



The Invicti AppSec Indicator

SPRING 2023:

Scan More. Fix More.
Reduce Risk.



REPORT OVERVIEW

- 04 INTRODUCTION AND METHODOLOGY
- 08 EXECUTIVE SUMMARY



YEAR-OVER-YEAR TRENDS

- 10 KEY TAKEAWAYS FOR SCANS
- 13 INDUSTRY SPOTLIGHT: MANUFACTURING
- 15 VULNERABILITY TAKEAWAYS



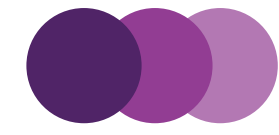
VULNERABILITY DEEP DIVE

- 19 REMOTE CODE EXECUTION (RCE)
- 20 OS COMMAND INJECTION (OSCI)
- 21 CROSS-SITE SCRIPTING (XSS)
- 22 SQL INJECTION (SQLI)
- 23 SERVER-SIDE REQUEST FORGERY (SSRF)
- 24 LOCAL FILE INCLUSION (LFI)



LOOKING AHEAD

- 26 THE BENEFITS OF DAST
- 28 THE FOUR PILLARS OF RELIABLE APPSEC
- 30 CONCLUSION



REPORT OVERVIEW

Software security: Increased scanning frequency drives a decrease in risk

Web application and API security isn't a one-and-done effort. Vulnerabilities are continuously introduced (or re-introduced) into web applications and APIs as code grows and changes, and environments evolve. For a successful security strategy to permeate your software development lifecycle, companies must shift their efforts both left and right by continuously scanning for security issues earlier in the software development pipeline and in production.

In our bi-annual AppSec Indicator report, we uncover insights and trends to guide best practices in vulnerability identification and remediation.

For this year's Spring edition of the Invicti AppSec Indicator, we analyzed data from 1.7 million scans conducted by the 1,700 customers that use our cloud dynamic application security testing (DAST) offering, representing approximately half of our entire customer base.

1.7
MILLION
SCANS

OVER
1,700
CUSTOMERS
ANALYZED

1
APPSEC
INDICATOR
REPORT

This cumulative data pull marked a significant change in volume from last year's report, using an expanded set of algorithms to analyze the data. We used different perspectives including small to medium businesses (those up to 1,999 employees) and enterprises (those with more than 2,000 employees) to better understand the breakdown of scanning frequencies and vulnerability discovery by company size, and also across industries to understand patterns from different customer groups.

We uncovered two key trends which indicate that companies are improving their security posture. Firstly, **enterprises are increasing their scanning activity**, scanning more web applications and APIs, and scanning them more often. And secondly, the **scans are uncovering fewer critical and high severity**

vulnerabilities, indicating that increased security testing improves security posture in the long run.

Continuous scanning is more critical than ever as the average cost of a data breach is on the rise, topping \$4.3 million globally and surpassing \$9.4 million in the United States alone.¹ Clearly, there's still a gap for some organizations that aren't investing in the necessary security solutions. According to research from ESG, when asked about security challenges that businesses face with faster development cycles, 43% of respondents noted a lack of visibility and control for security. Additionally, 35% said that there is a lack of consistency for their security processes.²

¹ IBM, *Cost of a Data Breach 2022*

² ESG, *Prioritizing Shift Left Security Solutions to Keep Up with Faster Release Cycles*



KEY DATA INSIGHTS FROM THIS REPORT

Attackers know that gaps in cybersecurity are a proverbial cash cow for them, and often web applications are a common entry point for such attacks. Verizon's 2022 Data Breach Investigations Report indicates that about **70% of security incidents in 2021 were connected to the hacking of web applications.**³ Tackling the issue of consistency in securing web applications is a three-pronged initiative for businesses large and small, requiring that they review and restructure their tools, processes, and culture. Let's look at what this year's data tells us.

1 Scanning activity is on a steady annual increase, up **50%** from 2019 to 2022, as customers are scanning their web applications and APIs more often.

2 After steady increases in prior years, the percentage of scans with severe vulnerabilities declined **19%** from 2021 to 2022.

3 Remote code execution (RCE) vulnerabilities show a notable increase, with the average percentage of apps with RCE flaws up **40%** since last year.

4 The percentage of scans with severe cross-site scripting (XSS) vulnerabilities continues to decline, dropping **12%** from 2021 to 2022.

³ Verizon, *2022 Data Breach Investigations Report*


HERE'S WHAT YOU NEED TO KNOW:

Scanning is steadily increasing year over year since 2019. There was a 50% increase in scan frequency per account over the last 4 years, showing a trend of companies scanning more internal and external web apps and APIs, and at greater frequency as they expand security testing left (in development) and right (more frequent production testing).



50%

Percentage of scans with a severe vulnerability declined 19% year over year. While application scanning increased, there was a decrease in the percentage of severe vulnerabilities overall as AppSec programs are able to find and fix more issues on a more frequent basis. The average percentage of scans with critical or high vulnerabilities decreased from 11.8% in 2021 to 9.6% in 2022.



19%


Percentage of scans with RCE vulnerabilities shows a significant increase.

Instances of remote code execution increased 40%, from 2021 to 2022. End goals such as ransomware and cryptojacking have increased the use of RCE vulnerability exploits with remote malware code.



40%

Percentage of scans with cross-site scripting vulnerabilities declines for the fourth year running. Percentage of scans with a severe cross-site scripting vulnerability declined 12% from 2021 to 2022. Consistent improvements in the prevalence of severe XSS vulnerabilities (from 2019 to 2022) are likely attributed to the rise of modern single-page applications (SPAs) written in JavaScript frameworks like React, Angular, and Vue, which provide more guardrails against XSS.



12%

EXECUTIVE SUMMARY

This year's Spring AppSec Indicator report shows us that organizations are scanning a much larger portion of their attack surface and scanning more frequently as well. **As a result, the data shows their security posture is improving.**

As more web apps are designed, built, deployed, and updated daily, risk profiles continue to grow.

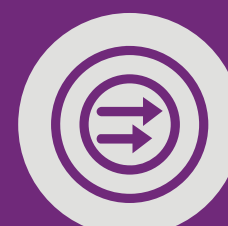
Following in the footsteps of Invicti customers who increase scanning to secure their application attack surface, organizations have an opportunity to shrink the risk they carry day to day. As evidenced by this year's data, the path to improved security posture includes:



Broader coverage by finding and testing more web applications and APIs, incorporating dynamic application security testing (DAST), software composition analysis (SCA), and interactive application security testing (IAST) for a deeper view of the entire app.



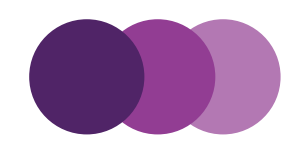
Efficiency from a fine-tuned strategy that enables teams to test and remediate code quickly by embedding AppSec into the software development lifecycle (SDLC) through integrated tools used by development, security, and operations teams.



Accuracy that instills confidence in the data and remediation guidance through capabilities like proof-based scanning and automatic feedback delivered right to the developers.



Scanning frequency to keep up with new vulnerabilities introduced from increasingly rapid software release programs, where applications and APIs are regularly scanned in production and development.

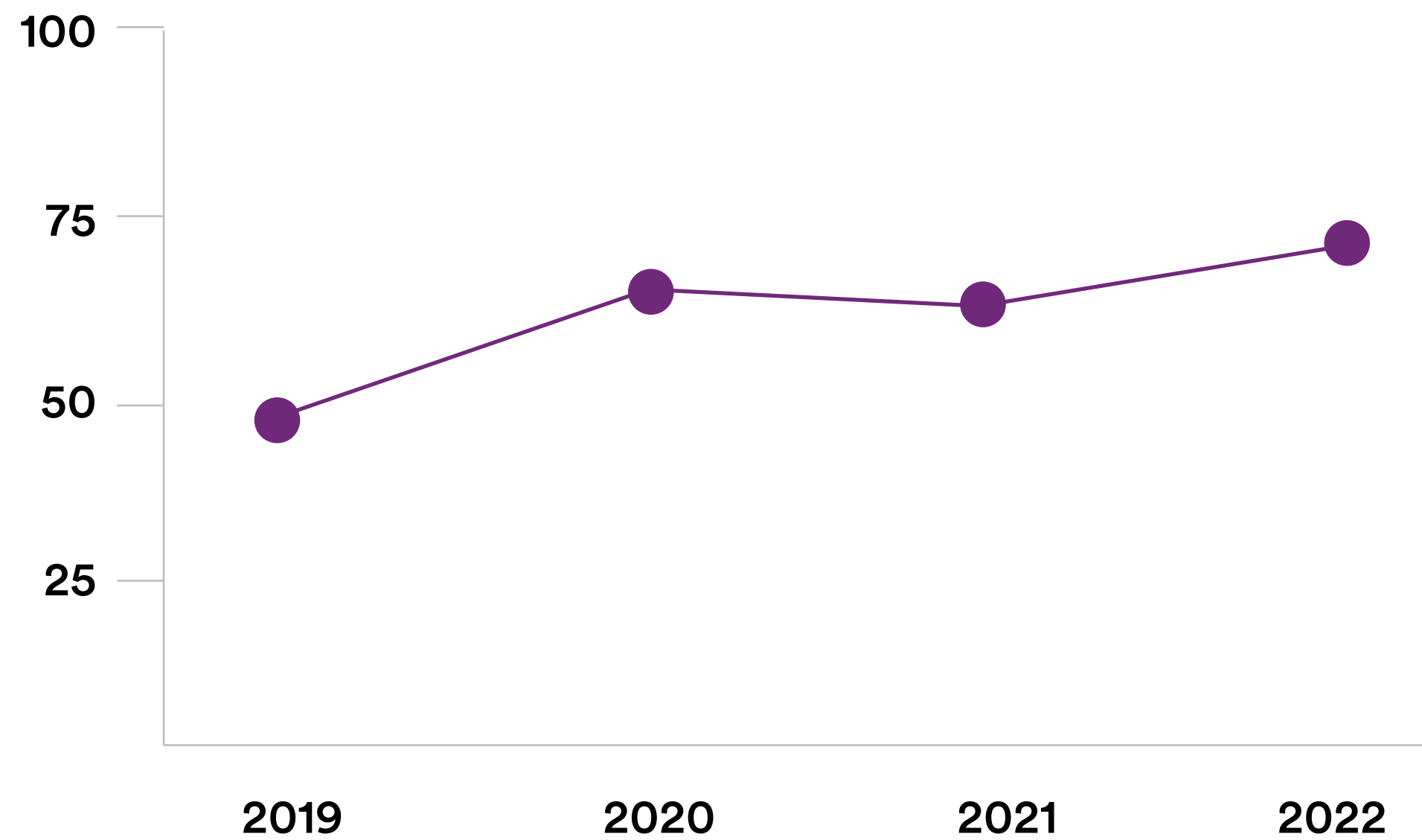


YEAR OVER YEAR TRENDS

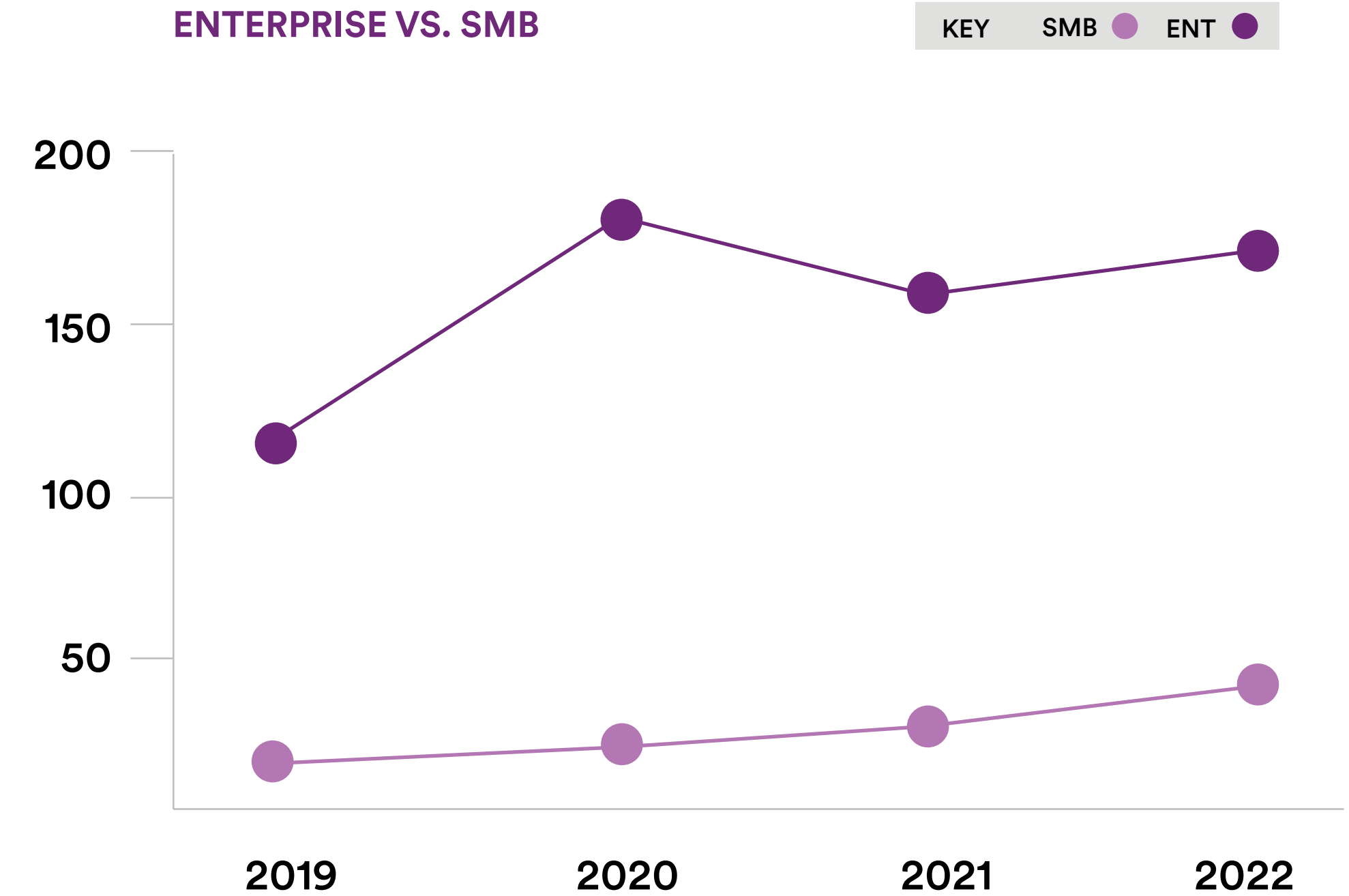
Scanning frequency is on the rise, increasing 50% since 2019.



AVERAGE SCANS PER ACCOUNT

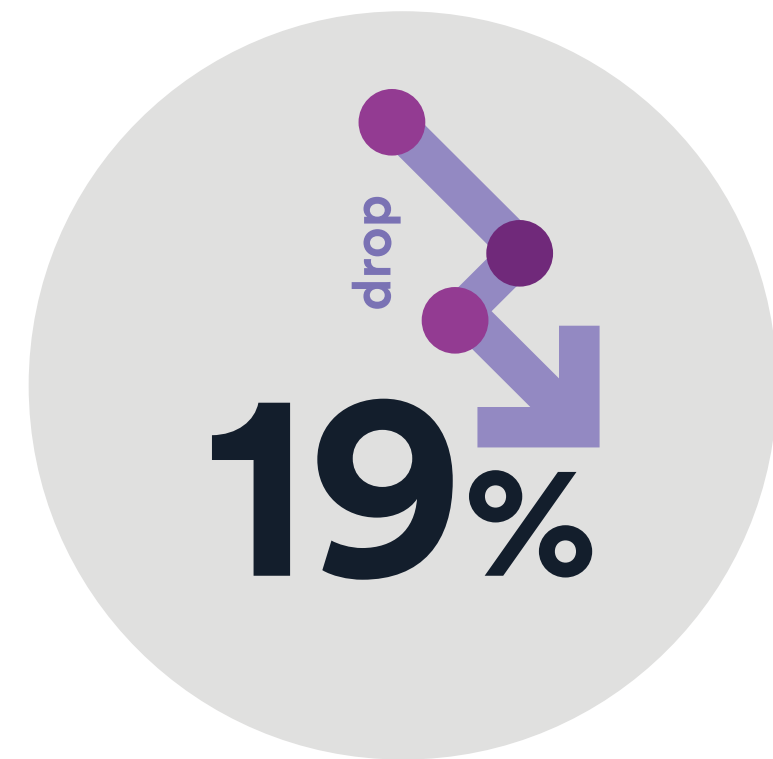


ENTERPRISE VS. SMB



As we dug deeper into the data on scan frequencies and findings, the trends uncovered something that we know deep in our very DNA: the more you scan, the more you uncover – and the more you fix, the fewer vulnerabilities you find. Our customers are scanning more applications and APIs more frequently than ever before, and the results are encouraging.

+ With a 50% increase in scans since 2019 and a 13% increase from 2021 to 2022, **organizations are now running an average of 73 scans per month** – up from an average of 49 scans per month in 2019. When we break it down even further, that 2019–2022 time period translates into a 41% increase in scans for our Enterprise customers and a massive 83% increase in scans for SMBs.

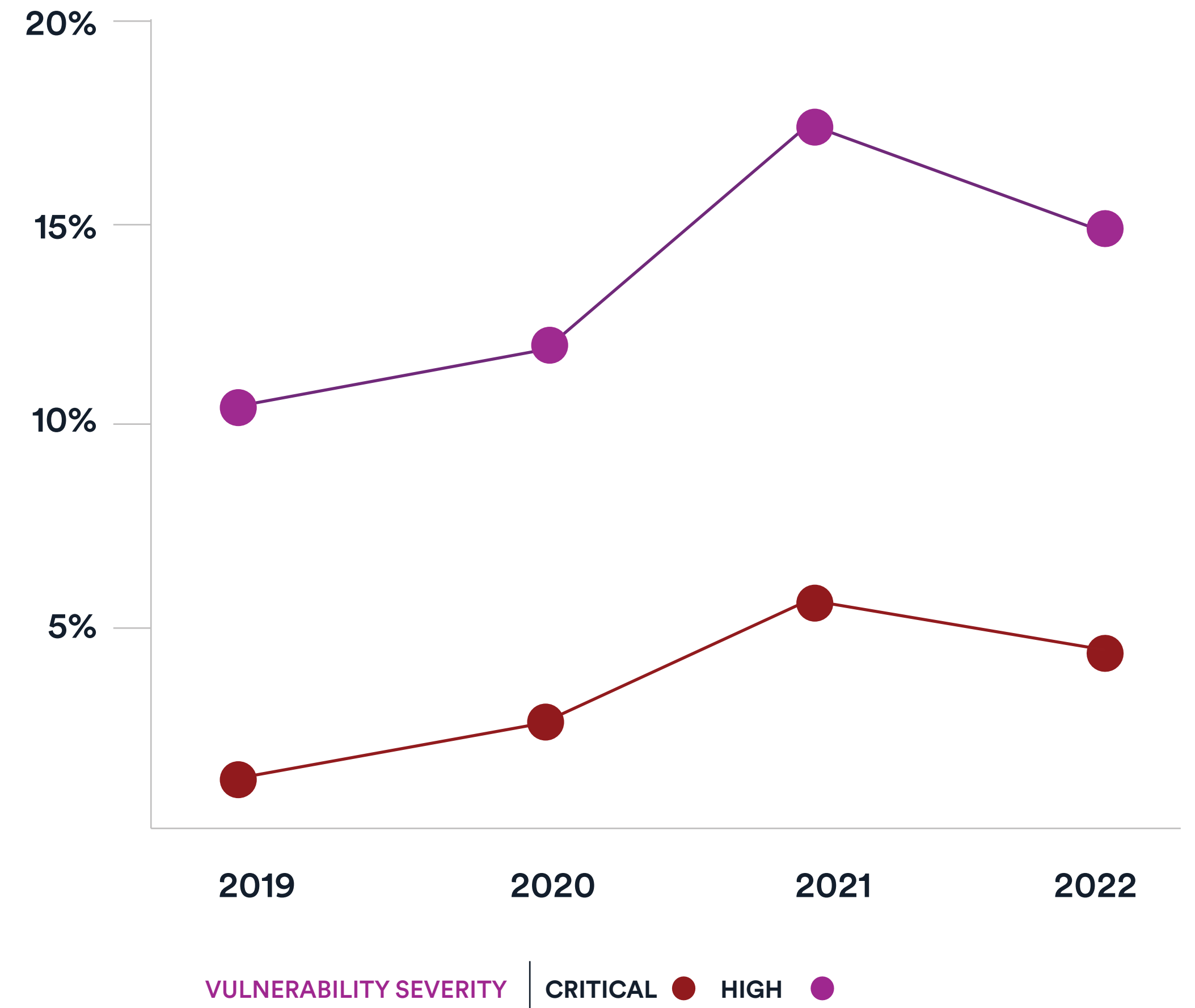


Frequent scanning leads to a reduced percentage of scans with severe vulnerabilities, **which dropped 19% from 2021 to 2022**

While the total number of severe vulnerabilities (critical and high severity) has been increasing yearly, the average percentage of severe vulnerabilities per scan has decreased 19% overall, from 11.8% in 2021 to 9.6% in 2022. After steady increases in severe vulnerability rates found per scan from 2019 to 2021, critical vulnerabilities declined 28% and high severity vulnerabilities decline 17% .

The data suggests that investments in maturing application security programs with DAST scanning throughout the SDLC are paying dividends. Scan frequency is increasing, so the percentage of scans with severe vulnerabilities is decreasing, thereby reducing the risk exposure for the organization. As the pace of software delivery has continued to increase, AppSec programs have adopted a shift-left and shift-right strategy using DAST to help mitigate the risk of a data breach in a continuous process.

AVERAGE PERCENTAGE OF SCANS WITH A VULNERABILITY

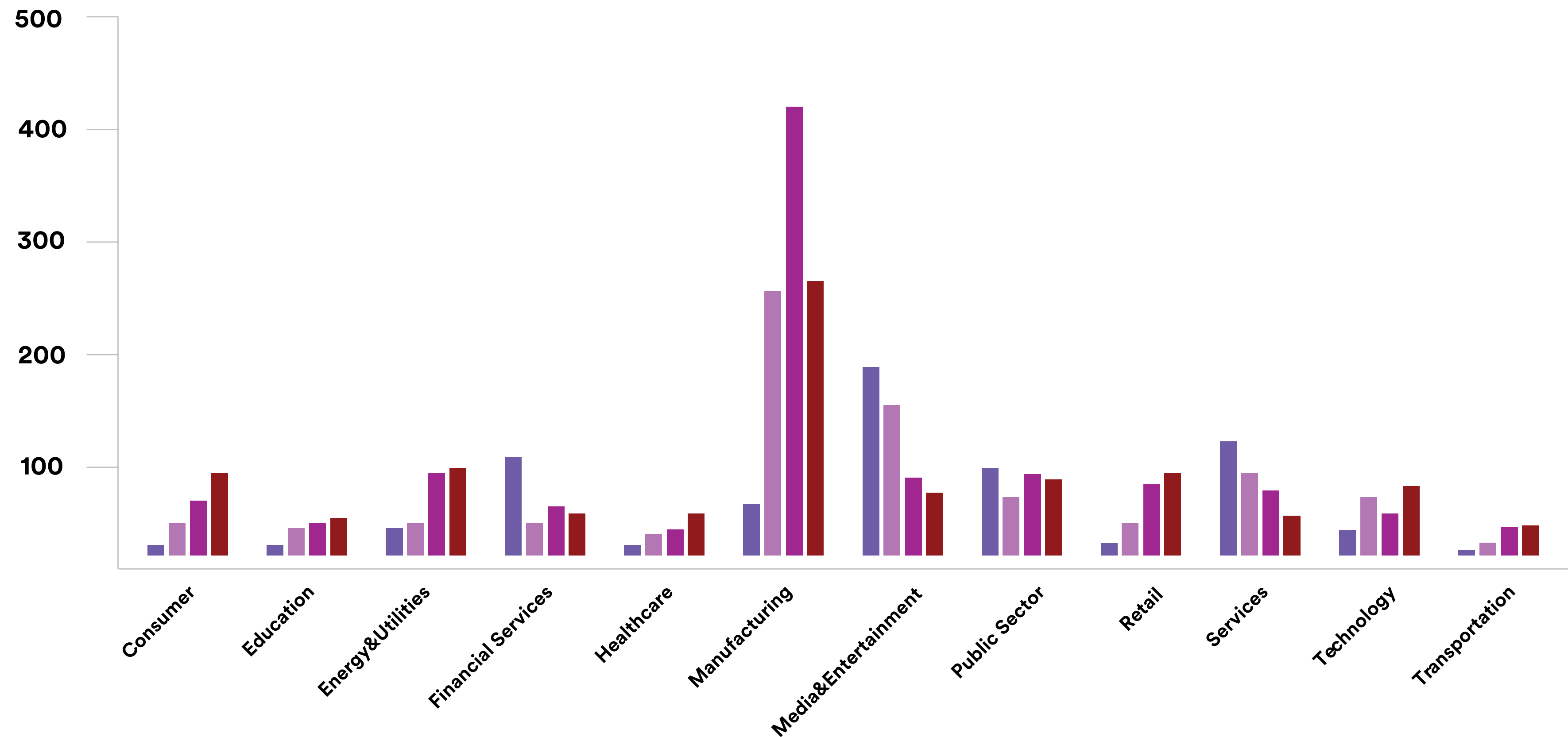


KEY TRENDS FOR SCANS

Automating DAST into the software development lifecycle with integrations into developer workflows is the path to keeping pace with software delivery. Approximately 50% of the Enterprise customer segment uses at least one integration, with issue trackers and CI/CD pipelines being the most common ones used. As customers expand scanning just from business critical applications to scanning their entire attack surface, automated scanning and remediation workflow triggered from the tools developers already use help customers scale their AppSec programs.

The data also showed a particularly large jump in integrations with CI/CD tools like GitHub and Azure DevOps, which increased by 512% and 278% respectively from 2020 to 2021. This shows that customers are shifting left with DAST and increasing their scan cadence overall.

AVERAGE NUMBER OF SCANS PER ACCOUNT BY INDUSTRY



“Our goal is to provide an environment where our products are safe-by-design. That means having our DevSecOps team focus on a shift-left approach where we use tooling to help fill the gap of security experts, while also involving the human element for efficient and precise triaging.”

– DRAGAN ILIEVSKI, Principal DevSecOps Engineer, Allocate Software

➔ [Read our case study with Allocate Software to learn how they shifted security left and right with Invicti](#)

Manufacturing leads a set of industries in growing scanning frequency.

Positive trends are emerging as many industries are showing increased average scanning rates per customer. The chart on the previous page shows average scans per customer across industry sectors.

Industries with a significant increase in scans from 2021 to 2022 include the Consumer, Healthcare, and Technology sectors. With the introduction of new data privacy regulations worldwide, these sectors have had to mature their application security programs, and many of our customers in these sectors are scanning their web applications and APIs in development and production. Of more concern are industries where scanning rates in 2022 were relatively flat from 2021, including the

Education, Retail, Financial Services, Public, and Transportation sectors, as well as sectors where scanning per customer actually declined, such as the Services sector.

Digging into the data, there was a sharp increase in scanning across all industries in Q4 2021 due to vulnerabilities in Log4j discovered in November 2021. In some cases, the increase in scanning activity in November and December greatly impacted the annual number, making the 2022 scanning activity appear artificially low by comparison.

While there are many factors that influence these numbers, they are relatively in line with industry trends.

For example, the Education sector continues to underinvest in cybersecurity. In November 2022, the Center for Internet Security found that the average school spends less than 1% of their IT budget on cybersecurity.⁴ Similarly, the Public sector also has underinvested in cybersecurity, but we hope to see that change in the U.S. in 2023 with the FY23 Omnibus Appropriations Package that allocates over \$2.9 billion for cybersecurity efforts under the Cybersecurity and Infrastructure Security Agency (CISA), which is a 12% increase from the FY22 budget request.⁵

⁴ Center for Internet Security, *K-12 Report*

⁵ Senate Appropriations Committee, *FY23 Omnibus Appropriations Package Topline Summary*

The standout industry in this year's data is Manufacturing, which is outpacing all other industries in our data set, in the range of 3x more scanning. The manufacturing sector spends less on cybersecurity as a percentage of their IT budget than most other industries.⁶ However, we have found that some of our manufacturing customers are actually quite mature in their application security practices. As the manufacturing industry has become more digitized and connected, it has become subject to more cybersecurity threats from financially and politically motivated hackers. The industry faces several regulations with regards to cybersecurity aimed to ensure

the protection of sensitive data and critical infrastructure. Some of the key regulations and industry standards include: Industrial Control Systems Cyber Emergency Response Team (ICS-CERT), NIST Cybersecurity Framework, and GDPR.

If we look at the customers that are scanning their applications and APIs most frequently and remediating vulnerabilities quickly, their behavior generally has less to do with their industry and more to do with the maturity of the application security team and of their engineering process.

⁶ VentureBeat, *Benchmarking Your Cybersecurity Budget in 2023*

Organizations that most effectively mitigate risk generally share a few common characteristics:

A security culture enforced from the top down and shared across the company.

Strong collaboration between security and development teams. Security champions are often a good way to infuse security knowledge into development teams.

Mature software development process with application security tools integrated into development pipeline and ticketing systems so that security is automated.

Severe vulnerabilities show improvement overall, driven by decreases in XSS, while offenders like RCE and SQL injection show clear increases.

When we delved into some of the regular offenders from our database of vulnerabilities, a handful bubbled to the surface with alarming increases year over year. And while some of these vulnerabilities are more severe than others, even the least critical flaws can lead to more serious breaches and incidents where attackers take a low-and-slow approach while inside sensitive systems. Fortunately, each of these vulnerabilities have quick fixes and best practices for prevention, which you can read in detail on *Invicti Learn*.

	SEVERITY	PREVALENCE	SCOPE	TECHNICAL IMPACT	WORST-CASE CONSEQUENCES	QUICK FIX
OS Command Injection	LFI	VERY SEVERE	Discovered rarely	Appears only in web-related software	Access to sensitive information	Remote code execution Do not use filenames from user input
		VERY SEVERE	Discovered rarely	May appear in all computer software	Command shell access	Full system compromise Do not call OS functions based on user input
	RCE	VERY SEVERE	Discovered regularly	May appear in all computer software	Execute arbitrary code	Full system compromise Do not evaluate code based on user input
	SQLi Injection	VERY SEVERE	Discovered often	May appear in software that uses SQL	Access to the database or system information	Full system compromise Use prepared statements also known as parameterized queries
	SSRF	SEVERE	Discovered regularly	May appear in all networked software	Access to privileged resources	Full system compromise Sanitize user data in calls to other servers
	XSS	SEVERE	Discovered very often	Websites and web applications	Malicious code run in the browser	Full system compromise Use user input filtration and encoding

Remote code execution severe vulnerability rates increased 40% from 2021 to 2022, and SQL injection severe vulnerability rates jump 91%.

While remote code execution and SQL injection increased, overall we finally saw notable declines in vulnerabilities from 2021 to 2022. Looking across all severity levels, across Critical, High and Medium severity vulnerabilities we saw declines, with Low severity up marginally.

While we see some increases in specific severe vulnerabilities, such as RCE and SQL injection, there was a lower overall prevalence of RCE and SQL injection, allowing for a decline in severe vulnerabilities overall. The improvement indicates more customers are taking action and increasing adoption of DAST tooling into the software development pipeline, improving the effectiveness and outcomes of AppSec programs. Typically, Enterprise customers are more advanced in their AppSec programs. We see this play out, for example, in SQL injection vulnerabilities which increased 24% year over year for our Enterprise customers; conversely SMB customers saw a dramatic 171% increase from 2021 to 2022. This suggests that advancing the adoption of DAST tooling results in improved application security and reduced risk.

SEVERE VULNERABILITY TRENDS



CROSS-SITE SCRIPTING (XSS)

Prevalence decreased **12%** from 2021 to 2022



REMOTE CODE EXECUTION (RCE)

Prevalence increased **40%** from 2021 to 2022



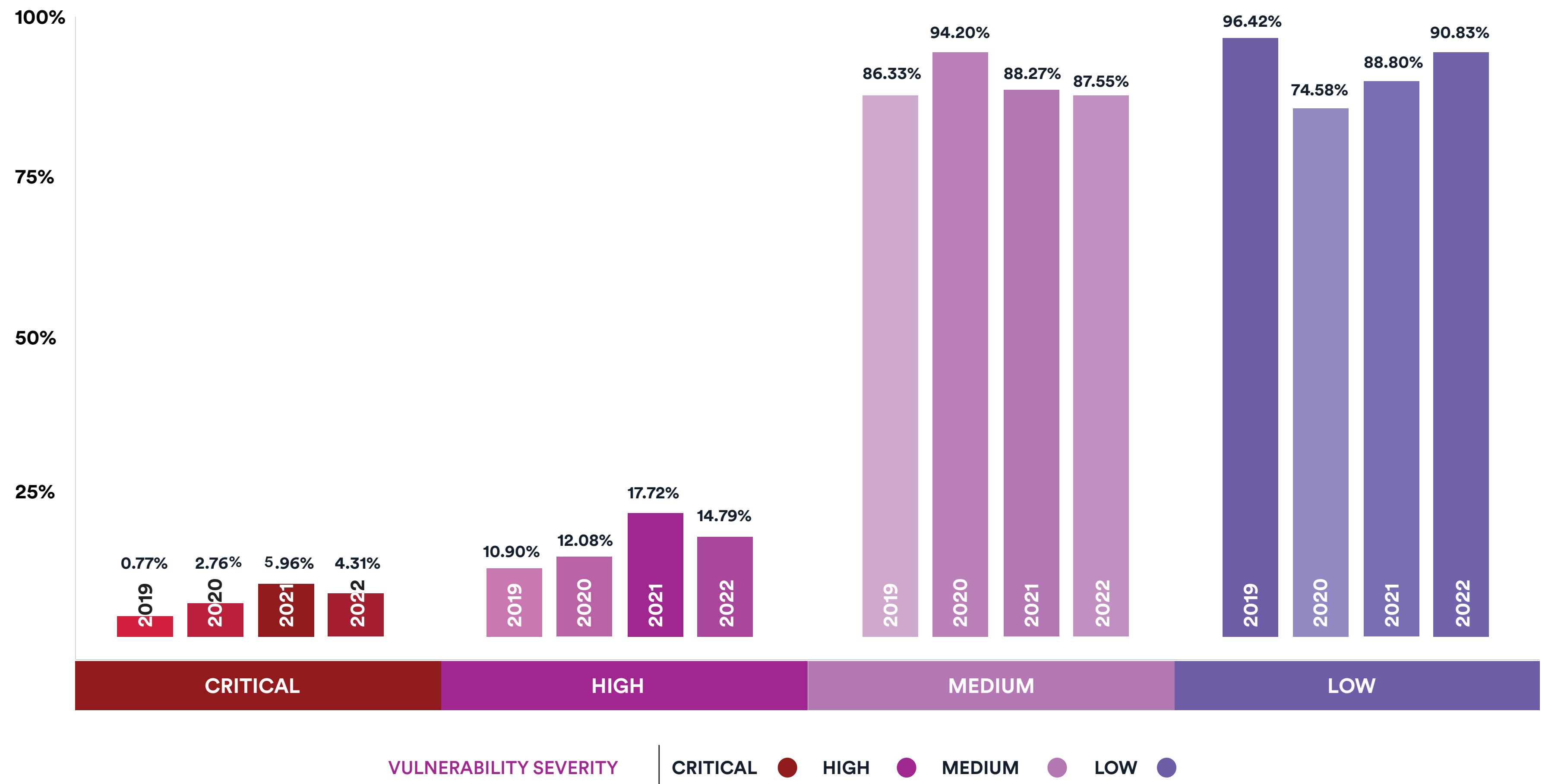
SQL INJECTION (SQLi)

Prevalence increased **91%** from 2021 to 2022

Looking back a year, we saw a clear spike in critical and high vulnerabilities found in **the last quarter of 2021 and first quarter of 2022**, corresponding to organizations intensely scanning for Log4Shell (CVE-2021-44228) and then settling down after the crisis was contained.

The scan spike corresponding to Log4Shell illustrates how mature AppSec programs that include DAST for regular vulnerability scanning are valuable in zero-day or crisis situations that require a need to quickly and automatically scan thousands of assets for a high-risk and remotely exploitable vulnerability.

AVERAGE PERCENTAGE OF SCANS WITH A VULNERABILITY

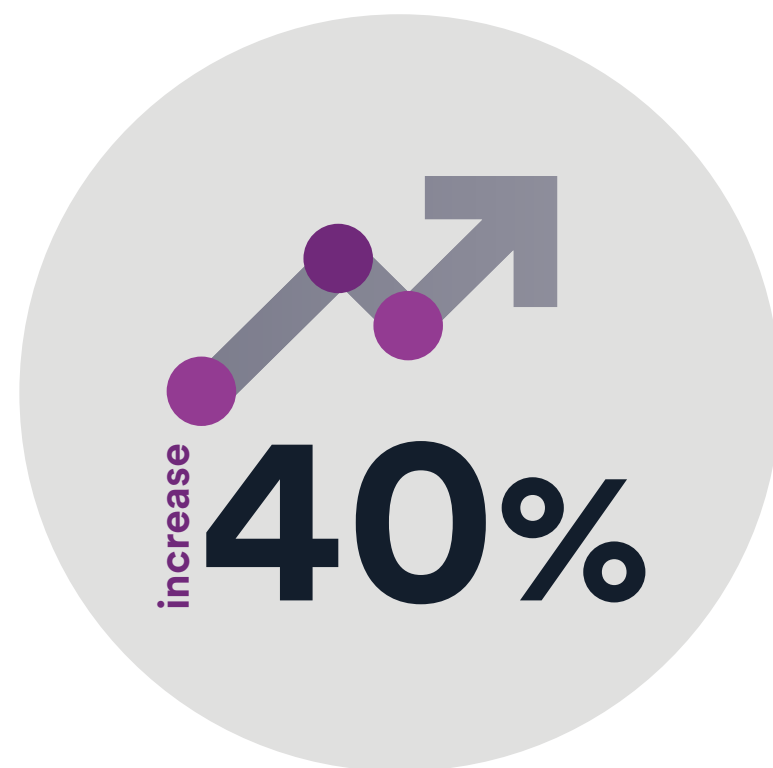




VULNERABILITY DEEP DIVE

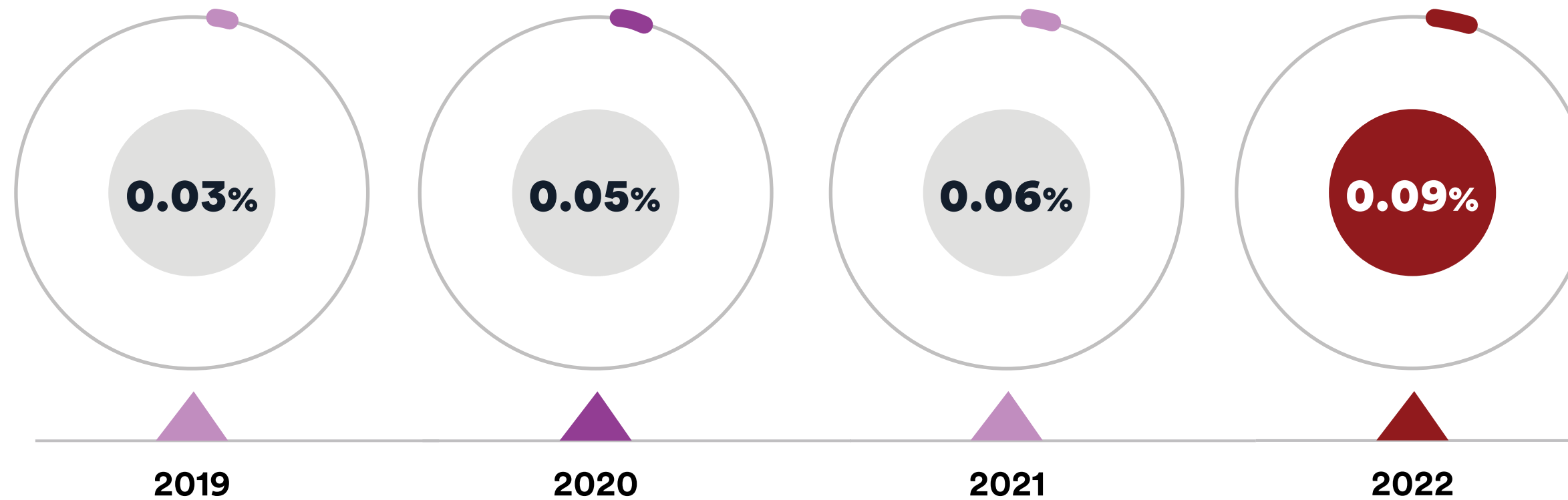
REMOTE CODE EXECUTION (RCE)

SEVERITY		Very severe
PREVALENCE		Discovered regularly
SCOPE		May appear in all computer software
TECHNICAL IMPACT		Command shell access
WORST-CASE CONSEQUENCES		Full system compromise
QUICK FIX		Do not evaluate code based on user input



The Log4Shell incident in Apache Log4j 2.x is an example of an RCE flaw. It impacted multiple Log4j versions and opened the door for threat actors to install ransomware, move between servers, and steal sensitive data.

AVERAGE PERCENTAGE OF SCANS WITH A SEVERE VULNERABILITY



Remote code execution rate increased **40% from 2021 to 2022, steadily increasing from 2019 to 2022.**

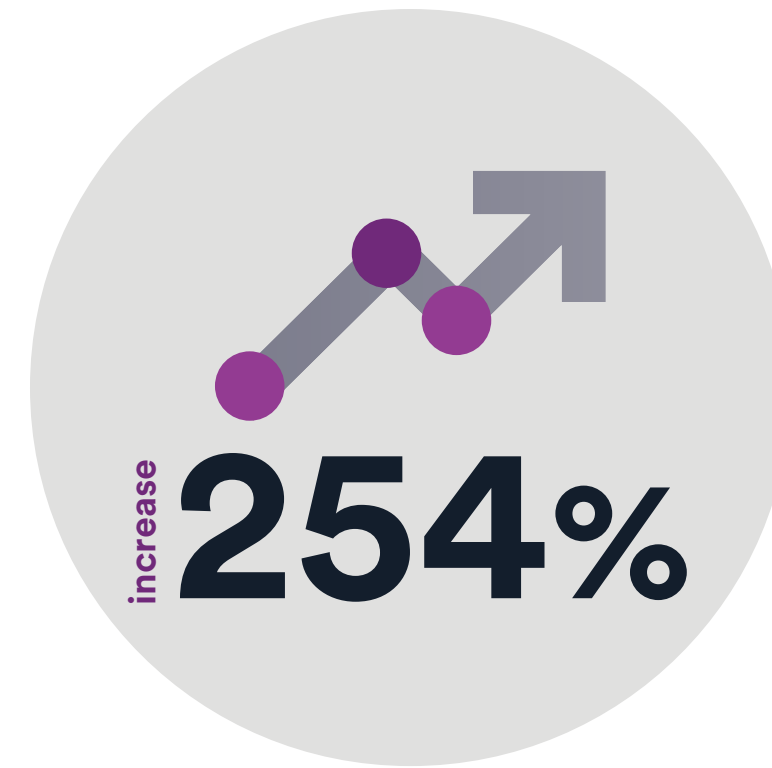
While at low prevalence overall, the values represent average prevalence by month and clearly show an increasing trend. The Log4Shell RCE vulnerability raises awareness of not just the software supply chain but also the severe consequences of unattended weaknesses in your

digital environments. By deploying applications across multiple containerized components, organizations can easily multiply their potential attack surfaces, which may increase the number of individual vulnerabilities discovered.

[LEARN MORE →](#)

Remote code execution is a vulnerability that allows an attacker to execute arbitrary code in the programming language in which the developer wrote the application. The term “remote” indicates that the attacker can do so from a location different from the system running the application. Additionally, RCE is sometimes referred to as code injection and remote code evaluation. Learn more about RCE and how to prevent it.

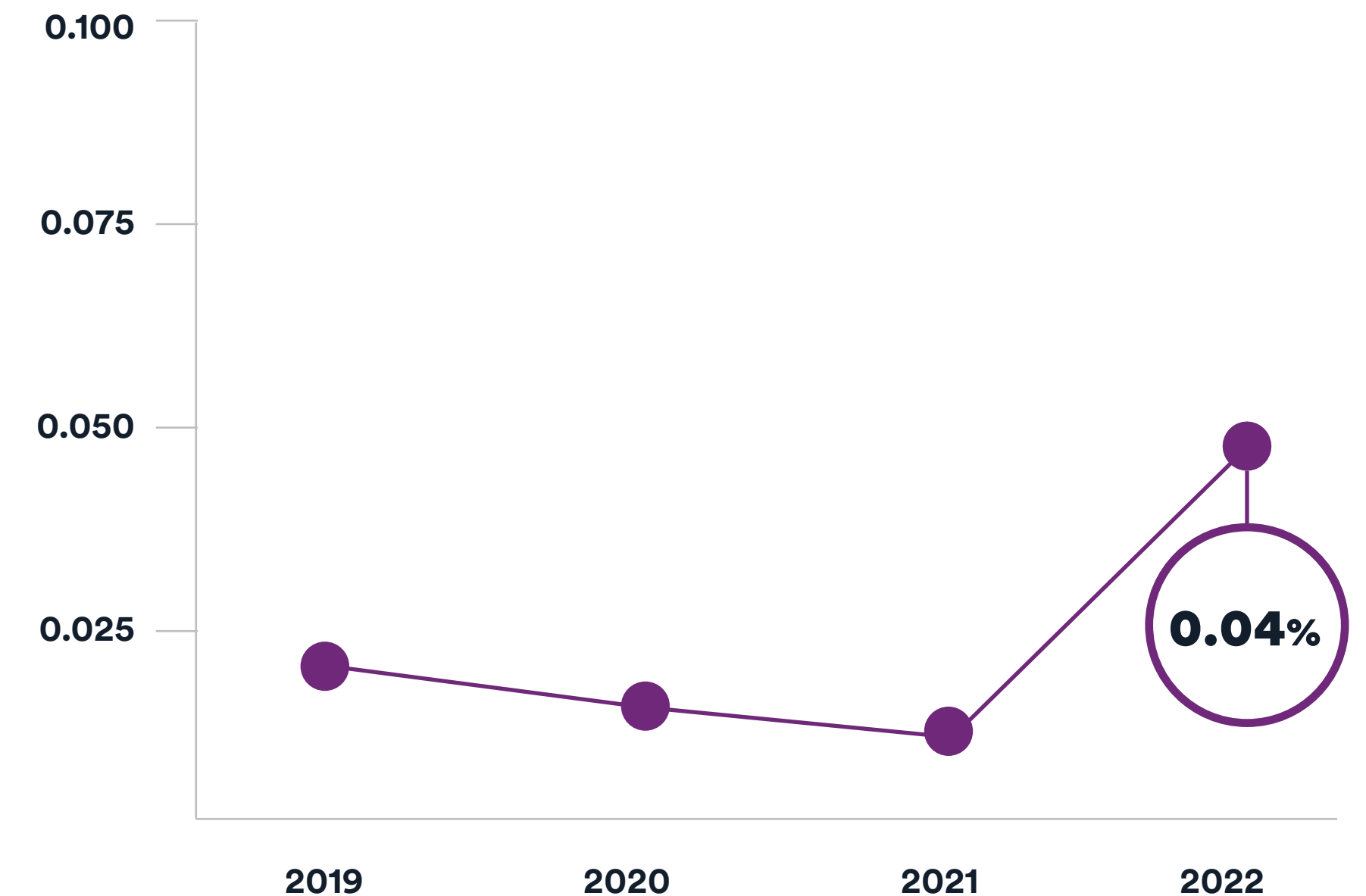
SEVERITY		Severe
PREVALENCE		Discovered rarely
SCOPE		May appear in all computer software
TECHNICAL IMPACT		Command shell access
WORST-CASE CONSEQUENCES		Full system compromise
QUICK FIX		Do not call OS functions based user input



OS command injection jumped **254% between 2021 to 2022**, from steady lower levels in prior years.

With virtualized and containerized web application deployments, the number of individual targets for OS command injection is on the rise – especially when compared to monolithic applications running on a single server. At the same time, successful exploitation is no longer synonymous with full system compromise as commands are more likely to be injected into an isolated and limited environment.

AVERAGE PERCENTAGE OF SCANS WITH A SEVERE VULNERABILITY



Shellshock is a famous vulnerability from 2014 in which attackers injected OS commands into web application software. Threat actors were able to exploit this flaw by creating computer botnets which then initiated millions of DDoS attacks across the globe.

LEARN MORE →

Sometimes also referred to as command injection or shell injection, OS command injection allows an attacker to trick an application into executing operating system (OS) commands. When using operating system call functions with insufficient input validation, it can lead to OS command injection. Without validation, a threat actor may inject malicious commands into user input and execute them on the host operating system. Learn more about OS command injection and how to prevent it.

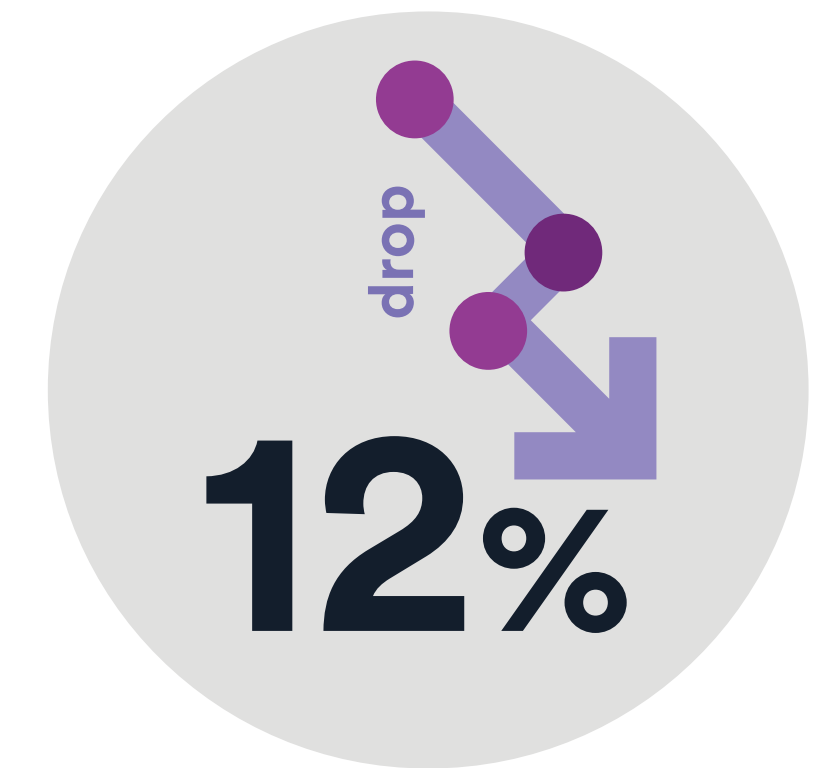
CROSS-SITE SCRIPTING (XSS)

SEVERITY	●●●●	Severe
PREVALENCE	●●●●●	Discovered very often
SCOPE	●●●●	Web sites and web applications
TECHNICAL IMPACT		Malicious code run in the browser
WORST-CASE CONSEQUENCES		Full system compromise
QUICK FIX		Use user input filtration and encoding

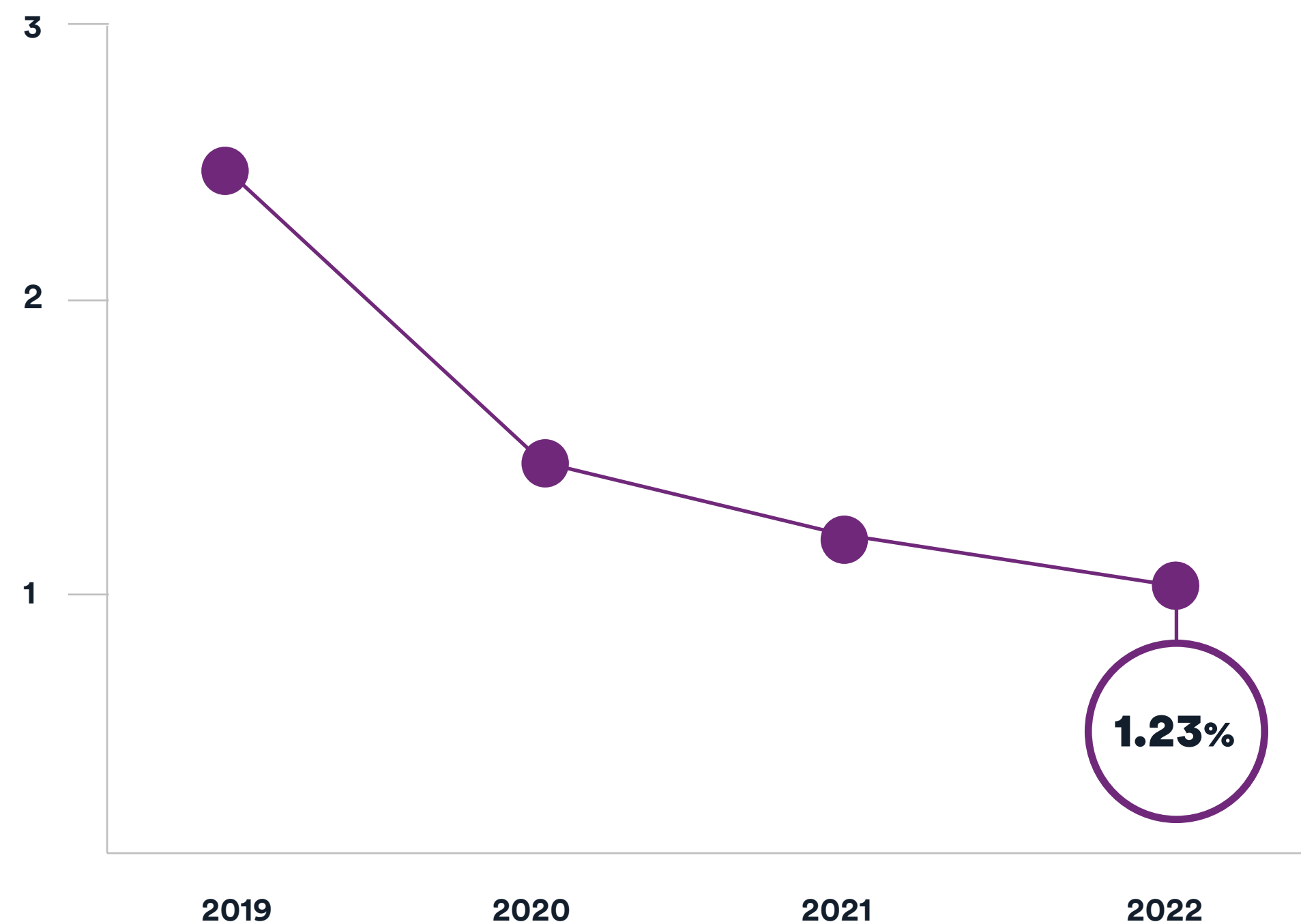
Because modern web applications rely on JavaScript not only for dynamic functionality but often as the primary development technology, more attack surface is exposed for XSS attempted attacks. Although many web frameworks now include constructs to prevent such script injections, the number of ways and places to inject JavaScript continues to grow, which makes sanitization crucial for all potential inputs – not just the expected ones.

●● A major European airline was impacted by a cross-site scripting flaw in 2018 that exposed 380,000 booking transactions. After investigating, it was suspected by security researchers that the attack on the airline had ties to Magecart, a group of hackers that uses malicious code injections and skimming techniques in their exploit attempts.

Encouraging news as cross-site scripting continues to decline, dropping 12% from 2021 to 2022.



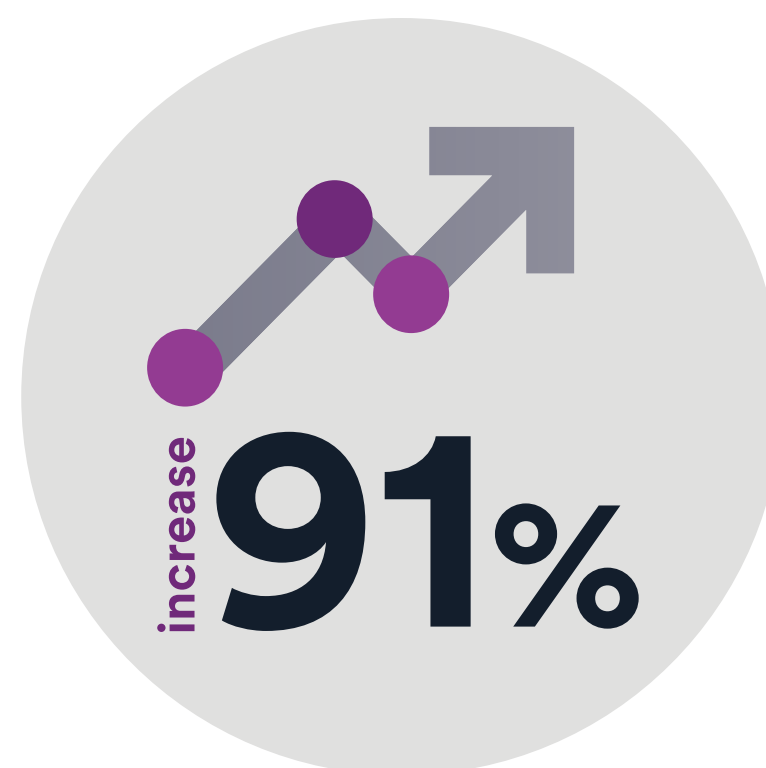
AVERAGE PERCENTAGE OF SCANS WITH A SEVERE VULNERABILITY



[LEARN MORE →](#)

Cross-site scripting allows threat actors to inject illegitimate commands into legitimate client-side code where it is then executed by a browser for the target application. There are two very common techniques used in XSS; reflected XSS (non-persistent XSS) and stored XSS (persistent XSS). DOM-based XSS and blind XSS are less common but can also occur. The best way to avoid XSS is through validation and sanitization of user input. Learn more about XSS and how to prevent it.

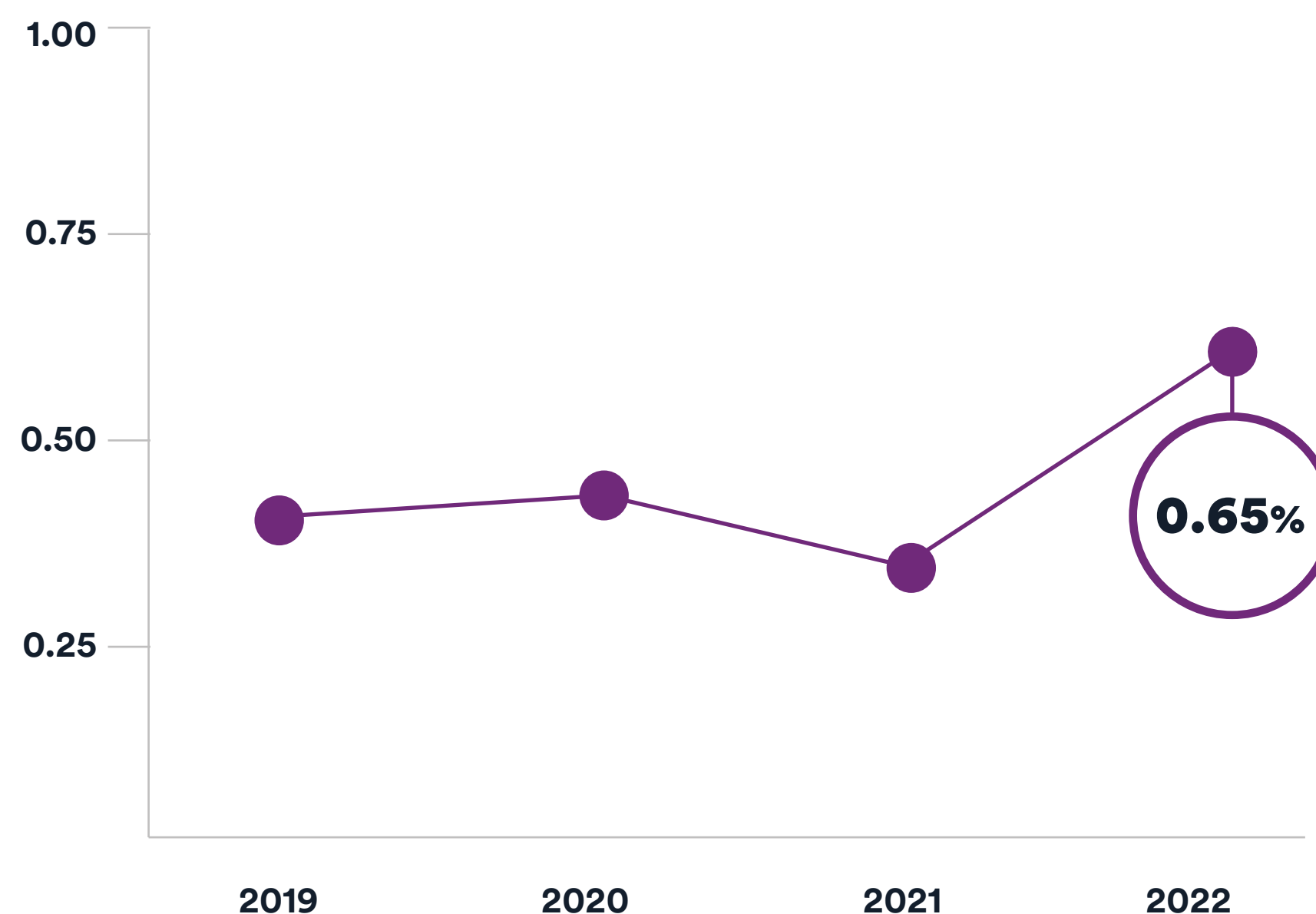
SEVERITY		Very severe
PREVALENCE		Discovered often
SCOPE		May appear in software that uses SQL
TECHNICAL IMPACT		Access to the database or system information
WORST-CASE CONSEQUENCES		Full system compromise
QUICK FIX		Use prepared statements also known as parameterized queries



Notable jump in SQL injection of 91% from 2021 to 2022.

Despite the rapid growth of special-purpose, non-relational databases, traditional SQL databases are the mainstay of web technology stacks and will remain a target for threat actors. Even though SQLi is one of the oldest web vulnerabilities caused by failures to sanitize input and is extremely easy to prevent, we theorize that it'll be an issue for years to come, even as numbers slowly decrease with increased scanning.

AVERAGE PERCENTAGE OF SCANS WITH A SEVERE VULNERABILITY



Back in 2019, tax information for millions of residents of an Eastern European country was stolen as the result of a SQLi flaw. Threat actors exposed records on underground forums, which included income, full names, personal identification numbers (EGN), and more.

[LEARN MORE →](#)

SQL injection is a vulnerability that lets a malicious hacker inject undesired SQL code into SQL queries that are then executed by the software. Such injection attacks occur when an attacker manipulates user input, and the user input is then implemented to directly construct queries sent to SQL databases. Then, attackers can potentially send additional or changed commands to databases and, depending on the scope of the vulnerability, obtain information from the database or modify it entirely. Learn more about SQLi and how to prevent it.

SERVER-SIDE REQUEST FORGERY (SSRF)

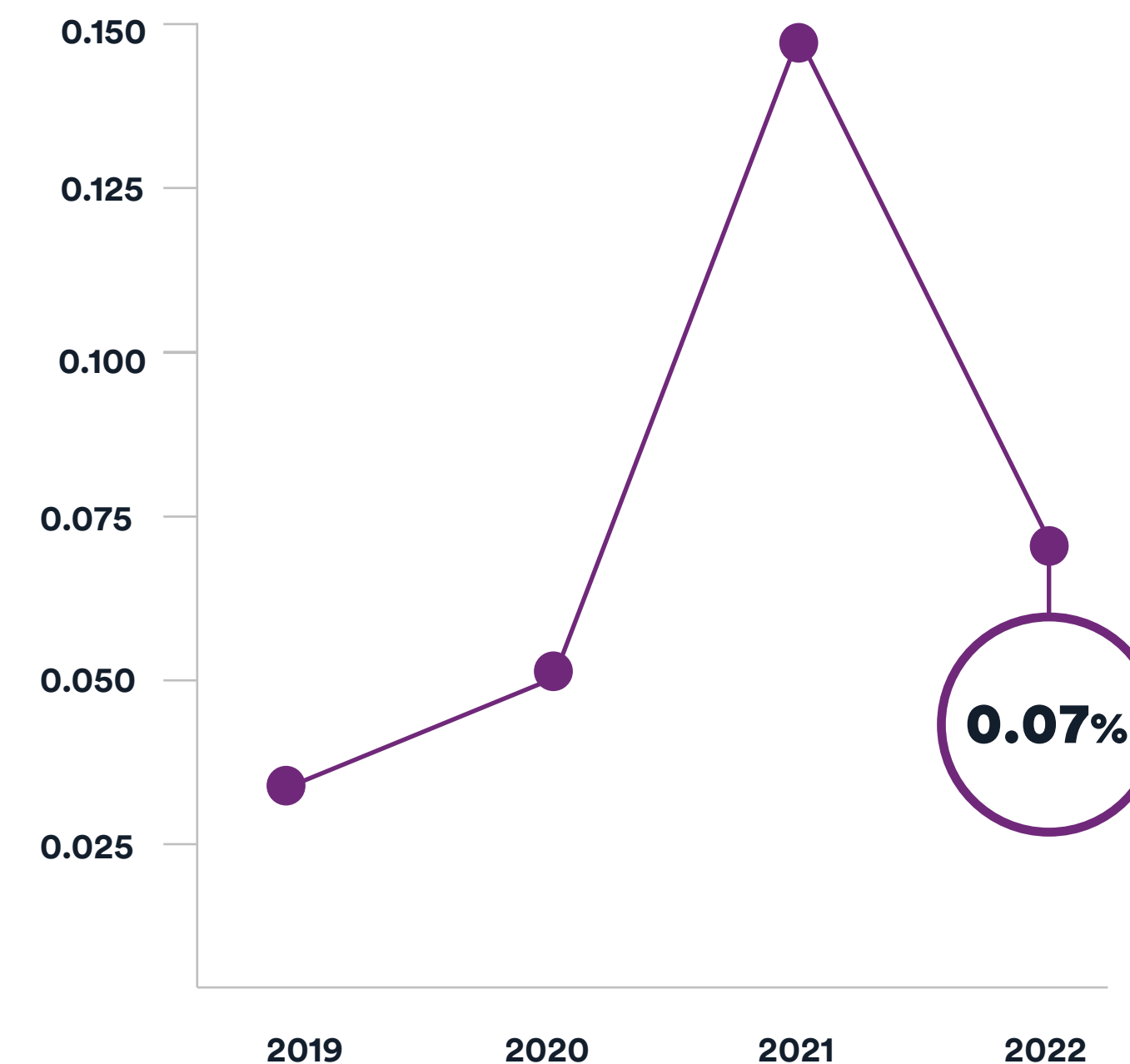
SEVERITY		Severe
PREVALENCE		Discovered regularly
SCOPE		May appear in all networked software
TECHNICAL IMPACT		Access to privileged resources
WORST-CASE CONSEQUENCES		Full system compromise
QUICK FIX		Sanitize user data in calls to other servers



Server-side request forgery is back down to 2020 levels after a **55% decline from 2021 to 2022.**

However, we expect them to stick around: widespread reliance on web APIs for communication with the backend means that a successful SSRF attack may provide attackers with access to internal APIs that serve up valuable assets. Apart from accessing sensitive data, bad actors can (and do) use SSRF to obtain cloud credits or spin up cloud service instances, for example for cryptocurrency mining. This evergreen vulnerability class will likely grow in importance and frequency as web applications become ever-more opaque.

AVERAGE PERCENTAGE OF SCANS WITH A SEVERE VULNERABILITY



[LEARN MORE →](#)

SSRF is a vulnerability that allows attackers to send requests from the back end of the software to another server, or to a local service. When done successfully, the server or service receiving that request believes that it originated from the application and assumes it is legitimate. Learn more about SSRF and how to prevent it.

A misconfigured web application firewall was an entry point for threat actor Paige Thompson to acquire AWS access keys by exploiting SSRF in 2019. This gave Thompson access to data from about 106 million customers of a major credit card company across the United States and Canada.

LOCAL FILE INCLUSION (LFI)

SEVERITY		Severe
PREVALENCE		Discovered rarely
SCOPE		Appears only web-related software
TECHNICAL IMPACT		Access to sensitive information
WORST-CASE CONSEQUENCES		Remote code execution
QUICK FIX		Do not use filenames from user input

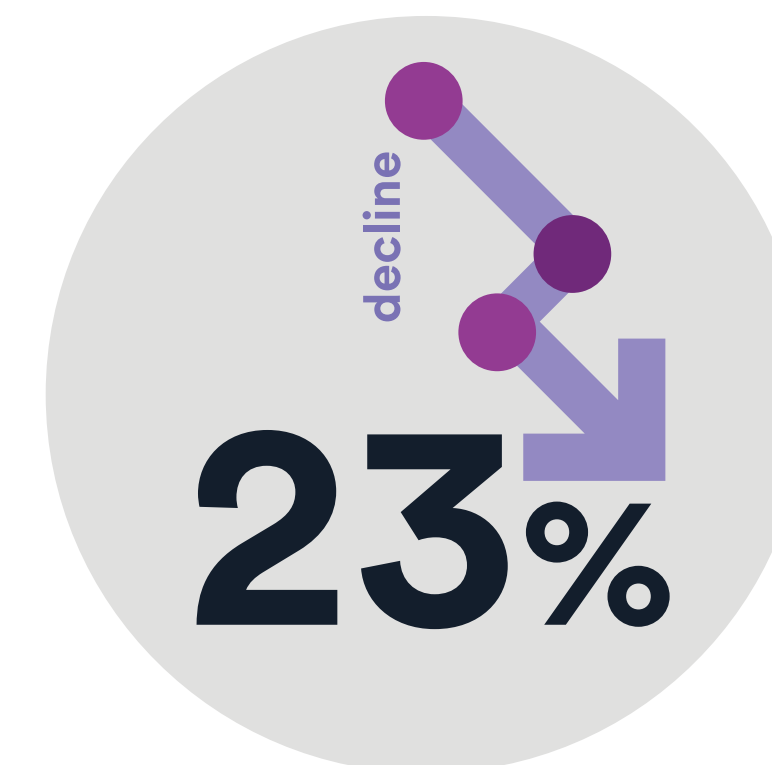
Local file inclusion rate declines **23%** from 2021 to 2022 but stays above levels in prior years.

The vast majority of LFI discovered by our vulnerability scanners involve PHP websites and applications. With the massive popularity of WordPress and other PHP-based web applications and plugins, LFI vulnerabilities are particularly a concern for smaller organizations relying on third-party products as they may not have the resources to ensure that those products are secure.

AVERAGE PERCENTAGE OF SCANS WITH A SEVERE VULNERABILITY



Looking back at 2016, an adult dating site experienced the repercussions of an LFI exploit. Attackers exploited this vulnerability to find information on over 400 million user accounts, which included passwords, confidential data about relationships, and emails.



[LEARN MORE →](#)

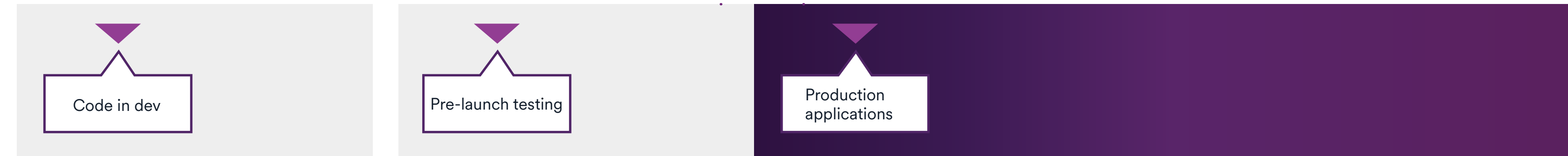
Local file inclusion is a web vulnerability that enables a malicious hacker to access, view, and include files located in the web server file system, within the document root folder. These vulnerabilities are exploited when a malicious user inserts an arbitrary file name or path within user input. Learn more about LFI and how to prevent it.



INSIGHTS: THE BENEFITS OF DAST

Shift left focuses security efforts here...

...and can leave production applications exposed



From brand damage to customer churn, the implications of a major security incident can last for years. The financial damages reach around \$4.35 million on average, stifling long term financial growth and even hurting employee retention as they seek greener (more security-focused) pastures.⁷

Getting and staying ahead of those breaches requires more than scanning your web applications and APIs once or twice a year. Because production applications make up the vast majority of an organization’s attack surface – and about 70% of security incidents are connected to the hacking of web apps– risk can grow with every new update.⁸ Modern web application development requires an approach that blends speed with accuracy, while also increasing frequency.

Historically, it hasn’t been easy for many organizations to increase their scanning frequencies. If organizations are using multiple methods and a handful of scanners, there is no single, unified view of all their vulnerabilities. They have to collect that information and filter it down manually, which is tedious. Accuracy is another roadblock – without knowing which results indicate a real vulnerability or how to go about fixing it quickly, it’s difficult to pivot and be proactive. An overall lack of resources attributes to this issue too, where budget constraints or legacy tools mean more errors and less accuracy.

Reliable DAST checks these boxes when built into the SDLC, providing more complete coverage for all applications in development and production with accurate results that cut down on manual

work. It’s fast and helps teams improve security over time by building security testing right into the development pipeline to find vulnerabilities before they reach production. Modern DAST solutions scan everything from single pages to multi-level forms using advanced crawling technology so that teams can find and fix vulnerabilities, fast.

Integrating DAST into development and operations translates into direct security improvements and fewer vulnerabilities in the long run.

⁷ IBM, *Cost of a Data Breach 2022*
⁸ Verizon, *2022 Data Breach Investigations Report*

Modern DAST tools with automation and accuracy built into their very design are the future of impactful AppSec. With accompanying technology like web asset discovery and vulnerability assessment aiding risk management, crafting a security strategy that actually delivers is achievable.



Unlike other testing types, you can run DAST scans at multiple stages of the SDLC, whether that means testing legacy apps that have already been deployed or apps in development pre-release with a CI/CD integration.



DAST works regardless of the language in which the application was written or the technology it runs on – if it communicates over HTTP, then DAST can scan it, and doesn't need access to the source code to get the job done.



The APIs that now form the backbone of web development and data access are often only testable using dynamic methods. Advanced DAST tools can scan a variety of API types, including REST, SOAP, and GraphQL endpoints.



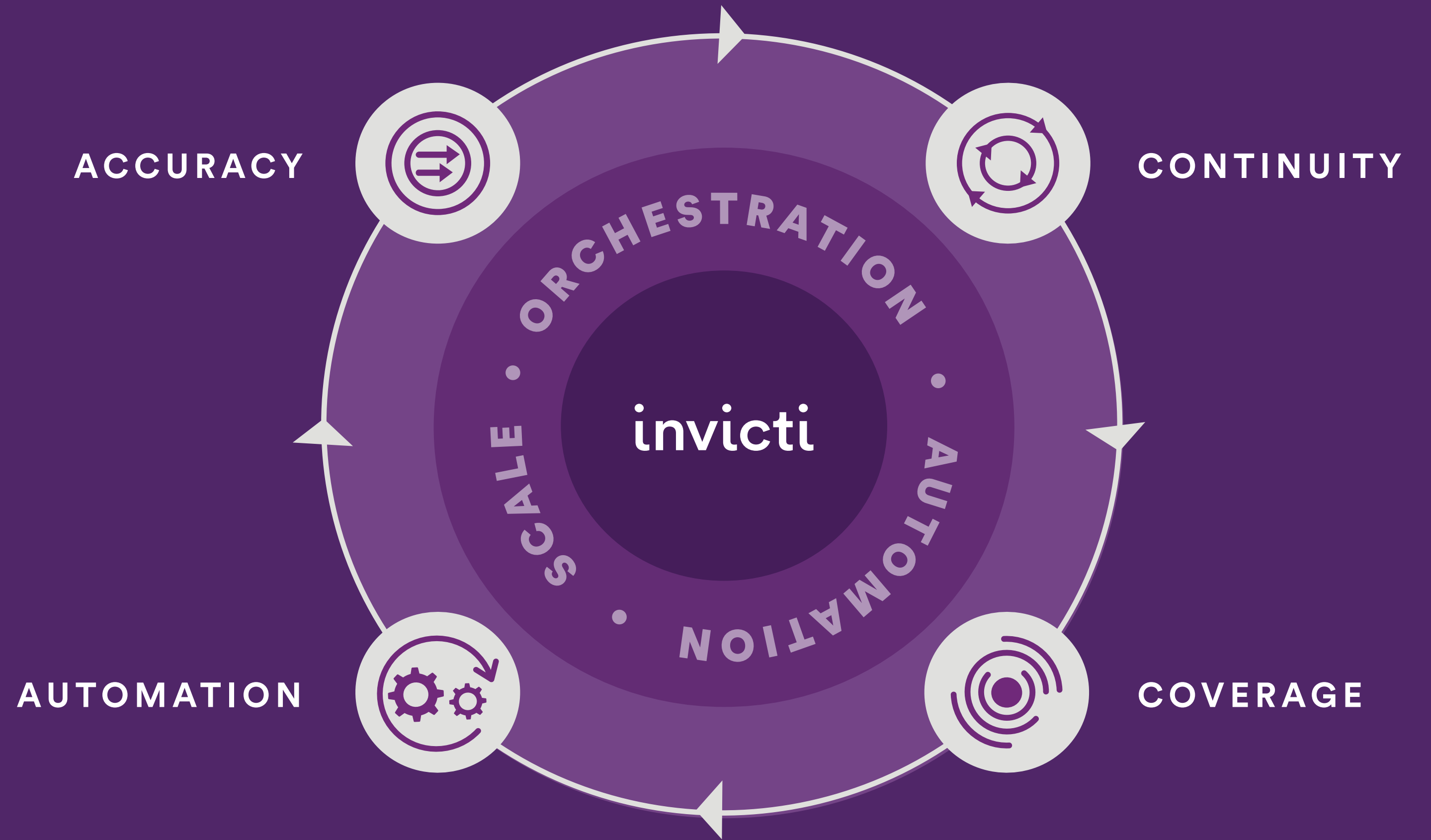
Simulating user actions for an outside-in view of how the application functions, DAST provides more accurate results than other tools (such as SAST and SCA) by finding vulnerabilities that are actually exploitable. Greater accuracy means less time wasted on chasing down false positives. Combined with IAST, Invicti's DAST also provides the location of the vulnerability, making remediation faster.



Accurate DAST results can be used to virtually patch by automatically creating WAF rules via its API for immediate mitigation.

Modern AppSec is built on **accuracy, continuity, coverage, and automation**. This checklist identifies critical elements of a successful AppSec program with actionable recommendations to get you there.

ACCURACY
CONTINUITY
COVERAGE
AUTOMATION



Web application security tools like Invicti – built with orchestration, automation, and scalability as foundational features – help you continuously shrink your threat landscape with less noise and more accuracy.



ACCURACY
CONTINUITY
COVERAGE
AUTOMATION

1 ACCURACY:

Feel confident in your data and the remediation guidance. Once test results are in hand, remediation begins. If teams don't feel confident in their scan results, it can slow everything down and even lead to missed security steps as developers race to meet tight deadlines.

- + Opt for scanning tools with features like proof-based scanning which provide validated results that DevSecOps teams don't need to re-verify.
- + Keep up with automated development toolchains and CI/CD pipelines with integrations that automatically deliver accurate feedback to developers.

2 CONTINUITY:

Scan consistently to keep up with unpredictable changes.

Applications are created and modified daily, and so what is safe today may not be safe tomorrow. To maintain complete coverage that protects against these security surprises, testing and remediation need to keep up no matter what happens.

- + Regularly scan applications and APIs in development and production, using automation to manage with frequent scan schedules.
- + Scan at every stage of the application lifecycle with regularly updated security checks to minimize risk and keep up with dynamic attacks.

3 COVERAGE:

Find everything, and test everything.

Teams can't fix what they don't know about. Knowing your entire attack surface and finding assets that have been lost, forgotten, or unauthorized is a first step to efficient coverage. With a full view of the entire application, it's easier to spot areas where attackers may infiltrate sensitive systems.

- + Incorporate dynamic application security testing (DAST) leveraging interactive application security testing (IAST) and software composition analysis (SCA) for a deeper view of the entire application including first and third-party code.
- + Select a security tool with features like web asset discovery to find lost, hidden, or unknown assets that attackers might exploit.

4 AUTOMATION:

Fine-tune your strategy to test and remediate faster. Manual processing of security issues, especially by short-staffed teams, is often a major bottleneck to quick development and releases. Security tools that integrate time-saving features ensure that tedious manual tasks happen quickly, accurately, and reliably so that teams can focus on innovation.

- + Embed your AppSec program with DAST deeply into the software development lifecycle (SDLC) and give teams the tools they need to release secure apps on schedule.
- + Implement integrations within existing issue trackers so that developers can resolve issues as they would any other software bug.

Learn more →

Organizations are scanning a larger part of their attack surface, scanning applications and APIs in development and production, and scanning more frequently. The result is an improved security posture and lower risk, measured by the decline in the percentage of scans with critical vulnerabilities found.

Our data showed that between 2019 and 2022, **Enterprise organizations increased scanning by 41% and SMBs increased by 83%**, indicating that Invicti customers are increasing their scanning rates and likely maturing their application security programs. Through frequent, automated scanning through the SDLC – which enables more frequent identification and remediation of vulnerabilities – we have observed in the last year a decline in the percentage of scans conducted that find a critical or high severity vulnerability. These proactive security measures mean fewer opportunities for threat actors to disrupt your daily business.

While certain incidents like Log4Shell and compliance mandates from the United States government have led to some spikes in scan increases, the steady upward trends from 2019 to 2022 reveal that continuous security is now an indispensable feature of successful modern AppSec - and overall cybersecurity-programs.



To more effectively reduce risk from potential cyber incidents, organizations should leverage modern application security programs, that include:



Know your risk exposure. Discover and scan your entire attack surface



Shift your application security program left and right by frequently scanning applications and APIs in development and production



Automate scan and remediation workflow via integrations with development and security tools, such as CI/CD pipelines and ticketing systems. This will allow you to embed security into the software development process



Embrace a security culture enforced from the top down and shared across the company. Build strong collaboration between security and development teams



CONTINUOUS SECURITY

With **continuous security** baked into the very design of the development process, companies – especially Invicti customers – are able to find and fix more high-severity flaws faster than ever before and improve their overall security posture.

invicti

AppSec with Zero Noise

Invicti Security - which acquired and combined AppSec leaders Acunetix and Netsparker - is on a mission: application security with zero noise. An AppSec leader for more than 15 years, Invicti delivers continuous application security, designed to be reliable for security, practical for development and serve critical compliance requirements. Customers choose Invicti's DAST, SCA and IAST solution to better secure and ultimately reduce risk across their web applications and APIs. Invicti operates globally with employees in over 11 countries and serves more than 4,000 customer organizations. For more information, visit www.invicti.com or follow us on LinkedIn.

www.invicti.com [in](#) [t](#) [f](#) [i](#)