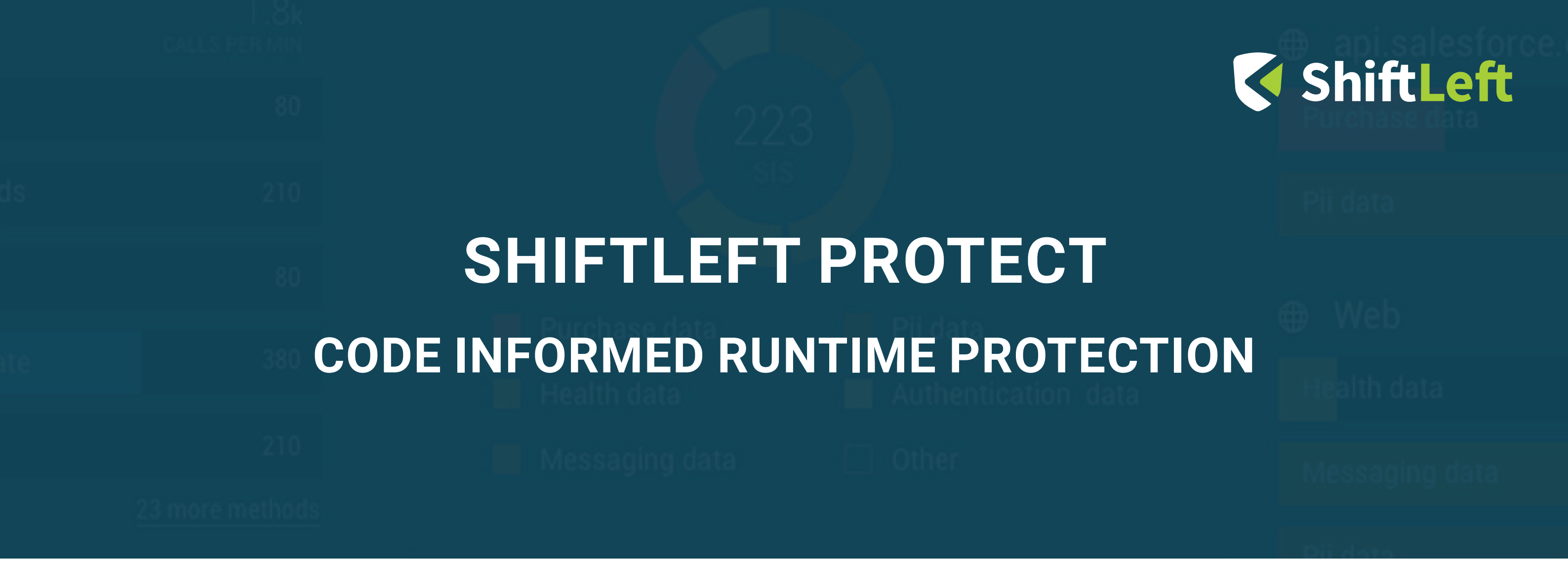


ShiftLeft

SHIFTLEFT PROTECT

CODE INFORMED RUNTIME PROTECTION

ShiftLeft Protect is the only code-informed Runtime Protection solution, providing an inside-out approach to automating vulnerability protections at runtime. Protect leverages "code-informed" insights into vulnerabilities discovered in each version of code during the dev cycle, establishing specific security policies to protect the vulnerabilities still present in the runtime environment.

This laser-focused approach to instrumenting security policies based on residual vulnerabilities enables Protect to operate with minimal footprint and overhead. It also enables users to tightly monitor, and automate "blocking or alerting" responses to, any exploitation attempts targeting these vulnerabilities. Where those instances of exploitation attempts occur, Protect informs DevOps so that vulnerabilities can be re-prioritized in the next or future versions of code.

AUTOMATICALLY PROTECT THE APPLICATION IN PRODUCTION (NO POLICIES REQUIRED)

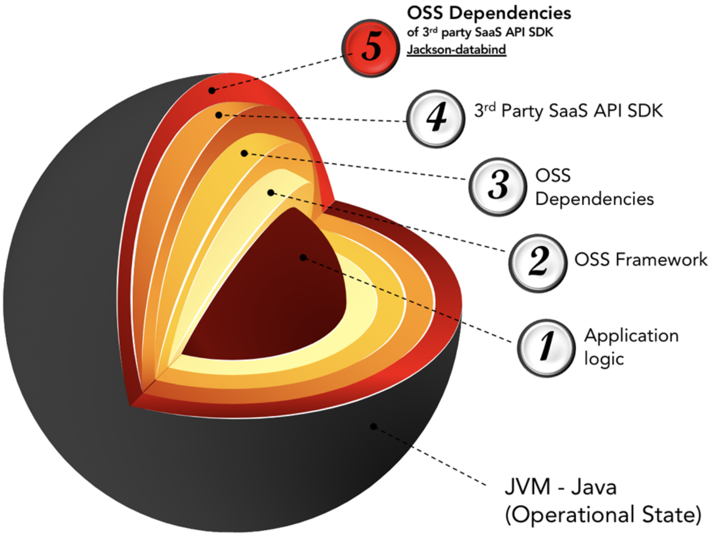
Traditional application security tools such as web application firewalls (WAFs) struggle to keep pace with today's continuous integration/continuous delivery (CI/CD) environments, where code is changing and being deployed at a rapid pace. These tools were built for more static environments as they require constant tuning to establish baselines, in order to understand anomalies. The use of these tools today is more of a security checkbox, versus a useful appliance, at best.

ShiftLeft solutions were built to fulfill today's DevOps requirements through rapid and deep code analysis performed by the code property graph (CPG) built into every ShiftLeft solution. The CPG identifies the vulnerabilities specific to each unique version of code, and it automates the creation of policies for use by Protect, for those vulnerabilities not remediated prior to the release cycle.

ShiftLeft Protect is designed to secure applications automatically at the speed of DevOps. During the development process, ShiftLeft identifies vulnerabilities and potential data leakage scenarios. These are presented to developers to fix; they will typically focus on a subset of the issues identified, depending on the organization's priorities. The ShiftLeft solution then creates a policy for the ShiftLeft application micro-agent to protect the residual issues in production. The process, from analyzing the new build to enforcing a new policy in production, is fully automated, and it takes just minutes.

SECURE YOUR ENTIRE APP (CUSTOM CODE, OPEN SOURCE LIBRARIES, FRAMEWORKS & COMMERCIAL SDKS)

Protect works with the in-depth analysis performed by the CPG, including custom code, frameworks, open source libraries, and commercial SDKs and all their dependencies. Through the holistic analysis performed by the CPG in the development stage, ShiftLeft Inspect is able to identify even the most complex vulnerabilities found in modern applications, such as multi-stage deserialization vulnerabilities stemming from the way individual components are able interact with each other. Inspect can then automatically inform Protect of any vulnerabilities that were not able to be remediated prior to production for tightly focused monitoring during runtime.



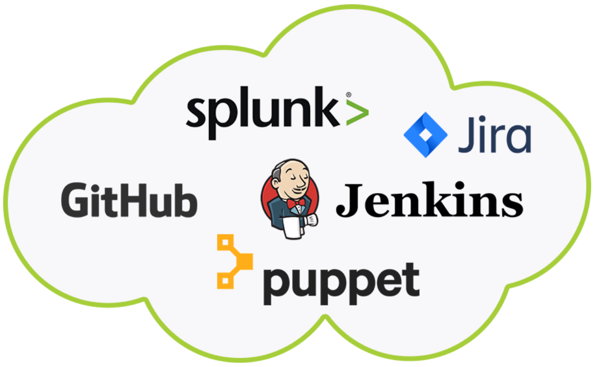
MANAGING SENSITIVE DATA AND COMPLIANCE

The adoption of microservices, open source libraries, commercial SDKs, and external APIs has dramatically increased the complexity of how data flows into, across, and out of modern applications. The results are regular headlines of major organizations accidentally leaking unencrypted critical data through external APIs, logging services, and/or cloud infrastructure. Furthermore, regulations such as GDPR and the California Consumer Privacy Act of 2018 have significantly expanded the types of data that must be protected.

ShiftLeft automatically identifies critical variables in source code using industry-specific natural language processing (NLP). This enables the CPG to map definitively the exact flow of critical variables across sources, transforms, and sinks. Thus, critical data violations are easily identified in development, before being pushed to production. Organizations using non-standard variable naming conventions can edit the critical data dictionary to customize it to their environments.

INTEGRATED WITH CI/CD

ShiftLeft Protect can analyze your code at several points in your integration and deployment pipeline, depending on your needs: at pull request, at code commit, or during the build process.



ACHIEVE CONTINUOUS SECURITY AT THE LOWEST OPEX IN THE INDUSTRY

Application security has traditionally been manual, time-intensive, and computationally expensive. In development, false positives from code analysis, writing QA tests, and patching require many hours from specialized talent. In production, false positives waste time and/or block legitimate traffic. Inline decryption/re-encryption required to inspect encrypted traffic is computationally expensive and adds to the overall complexity.

ShiftLeft significantly reduces these operational inefficiencies by eliminating false positives and automatically creating security profiles that protect the application in runtime. Vulnerabilities are confirmed via test or production traffic, so developers do not waste time fixing false positives. Confirmed vulnerabilities are always delivered to developers with their exact line(s) of code, which further decreases the mean time to repair (MTTR).

Lastly, the computational resources required in production are minimal because of the accuracy of the ShiftLeft solution. ShiftLeft's application micro-agent is informed by the security policy exactly how and where the application is vulnerable, so it is extremely precise in how it protects the application. There is no need to decrypt and re-encrypt production traffic because the ShiftLeft micro-agent sees what the application sees.

For more information on this subject, please download a copy of our whitepaper: The ShiftLeft Economic Value Generated by ShiftLeft's Approach to Modern Application Security (Code Analysis & Runtime Security) - **https://www.shiftleft.io/resources/whitepapers/ROI-shiftleft.pdf**

MINIMAL PERFORMANCE IMPACT

Unlike the agents associated with legacy application performance monitoring or security tools, the Protect micro-agent is extremely lightweight (Figure 1). The automated security profiles that Protect enforces are based on identification of the specific vulnerabilities not addressed prior to runtime. This enables Protect to understand exactly where the application is and is not vulnerable. With this automated understanding, Protect does not need to analyze all web traffic; instead, it only needs to focus on the tiny fraction that is going to the known vulnerable routes. This highly targeted precision enables Protect to secure the application with virtually no production latency and negligible RAM or CPU footprints.

Resource Impact	Legacy RASP + Analysis Tool	Legacy RASP Tool	Legacy WAF	ShiftLeft Protect
CPU Utilization	13% 🟧	3% 🟩	30% 🟥	2% 🟩
Memory Utilization	64 MB 🟧	50 MB 🟧	< 10% Variance 🟩	35 MB 🟩
Latency Impact	144 ms 🟥	11 ms 🟨	613 ms 🟥	2 ms 🟩

**Calculations based on SQL injection payload traffic for Java Vulnerable Lab Application*

Minimal 🟩 Low 🟨 Medium 🟧 High 🟥

Figure 1