maximus

Shift Left: Achieving Secure Application Modernization with DevSecOps

Federal Introduction

Modernizing legacy systems and software is key to reducing technical debt, enhancing customer experiences, and delivering services with greater speed and agility. New development methodologies, such as DevOps, have offered agencies significant advantages in recent years by shortening the timeline of the systems development lifecycle (SDLC).

But as federal agencies seek to meet the demands of an ever-evolving tech landscape, they must balance modernization needs with heightened cybersecurity protocols. By updating waterfall or DevOps processes to DevSecOps, federal technology leaders can rest assured that security is integrated into every step of the SDLC.

The Challenge: Modernizing Federal Legacy Systems

Effective technology is essential to keeping government operations running smoothly, but upgrading outdated systems is often easier said than done. As the Government Accountability Office notes, the federal government spends billions of dollars each year on information technology, most of which is "used to operate and maintain existing systems, including aging (also called 'legacy') systems. These systems can be more costly to maintain and vulnerable to hackers."

For some agencies, the concept of a major upgrade is particularly important due to the vitality and sensitivity of the programs they maintain. When an application supports missions that are essential to our nation, for example ensuring the integrity of financial systems, technology leaders can't afford to lag behind.

"It's just a different level of criticality," says Kynan Carver, Cybersecurity Lead, DoD at Maximus. "You're talking about the economy of the United States, for example, and the downstream effects of that, and the different companies and livelihoods that depend on that information."

At the same time, those essential missions are best supported by cutting-edge technologies and development processes. While it can be tempting to avoid downtime for a high-value program by building around an outdated legacy system, "at some point, you have to consider that the risk becomes too high," Carver says. "The phrase 'if it's not broken, don't fix it' doesn't apply to legacy systems," particularly when they also need to support newer cybersecurity initiatives, such as Zero Trust.

"The whole idea of just plugging in a few pieces and saying you're Zero Trust compliant – that doesn't work," Carver adds. "With Zero Trust, you have to modernize. You have to update all of your old systems so that they can even speak the Zero Trust language."

To keep up with the pace of innovation, agencies must reevaluate the very foundation upon which applications are built and updated. By shifting from the slower waterfall development methodology, or even the more modern DevOps, to DevSecOps processes, agencies can prioritize agility and efficiency in the SDLC with the added bonus of robust security embedded from start to finish.

The Game Changer: From Waterfall to DevOps

The traditional waterfall method, first established in the 1970s and later applied to software development, breaks the development process into discrete steps that occur sequentially: Requirements, design, implementation, verification, and maintenance. With no overlap between steps, the waterfall method can result in a siloed SDLC with limited communication and interoperability among development, operations, and QA teams.

“It wasn’t very agile and it wasn’t very fast,” Carver says. “It worked, but in order to keep up with growing demands and fast-moving markets and technologies, you had to move into DevOps.”

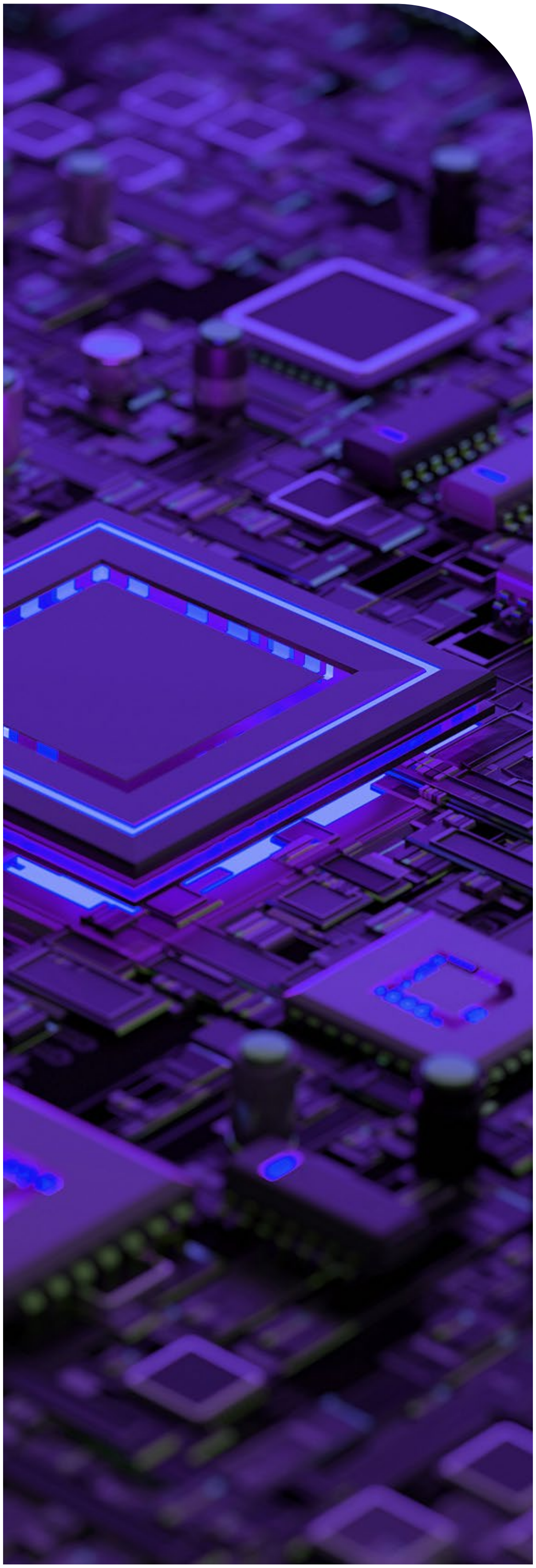
Rather than a series of individual steps, which tend to be downstream and separate from the previous, DevOps operates like an infinite loop: plan, code, build, test, release, deploy, operate, monitor and repeat. “Continuous” is the keyword – continuous integration and continuous deployment (CI/CD) pipelines feed the loop

“You develop faster using CI/CD pipelines, and you’re pushing applications quicker,” Carver says.

The Way of the Future: ‘Shift Left’ With DevSecOps

While DevOps provides the flexibility and agility to significantly speed time to delivery, it still leaves room for some security vulnerabilities. That’s where DevSecOps comes in. DevSecOps is essentially DevOps with a “shift left” approach to security, incorporating it from the beginning of the SDLC.

“By using code analysis tools and automated tests earlier in the development process, security vulnerabilities are more easily identified and resolved,” Carver says. “This allows for a smooth deployment of the software, as security is built in



and not just added on as an afterthought."

Federal agencies can reduce risk through DevSecOps because security flaws and other issues are detected earlier in the development process. Security is no longer an additional step outside of the development process, but an integral part of the infinite loop from planning to operating and monitoring.

"When you do DevSecOps properly, you're embedding security checkpoints and vetted security tools in the process of developing applications, so that at every step of the way, you're doing security tests as the application is being built," Carver says. "As a developer starts coding or pushing code to a repository, it gets scanned and tested immediately vs. waiting until the end to do the security tests.".

How to Make the Shift to DevSecOps

According to Carver, there are several steps agency leaders can take to begin the shift to DevSecOps and establish a more security-focused SDLC:

1 Assess current operations of all programs to determine which are still using the waterfall method and which have already employed DevOps practices. The latter will be much simpler to convert to DevSecOps. Based on the status of current operations, estimate the cost in time and money to make upgrades.

2 Consider the importance of all programs – which programs have the biggest impact on internal operations or citizens served, and which are most at risk due to dependencies on legacy systems?

> **The goal is to establish a culture where security is considered a fundamental aspect of application development, allowing developers to confidently focus on their work without feeling constantly monitored or blamed by security teams. This can be achieved by prioritizing and giving visibility to security within the organization, encouraging collaboration and trust between teams, and implementing processes that empower teams to take ownership of security.**
>
> **KYNAN CARVER**
> Cybersecurity Lead
> DoD

3 Prioritize programs by weighing the previous assessments – cost and value. The fastest and cheapest may not necessarily be the top priority based on level of importance or vulnerability.

For many technology leaders, legacy modernization can create a Catch-22 situation – they build around a dated system to avoid downtime. Meanwhile, as the system ages, downtime due to an outage or security breach becomes increasingly likely. But Carver and Maximus have solutions for upgrading without significant disruption. Take the following client, for example, looking to update a vital, top-priority system.

"We built an exact replica of how the system operates using DevSecOps, and we ran it in parallel," Carver said. "Every time a request would come through, we would process it through the traditional system that was still up and working, and we would double process it through our new system."

Carver and his team continued to process requests in tandem through the two systems until significant issues were resolved. Once they reached a satisfactory amount of time without errors, they were able to switch the client permanently to the DevSecOps platform and turn off the old version.

By "shifting left" the client will now see the benefits of a more flexible development process that strikes a balance between speed and security. Mature DevSecOps practices also break down security silos because all team members are responsible for security throughout development, rather than it being an isolated function.

"The integration of security into DevOps practices can greatly improve the impact and prevent many potential bottlenecks," Carver says. "With the increasing frequency of cyber threats and stricter regulations, incorporating security into DevOps is not just a choice but a necessity."

Learn more about how Maximus can help federal agencies achieve secure modernization to further their missions

→ **maximus.com/digital-modernization**

@maximus_news
maximus