Contrast SECURITY

Outdated Application Security Tools put Federal Agencies at Risk

Executive Overview

Agile and DevOps are being implemented in the Federal sector due to increasing pressure to accelerate the speed of missions. The Office of Management and Budget (OMB) reports that 60% of all major Federal IT projects are currently using an iterative (Agile or DevOps) approach.[1] Unfortunately, the benefits of adopting these processes have not been instantaneous; the Federal IT Dashboard, for example, shows average project duration is almost 500 days.[2] At the same time, applications remain the leading source for breaches—both in general and within the public sector. Government agencies need advanced application security designed for speed, agility, efficiency, and defensive requirements of their unique attack surface.

Application Vulnerabilities Pose a Serious Risk

Federal agencies are under pressure to improve operational efficiencies in application development. While meeting mission objectives and speed to deployment are both critical needs, these cannot come at the expense of mitigating cybersecurity risks as applications remain the leading target for hackers. Overall, the percentage of data breaches tied to application vulnerabilities doubled over the past year—hitting 43%. Within the Public Administration sector in particular, web applications were by far the most common breach pattern seen last year.[3]

In addition, the attack surface is getting bigger—with more than 111 billion lines of new software code being written each year that needs to be secured.[4] This is a pressing concern in the Federal sector. For example, within the Department of Defense (DoD) alone, a single fighter jet may include more than 8 million lines of code—and that total jumps to 24 million lines counting its ground-based Autonomic Logistics Information System (ALIS).[5]

> “In many circles, security is still seen as an impediment to software delivery, a blocker that is both maddening and inevitable. However, the truth is that security can actually promote speed.[6]

Contrast SECURITY

Vulnerability Mitigation Should Be a Benefit, Not a Barrier

Traditional methods for application security in Federal agencies can occupy a great deal of development time, even for the smallest changes to be approved. In the legacy model, a developer builds an application and then hands it off to security, which hands it off to operations, which finally deploys it in a completely different production environment. This convoluted process can take months or even years to complete.

Just as problematic, there are often varying layers of underlying software between the development and production environments that create complexity and increase risk. In addition, human workflow dependencies with these tools increase process inefficiency and the chances of one or more errors being made.[7]

> Legacy software acquisition and development practices in the federal government, and especially the dod, do not provide the agility to deploy new software "at the speed of operations." In addition, security is often an afterthought, not built in from the beginning of the life cycle of the application and underlying infrastructure.[8]

Legacy Application Security Tools Cause Multiple Devops Issues

The de facto tools that many government agencies have in place for vulnerability mitigation and other application security functions were not designed for today's Agile/DevOps environments. This causes problems in several critical areas:

The need for speed. Most Federal agencies still rely on penetration testing and scanning tools. These legacy approaches to application security impede the speed and agility of DevOps and require the tester to specifically know what to test for or they test lines of code that do not take into account how the data flows. While any development team today is measured on speed and efficiency, workflows like security testing bring significant inefficiencies—17 hours per week for each developer per one study.[9]

Traditional development processes more often incorporate security as a checkpoint that needs to be passed, but does not integrate security concerns throughout the process.[10] If security is not integrated at the front of software development, agencies that use Agile and DevOps must repeat long development cycles by putting security in place at the end of the process.[11]

Traditional application security approaches simply cannot keep pace with the speed of modern agile and devops. Constant security scans slow release cycles and increase developer inefficiencies.[12]

Limited deployment and scalability features. Outdated application security tools can be difficult to deploy and scale across an agency or even a department within an agency due to incompatibilities with current or planned DevOps and Agile infrastructure. They also typically have limited capacity to efficiently handle the size of many modern applications. With an application that may have upwards of 1 million lines of code, a single scan can take days. When developers make even a small change or vulnerability fix, they must then run the entire scan over again. This approach can delay release cycles by days, weeks, or even months.

Alert fatigue and lack of prioritization. The results from outdated testing methods are riddled with alerts. Many of them do not pose a risk (false positives). In these instances, security teams must triage and diagnose every alert to determine whether or not it presents a true vulnerability. With as many as 85% of these alerts as false positives, the noisy results cause alert fatigue among staff.[13]

These inaccuracies also interfere with effective prioritization of vulnerabilities—increasing an application's risk posture.[14] Tools that lack prioritization in terms of criticality make testing and remediation efforts a guessing game rather than a strategic process with time-sensitive goals. And when developers lose trust in the accuracy of their application security tools, they often begin to think that every alert is a false positive—allowing actual vulnerabilities to pass into production and increasing the risk of an eventual compromise.

Manual workflows. Many of the de facto application security tools used in Federal agencies are dependent on manual workflows—relying on human staff that is slow, expensive, and error-prone. This human dependency largely stems from the fact that siloed security solutions do not integrate with one another. This disconnection inhibits these tools from providing contextual awareness of a vulnerability, which slows mitigation processes. Lack of integration also inhibits automated security responses for detection and prevention of attacks.

At the same time, most organizations struggle to find enough skilled security staff to keep pace with demand. Nearly 75% of organizations report difficulty finding and retaining application security engineers for their DevOps environments.[15] Over half of cybersecurity professionals indicate their organization is at moderate or extreme risk due to staff shortages, and application security is an area where the gaps are the most glaring.[16]

> Government agencies can deliver higher performing and better quality applications by embedding automated security controls and tests into their pipelines. They can also avoid bottlenecks and deliver capabilities faster by Automating the tasks and approval gates that really don't need a human in the loop.[17]

Bureaucratic bottlenecks. Interdepartmental dependencies are bogging down development and creating logjams within the public sector. Outdated application security methods are designed for a back-and-forth workflow—handing application code off to the security team for testing then back to development for remediation.

This creates an adversarial relationship between security and development teams, which can subsequently cause project delays and disrupt broader mission objectives. More than half (55%) of security professionals report difficulties in getting development teams to prioritize remediation of vulnerabilities—even if security is a performance metric for developers.[18]

> Performing a security review or a compliance check as a late-stage task before go-live can lead to serious inefficiencies, as key design decisions may have to be revisited, delaying the final delivery.[19]

Increased risk and remediation costs. False negatives (vulnerabilities that pass through testing undetected into production) elevate the risk of a breach occurring while increasing the cost of eventual vulnerability remediation. It costs six times more to fix a bug found during implementation rather than during the design phase; 15 times more if it's identified in testing; and 100 times more once the code is in production.[20]

Compliance. Many outdated application security tools may not provide sufficient protection to ensure compliance with the latest standards, regulations, and privacy laws. This leads to many agencies lagging behind the curve when it comes to meeting the current compliance requirements. For example, the Office of Management and Budget gave agencies one year to update systems to meet new National Institute of Standards and Technology (NIST) standards. But after nearly two years, some agencies have yet to upgrade a single application.[21]

To comply with a combination of evolving requirements that apply to public sector applications, security teams need transparency and useful insights at every step of the software development life cycle (SDLC)—without waiting for developers to generate and share reports post-development.[22] Scanning lines of code rather than the flow of data across all parts of the application inhibits real-time visibility of the application once it is in production. This obstructs application monitoring and threat prevention/protection capabilities.

Many compliance regulations also emphasize clear documentation of business processes and how incidents should be handled.[23] In these instances, outdated tools may not offer sufficient monitoring, logging, and remediation features to support auditing and reporting needs.

The NIST cybersecurity framework is quickly becoming the default standard used in the public and private sectors in the united states. All U.S. Federal government agencies are now mandated to comply with NIST.

Agencies Need Devsecops for Speed, Efficiency, and Security

Yesterday's methods of application security testing (i.e., penetration testing and legacy scan-based tools) were not designed for the demands of modern applications. This incompatibility slows down Federal development projects and puts mission goals in jeopardy while simultaneously increasing the chances of a significant breach event downstream.

Government agencies need to adopt new application security methodologies to evolve from DevOps to a true DevSecOps model—accelerating time to delivery, achieving mission goals, and ensuring comprehensive security testing early in the SDLC to reduce the risk of application-based breaches. It is now critical to involve security teams in pipeline-related discussions to ensure that the right security steps are built in from the beginning.[24]

1. Jason Miller, "DevOps methodology helps agencies achieve citizen expectations," Federal News Network, January 8, 2020.
2. Jason Miller, "DevOps methodology helps agencies achieve citizen expectations," Federal News Network, January 8, 2020.
3. "2020 Data Breach Investigations Report," Verizon, June 2020.
4. Pieter Danhieux, "What WON'T Happen in Cybersecurity in 2020," Dark Reading, February 4, 2020.
5. Kevin McCaney, "DoD Software Behind the Times? DevSecOps to the Rescue," GovLoop, June 17, 2020.
6. Michael Wright, "How DevSecOps Helps the U.S. Federal Government Achieve Continuous ATO," The New Stack, April 7, 2020.
7. John Morello, "Shift Left: DevSecOps and the Path to Continuous Authority to Operate," Nextgov, July 27, 2020.
8. "DoD Enterprise DevSecOps: Reference Design," U.S. Department of Defense, August 12, 2019.
9. "The Developer Coefficient: a $300B opportunity for businesses," Stripe, accessed September 22, 2020.
10. Chris Yates, "Achieving gains in government IT performance with DevSecOps," FCW, August 12, 2020.
11. Phil Goldstein, "NIST Considers DevSecOps Framework for Agencies," FedTech, April 7, 2020.
12. Jeff Williams, "New NIST Standards on IAST and RASP Deliver State-of-the-Art AppSec," Security, June 19, 2020.
13. "Digital Transformation Thwarted: When Your AppSec Tools, Scanning, and Resources Become Your Mr. Hyde," Contrast Security Webinar, July 23, 2020.
14. Augusto Barros, "From my Gartner Blog – Considering Remediation Approaches For Vulnerability Prioritization," Security Boulevard, May 2, 2019.
15. "Priorities and Challenges for Modern Software Developers," Contrast Security, September 2020.
16. "Strategies for Building and Growing Strong Cybersecurity Teams," (ISC)2 Cybersecurity Workforce Study 2019, accessed February 10, 2020.
17. Michael Wright, "How DevSecOps Helps the U.S. Federal Government Achieve Continuous ATO," The New Stack, April 7, 2020.
18. "2019 Global Developer Report: DevSecOps," GitLab, July 2019.
19. Andrew Davis, "Get security and compliance with DevSecOps: 4 key components," TechBeacon, March 4, 2020.
20. Mukesh Soni, "Defect Prevention: Reducing Costs and Enhancing Quality," iSixSigma, accessed April 16, 2020.
21. Jack Corrigan, "IRS' Outdated App Security Leaves Taxpayers at Risk of Identity Theft, Watchdog Says," Nextgov, April 24, 2019.
22. Michael Wright, "How DevSecOps Helps the U.S. Federal Government Achieve Continuous ATO," The New Stack, April 7, 2020.
23. Andrew Davis, "Get security and compliance with DevSecOps: 4 key components," TechBeacon, March 4, 2020.
24. Michael Wright, "How DevSecOps Helps the U.S. Federal Government Achieve Continuous ATO," The New Stack, April 7, 2020.

Contrast SECURITY

Contrast Security provides the industry's most modern and comprehensive Application Security Platform, removing security roadblocks inefficiencies and empowering enterprises to write and release secure application code faster. Embedding code analysis and attack prevention directly into software with instrumentation, the Contrast platform automatically detects vulnerabilities while developers write code, eliminates false positives, and provides context-specific how-to-fix guidance for easy and fast vulnerability remediation. Doing so enables application and development teams to collaborate more effectively and to innovate faster while accelerating digital transformation initiatives. This is why a growing number of the world's largest private and public sector organizations rely on Contrast to secure their applications in development and extend protection in production.

240 3rd Street
2nd Floor
Los Altos, CA 94022
Phone: 888.371.1333
Fax: 650.397.4133

Contrast SECURITY

contrastsecurity.com