LTIMindtree

POV

Merge speed of delivery with security using DevSecOps

Author **Chandra Deo Kumar**
Security Analyst – Cyber Security Practice, LTIMindtree

In order to ramp up the software application development and implementation process, most organizations are turning to DevOps. Though DevOps speeds up the process of application development, the need for security measures often takes a backseat. However, it is crucial to integrate security at an early stage, and that is where DevSecOps comes into the process.

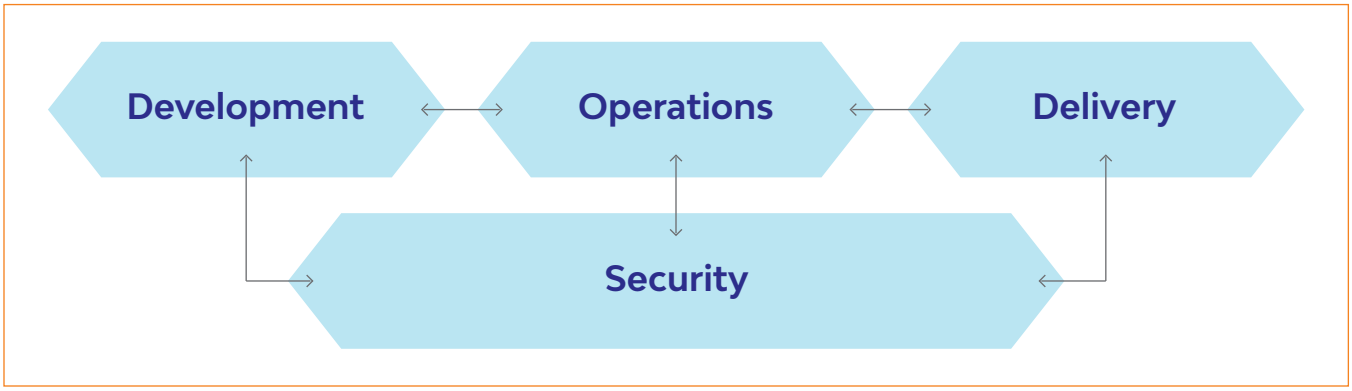
DevSecOps plays a vital role in ensuring true collaboration between the development, operations, and infosec teams in the organization, and helps weave security into the company culture across all the teams involved. With DevSecOps, organizations can integrate security at an early stage to avoid delays in delivery or updates, and bring down security breaches, considering the updates are tested and deployed well in advance.

The need for DevSecOps

With the advent of enhanced SDLC velocity and agile software models, the threat of losing sensitive and critical data is higher. One recent example of such a breach is the case of a major hotel chain, where in February 2020, the hotel confirmed that personal details of more than 10.6 million guests were published on a hacking forum for sale. Security breaches happen every year because of the rise in cyber attacks targeting major organizations. However, a common element that stands out is that almost 85% of the breaches fall in the OWASP Top 10 (OWASP top 10 is a globally accepted security testing methodology).

How DevSecOps works

DevSecOps, an innovative alternative to traditional agile and security audits, involves incorporating automated security gates at each phase of the DevOps pipeline so that the application code is blocked at each gate if it does not match the criteria set. DevSecOps empowers all teams, involved in the DevOps pipeline, to be cognizant about security and accept it as a culture. It removes unnecessary friction among development, operations, delivery, and security teams and introduces a seamless and collaborative environment.



Methodologies in DevSecOps

Each DevOps cycle begins with a threat modeling exercise to identify potential security threats and vulnerabilities, assess them for their seriousness and prioritize them for mitigation. This allows the code development to be more security-oriented.

Transitioning from DevOps to DevSecOps

Process

Policy for each task for Development & Ops teams

Technology

Automation of security gates at each major phase in DevOps

People

"Security is everyone's responsibility" mindset and accommodate

Based on observed errors, the infosec team introduces the below-mentioned methodologies:

1

Static Application Security Testing (SAST)

Developers inadvertently introduce security weakness by not writing secure code as per set guidelines and best practices. Static Application Security Testing helps review and analyze the code behind (source code). During coding, the developer's IDE must be integrated with SAST plugins to alert the developers of any insecure code syntax.

Software Composition Analysis (SCA)

2

Most development teams today use open source libraries and frameworks extensively that are built by a vast community of developers, and they may contain significant security vulnerabilities. In order to identify those vulnerabilities, they must use Software Composition Analysis that tracks the open source components in applications and flag the vulnerabilities within them.

Benefits of DevSecOps

- Security teams can educate the development team(s) to integrate the security plugins in IDE that helps in secure coding.
- Awareness of process or policy changes due to security.
- Automation reduces the risk of manual testing and helps to get accurate results via AI and ML-like techniques with less time spent.

  Continuous and round-the-clock monitoring of security reports.
- Easy integration of security processes as culture change across organizations in the workflow.
- Improving collaboration within teams by changing the mindset and reducing the communication gaps.
- Enhanced software delivery speed and collaboration reduces data breaches.
- Reduction in cost with proper documentation.
- Expedite change in software code behind with security as per business needs within the given constraints.

Container Security

3

Many organizations use containerization to deploy the code in the QA environment and production. The application code is packaged and built, and its container images are created for deployment. Although these containers are short spanned, they must be secured with Container Security tools that scan the container images, registry and running containers to identify the vulnerabilities within them.

Dynamic Application Security Testing (DAST)

4

Dynamic Application Security Testing is performed in the QA phase, along with another functional testing. Once the application is fully built and is in its running state, DAST identifies security vulnerabilities using a real-world hacker's approach. In order to perform DAST, developers must leverage the OWASP top 10 and SANS 25 standards, so that vulnerabilities known to attackers are re-mediated before the application goes live.

Interactive Application Security Testing (IAST)

5

Some organizations have been using Interactive Application Security Testing in the testing phase, which has its benefits. IAST analyzes the application code for vulnerabilities when the application is running, and functional changes and interactions are happening.

Continuous Security in Production

6

Most security breaches in the world happen to the applications running in production. Hence, organizations should perform periodic security audits such as SAST, DAST, SCA, and Penetration Testing, along with recurring compliance validations. In order to safeguard the applications from foreign attacks, a self-defence mechanism known as Runtime Application Self Protection (RASP) is implemented. RASP continuously monitors the application for threats, blocks the illegitimate request, records logs, and issues alerts.

AI and ML-Driven Automation

7

When we implement Security in the DevOps pipeline, we implement different types of security tools at each phase. Many organizations use a mix of open source and commercial tools for better vulnerability coverage. However, when the security teams use multiple tools, they end up getting a vast number of vulnerabilities, and the biggest hurdle is to manually weed out the enormous amounts of false positives and duplicates. To expedite the process, developers can automate the processes of orchestration and correlation of vulnerabilities. An automated mechanism will ensure speedy normalization and deduplication to present us a short and concise list of valid vulnerabilities. It is also essential to bring in a single, unified dashboard to show the real-time application security posture. AI and ML-driven initiatives will inevitably come in to enhance the outcomes of the DevSecOps journey. AI and ML-driven initiatives can be leveraged to minimize false positives and automated security tools selection.

Secure Coding Training

8

Development teams in organizations must perform coding in a secure manner so that fewer vulnerabilities are introduced in the application code. In order to empower the developers to deliver secure code, we must provide developers with secure coding training. The secure coding training must often be curated to the requirements of the business needs.

Conclusion

Security in DevOps must become a culture, and each team must adopt security as its responsibility. DevSecOps is the next evolution in the application security space, allowing for security to be automated at each phase to complement the agile velocity of DevOps. Security must be embedded at every stage of the DevOps pipeline, so that we ensure that no vulnerable code can move ahead. Security must be automated at each phase and needs to complement the agile velocity of DevOps. With teams under pressure to develop software products fast, the need to adopt DevSecOps is urgent so that developers can address code vulnerabilities before a hacker. DevSecOps is a collaborative approach towards delivering a secure software product in a fast and agile way.

About the Author

Chandra Deo Kumar

Security Analyst – Cyber Security Practice, LTIMindtree

Chandra is a Cyber Security professional with diverse experience in Application Security Consulting, DevOps, and Application Development. He has worked extensively in managing Corporate Application Security for global clients across domains, including banking and insurance, retail, renewable energy, and healthcare. At LTIMindtree, Chandra is responsible for identifying customers' technical and business requirements to design Application Security solutions as per industry standards.

About LTIMindtree

LTIMindtree is a global technology consulting and digital solutions company that enables enterprises across industries to reimagine business models, accelerate innovation, and maximize growth by harnessing digital technologies. As a digital transformation partner to more than 700 clients, LTIMindtree brings extensive domain and technology expertise to help drive superior competitive differentiation, customer experiences, and business outcomes in a converging world. Powered by 84,000+ talented and entrepreneurial professionals across more than 30 countries, LTIMindtree — a Larsen & Toubro Group company — solves the most complex business challenges and delivers transformation at scale. For more information, please visit **https://www.ltimindtree.com/.**