
# INSTITUTE OF TECHNOLOGY SCHOOL OF COMPUTING

# DEPARTMENT OF SOFTWARE ENGINEERING

**COURSE TITLE: SOFTWARE ENGINEERING TOOLS AND PRACTICE**

**COURSE CODE: SEng3051**

**STUDENT NAME** **ID**

1. **ASEFU MOLLA ----------145461**

**SUBMMITED TO: Esmail M.**

**SUBMITION DATE: 5/29/2024 GC**

## Contents

## Introduction

DevSecOps, an overall new term in the application security (AppSec) space, is associated with presenting security before in the thing improvement life cycle (SDLC) by fostering the nearby coordinated effort among movement and activities packs in the DevOps headway to join security bundles too. The initiation of devsecops was largly driven by need to address several software engineering problems. DevSecOps in aims to break down silos between development, operations, and security teams, fostering collaboration and communication throughout the software development lifecycle. When it comes to DevSecOps, the integration of security practices within the DevOps workflow is essential for ensuring secure software development and deployment processes. There are several well-known DevSecOps tools that help organizations automate security checks, vulnerability management, compliance monitoring, and overall security posture. It enables a development team to deliver and deploy code quickly without sacrificing security. Implementing DevSecOps practice offers numerous benefits that enhance security, collaboration, efficiency, and reliance throughout the software development lifecycle.it is rapidly field with a wide range of career opportunities and career path available locally and internationally.

## 1. What are Software engineering problems which was cause for initiation of DevSecOps?

Increasing cybercrime and cyber security threats in recent years have brought about the new term DevSecOps in the software industry. To keep up with the modern application and software development needs, it is critical for developers and enterprises to adopt DevSecOps.

As technology builders and maintainers, we build, deploy, and maintain applications in order to help our end users by making their day-to-day existence a little bit easier and more streamlined. These end users trust us with their time and data, so it's important that we take all necessary steps to protect them in their online journeys.

DevSecOps was initiated in response to the growing need for integrating security practices into the software development lifecycle. Traditionally, security was often treated as an afterthought, leading to vulnerabilities and breaches in applications.

- According to "The State of DevSecOps Report" by GitLab, some of the key software engineering problems that led to the initiation of DevSecOps include lack of collaboration between development, security, and operations teams, slow and manual security processes, and inadequately secured code deployments.

- Explore specific incidents or breaches that highlighted the need for integrating security early in the development process.

- Discuss common challenges faced by software engineering teams that prompted the adoption of DevSecOps practices.

## 2. What is DevSecOps?

DevSecOps represents a natural and necessary evolution in the way development organizations approach security. In the past, security was 'tacked on' to software at the end of the development

cycle, almost as an afterthought. A separate security team applied these security measures and then a separate quality assurance (QA) team tested these measures.

This ability to handle security issues was manageable when software updates were released just once or twice a year. But as software developers adopted Agile and DevOps practices, aiming to reduce software development cycles to weeks or even days, the traditional 'tacked-on' approach to security created an unacceptable bottleneck.

DevSecOps integrates application and infrastructure security seamlessly into Agile and DevOps processes and tools. It addresses security issues as they emerge, when they're easier, faster, and less expensive to fix, and before deployment into production.

Additionally, DevSecOps makes application and infrastructure security a shared responsibility of development, security and IT operations teams, rather than the sole responsibility of a security silo. It enables “software, safer, sooner”—the DevSecOps motto–by automating the delivery of secure software without slowing the software development cycle.

## 3. Briefly explain DevSecOps lifecycle?

The DevSecOps lifecycle encompasses a series of stages where security practices are integrated into the software development process from planning to deployment and beyond. Here is an overview of the typical DevSecOps lifecycle stages:

**Plan:**

The planning phases of DevSecOps integration are the least automated, involving collaboration, discussion, review, and a strategy for security analysis. Teams must conduct a security analysis and develop a schedule for security testing that specifies where, when, and how it will carry it out. IriusRisk, a collaborative threat modelling tool, is a well-liked DevSecOps planning tool. There are also tools for collaboration and conversation, like Slack, and solutions for managing and tracking issues, like Jira Software.

**Code:**

Developers can produce better secure code using DevSecOps technologies during the code phase. Code reviews, static code analysis, and pre-commit hooks are essential code-phase

security procedures. Every commit and merge automatically starts a security test or review when security technologies are directly integrated into developers' existing Git workflow. These technologies support different integrated development environments and many programming languages. Some popular security tools include PMD, Gerrit, SpotBugs, CheckStyle, Phabricator, and Find Security Bugs.

**Build:**

The ' build ' step begins once developers develop code for the source repository. The primary objective of DevSecOps build tools is automated security analysis of the build output artifact. Static application software testing (SAST), unit testing, and software component analysis are crucial security procedures. Tools can be implemented into an existing CI/CD pipeline to automate these tests. Dependencies on third-party code, which may come from an unidentified or unreliable source, are frequently installed and built upon by developers. In addition, dependencies on external code may unintentionally or maliciously involve vulnerabilities and exploits. Therefore, reviewing and checking these dependencies for potential security flaws during the development phase is crucial. The most popular tools to create build phase analysis include Checkmarx, SourceClear, Retire.js, SonarQube, OWASP Dependency-Check, and Snyk.

**Test:**

The test phase is initiated once a build artifact has been successfully built and delivered to staging or testing environments. Execution of a complete test suite requires a significant amount of time. Therefore, this stage should fail quickly to save the more expensive test tasks for the final stage. Dynamic application security testing (DAST) tools are used throughout the testing process to detect application flows such as authorization, user authentication, endpoints connected to APIs, and SQL injection. Multiple open-source and paid testing tools are available in the current market. Support functionality and language ecosystems include BDD Automated Security Tests, Boofuzz, JBro Fuzz, OWASP ZAP, SecApp suite, GAUNTLET, IBM AppScan, and Arachi.

**Release:**

The application code should have undergone extensive testing when the DevSecOps cycle is released. The stage focuses on protecting the runtime environment architecture by reviewing

environment configuration values, including user access control, network firewall access, and personal data management. One of the main concerns of the release stage is the principle of least privilege (PoLP). PoLP signifies that each program, process, and user needs the minimum access to carry out its task. This combines checking access tokens and API keys to limit owner access. Without this audit, a hacker can come across a key that grants access to parts of the system that are not intended. In the release phase, configuration management solutions are a crucial security component. Reviewing and auditing the system configuration is then possible in this stage. As a result, commits to a configuration management repository may use to change the configuration, which becomes immutable. Some well-liked configuration management tools include HashiCorp Terraform, Docker, Ansible, Chef, and Puppet.

**Deploy:**

If the earlier process goes well, it's the proper time to deploy the build artifact to the production phase. The security problems affecting the live production system should be addressed during deployment. For instance, it is essential to carefully examine any configuration variations between the current production environment and the initial staging and development settings. In addition, production TLS and DRM certificates should be checked over and validated in preparation for upcoming renewal. The deploy stage is a good time for runtime verification tools such as Osquery, Falco, and Tripwire. It can gather data from an active system to assess if it functions as intended. Organizations can also apply chaos engineering principles by testing a system to increase their confidence in its resilience to turbulence. Replicating real-world occurrences such as hard disc crashes, network connection loss, and server crashes is possible.

**Operation**

Another critical phase is operation, and operations personnel frequently do periodic maintenance. Zero-day vulnerabilities are terrible. Operation teams should monitor them frequently. DevSecOps integration can use IaC tools to protect the organization's infrastructure while swiftly and effectively preventing human error from slipping in.

**Monitor**

A breach can be avoided if security is constantly being monitored for abnormalities. As a result, it's crucial to put in place a robust continuous monitoring tool that operates in real-time to maintain track of system performance and spot any exploits at an early stage.

In this article, we will cover the top 10 DevSecOps tools for application security and explore their key features such as application security testing, vulnerability scanning, integration, and reporting. DevSecOps tools is a broad category of solutions, and so in this article we will look at a range of services, including platforms which may cover DevSecOps capabilities as well as having other capabilities.

## 4. How dose DevSecOps works?

To implement DevSecOps, software teams must first implement DevOps and continuous integration.

**DevOps**

DevOps culture is a software development practice that brings development and operations teams together. It uses tools and automation to promote greater collaboration, communication, and transparency between the two teams. As a result, companies reduce software development time while still remaining flexible to changes.

**Continuous integration**

Continuous integration and continuous delivery (CI/CD) is a modern software development practice that uses automated build-and-test steps to reliably and efficiently deliver small changes to the application. Developers use CI/CD tools to release new versions of an application and quickly respond to issues after the application is available to users. For example, AWS Code Pipeline is a tool that you can use to deploy and manage applications.

DevSecOps

DevSecOps introduces security to the DevOps practice by integrating security assessments throughout the CI/CD process. It makes security a shared responsibility among all team members who are involved in building the software. The development team collaborates with the security team before they write any code. Likewise, operations teams continue to monitor the software for security issues after deploying it. As a result, companies deliver secure software faster while ensuring compliance.

DevSecOps compared to DevOps

DevOps focuses on getting an application to the market as fast as possible. In DevOps, security testing is a separate process that occurs at the end of application development, just before it is deployed. Usually, a separate team tests and enforces security on the software. For example, security teams set up a firewall to test intrusion into the application after it has been built.

DevSecOps, on the other hand, makes security testing a part of the application development process itself. Security teams and developers collaborate to protect the users from software vulnerabilities. For example, security teams set up firewalls, programmers design the code to prevent vulnerabilities, and testers test all changes to prevent unauthorized third-party access.

- DevSecOps incorporates security practices throughout the software development lifecycle, from planning to deployment and operation.

- It emphasizes automation of security controls and testing processes to identify and remediate security vulnerabilities early in the development cycle.

- Collaboration between development, security, and operations teams is essential to ensure timely security assessments and responses.

DevSecOps relies on automation tools for security testing, configuration management, vulnerability scanning, and continuous monitoring of applications.

- By automating security processes and integrating security tools into the development pipeline, DevSecOps enables rapid detection and remediation of security issues.

## 5. Explain well known DevSecOps tools

DevSecOps tools are a set of software and applications that facilitate the integration of security practices into the software development and operations lifecycle. These tools play a pivotal role in ensuring that security measures are seamlessly woven into every step of the development process – from code creation to deployment and beyond.

**Continuous Integration & Continuous Deployment (CI/CD)tools :** solutions play a vital role in the DevSecOps approach by facilitating the automation of application build, test, and deployment processes. By streamlining workflows and emphasizing security at every stage of development, these tools contribute to a seamless and effective software delivery lifecycle.

- **Jenkins** is a widely adopted, free (open-source) automation server that helps automate various aspects of software development, specifically focusing on continuous integration and continuous delivery (CI/CD). In a DevSecOps context, Jenkins plays a critical role in streamlining the build, testing, and deployment stages, ensuring that security checks are seamlessly integrated throughout the development lifecycle.

***Unique features:***

- Wide range of supported programming languages and platforms for diverse development ecosystems.
- Robust plugin ecosystem for additional functionality and customization.
- Extensive library of integrations with other DevSecOps tools.

- **GitLab** free for GitLab Core users and paid options for additional features and support. GitLabCI/CD serves as a fundamental component of the GitLab platform, providing a comprehensive and cohesive CI/CD experience. With the aim of automating the complete application lifecycle, GitLab CI/CD guarantees that the code is constructed, examined, and deployed with a focus on security.

***Unique features:***

- Support for various languages, platforms, and frameworks.

- Built-in container registry for easy management of Docker images.
- Auto DevOps feature for automatic CI/CD pipeline configuration based on best practices.

**Static Application Security Testing (SAST) tools** are important in examining your source code and compiled applications to uncover potential security vulnerabilities. By employing these tools in your development pipeline, you can proactively detect and address security issues early on, mitigating risks and protecting your applications and users from potential threats.

- **SonarQube** is an open-source platform designed to continuously inspect code quality and security throughout the entire development lifecycle. It performs a static code analysis to detect vulnerabilities, code smells, and bugs across a wide range of programming languages, empowering developers and security teams to address issues before they reach production environments.

***Unique features**:*

- Supports over 20 programming languages.
- Customizable rules and quality profiles tailored to organizational requirements.
- Extensive integration capabilities with popular CI/CD tools.
- Provides historical data and trends for code quality and security metrics.

- **FindSecBugs** is an open-source security plugin by OWASP for the **Find Bugs static analysis tool**, specifically targeting **Java applications**. By analyzing byte code, FindSecBugs is language-independent and capable of detecting issues in source code and third-party libraries. It seamlessly integrates with popular IDEs, enabling developers to identify and address vulnerabilities early in the development process.

***Unique features**:*

- Detects a wide range of vulnerability categories, including injection flaws, insecure randomness, and weak cryptography.
- High accuracy and low false positives, make it a reliable choice for Java projects.

- IDE integration allows for real-time vulnerability detection during development.
- Supports custom rules and configurations to meet specific project needs.

**Dynamic Application Security Testing (DAST) Tools** play a pivotal role in uncovering security vulnerabilities in web applications as they operate. By simulating genuine attack scenarios, these tools provide valuable insights into potential weaknesses that could be targeted by cyber criminals, thus empowering security professionals to proactively address and remediate vulnerabilities.

- **The OWASP Zed Attack Proxy (ZAP)** offers an all-inclusive **web application security testing** solution that allows you to identify vulnerabilities in your applications. Developed with a strong focus on DevSecOps from one of the leading web application projects, ZAP features an array of automated scanners and manual testing tools, making it an indispensable asset for security experts across all stages of the software development process.

***Unique features:***

- API for automation and customization, enhancing integration with other DevSecOps tools
- Extensive collection of scripts and add-ons to expand the tool's capabilities
- Spider and AJAX Spider for crawling applications to discover their structure and content
- Passive and active scanning techniques for thorough vulnerability detection

- **Burp Suite** is a powerful web application security testing framework that combines manual and automated testing techniques. Designed to integrate seamlessly into the DevSecOps pipeline, it helps security professionals identify vulnerabilities, understand their impact, and prioritize remediation efforts for more secure applications.

***Unique features:***

- Intruder tool for crafting customized attacks and testing custom payloads
- Repeater tool to manipulate and resend individual requests, examining application responses
- Extensibility through the BApp Store, allowing for additional functionality via third-party add-ons

- Proxy feature for intercepting and modifying HTTP and WebSocket traffic between the browser and the target application

  - **Container security :** plays a vital role in DevSecOps, as it emphasizes safeguarding containerized applications and the infrastructure they rely on. By adopting stringent container security practices, you can shield your applications against a wide array of threats and vulnerabilities during every stage of development and deployment.
  - **Aqua Security** is a platform designed to provide complete container security, ensuring the protection of your containerized applications at every stage of the development process.

  With seamless integration capabilities for Docker, Kubernetes, and other container technologies, Aqua Security empowers you to effectively safeguard and monitor your containerized applications as they transition from development to live production environments.

  ***Unique features****:*

  - In-depth visibility into container activity and risk assessment.
  - Automated remediation of vulnerabilities.
  - Image assurance and drift prevention.
  - Runtime security controls.
  - Compliance enforcement and reporting.

  - ***Sysdig Secure*** *is a comprehensive container security solution that delivers vulnerability scanning, runtime protection, and forensics capabilities for your containerized applications. Designed to work seamlessly with Kubernetes, Docker, and other container technologies, Sysdig Secure ensures that your containerized applications remain secure and compliant from development to production*

    ***Unique features****:*

    - Process-level visibility into container activity.
    - Policy-driven protection and automated incident response.
    - Runtime threat detection and response.

- Compliance and risk management.
- Integration with Kubernetes for enhanced security monitoring.

**Infrastructure as Code (IaC) Security Tools** plays a vital role in managing and safeguarding your cloud infrastructure. These tools empower you to automate resource provisioning and configuration processes while adhering to security best practices and industry standards. By leveraging IaC Security Tools, you can streamline your infrastructure management tasks and fortify the security posture of your entire environment.

**Terraform** is an open-source tool in the Infrastructure as Code category, created to support DevSecOps teams with automating tasks related to provisioning, compliance, and management of infrastructure resources across multiple cloud platforms and on-premises settings. Terraform offers the ability to define the target infrastructure state, thus streamlining the ongoing maintenance and adaptation of the infrastructure.

**Unique features:**

- Robust plugin system for third-party tool and service integration.
- State management system for consistent infrastructure deployment across teams.
- Support for various cloud providers and on-premises environments.

**Checkov** is an open-source static code analysis tool designed to help DevSecOps teams identify and remediate misconfigurations and compliance violations in Infrastructure as Code (IaC) files. With support for Terraform, CloudFormation, Kubernetes, and other IaC files, Checkov provides comprehensive coverage for multiple IaC frameworks, helping ensure that your infrastructure is secure and compliant.

Unique features:

- A graph-based approach for more accurate and efficient IaC file analysis.
- Support for multiple IaC frameworks.
- An extensive list of built-in policies and the capability to create custom policies.

**Pulumi** is an innovative Infrastructure as Code platform tailored to DevSecOps teams that allows you to use familiar programming languages like Python, TypeScript, and Go to automate provisioning, compliance, and management of cloud infrastructure resources. By utilizing

existing programming skills, Pulumi makes it more accessible for developers to define, deploy, and manage cloud infrastructure while ensuring security and compliance.

**Unique features:**

- Support for popular programming languages (Python, TypeScript, Go, etc.).
- Real-time feedback during infrastructure deployments.
- Policy as Code feature for defining and enforcing security and compliance policies across the infrastructure.

**Secrets Management Tools**:Tools for managing secrets are essential in securely storing, handling, and providing access to sensitive data like API keys, tokens, and passwords throughout your applications and infrastructure. By using these solutions, you can make certain that confidential information stays protected and is only made available to authorized users or services.

**HashiCorp Vault** is an open-source secrets management solution that enables secure storage, management, and controlled access to sensitive data such as API keys, tokens, and passwords. With its dynamic secret generation and encryption as a service capabilities, Vault plays a crucial role in the DevSecOps pipeline by ensuring that sensitive data is protected and accessible only to authorized services and users, enhancing overall security.

**Unique features:**

- Dynamic secrets generation, creating short-lived credentials on-demand.
- Encryption as a service, allows data encryption without managing cryptographic keys.

  Support for multiple secret storage backends.

- Extensive API for seamless integration with other tools in the DevSecOps ecosystem.

**CyberArk Conjur** is a secrets management platform specifically designed to secure sensitive data, such as credentials and encryption keys, throughout the CI/CD pipelines and cloud-native environments. By enabling granular access control policies and centralized secrets management, Conjur helps DevSecOps teams safeguard sensitive information and maintain compliance while streamlining the development process.

**Unique features**:

A policy-as-code approach using human-readable YAML files for defining and managing access control policies.

- Seamless integration with other CyberArk products for a comprehensive security solution.
- Built-in high availability and scalability for large-scale deployments.
- Robust API for integration with DevSecOps tools and workflows.

**Infrastructure security tools**: are designed to safeguard your organization's digital assets as they monitor, detect, and mitigate potential risks to your networks and systems. They address vulnerabilities and ensure adherence to multiple security standards.

**Cloudflare** is an extensive and popular cloud platform providing a suite of security and performance services designed to safeguard web applications and infrastructure. With features such as **DDoS** mitigation, a web application firewall (WAF), and secure DNS services, Cloudflare helps you proactively defend your applications and infrastructure in a DevSecOps context, delivering top-notch protection against cyber threats.

***Unique features:***

- Cloudflare's global network spans 200+ cities, reducing latency and improving website performance.
- Advanced analytics and insights to help you fine-tune your security settings and configurations.
- Automatic SSL encryption for all your web applications.
- Built-in serverless computing capabilities with Cloudflare Workers.

➢ **Wazuh** serves as a versatile open-source security monitoring and compliance tool tailored for both cloud and on-premises infrastructures. Equipped with an array of capabilities like intrusion detection, log analysis, and vulnerability detection, Wazuh assists you in safeguarding your infrastructure and ensuring compliance. In the context of DevSecOps, Wazuh delivers real-time insights into your environment.

***Unique features:***

- Flexible and modular architecture, allowing for customization and scalability.
- Comprehensive file integrity monitoring for detecting unauthorized changes to critical files.
- Integration with popular security tools, such as the ELK Stack, Suricata, and more.
- Support for a wide range of industry standards, including PCI-DSS, HIPAA, and NIST.

**Compliance and Governance Tools**: play an important role in the DevSecOps ecosystem, helping organizations maintain compliance with industry standards, regulatory requirements, and best practices. These tools also foster uniform security policies across applications and infrastructure, making them indispensable for a comprehensive security approach.

- **OpenSCAP** is an open-source solution designed for compliance auditing and security configuration management. This tool assists organizations in meeting a variety of security standards, including PCI-DSS, HIPAA, and NIST. By incorporating OpenSCAP you can effectively evaluate, establish, and uphold security baselines while streamlining the process of compliance checks.

  ***Unique features:***

  - Integration with popular configuration management tools like Ansible, Puppet, and Chef.
  - Generates human-readable reports and system remediation guides.
  - Supports SCAP (Security Content Automation Protocol) standard for maintaining security policies.
  - Extensive library of pre-built security profiles for different standards.

- **InSpec by Chef** is an open-source, language-based framework designed for automating compliance checks and enforcing security policies across infrastructure and applications in a DevSecOps environment. It allows you to define and test security and compliance rules using a code-like syntax, ensuring that your systems meet specific requirements.

***Unique features:***

- Supports both Linux and Windows platforms.
- Integrates with popular cloud platforms like AWS and Azure.
- Allows creation of custom compliance profiles.
- Offers executable compliance documentation.
- Can be integrated with Chef Automate for end-to-end infrastructure and application management.

**Identity and Access Management (IAM) Tools:** Within the cyber security landscape, Identity and Access Management (IAM) solutions are essential for overseeing user identities and regulating access to critical resources. By making certain that only authorized individuals gain access to the appropriate systems and information, IAM tools boost security measures and minimize the likelihood of unauthorized access.

- **Okta** is a comprehensive identity management platform designed to streamline secure access control and identity federation for both cloud and on-premises applications from a DevSecOps perspective. Okta simplifies the process of managing user access, providing a centralized solution for Single Sign-On (SSO), Multi-Factor Authentication (MFA), and user provisioning across your organization's applications and infrastructure.

  ***Unique features:***

  - Adaptive Multi-Factor Authentication adjusts authentication requirements based on user risk profiles, devices, and locations.
  - Robust API for custom integration and automation
  - Extensive range of pre-built integrations with popular third-party applications and services.

- **Keycloak** is a powerful, open-source Identity and Access Management platform that facilitates secure authentication, authorization, and user management for web and mobile applications in a DevSecOps environment.

Supporting a variety of authentication protocols, including SAML and OpenID Connect (OIDC), Keycloak streamlines user access management, providing a unified solution with Single Sign-On (SSO), Multi-Factor Authentication (MFA), and identity brokering capabilities.

***Unique features:***

- Easy integration with social logins, such as Facebook, Google, and Twitter.
- Policy-based authorization system for simplified access control management.
- Highly customizable and scalable to accommodate diverse organizational requirements.

**Endpoint security tools:** solutions play a critical role in safeguarding your devices and networks from the ever-growing landscape of cyber threats. By employing these tools, you can effectively monitor, identify, and address potential security incidents on a wide range of endpoints, including desktop computers, laptops, and mobile devices. This proactive approach helps ensure your organization's valuable assets and data remain secure.

- **CrowdStrike Falcon** is a cloud-native endpoint protection platform that delivers a comprehensive set of capabilities for threat detection, incident response, and proactive prevention. It leverages advanced machine learning and behavioral analysis to identify and block known and unknown threats. From a DevSecOps perspective, this integration with other security tools and platforms enhances its ability to safeguard your endpoints and workloads.

***Unique features:***

- Advanced machine learning and behavioral analysis for detecting and blocking threats.
- The cloud-native architecture ensures seamless scalability and easy deployment.
- "**1-10-60**" rule for rapid detection (within 1 minute), investigation (in 10 minutes), and remediation of security incidents (in 60 minutes).
- Integration with other security tools and platforms.

- **Microsoft Defender for Endpoint** serves as a comprehensive endpoint security solution, offering cutting-edge threat protection, automated analysis, and response capabilities for Windows, MacOS, and Linux endpoints. This platform is specifically engineered to integrate smoothly with Microsoft 365 and other Microsoft security offerings, creating a cohesive security experience for your organization.

  In the context of DevSecOps, Microsoft Defender for Endpoint plays a vital role in safeguarding endpoints while identifying potential threats throughout the entire development and deployment pipeline.

  ***Unique features:***

  - Deep integration with the Microsoft ecosystem for a unified security experience.
  - Advanced behavioral analysis, threat intelligence, and automated investigation and response.
  - Microsoft Threat Experts service for expert-level threat monitoring and analysis.
  - Supports Windows, MacOS, and Linux endpoints.

**Incident Response and Forensics Tools:** Tools for incident response and digital forensics play a large role in the arsenal of cyber security professionals. They assist in the examination, inquiry, and resolution of security events, offering a vital understanding of harmful actions while contributing to the deterrence of subsequent assaults.

- **Volatility** is an open-source memory forensics framework designed for incident response and digital investigations. It helps cyber security professionals analyze volatile memory (RAM) from a wide range of systems, such as Windows, Linux, and macOS.

  ***Unique features****:*

  - Extensive range of plugins to enhance functionality and cater to specific analysis requirements.
  - Support for memory dumps from various sources, ensuring versatility in different incident response scenarios.
  - Active development and community contributions, maintaining an up-to-date and effective tool.

- **GRR Rapid Response** is an advanced, open-source remote live forensics framework that enables organizations to swiftly investigate and respond to security incidents. It provides DevSecOps teams with the ability to examine systems remotely, collect crucial forensic data, and execute actions across multiple endpoints simultaneously.

  **Unique features:**

  - Scalability for handling large environments and extensive IT infrastructures.
  - Remote live analysis capabilities without requiring physical access to the systems.
  - A web-based user interface for simplified management and collaboration among incident response team members.

**Network Security Tools:** Network Security Tools shield your network from potential hazards. By keeping an eye on, examining, and warding off vulnerabilities, intrusions, and harmful activities, these tools contribute to establishing a secure environment, enabling you to tackle risks and maintain the integrity of your network infrastructure.

- **Suricata** is a top-tier, open-source network threat detection engine delivering real-time intrusion detection and prevention, network monitoring, and threat-hunting capabilities. By employing an advanced rules language and a robust signature-based detection engine, Suricata plays a critical role in DevSecOps, ensuring network infrastructure security and proactively identifying possible threats.

  ***Unique Features:***

  - File extraction: Capture and analyze files transferred over your network.
  - Integration with threat intelligence platforms: Enhance detection and prevention capabilities by connecting with popular platforms like MISP.

- **Wireahark:** Wireshark is a leading network protocol analyzer, extensively utilized for network troubleshooting, analysis, software development, and communication protocol assessments. As a key component in a DevSecOps pipeline, it empowers security teams

to delve into network traffic, pinpoint potential vulnerabilities, and oversee interactions between applications and services, ultimately fostering a more secure environment.

***Unique Features:***

- Custom filters: Create and apply filters to focus on specific network traffic or protocols.
- Decryption support: Decrypt various encrypted protocols for a more in-depth analysis of secure communications.

By incorporating these well-known DevSecOps tools into the software development lifecycle, organizations can effectively integrate security into their DevOps practices, automate security checks, and prioritize remediation efforts. Each tool brings unique features and capabilities to enhance the security posture of applications and infrastructure.

## 6. What are the benefits of DevSecOps?

"The purpose and intent of DevSecOps is to build on the mindset that everyone is responsible for security , with the goal of safely distributing security decisions at speed and scale to those who hold the highest level of context without sacrificing the safety required," describes Shannon Lietz, co-author of the "DevSecOps Manifesto."

**Rapid, cost-effective software delivery**: When software is developed in a non-DevSecOps environment, security problems can lead to huge time delays. Fixing the code and security issues can be time-consuming and expensive. The rapid, secure delivery of DevSecOps saves time and reduces costs by minimizing the need to repeat a process to address security issues after the fact.

This process becomes more efficient and cost-effective since integrated security cuts out duplicative reviews and unnecessary rebuilds, resulting in more secure code.

**Improved, proactive security:** DevSecOps introduces cybersecurity processes from the beginning of the development cycle. Throughout the development cycle, the code is reviewed, audited, scanned and tested for security issues. These issues are addressed as soon as they are identified. Security problems are fixed before additional dependencies are introduced. Security issues become less expensive to fix when protective technology is identified and implemented early in the cycle.

Additionally, better collaboration between development, security and operations teams improves an organization's response to incidences and problems when they occur. DevSecOps practices reduce the time to patch vulnerabilities and free up security teams to focus on higher value work. These practices also ensure and simplify compliance, saving application development projects from having to be retrofitted for security.

**Accelerated security vulnerability patching**: A key benefit of DevSecOps is how quickly it manages newly identified security vulnerabilities. As DevSecOps integrates vulnerability scanning and patching into the release cycle, the ability to identify and patch common vulnerabilities and exposures (CVE) is diminished. This capability limits the window that a threat actor has to take advantage of vulnerabilities in public-facing production systems.

**Automation compatible with modern development**: Cyber security testing can be integrated into an automated test suite for operations teams if an organization uses a continuous integration/continuous delivery pipeline to ship their software.

Automation of security checks depends strongly on the project and organizational goals. Automated testing can ensure that incorporated software dependencies are at appropriate patch levels, and confirm that software passes security unit testing. Plus, it can test and secure code with static and dynamic analysis before the final update is promoted to production.

**A repeatable and adaptive process**: As organizations mature, their security postures mature. DevSecOps lends itself to repeatable and adaptive processes. DevSecOps ensures that security is applied consistently across the environment, as the environment changes and adapts to new requirements. A mature implementation of DevSecOps will have a solid automation,

configuration management, orchestration, containers, immutable infrastructure and even server less compute environments.

**Reduced security error and associated costs:** By addressing security issues throughout the development lifecycle rather than after release, organizations can catch and address issues earlier, when the time and cost to resolve them is much lower. These costs include lost dollars, additional effort and rework, and customer goodwill

**Reduced time to deploy:** Catching flaws and vulnerabilities through constant testing during the lifecycle reduces time to deploy, reduces time spent after deployment on error response, and improves your readiness to deploy.

- Improved security posture with proactive identification and mitigation of security vulnerabilities.

- Faster delivery of secure software through automation and integration of security processes.

- Enhanced collaboration between development, security, and operations teams.

- Reduced security risks and compliance issues through continuous security monitoring and testing.

## 7. About Local and international DevSecOps career opportunities, career path.

A DevSecOps career can offer you the chance to work with cutting-edge technologies, learn valuable workplace skills, and help organizations streamline and enhance their development processes. With different routes into this career, you'll find various DevSecOps certifications available that can provide your resume with a boost to help you get onto a DevSecOps career path.

DevSecOps combines information security best practices with the ability to integrate and deploy software changes continuously. The combination of DevOps and Sec can improve software reliability, security, and quality. DevSecOps is an approach to development that grew out of

DevOps. Rather than considering security in late development and post-development phases, DevSecOps makes security integral to development activities through the development lifecycle.

**What does a DevSecOps professional do?**

A DevSecOps professional is responsible for the security of the software development process, including automating scans, code verification, and developing security protocols. In this role, you'll work with operations staff and developers to ensure that teams design security into the software from the start and that the software environment is secure and monitored continuously.

**Professional Certificate**

**Microsoft Cybersecurity Analyst**

Launch your career as a cybersecurity analyst. Build job-ready skills for an in-demand career in the field of cybersecurity in as little as 6 months. No prior experience required to get started. Skills you'll build:

Cloud Computing Security, Computer Security Incident Management, Network Security, Penetration Test, Threat mitigation, Computer Architecture, Cybersecurity, Cloud Computing, Operating Systems, Network Monitoring, Computer Network, Information Security (INFOSEC), Encryption techniques, threat intelligence, Compliance techniques, Authentication Methods, Access Management, Enterprise security, Identity governance, Event Management, Security Response, System Testing, Security Testing, Cybersecurity planning, Record management, Data Management, Cloud Architecture, Threat Model, Access Control, Asset Management, Cybersecurity strategies, Regulatory Compliance, Security Analysis

**How do you start a career in DevSecOps?**

Experience is highly prized when employers are looking at DevSecOps job applicants. You'll find different routes to working in this function. You can take various jobs to help you prepare for a DevSecOps role. The important thing is to get some valuable experience before moving into the pressure of a security-focused role.

For example, working as a software developer can help you build experience with coding and developing applications. This job can give you experience in the Dev side of the role. Working in operations or a security role will provide you with experience with the business tools, systems, and processes used to manage and secure software applications.

Should you opt to pursue a college degree, research which major would be most beneficial for your career goals. Depending on the roles you're targeting, you might choose a degree that focuses on cybersecurity or a degree that is more software development-focused.

Attending conferences and workshops can demonstrate that you're keeping up with the latest security trends. Additionally, you can enhance your resume by taking courses and certifications. You'll want to make your resume as appealing as possible to potential employers.

**Types of jobs in DevSecOps**

You'll find many types of jobs in which you can build a career in DevSecOps. For example, you could become a developer, a tester, an operations engineer, or a security analyst. Here are some roles advertised in DevSecOps environments and their average annual salaries.

- DevSecOps engineer:
- DevSecOps software engineer:
- Cloud security engineer:
- Cloud and DevSecOps architect:
- Senior DevSecOps engineer:
- DevSecOps lead:

Skills needed in DevSecOps jobs

When you work in DevSecOps, you'll bring security to the heart of software development and deployment. You'll need an understanding of the organization's development and operational side and will have programming and infrastructure knowledge to ensure that security becomes a vital part of the software lifecycle. To get a DevSecOps job, you'll need to demonstrate both technical and workplace competencies that map to your target role.

Technical skills

You must quickly adapt and learn new technologies in the ever-changing business and technology landscape. Having the capacity to troubleshoot and resolve technical issues fast is critical in this role. Here are some of the top DevSecOps skills you'll see in job advertisements.

- Understanding of code development and scripting languages like Java, C++, XML, and JSON
- Familiarity with automation tools like Puppet, Chef, and Ansible
- Experience with cloud technologies for cloud DevSecOps
- Working knowledge of security concepts and tools like firewalls, intrusion detection/prevention systems, and encryption
- Configuration management expertise
- Familiarity with basic Linux commands
- A keen understanding of networking concepts
- Cloud computing
- Continuous integration and continuous delivery (CI/CD)
- Coding skills in at least one common scripting language, such as Python or Ruby
- Ability to use a text editor, such as Vim or Emacs
- Familiarity with basic Linux commands
- Ability to use a terminal emulator, such as PuTTY or iTerm2

Workplace skills

It's also crucial that you have strong workplace skills. The following skills can help you be more successful in your DevSecOps career and help you positively impact your organization.

- Strong communication and interpersonal skills
- Ability to manage and prioritize tasks
- Knowledge of top-level cybersecurity subjects and issues
- Ability to research threats and draw up logical conclusions through well-thought-out, unbiased processes
- Ability to troubleshoot and solve problems
- Ability to learn new technologies quickly

- Ability to bring together data from diverse sources and articulate it into simple and concise information

What is the future of DevSecOps?

With the ever-growing need for speed and agility, organizations are turning to DevSecOps to help deliver software with greater security and get it to the market faster. By automating security controls, integrating them into the software development process, and taking a more strategic approach to security, companies can mitigate the increasing risk posed by cyber threats.

**Career paths**

1. **DevSecOps Engineer:** DevSecOps Engineers are responsible for designing, implementing, and maintaining secure development practices within the organization. They work closely with development, security, and operations teams to integrate security tools, automate security processes, and monitor security vulnerabilities throughout the development lifecycle.

Skills required: Strong knowledge of security principles, proficiency in scripting and automation tools, expertise in cloud security, and experience with DevOps practices.

2. **Security Automation Specialist:** Security Automation Specialists focus on automating security tasks, processes, and controls to improve efficiency and reduce manual errors.They develop scripts, tools, and integrations to automate security testing, compliance checks, and incident response workflows.

Skills required: Proficiency in scripting languages (e.g., Python, Ruby), experience with automation tools (e.g., Ansible, Puppet), knowledge of security controls and compliance standards.

3. **Security Analyst (DevSecOps):** Security Analysts in DevSecOps roles analyze security threats, vulnerabilities, and incidents within the development pipeline. T hey conduct security testing, vulnerability assessments, and code reviews to identify and remediate security issues proactively.

Skills required: Knowledge of penetration testing tools, security testing methodologies, understanding of secure coding practices, and experience with threat modeling.

4. **Security Architect (DevSecOps):** Security Architects design and implement secure architecture for applications, infrastructure, and cloud environments within the DevOps framework. They establish security controls, define security policies, and provide guidance on security best practices to ensure a robust security posture.

Skills required: Expertise in security architecture design, experience with cloud security solutions, understanding of security frameworks (e.g., OWASP), and knowledge of compliance requirements.

**5. DevSecOps Manager/Director:** DevSecOps Managers or Directors oversee the implementation of security practices across the organization's development teams.They define security strategies, manage security initiatives, and ensure compliance with security policies and regulations.

Skills required: Leadership and project management skills, strategic planning abilities, understanding of security governance, risk management, and compliance.

**6. Compliance Analyst (DevSecOps):** Compliance Analysts in DevSecOps roles focus on ensuring that security practices align with regulatory requirements and industry standards. They conduct audits, assessments, and reviews to verify compliance with security regulations and provide recommendations for improvement.

Skills required: Knowledge of security compliance frameworks (e.g., GDPR, HIPAA), experience with compliance audits, understanding of data protection laws.

## Conclusion

In conclusion, DevSecOps addresses the need for integrating security practices into software development, with a focus on collaboration, automation, and continuous testing. By following the core principles of DevSecOps, implementing the DevSecOps lifecycle, leveraging technical tools, and reaping the benefits of improved security, organizations can enhance their overall security posture and create rewarding career opportunities in the field of DevSecOps.

DevSecOps stands at the intersection of development, security, and operations, offering a holistic approach to building and maintaining secure software systems. By following the DevSecOps lifecycle, leveraging automation tools, and embracing a culture of shared responsibility for security, organizations can effectively mitigate risks, identify vulnerabilities early, and respond to security threats proactively. The benefits of DevSecOps extend beyond improved security to include enhanced collaboration, agility, and overall efficiency in the software development process. As DevSecOps continues to gain prominence in the industry, it presents both challenges and opportunities for professionals to make a significant impact in creating a more secure and resilient digital ecosystem.

# Reference

[1]. https://doi.org/10.58012/fywc-yq50

[2]. https://www.veritis.com/solutions/devops/

[3]. https://expertinsights.com/insights/the-top-10-devsecops-tools-for-application-security/

[4]. https://aws.amazon.com/what-is/devsecops/

[5]. https://www.coursera.arg/articles/devsecops/