Industry trends Built-in security

4 min read

Evolving Microsoft Security Development Lifecycle (SDL): How continuous SDL can help you build more secure software

By David Ornstein, Principal Software Engineering Manager
Tony Rice, Principal Security PM Manager, Customer Security and Trust

March 7, 2024

The software developers and systems engineers at Microsoft work with large-scale, complex systems, requiring collaboration among diverse and global teams, all while navigating the demands of rapid technological advancement, and today we're sharing how they're tackling security challenges in the white paper: "Building the next generation of the Microsoft Security Development Lifecycle (SDL)", created by pioneers of future software development practices.

Two decades of evolution

It's been 20 years since we introduced the Microsoft Security Development Lifecycle (SDL)—a set of practices and tools that help developers build more secure software, now used industry-wide. Mirroring the culture of Microsoft to uphold security and born out of the Trustworthy Computing initiative, the aim of SDL was—and still is—to embed security and privacy principles into technology from the start and prevent vulnerabilities from reaching customers' environments.

In 20 years, the goal of SDL hasn't changed. But the software development and cybersecurity landscape has—a lot.

With cloud computing, Agile methodologies, and continuous integration/continuous delivery (CI/CD) pipeline automation, software is shipped faster and more frequently. The software supply chain has become more complex and vulnerable to cyberattacks. And new technologies like AI and quantum computing pose new challenges and opportunities for security.

SDL is now a critical pillar of the Microsoft Secure Future Initiative, a multi-year commitment that advances the way we design, build, test, and operate our Microsoft Cloud technology to ensure that we deliver solutions meeting the highest possible standard of security.

Next generation of the Microsoft SDL

Learn how we're tackling security challenges.

Read the white paper >

Continuous evaluation

Microsoft has been evolving the SDL to what we call “continuous SDL”. In short, Microsoft now measures security state more frequently and throughout the development lifecycle. Why? Because times have changed, products are no longer shipped on an annual or biannual basis. With the cloud and CI/CD practices, services are shipped daily or sometimes multiple times a day.

Data-driven methodology

To achieve scale across Microsoft, we automate measurement with a data-driven methodology when possible. Data is collected from various sources, including code analysis tools like CodeQL. Our compliance engine uses this data to trigger actions when needed.

> **CodeQL**: A static analysis engine used by developers to perform security analysis on code outside of a live environment.

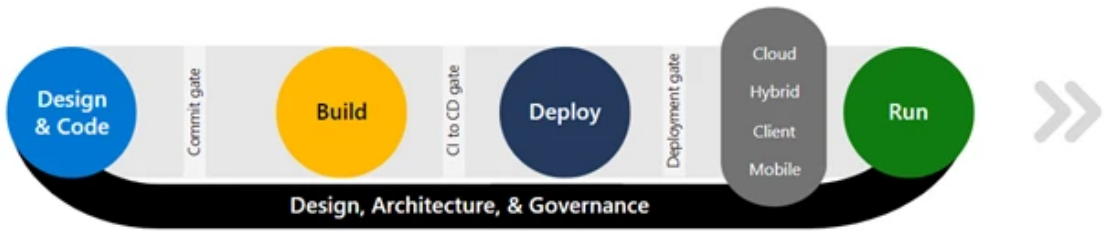
While some SDL controls may never be fully automated, the data-driven methodology helps deliver better security outcomes. In pilot deployments of CodeQL, 92% of action items were addressed and resolved in a timely fashion. We also saw a 77% increase in CodeQL onboarding amongst pilot services.

Transparent, traceable evidence

Software supply chain security has become a top priority due to the rise of high-profile attacks and the increase in dependencies on open-source software. Transparency is particularly important, and Microsoft has pioneered traceability and transparency in the SDL for years. Just as one example, in response to Executive Order 14028, we added a requirement to the SDL to generate software bills of material (SBOMs) for greater transparency.

But we didn't stop there.

To provide transparency into *how* fixes happen, we now architect the storage of evidence into our tooling and platforms. Our compliance engine collects and stores data and telemetry as evidence. By doing so, when the engine determines that a compliance requirement has been met, we can point to the data used to make that determination. The output is available through an interconnected "graph", which links together various signals from developer activity and tooling outputs to create high-fidelity insights. This helps us give customers stronger assurances of our security end-to-end.



Modernized practices

Beyond making the SDL automated, data-driven, and transparent, Microsoft is also focused on modernizing the practices that the SDL is built on to keep up with changing technologies and ensure our products and services are secure by design and by default. In 2023, six new requirements were introduced, six were retired, and 19 received major updates. We're investing in new threat modeling capabilities, accelerating the adoption of new memory-safe languages, and focusing on securing open-source software and the software supply chain.

We're committed to providing continued assurance to open-source software security, measuring and monitoring open-source code repositories to ensure vulnerabilities are identified and remediated on a continuous basis. Microsoft is also dedicated to bringing responsible AI into the SDL, incorporating AI into our security tooling to help developers identify and fix vulnerabilities faster. We've built new capabilities like the AI Red Team to find and fix vulnerabilities in AI systems.

By introducing modernized practices into the SDL, we can stay ahead of attacker innovation, designing faster defenses that protect against new classes of vulnerabilities.

How can continuous SDL benefit you?

Continuous SDL can help you in several ways:

- **Peace of mind**: You can continue to trust that Microsoft products and services are secure by design, by default, and in deployment. Microsoft follows the continuous SDL for software development to continuously evaluate and improve its security posture.
- **Best practices**: You can learn from Microsoft's best practices and tools to apply them to your own software development. Microsoft shares its SDL guidance and resources with the developer community and contributes to open-source security initiatives.
- **Empowerment**: You can prepare for the future of security. Microsoft invests in new technologies and capabilities that address emerging threats and opportunities, such as post-quantum cryptography, AI security, and memory-safe languages.

Where can you learn more?

For more details and visual demonstrations on continuous SDL, read the full white paper by SDL pioneers Tony Rice and David Ornstein.

Learn more about the Secure Future Initiative and how Microsoft builds security into everything we design, develop, and deploy.