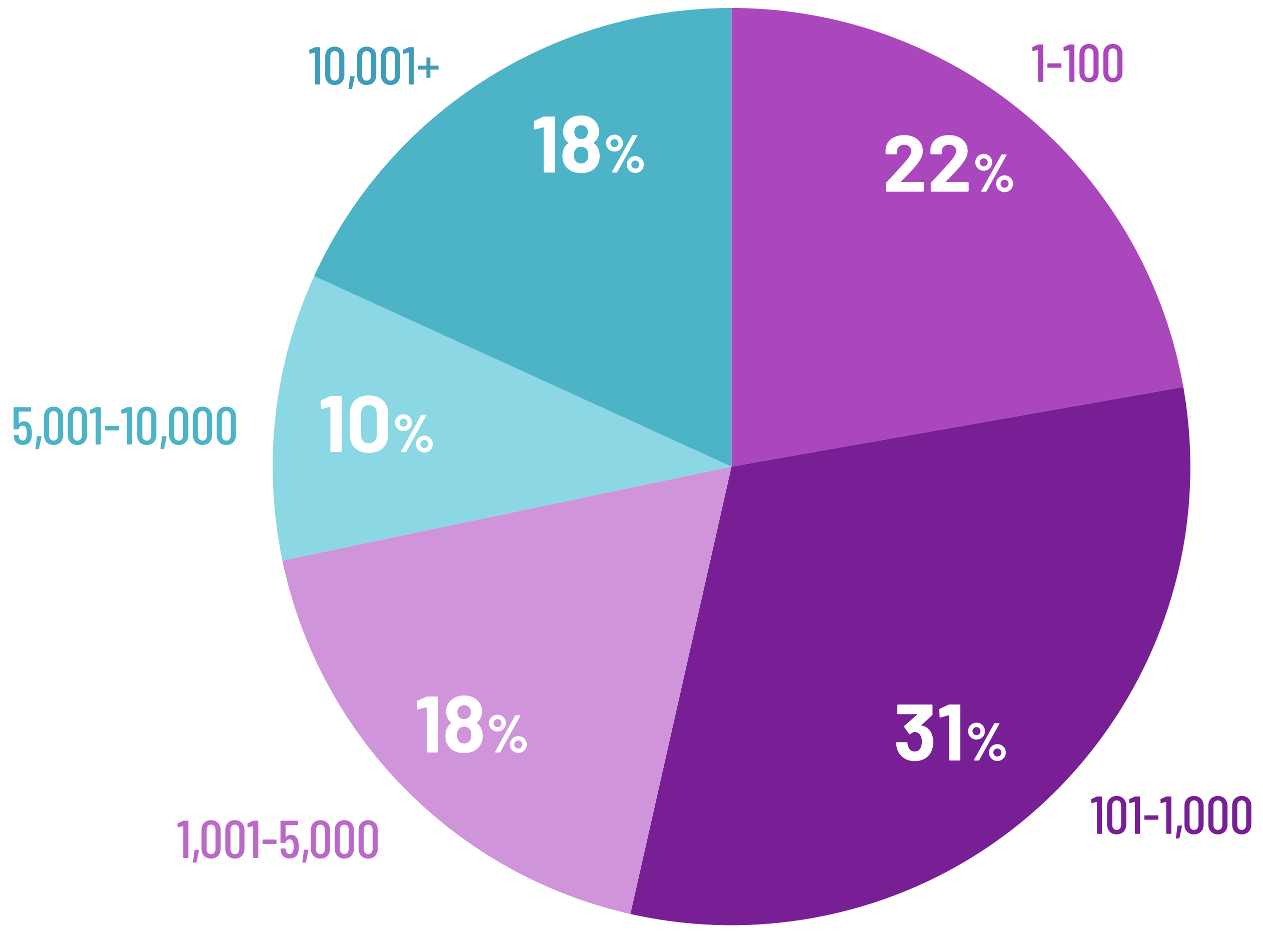
Demographics

The findings in this "State of API Security Report" pull from both empirical data from the Salt Security SaaS platform and survey responses of more than 250 respondents. The survey respondents are well distributed across a range of job responsibilities, industries, and company sizes. More than half **(55%) hold roles related to security**, 19% hold VP or C-level titles, **13% sit on API platform teams**, and 17% are in DevOps. The technology industry represents the largest share of responses at 38%, followed by financial services and insurance companies at 17%. Organizations large and small are evenly represented.

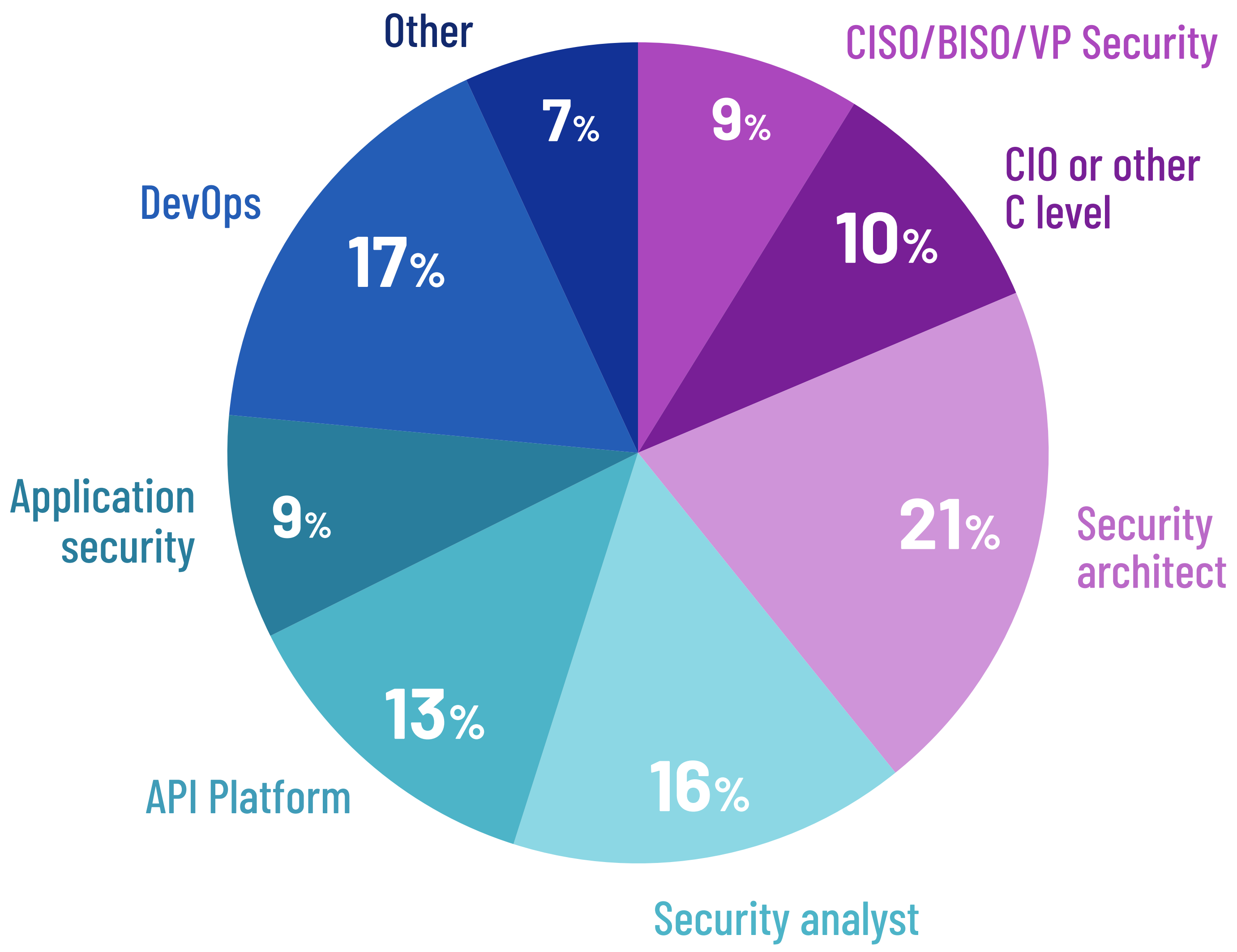
Industry



Size of company (employee count)



What area best represents your functional role?



Recommendations to help mitigate the threat of API attacks

Define a robust API security strategy

The gaps that WAFs and API gateways leave in defending against API attacks are clear, so companies need to define and execute against an API security strategy that covers the full lifecycle of APIs and addresses cross-functional responsibilities. A minimum scope includes API design analysis and drift analysis, automatic and continuous discovery, augmented runtime protections, a feedback loop for developers to use runtime insights to harden APIs, training for SecOps teams to understand and triage API security incidents, and a clear model for shared responsibility across functional groups.

Assess your current level of risk

Validate current API designs against API security best practices, checking whether authentication and authorization controls are in place throughout the sequence of API calls for a given business function, for example. Launch attacks based on the OWASP API Security Top 10 list and see whether your WAF or API gateway can detect them. Emulate the tactics of well-known API security incidents of 2021 to see whether similar business logic flaws exist in your APIs.

Enable frictionless API security across all your application environments

With APIs being the foundation of all application development today, you can't afford to leave some of your environments unprotected. You must be able to apply API discovery and runtime protection on prem and in the cloud and on legacy apps as well as your container and Kubernetes deployments. How you connect the API security tooling into your environments is also crucial – avoid inline deployments, agents, or the need to instrument code to keep your API security platform from being blamed for any application impact.

Tap the power of cloud-scale big data, AI, and ML to pinpoint the subtle probing of API attackers

Since every API is unique, bad actors have to perform extensive reconnaissance to understand how each API works and identify vulnerabilities or gaps in business logic they can exploit. Attackers know how to keep their probing subtle to avoid tripping coarse security protections such as rate limiting on WAFs. To see these nefarious activities, an API security platform must be able to capture millions of data points over a long period of time, since API attacks can take weeks and months to unfold. Then, the platform must tap AI and ML to discern the recon activities of a bad actor and correlate them into a single attacker profile to avoid alerting on each bad action. Such robust data collection and correlation requires cloud-scale big data and cannot by achieved using VM-based collection.

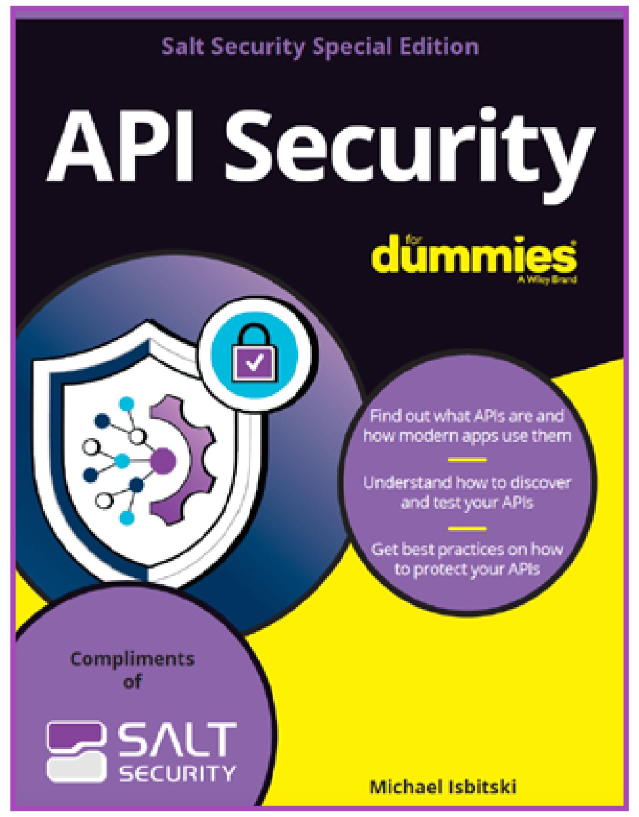
Don't over-rotate on shift-left tactics

Shift-left and secure build pipeline approaches have their merits. But many API security gaps can't be detected as part of code review – they can be detected only in runtime. Look for an API security platform that complements pipeline testing and analysis with robust runtime protection. Shift-left tactics take much longer to deliver value and ultimately offer only limited value, since they identify only a fraction of API security risk and and they leave your security teams dependent on developers to work through a backlog of vulnerability fixes. Get your APIs protected today with runtime security – then you can make hardening APIs over time a realistic goal.

Get a Salt Security demo »

Resources to help you get started

Check out these API security guides to help your team upskill in API security and develop a game plan for effective deployment in your environment.



Salt Security Special Edition

API Security

for dummies A Wiley Brand

Find out what APIs are and how modern apps use them

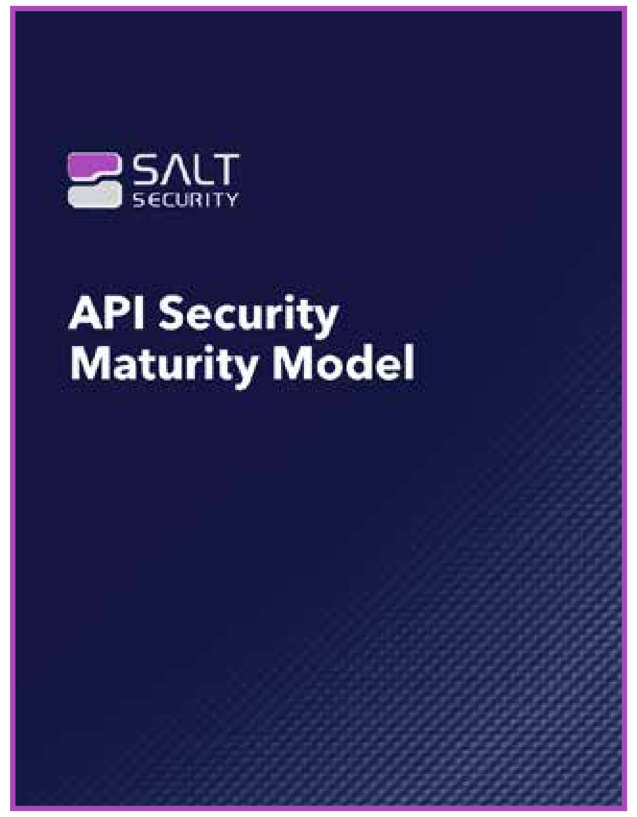
Understand how to discover and test your APIs

Get best practices on how to protect your APIs

Compliments of SALT SECURITY

Michael Isbitski

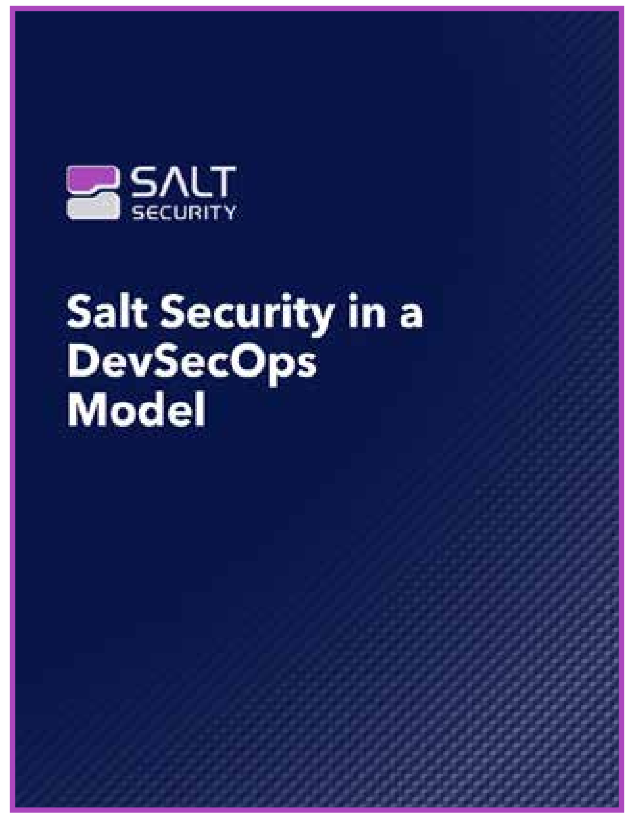
Salt Security Special Edition API Security for Dummies eBook »



SALT SECURITY

API Security Maturity Model

Salt API Security Maturity Model »



SALT SECURITY

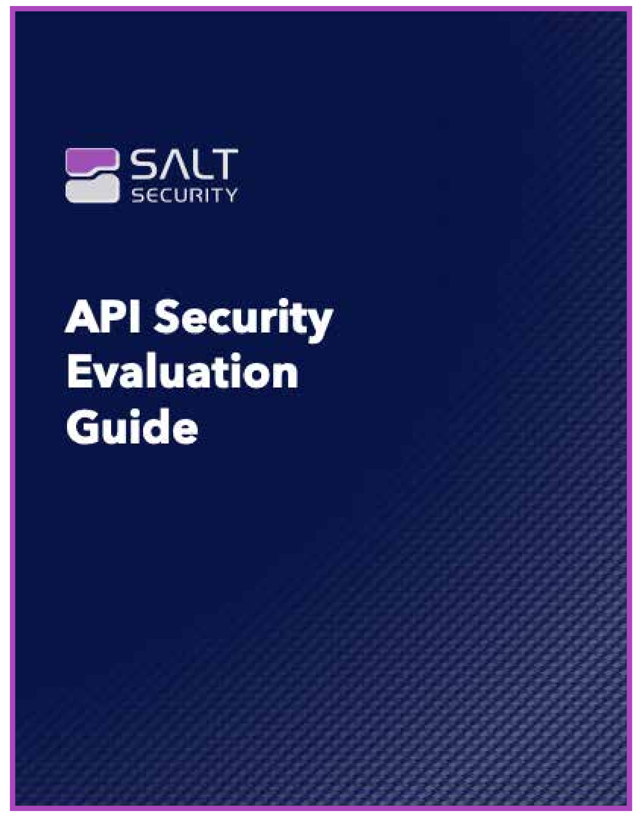
Salt Security in a DevSecOps Model

Salt Security in a DevSecOps Model »

SALT SECURITY

The Limits of Shift-Left in Enabling API Security

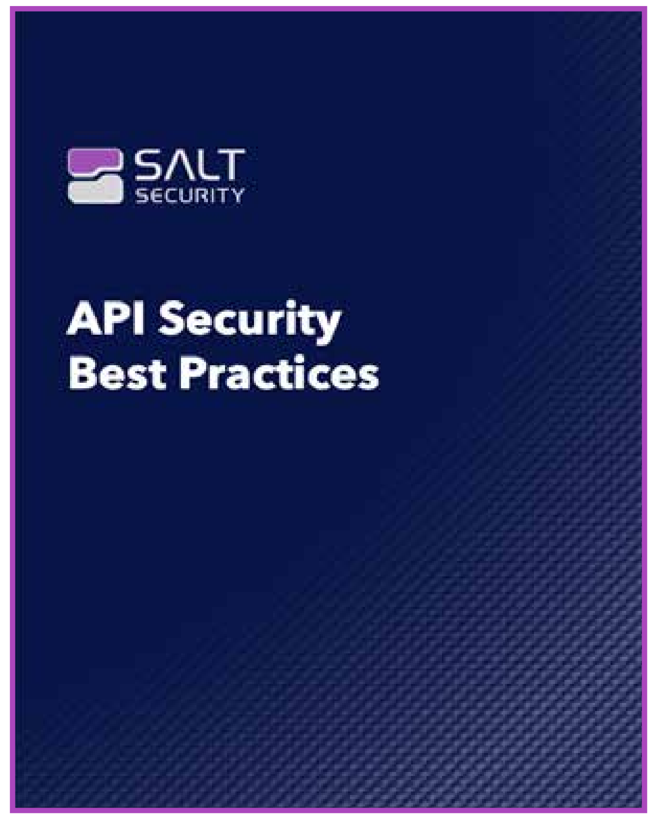
Salt Security Limits of Shift-Left in Enabling API Security »



SALT SECURITY

API Security Evaluation Guide

Salt Security API Security Evaluation Guide »



SALT SECURITY

API Security Best Practices

Salt Security API Security Best Practices Guide »

About Salt Security

Salt Security protects the APIs at the heart of all your modern applications.

The Salt Security API Protection Platform is the industry's only patented solution to prevent the next generation of API attacks. Only Salt delivers the context you need to protect your APIs across build, deploy, and runtime phases, using machine learning and AI to automatically and continuously identify and protect APIs. Deployed in minutes, the Salt platform learns the granular behavior of your APIs and requires no agents, configuration, or customization to pinpoint and stop API attackers.

The Salt platform provides critical advantages that enable complete protection of APIs across across the full development lifecycle:

- Complete coverage – we cover all your APIs across all your environments, including load balancers, API gateways, WAFs, and Kubernetes clusters, running on prem or in the cloud. We deploy out of band to avoid any performance impact on your critical applications.
- Unparalleled detection – our API Context Engine (ACE) architecture uses our cloud-scale big data with ML and AI to correlate activity across millions of users and API calls. We baseline the unique logic of your APIs, and against that backdrop of normal behavior, the reconnaissance activities of bad actors become easy to pinpoint. We're able to identify and stop attackers before they can achieve their objective.
- Proactive security and a DevOps feedback loop – we provide API design analysis, API drift analysis, attack simulation for pipeline validation, and runtime insights with extensive remediation details developers can use to harden APIs and improve API security posture over time.

SALT LABS

About Salt Labs

Salt Labs identifies API threats and vulnerabilities in customer deployments and in the wild. Our in-depth API threat research reports document the steps of an exploit, including the processes and tooling, to reveal an attacker's approach, the details of an exploit, the risk to the business, and the steps an organization can follow to avoid becoming victim to a similar attack. We also apply our research findings to improve the ML and AI algorithms at the heart of our API security platform, so all our customers benefit from our on-going research. Our industry reports, such as this State of API Security Report, tap empirical and survey data to educate the market on API security trends.

SALT
SECURITY