Checkmarx

Checklist

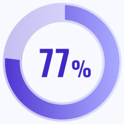
THE COMPLETE Enterprise Application Security Checklist

Printable Checklist
Code to Cloud Pipeline Coverage
Cloud-Based Platform
Single SaaS Cloud platform
Scalability Process
Integration for Correlation and Fix Prioritization
Correlation and Prioritiz
Solve for F
Reports, Dashboards, And Notifications

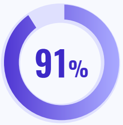
The Need for
Secure App Development

Cloud-native development is shifting everything about how enterprises think about applications: what they are made of; how they are packaged; and, how they are deployed. The result is an increase in developers and distributed teams "writing" billions of lines of code, across hundreds or thousands of apps and microservices. Amongst all this change, AppSec teams continue to carry the burden of ensuring that this is all done securely.

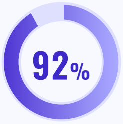
The stakes are high:
In a recent Checkmarx survey of 200 CISOs



of respondents said that **over half of their organization's revenue comes from applications that they're responsible for securing.**



Furthermore, 91% of organizations have **deployed known-vulnerable code in production** to meet business or feature deadlines;



and 92% of organizations experienced at least one breach in the last year **as a result of a vulnerable application they developed.**

AppSec practitioners have turned to a range of tools to support their programs in order to ensure the success of application security initiatives. Due to an increasing number of tools, AppSec vendors have begun to consolidate on platforms. The right AppSec platform will allow security teams to manage the challenges of continuous and secure development. If consolidating on an AppSec platform is in your plans, here is a checklist for what to keep in mind when evaluating different vendors.

The Challenges
of AppSec

Why are you looking for an AppSec platform?
Here are the reasons we see in the market:

Consolidation

Securing your enterprise's applications effectively is complex. AppSec teams that started with just SAST have moved on to owning and managing multiple Application Security Testing (AST) solutions that are all aimed at different areas of the software development lifecycle (SDLC). Ten years ago you likely had Static Application Security Testing (SAST) and a penetration testing contractor; now you probably have Software Composition Analysis (SCA), Dynamic Application Security Testing (DAST), API security, and much more.

Managing multiple vendors, products, logins, and billing cycles has led many AppSec practitioners to decide that it is time to consolidate on a platform. Many organizations look for an enterprise AppSec platform that will give the following benefits: reduced total cost of ownership (TCO), simplified vendor management, and a "single pane of glass" for AppSec solutions. However, this does not address all the issues around multiple products. We recommend that when looking for an AppSec platform, you ensure that it does more to help reduce your application risk.

Developer Experience

Application security is not just about finding vulnerabilities – it's about fixing them. These two critical roles—finding and fixing—are typically performed by two teams: AppSec and developers. Those teams don't always work well together and the reason why is typically rooted in trust. Do developers trust that the results they see are real? Are they business critical? Will they be supported in fixing them? These are the hurdles that CISOs and AppSec managers often face when persuading developers to remediate the vulnerabilities they find.

Time to Value

AppSec teams are increasingly asked to do more with less. In large enterprises, they are often outnumbered by developers 150:1. The need to get AppSec solutions onboarded, then tuned properly, all while seamlessly integrating into developer workflows to minimize impact on time to market is a massive challenge. With DevSecOps and continuous secure development as the goal – time to value is essential when considering AppSec tools.

Secure Your Applications
from Code to Cloud

In Application Security, the software development lifecycle (SDLC) is your attack surface. No matter where you are in your AppSec journey, the platform you purchase should come equipped with a full suite of application security tools. Your platform should also include technology to assist in the correlation and prioritization of results from different solutions.

Code to Cloud Pipeline Coverage

Make sure your platform has access to these tools,
and that they are fully integrated

Static Application Security Testing (SAST)

Conduct fast and accurate scans to identify risk in your custom code. Cover all your applications with both deep scans of critical apps and quick-and-wide scans for the rest of your footprint.

Software Composition Analysis (SCA)

Identify security and license risks in open source software used in your applications, while also allowing you to prioritize which vulnerabilities are actually exploitable.

Software Supply Chain Security (SSCS)

Proactively identify supply chain attacks and secure developer environments with open-source vulnerability and malicious package detection, SBOMs, and secrets detection in your code and development sources.

API Security

Discover every API in your application and compare the full API inventory against your API documentation to help you eliminate shadow and zombie APIs and mitigate API-specific risks.

Dynamic Application Security Testing (DAST)

Identify vulnerabilities only seen in running applications and assess their behavior by testing against a broad range of web application attacks.

Container Security

Scan static container images, check configurations, and determine what open source packages are called and identify vulnerabilities pre-production.

Infrastructure-as-Code (IaC) Security

Automatically scan your IaC files to find security vulnerabilities, compliance issues, and infrastructure misconfigurations, using thousands of predefined queries.

Runtime Security

Extend AppSec into production to correlate pre-production data with runtime insights to better prioritize which critical vulnerabilities to fix first.

AppSec Training for Developers

Train developers on secure coding practices with personalized learning paths and just-in-time training to reduce risk from the first line of code.

↘

Integration for Correlation and Prioritization

As more companies embrace a code-to-cloud mentality, they have continued to purchase AppSec tools that place security controls across the SDLC. But the increase in tools hasn't resulted in an increase in time or other resources. AppSec teams now have too many disparate security products, tools, and controls that they need to deploy and manage, and not enough staff or resources to manage them effectively.

An enterprise AppSec platform should enable the correlation of results across individual AppSec tools, so that you can easily prioritize remediation for the most important vulnerabilities and maximize your team's resources. Here are some things you should consider in a platform:

Correlation and Prioritization

Your AppSec platform should be able to correlate data across multiple tools, particularly the two most common AppSec solutions – SAST and SCA – and use the aggregate data to identify your riskiest apps. This will give you better visibility into your application security posture.

Solve for Exploitability

Application security solutions will deliver results. But how do you prioritize them? Your platform should evaluate vulnerabilities in open source libraries and analyze whether they are actually called by your application's code. If they are not called, they are not exploitable. Filtering these out will help you focus on vulnerabilities that impact your business.

Correlate Runtime Protection

40% of vulnerabilities found by AppSec teams happen once an application is in production. Ensure that your platform can deliver runtime insights to get a full picture of your applications in use. Your platform should connect the dots between pre-production and deployment, giving your team clear visibility into the vulnerabilities that exist in workloads running in production.

↘

Reports, Dashboards and Notifications

AppSec programs live and die by their access to information. The ability to visualize results, correlate findings, and prioritize results – across the entire SDLC from code to cloud - is essential to the functioning of a successful AppSec team.

Unified Dashboard

One of the primary benefits – and the easiest tests – of a true consolidated AppSec platform is whether you can see all the vulnerabilities discovered across different tools in the same place. Seeing all your vulnerabilities in one place makes your job of managing, analyzing, and triaging vulnerabilities easier.

Application Risk Management

Your AppSec platform should be able to take its identified list of riskiest apps and present it in an easy-to-read visualization. This visualization should both be easily shareable, and easy to work directly from.

Configurable Dashboard

Different users or roles within the same organization will want a different view into security data and metrics to make their job of managing, analyzing, or triaging security issues easier. Available dashboards should be configurable to meet the needs of all your users.

Export Data to PDF/CSV/JSON

Data should be easily downloadable, exportable, and consumable so it is useful across your organization, whether shared by humans or integrated into broader vulnerability management processes and tools

Recent Changes

It should be possible to compare historical scans and quickly derive the differences both at a high level (new, recurrent, and mitigated vulnerabilities), and in greater detail (location in code).

Executive Summaries and Dashboard

Your platform should be built to report to management, including the CISO and the Board. It should easily display application rating scores, mitigation and vulnerability trends, and historical data.

Notification Alert Support

Every team uses different tools to communicate, like email or Slack. Your AppSec platform should be able to send notification alerts through the same tools you already use to fit seamlessly into your daily routine.

Policy Management

An AppSec program is only as strong as its policy. Your AppSec platform should support your program with robust policy management capabilities and notify AppSec teams on significant events.

API Support

Your team may already have your own custom-built dashboards or vulnerability management programs. Your AppSec platform should support them with APIs to allow you to funnel data directly into your existing toolset.

↘

Analysis and Triage

Bulk Marking Results

A platform should make it easy to triage large scan volumes by bulk-marking results for simplified management.

Permanent Results Modification

Support for changes to the state or severity of individual (or bulk) results.

Vulnerability Assignment

Vulnerabilities should not just live in a general pool waiting for a developer to remediate. Your platform should allow the specific assignment of vulnerabilities to developers as appropriate

De-Duplication of Similar Findings

Avoid creating multiple tickets for the same or similar vulnerabilities via correlation of findings and triage status.

Data Flow Visualization

Display data flow (source to sink, across files) to offer optimal triage and remediation of findings.

Aggregate Vulnerability Graphs

Support visualization of aggregated vulnerabilities to show the same source and destination for optimal triage.

Collaborative Auditing

Security issues often require further analysis to understand, and many require deep understanding of the application at hand. This means that analyzing a potential vulnerability may require multiple stakeholders. An AppSec platform should support the ability for multiple users and developers to collaborate and comment on audit findings.

Enterprise-Grade Application Security

AppSec programs live and die by their access to information. The ability to visualize results, correlate findings, and prioritize results – across the entire SDLC from code to cloud - is essential to the functioning of a successful AppSec team.

Supports 1000s of Repositories

Large enterprises can have 1000s of developers and hundreds of development teams all working on different applications. An AppSec platform is supposed to enable a consistent AppSec posture across your entire application footprint. This means it needs to provide the ability to integrate with and scan 1000s of applications and/or software repositories.

Supports 5 Million of Lines of Code

Enterprise applications are larger and more complex. While there are many ways to measure scale and complexity, an easy metric is how many millions of lines of code it can cover. Your AppSec platform should be able to scan individual codebases exceeding 5 million lines of code.

Incremental Scans

Scanning large codebases can take a lot of time. However, most code isn't changed between scans. To minimize the impact of security on your development teams, your platform should accelerate scan time by only scanning code that has been modified since the previous scan.

Concurrent Scans

With many applications being developed simultaneously, an AppSec platform needs the ability to scan many different applications at the same time in order to not slow down development schedules.

Fast Scans

Your platform should give quick scans that return shorter lists of only the most actionable vulnerabilities, with minimal false positives. Quickly get a handle on the top threats and fix those first.

Incremental Scans

Scanning large codebases can take a lot of time. However, most code isn't changed between scans. To minimize the impact of security on your development teams, your platform should accelerate scan time by only scanning code that has been modified since the previous scan.

Concurrent Scans

With many applications being developed simultaneously, an AppSec platform needs the ability to scan many different applications at the same time in order to not slow down development schedules.

Deep Scans

While fast scans make it easy to start and allow you to cover your entire application footprint more easily, you need to identify all the risk in your most critical applications. Does an AppSec platform support in-depth scanning in addition to fast scans?

Project Exclusion

Along with incremental scans, your platform should accelerate scan times by allowing you to exclude projects and folders from specific scans, which results in greater precision and optimization of your time.

Taxonomy of Supported Vulnerabilities

You want to make sure that any platform you choose can provide broad, comprehensive identification of as many vulnerabilities as possible. Ask the vendor if they can provide a searchable catalogue of the vulnerabilities they detect.

↘

Cloud-Based Platform

Your platform should be built on the cloud, for the cloud: that means flexibility. You should be able to tailor the type of scan and results you need depending on the application you are scanning. Performance matters, and you get to dictate what performance means to you.

Single SaaS Cloud Platform

To properly correlate results across tools, your AppSec platform should be fully integrated in the cloud.

Private Cloud Deployment Option

When you think of cloud, you might think of multi-tenant or public cloud deployments. However, some enterprises have stricter security requirements that limit them to private cloud deployments. If you have that requirement, make sure that the AppSec vendor supports a private cloud deployment option.

On-Premises Deployment Option

While enterprises are moving to the cloud, there are legitimate reasons for selecting an on-premises solution. Does the AppSec vendor support multiple deployment models, including both cloud-based SaaS and on-premises?

Scalability Process

Most organizations move to a cloud-based platform to make it easier to scale a solution to meet their needs. Ask every AppSec vendor to describe the process to scale up their environment if your needs grow after you've deployed.

Planned Release Cycle and Documentation

Every technology vendor provides regular updates to its solution, such as to add new capabilities or cover the latest disclosed vulnerabilities. How does the solution manage regular scheduled release patches and documentation?

Role-based Access Control

An AppSec platform should support the needs of many different teams and stakeholders throughout your organization. Every type of Users should have access to the tools and data they need to perform their functions.

Security Certifications

You need to comply with various regulations and standards, such as ISO 27001, SOC 2 Type II, and others. That means that your AppSec platform needs to as well.

Building an Excellent
Developer Experience

To fully secure applications, collaboration between the AppSec teams who identify security weaknesses, and the developers who fix them is crucial. Addressing application security requires a holistic approach that bridges the gap between these essential players. For AppSec leaders: you must understand that the road to success is through the developers themselves. Any AppSec effort relies on the buy-in from and the participation of developers. Successful AppSec leaders recognize the importance of putting the developer experience at the center of their program.

There are three pillars that organizations should recognize as elements of a strong developer experience:

✓

Meet developers where they live

Seamlessly integrate AppSec into developers' ecosystem and workflow to make it easier for them to fix vulnerabilities without slowing down

✓

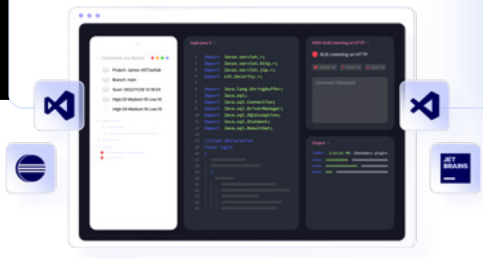
Prioritize the greatest impact

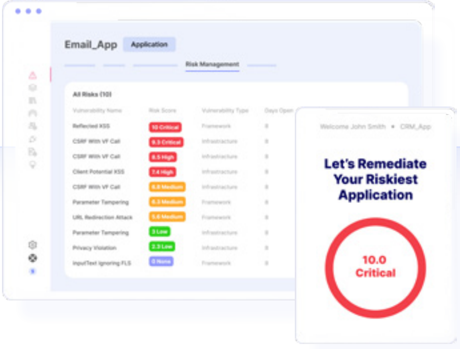
For AppSec teams, there are vulnerabilities that represent the biggest risk. But for developers, it's where they can make the greatest impact

✓

Equip them with tools and knowledge

For AppSec teams, there are vulnerabilities that represent the biggest risk. But for developers, it's where they can make the greatest impact





Email_App Application

Risk Management

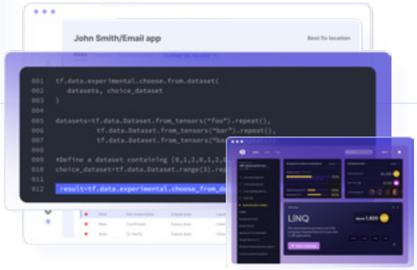
All Risks (10)

Vulnerability Name	Risk Score	Vulnerability Type	Days Open
Reflected XSS	10 Critical	Framework	[illegible]
CSRF With VF Call	9.3 Critical	Infrastructure	[illegible]
CSRF With VF Call	8.5 High	Infrastructure	[illegible]
Client Potential XSS	7.4 High	Infrastructure	[illegible]
CSRF With VF Call	6.8 Medium	Infrastructure	[illegible]
Parameter Tampering	6.3 Medium	Framework	[illegible]
URL Redirection Attack	5.6 Medium	Framework	[illegible]
Parameter Tampering	3 Low	Infrastructure	[illegible]
Privacy Violation	2.3 Low	Infrastructure	[illegible]
InputText Ignoring FLS	0 None	Framework	[illegible]

Welcome John Smith • CRM_App

Let's Remediate Your Riskiest Application

10.0 Critical



John Smith/Email app

Best fix location

```
tf.data.experimental.choose.from.dataset(
  datasets, choice_dataset
)

datasets=tf.data.Dataset.from_tensors("foo").repeat(),
         tf.data.Dataset.from_tensors("bar").repeat(),
         tf.data.Dataset.from_tensors("ba

#Define a dataset containing [0,1,2,0,1,2,0
choice_dataset=tf.data.Dataset.range(3).rep

result=tf.data.experimental.choose_from_da
```

↘

Meet Developers Where They Live

No developer wants to stop coding mid-flow, log into a separate tool, copy-paste their code, and manage a new interface to fix their code. If you want developers to sign on to your AppSec program, you need to integrate seamlessly into their workflow. The right AppSec platform will do this:

IDE Integration

Import scan results directly into any IDE to maintain developer workflow. Direct developers automatically to the line of vulnerable code, provide remediation guidance and links to interactive security training. This gives developers the information they need without making them leave their environment. Your platform should integrate with common IDEs like Eclipse, JetBrains, Visual Studio, and VS Code.

SCM Integrations with Pull Request (PR) Decoration

An AppSec platform should integrate directly with the repo to scan uncompiled code, as early as check-in. You should be able to set triggers for scans based on events, such as the PR, allowing your organization to shift further left while staying within developers' preferred workflow

CI/CD and Build Tool IntegrationDecoration

Automate scanning as part of your CI/CD pipeline by integrating with all major tools automating development, deployment, and testing.

Feedback Tool Integration

Put your vulnerabilities in context for your developers by treating them like any bug. Auto-create tickets in a bug-ticketing system, like JIRA, automatically assign to developers, and automatically close tickets when vulnerabilities are resolved. Vulnerability information, remediation guidance, and knowledge links should all automatically be populated in each ticket.

GenAI Plug-ins

70% of developers currently use GenAI to write code. The most advanced AppSec platforms include GenAI plug-ins to scan generated code for vulnerabilities and malicious packages in the GenAI tool itself before code is exported.

Command Line Interface (CLI) Agent

Provide CLI agents for key platforms (Windows, Liunux, etc.) enabling developers to perform the same operations available through the Web UI (creating projects, managing scans, managing triage results, viewing results, etc.)

Comprehensive Language Support

Developers utilize a vast range of programming languages depending on the type of application they are building. Providing broad, out-of-the-box coverage of both new and legacy programming languages, like .NET, Java, Python, or Dart, is essential.

Development Framework Support

Your AppSec platform should support your development teams in how they work together, with broad support across major frameworks, including the latest frameworks like Flutter.

Compiler-Agnosticism

Within any enterprise different development teams or business units often use different tools, including compilers. An AppSec platform that can support your entire development organization needs to be compiler-agnostic for various development environments.

↘

Prioritize for Greatest Impact

Prioritization is one of the core challenges of building trust between AppSec and development teams. Devs need to be able to trust that they are working on the right issues and are truly making an impact on the business.

↘

Customizable Rules and Rulesets

A fundamental rule of AppSec is that security must be tailored to the application. This is because the root cause behind false positives and false negatives has as much to do with the variation between your applications as it does with the AppSec solution. Different programming languages have different nuances, and what creates a vulnerability in one application may not in another. Tuning your AppSec controls to each unique application is the only way to account for the differences in your applications.

Your platform should come equipped with the following capabilities:

Customizable Rules

Augment prebuilt rules and write custom rules based on unique characteristics of your specific applications.

Customizable Rulesets

Modify existing rulesets, or build custom rulesets, to make it easier to apply different sets of queries to your applications based your AppSec goals.

AI Query Builder

Leverage the power of AI to enable teams with little or no AppSec expertise to build custom rules and tailor security coverage to your applications.

↘

Accurate Results and Prioritization

Improving accuracy and then prioritizing the most critical vulnerabilities is vital in building trust between AppSec and development teams. An AppSec platform should have multiple features to get developers working on the right problems, right away.

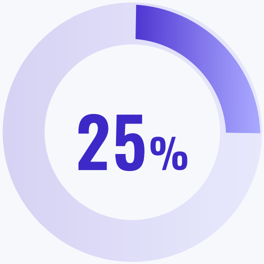
False Positive and False Negative Ratios

Providing accurate results is critical to maintaining a positive developer experience. An AppSec platform needs to provide metrics or control measures for false positives and false negatives.

Best-Fix Location (BFL)

Automatically guide developers to the line of code from which to best fix a vulnerability. Using BFL often results in resolving multiple vulnerabilities with one action, saving developers time and effort.

Equip Developers with Tools and Knowledge



of developers say that they’re confident they can write secure code.

This is not surprising, as none of the top undergraduate computer science programs in the US require a secure coding or secure application design class. Training developers to write more secure code and better understand how to create more secure applications is critical.

Your new platform should have the tools to give developers an interactive security education. It must empower developers and give them the knowledge they need to write secure code from the first line.

Learning Paths

Provide personalized secure code training journeys, designed to equip individual developers with role-specific knowledge, making security training both relevant and effective.

Customize Learning

Easily assign developers the most relevant training, based on vulnerabilities discovered in their own code. Link the lessons to their own vulnerabilities.

Just-in-Time Training

Connect developers to learning modules directly through their vulnerability tickets, eliminating the need for tedious research and directing them immediately towards solutions.

Remediation Recommendations

Fixing vulnerabilities is developers’ responsibility. However, most developers don’t have enough security training or knowledge to know how to remediate a vulnerability without additional research and analysis – which takes a lot of time! An AppSec platform should make it easier for developers to take the actions they need to, by providing actionable and easy-to-understand remediation recommendations on every vulnerability.

AI-Guided Remediation

Utilize AppSec-trained GenAI to suggest remediation steps for identified vulnerabilities, helping developers with less security knowledge reduce the time to identify and fix security flaws.

Accelerate
Time to Value

AppSec teams are increasingly asked to do more with less. Your AppSec platform should offer comprehensive professional services to help you build, refine, and manage your AppSec program. No matter what stage of development your program is in.

Deploying AppSec at enterprise scale isn't a self-service proposition. Your platform vendor should be prepared with a range of services to get you up and running quickly and to help your team continue to mature its AppSec program.

Onboarding

Your vendor should have experts in place to help you understand how long onboarding will take, who needs to be involved, and what the right steps are to proceed.

AppSec Team Training

If you are building out your first AppSec team, it will help to have experts along for the ride. The right platform should equip your team with the tools they need to build a successful program, utilize the tools in the platform, and plan to grow your effort.

Troubleshooting

Got a problem? Your vendor's services team should be easily available to help you sort through it.

AppSec Program Consulting

Your platform vendor should have a standardized framework for you to use in either building or improving your AppSec program. It should be able to easily assess your current state and recommend actionable steps for your team to move forward.

Developer Workshops

Change management, particularly when working across functions such as with development teams, can be a significant challenge. Your platform vendor should offer developer workshops to help with onboarding, training, and engagement.

AppSec Program Management

Your platform vendor should offer a turnkey AppSec service. The managed services team should be able to help set up your program and run it, from configuration and integration to threat modeling, change management, and scanning optimization. It should also be able to plug its own expert AppSec engineers into your existing team to cover gaps in knowledge, skills, or time.

Query Tuning and Preset Optimization:

AppSec should be customized to each individual organization. While having a platform work "out-of-the-box" is great, it should offer expert services to help you get the most out of the platform.

Results Triage and Optimization Optimization:

Your platform should offer services to increase developer adoption and focus developers on exploitable vulnerabilities through scan optimization, triage, and false positive removal. Experts should be able to help your teams better prioritize results for maximum business impact.

General

Vendor Requirements

All else being equal (including meeting all the solution requirements above), organizations always want to purchase solutions from the most experienced and reputable vendor. While there are many ways to evaluate vendors, here are simple ways to do so:

Leader in the Gartner MQ or Forrester Wave

As with any technology solution, it's often difficult to understand how well a vendor can meet your requirements. A list of capabilities on paper is often (and unfortunately) different than your experience in practice. A good starting point is an assessment of a leading analyst firm like Gartner and Forrester who has already done the analysis for you.

Years of Experience

The technology world is made up of a wide range of companies, from yesterday's startups to vendors who have been in business for decades. Startups may excel at bringing the latest, cutting-edge technologies to market, but often suffer in terms of scalability and reliability. When it comes to an AppSec platform, you want a good foundation to build on. Start with a vendor that has at least 10 years of experience in securing software development.

All Your Boxes

Checked

Checkmarx One is the essential enterprise AppSec platform. With an evolving set of correlation and prioritization features, Checkmarx One helps consolidate your expanding list of AppSec tools, and helps you make sense of the results. Designed with developer experience and AppSec teams in mind, Checkmarx One will also help you build the trust needed to ensure the success of your AppSec program investment.

Check all the boxes with

Checkmarx One

Request a Demo

Checkmarx

Checkmarx is the leader in application security and ensures that enterprises worldwide can secure their application development from code to cloud. Our consolidated platform and services address the needs of enterprises by improving security and reducing TCO, while simultaneously building trust between AppSec, developers, and CISOs. At Checkmarx, we believe it's not just about finding risk, but remediating it across the entire application footprint and software supply chain with one seamless process for all relevant stakeholders.

We are honored to serve more than 1,800 customers, which includes 60% of all Fortune 100 companies including Siemens, Airbus, SalesForce, Stellantis, Adidas, Wal-mart and Sanofi.

Printable Checklist

Code to Cloud Pipeline Coverage

- □ Static Application Security Testing (SAST)
- □ Software Composition Analysis (SCA)
- □ Software Supply Chain Security (SSCS)
- □ API Security
- □ Dynamic Application Security Testing (DAST)
- □ Container Security
- □ Infrastructure-as-Code (IaC)
- □ Runtime Security
- □ AppSec Training for Developers

Integration for Correlation and Fix Prioritization

- □ Correlation and Prioritization
- □ Solve for Exploitability
- □ Correlate Runtime Protection

Reports, Dashboards, and Notifications

- □ Unified Dashboard
- □ Application Risk Management
- □ Configurable Dashboard
- □ Export Data to PDF/CSV/JSON
- □ Recent Changes
- □ Executive Summaries and Dashboard
- □ Notification Alert Support
- □ Policy Management
- □ API Support

Analysis and Triage

- □ Data Flow Visualization
- □ Bulk Marking Results
- □ Permanent Results Modification
- □ Vulnerability Assignment
- □ De-Duplication of Similar Findings
- □ Aggregate Vulnerability Graphs
- □ Collaborative Auditing
- □ Enterprise-Grade Application Security
- □ Supports 1000s of repositories
- □ Supports 5 Millions Lines of Code
- □ Incremental Scans
- □ Concurrent Scans
- □ Fast Scans
- □ Deep Scans
- □ Project Exclusion
- □ Taxonomy of Supported Vulnerabilities

Cloud-Based Platform

- □ Single SaaS Cloud platform
- □ Private Cloud Deployment Option
- □ On-Premises Deployment Option
- □ Scalability Process
- □ Planned Release Cycle and Documentation
- □ Role-Based Access Control
- □ Security Certifications

Printable Checklist

Meet Developers Where They Live

- □ IDE Integration
- □ SCM Integrations with Pull Request (PR) Decoration
- □ CI/CD and Build Tool Integration
- □ Feedback Tool Integration
- □ GenAI Plug-ins
- □ Command Line Interface (CLI) Agent
- □ Comprehensive Language Support
- □ Development framework Support
- □ Compiler-Agnosticism

Prioritize for Greatest Impact

- □ Customizable Rules
- □ Customizable Rulesets
- □ AI Query Builder
- □ False Positive and False Negative Ratios
- □ Best-Fix Location (BFL)

Equip Developers with Tools and Knowledge

- □ Learning Paths
- □ Customized Learning
- □ Just-in-Time Training
- □ Remediation Recommendations
- □ AI-Guided Remediation

Accelerate Time to Value

- □ Onboarding
- □ AppSec Team Training
- □ Technical Training
- □ Troubleshooting
- □ AppSec Program Consulting
- □ Developer Workshops
- □ AppSec Program Management
- □ Query Tuning and Preset Optimization
- □ Results Triage and Optimization

General Vendor Requirements

- □ Leader in the Gartner MQ or Forrester Wave
- □ 10+ years of experience in AppSec market