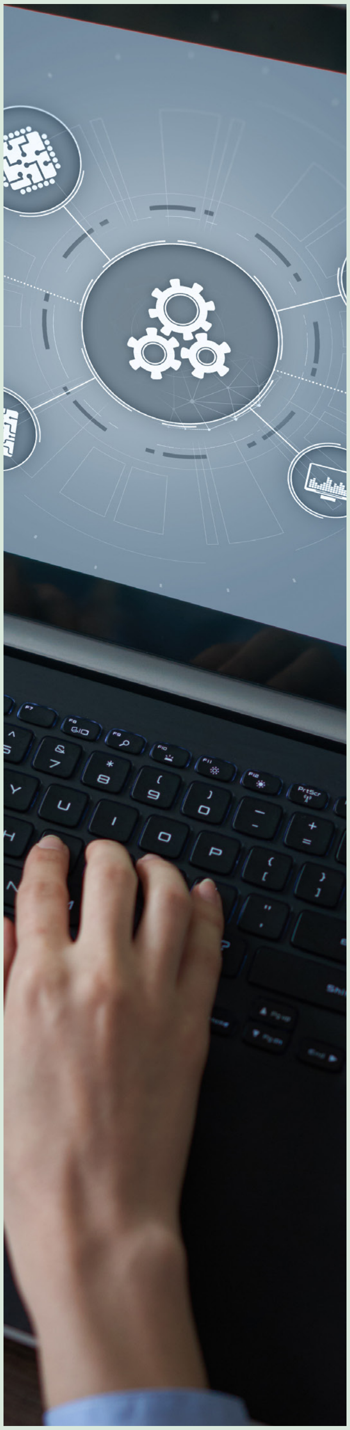
WHITE PAPER



BUILD A SECURE DEVOPS PROGRAM USING INFOSYS DEVSECOPS FRAMEWORK

Abstract

Despite organizations adopting DevOps practices to improve enterprise agility, the task of ensuring application security often resides with separate teams during specific testing phases. As the trend of DevSecOps gains momentum, organizations need better ways to infuse security into CI/CD pipelines to ensure high code quality and protect application data and infrastructure. This paper outlines six key themes for application security. It also describes how the Infosys DevSecOps framework leverages people, processes and technologies to enhance software security in an automated, integrated and transparent manner.

Infosys®
Navigate your next

Introduction

Ensuring software security is a critical part of software development. However, traditional methodologies are often costly, time-consuming and use a piecemeal approach. The responsibility of application security often lies with a separate team and most issues are identified only during the testing phase. Moreover, traditional DevSecOps implementations have fewer security controls added to the CI/CD pipeline. These include static application security testing (SAST), dynamic application security testing (DAST) and open source security analysis.

The biggest challenge, however, is the lack of a definitive framework that provides organizations with guidelines on how to drive DevSecOps programs.

6 vital DevSecOps themes

A recent article by Contrast Security describes six key themes that must be used to drive DevSecOps programs. These themes act as guidelines to help organizations understand what security activities, processes and technologies should be integrated into a DevOps program. The six themes are:

- Empower development and operations teams to deliver secure applications
- Make security activities visible so that they can be tracked, tasked and measured
- Test early by using shift left testing right from the beginning of the software delivery lifecycle (SDLC)
- Leverage security-as-code by automating security actions into actionable test cases, tools and configurations that can be continuously verified
- Enable continuous security, like continuous delivery and deployment, by integrating security with the threat landscape and intelligence
- Identify a balance between prevention (SecOps) and protection (DevOps) to minimize security issues in the SDLC

Infosys approach: Mapping themes to activities

Based on the above themes, Infosys has developed a unique framework that streamlines security integration and deployment. Infosys DevSecOps Framework maps security processes, activities and technologies to each of the above themes. The framework helps secure the application layer as well as the overall infrastructure.

Here are some of the benefits of the Infosys DevSecOps Framework across the three dimensions of people, process and technology:

- **People** – Introduces shared responsibility among the stakeholders, improves software delivery momentum and establishes live communication among teams
- **Process** – Describes the processes to be followed at each stage of the SDLC as well as during the provisioning of infrastructure for deployment
- **Technology** – Recommends key security tools/technologies to be used at each stage in order to enhance the security posture of the developed software

Infosys DevSecOps Framework

Infosys DevSecOps Framework breaks down each theme with a list of security activities that includes the tools and technologies needed, the processes around each tool and people/teams involved. Each step is described below:

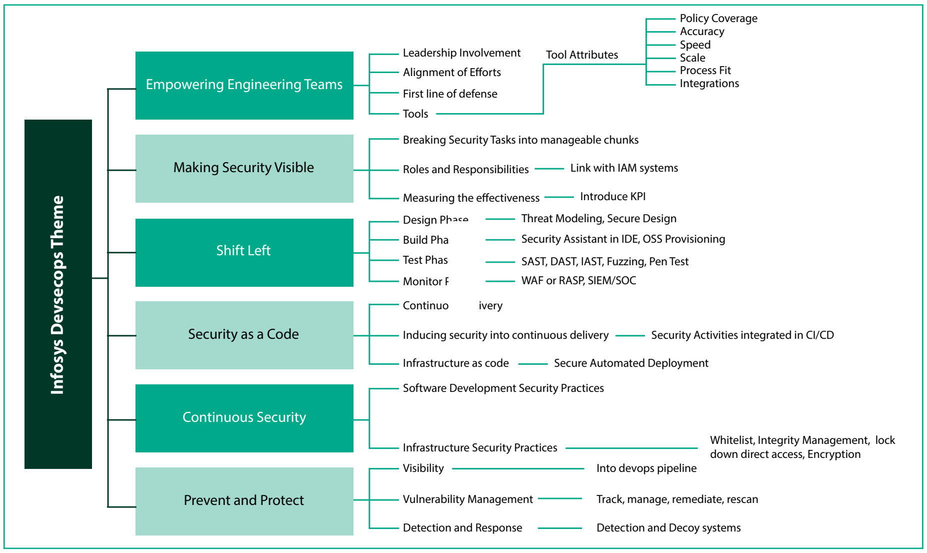

Infosys Devsecops Theme
Empowering Engineering Teams
Leadership Involvement
Alignment of Efforts
First line of defense
Tools
Tool Attributes
Policy Coverage
Accuracy
Speed
Scale
Process Fit
Integrations
Making Security Visible
Breaking Security Tasks into manageable chunks
Roles and Responsibilities
Link with IAM systems
Measuring the effectiveness
Introduce KPI
Shift Left
Design Phase
Threat Modeling, Secure Design
Build Pha
Security Assistant in IDE, OSS Provisioning
Test Phas
SAST, DAST, IAST, Fuzzing, Pen Test
Monitor F
WAF or RASP, SIEM/SOC
Security as a Code
Continuo ivery
Inducing security into continuous delivery
Security Activities integrated in CI/CD
Infrastructure as code
Secure Automated Deployment
Continuous Security
Software Development Security Practices
Infrastructure Security Practices
Whitelist, Integrity Management, lock down direct access, Encryption
Prevent and Protect
Visibility
Into devops pipeline
Vulnerability Management
Track, manage, remediate, rescan
Detection and Response
Detection and Decoy systems

***Fig 1:** How Infosys DevSecOps Framework maps themes to security activities*

1. **Empower engineering teams** – This is the foremost step in the framework. It includes getting leadership involvement and stakeholder buy-in. It also aims to align effort across all teams and empower development as well as operations teams on security aspects. In this step, the framework provides guidelines on what tools need to be used. For instance, the chosen tool should cover at least open web application security project (OWASP) Top 10 2017 and SANS top 25 vulnerabilities. Further, vulnerability management should follow OWASP benchmarks. The tool should also be user-friendly with high accuracy, scalability, and adaptability to integrate with the ecosystem while delivering faster results.

2. **Make security visible** – Security activities should be transparent, manageable, prioritized, and measurable in order to track program effectiveness and success. Infosys DevSecOps Framework can be integrated with identity access management (IAM) systems to provide automated access control, thereby simplifying user login and improving user performance. The framework also helps organizations identify and develop key metrics to measure the success of the program. These metrics include parameters like:

    - Reduction in the number of reported security issues and security-related build delays
    - Number of security-failed builds and missed security findings during build
    - Manual effort and time spent resolving security issues
    - Compliance KPIs like time to compliance, percentage of independent assessments passed and number of corrective measures needed post-audit

3. **Enable shift left testing** – The framework allows organizations to adopt DevOps security measures early on, right from the design phase. For instance, threat modeling ensures the requisite controls are in place before building software, thereby enabling secure design. Some common methodologies used here are STRIDE, DREAD, PASTA, TRIKE, VAST, and OCTAVE. The framework also includes secure coding practices and solutions that integrate with the development environment (IDE). To maximize value, the framework recommends appointing a security champion for each team or business unit who acts as the first line of support when remediating vulnerabilities. Some of the key processes enabled here are:

    - Conducting static analysis, third-party analysis and open source software (OSS) scanning
    - Checking all applications, virtual machines and containers for unknown, embedded or vulnerable components
    - Performing DAST, interactive application security testing (IAST) , abuse use case testing, fuzzy testing, and penetration testing after the build
    - Leveraging web application firewall or run time application security protection to monitor or block malicious traffic
    - Deploying security information and event management (SIEM) solutions that enable the incident response teams to take action. These solutions also feed real-time threat intelligence into systems so that the processes can continuously evolve

4. **Use security as a code** – Most security activities are manual in nature. One of the key differentiators of the Infosys DevSecOps Framework is the use of automation for continuous delivery. Typically, DevSecOps pipelines include development and operations activities like code packaging, auditing and performance tests. Apart from this, Infosys DevSecOps Framework includes security tests in the following stages:

    - Pre-commit stage: Lightweight threat modeling, SAST in IDE and peer code reviews
    - Commit stage: Software component analysis, compile and build time checks, incremental secure code analysis, and digital signing of binary artifacts
    - Acceptance stage: Secure automated provisioning of run time, smoke tests, DAST, functional and integration testing of security features, and automated security attacks
    - Production, deployment and post-deployment stages: Secure automated provisioning of run time, post-deployment checks, compliance checks, run time defense, red teaming, and bug bounty

    While some of these activities can be automated, others can be executed manually but still as part of DevSecOps pipe line.

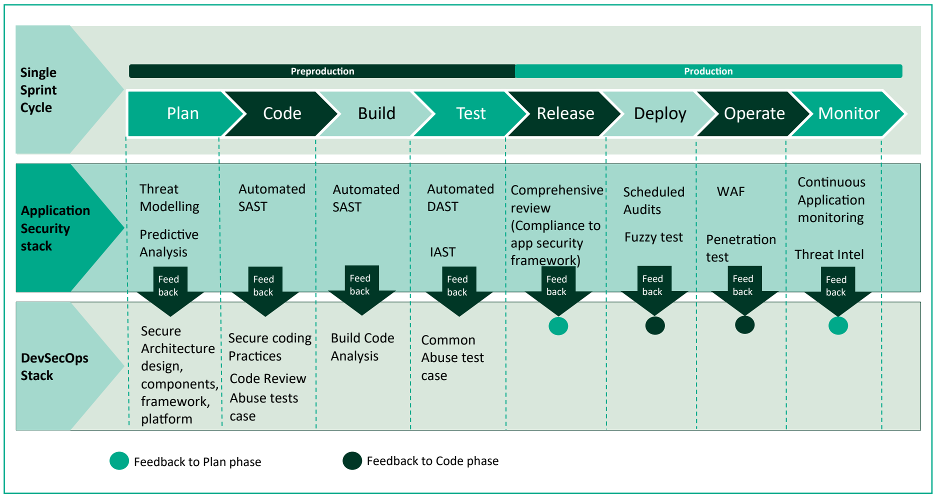
5. **Continuous security** – Continuous security is the ability to respond effectively to threats as they emerge. With Infosys DevSecOps Framework, organizations benefit from a holistic approach to security whereby security activities are continuously performed using an integrated approach that caters to an ever-expanding threat landscape.

6. **Prevent and protect** – While the above security activities can ensure high quality software, these alone do not safeguard software from cyber-attacks. Thus, organizations must adopt best-in-class prevention and protection systems that monitor user login/logout, transactions and interactions as well as network and system activities. The Infosys framework recommends using deception and decoy systems. It also suggests monitoring all system elements and determining the baseline so that meaningful deviations and violations can be immediately detected. Further, all teams should work in tandem to detect the root causes of unusual activities.

Integration model

Infosys DevSecOps Framework enables shared responsibility among the security, development and operations team as shown in Fig 2.

The application security team is responsible for security activities like threat modeling, predictive analysis, SAST, DAST, IAST, fuzzy testing, penetration testing, monitoring, firewalls, and operations. The development team is responsible for implementing robust and secure architecture and components, following secure coding practices, conducting peer code review, and performing common abuse test cases.



	Plan	Code	Build	Test	Release	Deploy	Operate	Monitor
Single Sprint Cycle	Preproduction				Production			
Application Security stack	Threat Modelling Predictive Analysis Feed back	Automated SAST Feed back	Automated SAST Feed back	Automated DAST IAST Feed back	Comprehensive review (Compliance to app security framework) Feed back	Scheduled Audits Fuzzy test Feed back	WAF Penetration test Feed back	Continuous Application monitoring Threat Intel Feed back
DevSecOps Stack	Secure Architecture design, components, framework, platform	Secure coding Practices Code Review Abuse tests case	Build Code Analysis	Common Abuse test case	🟢	●	●	🟢

🟢 Feedback to Plan phase ● Feedback to Code phase

***Fig 2:** The integration model of Infosys DevSecOps Framework and application security activities*

Integration model

Organizations that leverage DevOps for enterprise agility often struggle with ensuring software and application security. Infosys has developed a DevSecOps framework that maps critical security activities across the software delivery lifecycle to six key themes for application security. The framework acts as a roadmap to securing application data and infrastructure by empowering engineering teams, making security visible, enabling shift-left testing, using security as code, enabling continuous security, and deploying prevention and protection solutions. With Infosys DevSecOps Framework, information owners as well as business users who handle application data can benefit from comprehensive protection, automation, real-time threat intelligence, and user-based controls.

About the Author

Celia Aloysius

Senior Technology Architect

Biography: Celia Aloysius comes with over 12 years of experience, of which over 7 years has been in architecting Application Security solutions in customer ecosystems across USA and EMEA regions. She has led several Application Security engagements in the space of Secure Code Analysis, Threat Modeling, Application Penetration Tests, DevSecOps and Maturity Assessments. She is currently focusing on delivering a unified vulnerability management platform that offers risk prioritization based on asset criticality. Celia also has Product Development experience and has worked in Open Source Technologies to deliver products in the field of web analytics, support and content management.

She holds a Bachelors degree in Electronics and Telecommunication Engineering from University of Madras and has written various articles on Application Security.

For more information, contact askus@infosys.com

Navigate your next