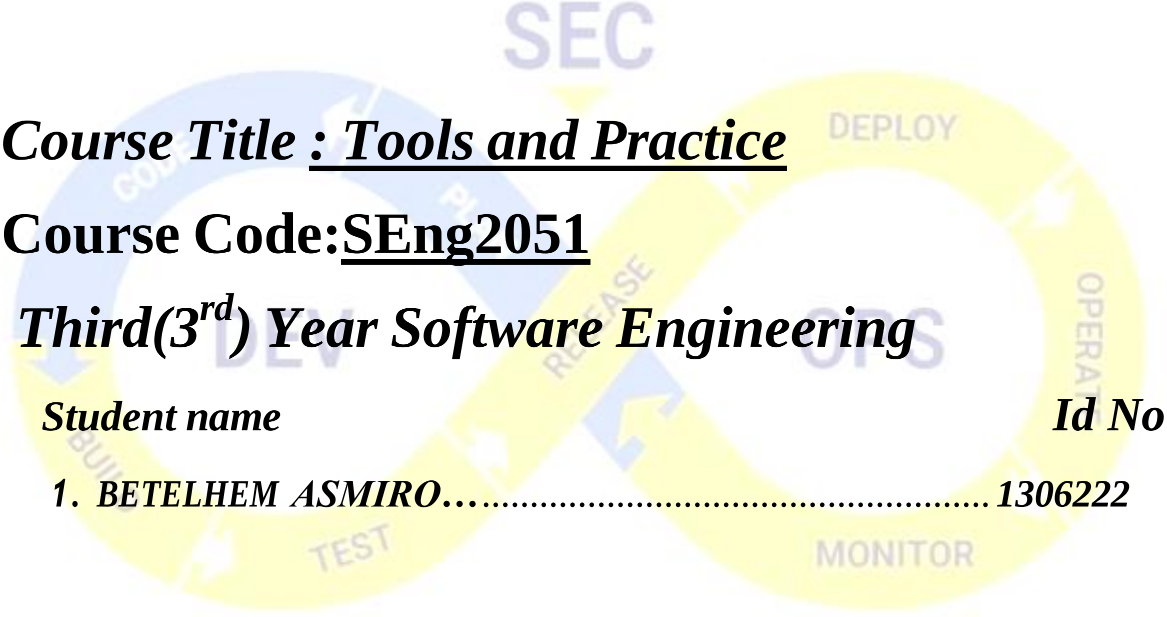
WOLDIA UNIVERSITY

INSTITUTE OF TECHNOLOGY SCHOOL OF COMPUTING

DEPARTMENT OF SOFTWARE ENGINEERING



Course Title : Tools and Practice

Course Code:SEng2051

Third(3rd) Year Software Engineering

Student name ***Id No***

1. *BETELHEM ASMIRO…………………………………………………1306222*

Submitted to:-**MR.Esmail**
Deadline date:- **15/03/2024**

TABLECONTENT **Page**

Introduction……………………………………………………………………………………..3

1. Cause for initiation of DevSecOps…………………………………………………………………...4

2. What is DevSecOps……………………………………………………………………………...5

3. DevSecOps lifecycle………………………………………………………………………………5

4. How does DevSecOps work……………………………………………………………………….7

5. Explain well known DevSecOps tools………………………………………………………....7

6. What are benefit of DevSecOps…………………………………………………………………22

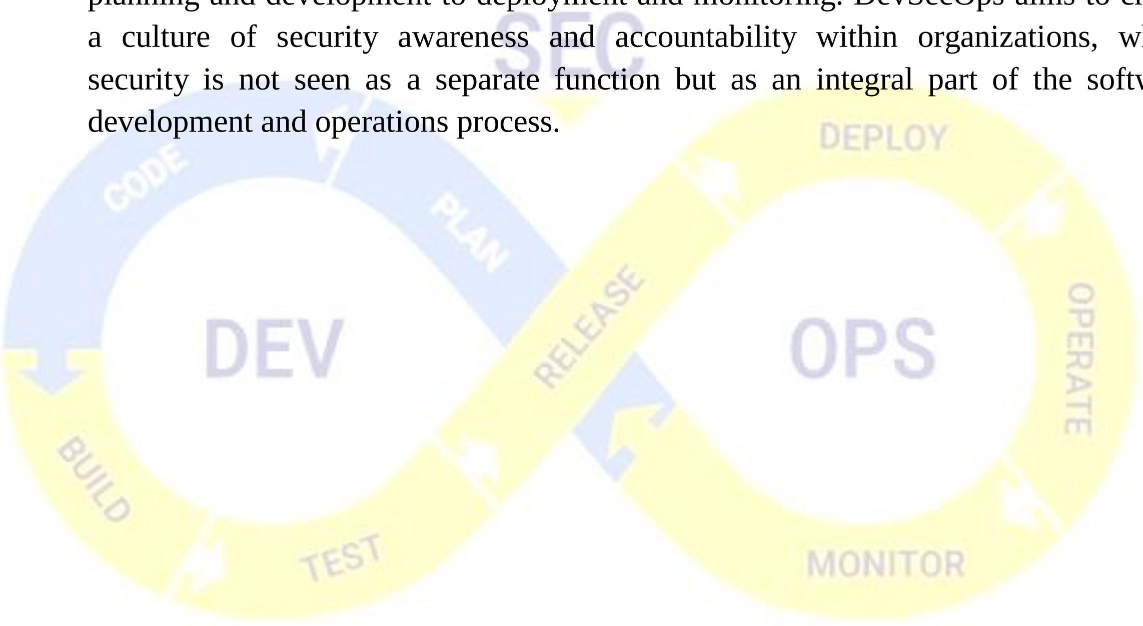
7.local and international DevSecOps career opportunities, career path………………………………………………………………………………………23

Conclusion……………………………………………………………………………………26

Reference……………………………………………………………………………………….27

INTRODUCTION

DevSecOps is a methodology that combines software development (Dev), security (Sec), and IT operations (Ops) to integrate security into every phase of the software development pipeline. It aims to build security into the development process from the very beginning, rather than treating it as an afterthought. By incorporating security practices and tools early on, DevSecOps helps organizations build more secure and resilient software applications. This approach promotes collaboration between development, security, and operations teams to ensure that security is prioritized throughout the development lifecycle. DevSecOps works by integrating security practices into every stage of the software development and operations process, from planning and development to deployment and monitoring. DevSecOps aims to create a culture of security awareness and accountability within organizations, where security is not seen as a separate function but as an integral part of the software development and operations process.



1. *What are software engineering problems which was cause for initiation of DevSecOps*.

There are several software engineering problems that led to the initiation of DevSecOps. Some of these include:

1. Lack of security awareness: Traditionally, security has been an afterthought in the software development process, leading to vulnerabilities being introduced during development.

2. Siloed teams: In many organizations, security teams operate separately from development and operations teams, leading to a lack of communication and collaboration between these groups.

3. Slow security testing: Traditional security testing processes are often slow and manual, leading to delays in the release of secure software.

4. Lack of automation: Manual security processes are error-prone and time-consuming, making it difficult to maintain security across a large number of applications.

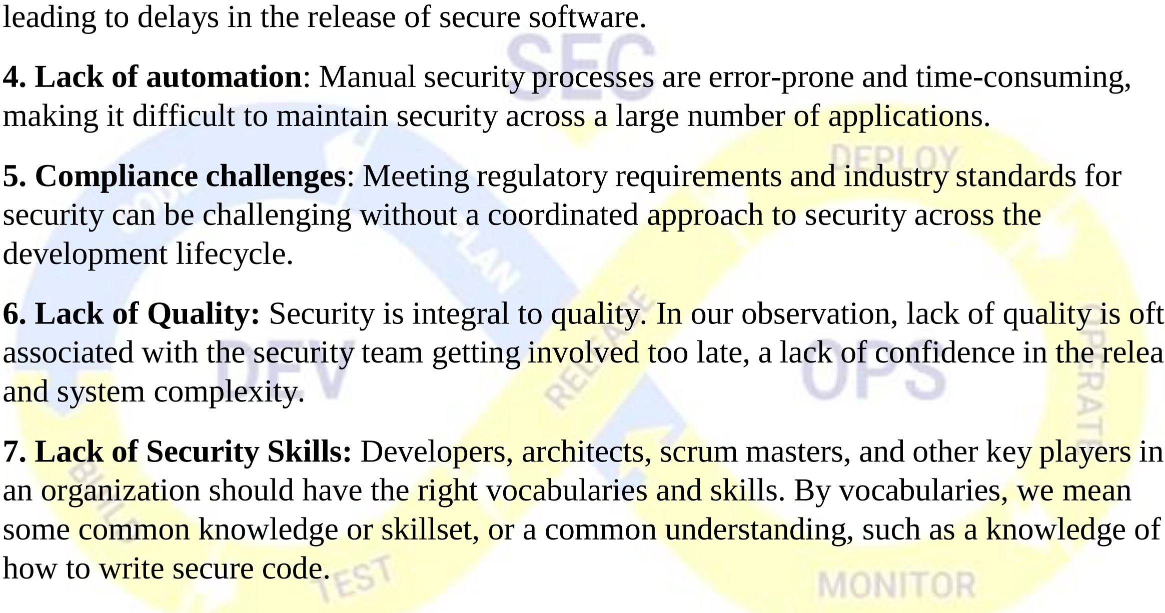
5. Compliance challenges: Meeting regulatory requirements and industry standards for security can be challenging without a coordinated approach to security across the development lifecycle.

6. Lack of Quality: Security is integral to quality. In our observation, lack of quality is often associated with the security team getting involved too late, a lack of confidence in the release, and system complexity.

7. Lack of Security Skills: Developers, architects, scrum masters, and other key players in an organization should have the right vocabularies and skills. By vocabularies, we mean some common knowledge or skillset, or a common understanding, such as a knowledge of how to write secure code.

8. Compliance and regulatory requirements : often add an extra layer of complexity to the software development process, making it difficult to ensure security and compliance at the same time.

9. Complexity: The complexity of modern software systems and the increasing use of third-party components and libraries create new attack surfaces for potential security breaches.



2. *What is DevSecOps?*

DevSecOps, which is short for *development*, *security* and *operations*, is an application development practice that automates the integration of security and security practices at every phase of the software development lifecycle, from initial design through integration, testing, delivery and deployment.

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?*

Steps in the Devsecops Lifecycle

DevSecOps is a software development methodology that emphasizes security and collaboration between development, security, and operations teams throughout the software development lifecycle. DevSecOps works best with teams that use CI/CD, or continuous integration and delivery process, meaning code changes are integrated and released as part of an automated process.

The DevSecOps lifecycle can be broken down into the following steps, with the development, testing, and deployment stages often happening in a loop as software updates are made and new features are added:s

1. Plan

In the planning phase, development teams work with security and operations teams to identify potential security risks and develop a security strategy. This includes identifying security requirements, defining security policies, and selecting the appropriate security testing tools

2. Develop

During the development phase, development teams both build and test the application. This includes integrating automated security testing into the development process, conducting code reviews, and ensuring that security requirements are met.

Since development and testing happen together in the DevSecOps lifecycle, less secure components, such as third-party code, can be tested as they are put into place.

This is where the continuous integration part of the CI/CD process comes in. Code changes are automatically integrated into a shared repository on a regular basis, allowing developers to identify and address conflicts and issues early in the development process.

Optional: Test

Since testing happens during development, a separate testing phase is not necessary in a DevSecOps approach. When it is included, testing takes much less time than it does in a traditional testing process.

During the testing phase, security teams test the application for security weaknesses, vulnerabilities, and threats using penetration testing, vulnerability scanning, and other security testing techniques.

3. Deploy and Monitor

In a traditional process, the operation team would have deployed the application to production. However, the DevSecOps lifecycle follows the DevOps approach, which shifted the responsibility of deploying the application from operations teams to development teams.

The process of deploying to production includes configuring and securing the infrastructure, implementing access controls, and monitoring the environment for security threats.

Today, many development teams trigger deployments using continuous delivery. This involves the use of tools and processes to automatically build, test, and deploy code changes to production environments.

After deployment, teams then monitor the application for security threats and respond to any incidents that occur.

4. *How does DevSecOps works*?

DevSecOps

DevSecOps works by integrating security practices into every stage of the software development and operations process, from planning and development to deployment and monitoring. The key principles and practices that guide DevSecOps implementation include:

1**.** ***Shift Left***: DevSecOps emphasizes shifting security practices and responsibilities to the left in the software development lifecycle, meaning that security is integrated early in the process. By addressing security considerations from the beginning, teams can identify and remediate vulnerabilities sooner, reducing the risk of security incidents in later stages.
2**.** ***Automation***: Automation plays a crucial role in DevSecOps by enabling teams to implement security controls consistently and at scale. Automated tools are used for tasks such as vulnerability scanning, code analysis, configuration management, and deployment, helping to streamline security processes and reduce manual errors.
3. ***Collaboration***: DevSecOps promotes collaboration and communication between development, operations, and security teams. By breaking down silos and fostering cross-functional teamwork, organizations can ensure that security is a shared responsibility and that all team members are aligned on security objectives and practices.
4**.** ***Continuous Improvement***: DevSecOps emphasizes continuous improvement through feedback loops and iterative processes. By regularly assessing security practices, monitoring for vulnerabilities, and implementing lessons learned from security incidents, teams can adapt and enhance their security posture over time.
5. ***Security as Code*:** DevSecOps encourages treating security practices as code, meaning that security controls are defined, implemented, and managed using code-based configurations. This approach allows teams to version control security policies, automate security testing, and integrate security into the same pipelines used for development and operations.
6. ***Risk Management***: DevSecOps incorporates risk management principles to prioritize security efforts based on the potential impact of vulnerabilities. By conducting risk assessments, threat modeling, and prioritizing security activities based on risk levels, teams can focus on addressing the most critical security issues first.
Overall, DevSecOps aims to create a culture of security awareness and accountability within organizations, where security is not seen as a separate function but as an integral part of the software development and operations process. By adopting DevSecOps practices, organizations can improve the security of their applications, reduce the likelihood of security incidents, and build more resilient and secure software systems.

5. *Explain well known DevSecOps Tools*?

- **Continuous Integration & Continuous Deployment (CI/CD) Tools**

Continuous Integration & Continuous Deployment (CI/CD) 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

Jenkins is a widely adopted, 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.

Availability

Free (Open-source)

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 CI/CD

GitLab CI/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.

In the context of DevSecOps, GitLab CI/CD allows teams to incorporate security practices during the entire development cycle, ultimately minimizing the chances of vulnerabilities in the end product.

Availability

Free for GitLab Core users and paid options for additional features and support.

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

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.

This approach fosters a more secure and robust software development environment, ultimately enhancing the overall security posture of your applications.

SonarQube

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.

Availability

Open-source

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

FindSecBugs is an open-source security plugin by OWASP for the **FindBugs static analysis tool**, specifically targeting **Java applications**. By analyzing bytecode, 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.

Availability

Open-source

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*

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.

OWASP ZAP

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

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.

Availability

Free Community Edition with limited features and a paid Professional Edition with advanced functionality.

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 Tools

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

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.

Availability

Free for individual use, paid options are available for teams and enterprises.

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

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.

Availability

Paid with tiers.

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

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

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.

Availability

Open-source, Free.

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

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.

Availability

Open-source, Free.

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

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.

Availability

Free for individuals and small teams and paid options for organizations.

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

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.

Availability

Open-source, free. Enterprise version available with additional features and support.

<u>Unique features:</u>

- 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

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.

Availability

The open-source version (Conjur Open Source) is free, while the enterprise version (CyberArk Dynamic Access Provider) has additional features and paid support options.

<u>Unique features:</u>

- 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

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

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.

Availability

Free for personal use and paid tiers for enterprise users

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

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.

Availability

Paid options with a free trial.

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

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

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.

Availability

Open-source, free.

<u>Unique features:</u>

- 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

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.

Availability

Open-source, free

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

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.

Availability

Paid tiers with a free trial.

Unique features:

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

Keycloak

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.

Availability

Open-source, free

<u>Unique features:</u>

- 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

Endpoint security 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

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.

Availability

Paid tiers, with a free trial.

<u>Unique features:</u>

- 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

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

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.

With its advanced memory analysis capabilities, Volatility is specifically tailored for uncovering artifacts left behind by malware, investigating memory-based attacks, and gathering valuable evidence during incident response.

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

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.

<u>Unique Features:</u>

- 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.

Wireshark

<u>Wireshark is a leading network protocol analyzer</u>, 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.

<u>Unique Features:</u>

- 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.

6. what are the benefits of DevSecOps ?

The two main benefits of DevSecOps are speed and security. Therefore, development teams deliver better, more-secure code faster and cheaper.

“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.

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 teteams 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

Cybersecurity 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 serverless compute environments.

Best practices for DevSecOps

DevSecOps should be the natural incorporation of security controls into your development, delivery and operational processes.

Shift left

'Shift left' is a DevSecOps mantra: It encourages software engineers to move security from the right (end) to the left (beginning) of the DevOps (delivery) process. In a DevSecOps environment, security is an integral part of the development process from the beginning.

An organization that uses DevSecOps brings in their cybersecurity architects and engineers as part of the development team. Their job is to ensure every component, and every configuration item in the stack is patched, configured securely, and documented.

Shifting left allows the DevSecOps team to identify security risks and exposures early and ensures that these security threats are addressed immediately. Not only is the development team thinking about building the product efficiently, but they are also implementing security as they build it.

Security education

Security is a combination of engineering and compliance. Organizations should form an alliance between the development engineers, operations teams and compliance teams to ensure that everyone in the organization understands the company's security posture and follows the same standards.

Everyone involved with the delivery process should be familiar with the basic principles of application security. They should understand the Open Web Application Security Project (OWASP) top 10, application security testing and other security engineering practices. Developers need to understand threat models, compliance checks and have a working knowledge of how to measure risks, exposure, and implement security controls

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

DevSecOps is a rapidly growing field that offers a wide range of career opportunities both locally and internationally. As organizations increasingly prioritize security in their software development and operations processes.Here are some insights into local and international DevSecOps career opportunities and potential career paths:

➢ Local DevSecOps Career Opportunities:

1. **Security Engineer:** Security engineers play a crucial role in implementing security measures, conducting security assessments, and ensuring the overall security of software systems. They work closely with development and operations teams to integrate security practices into the software development lifecycle.

2. **DevSecOps Engineer:** DevSecOps engineers are responsible for automating security processes, implementing security controls, and monitoring security metrics in the CI/CD pipeline. They collaborate with cross-functional teams to ensure that security is embedded throughout the development and operations process.

3. **Security Analyst:** Security analysts assess security vulnerabilities, conduct penetration testing, and analyze security incidents to identify potential threats and risks. They work to enhance the security posture of organizations by providing insights and recommendations for improving security practices.

4. **Security Consultant:** Security consultants provide advisory services to organizations on implementing DevSecOps practices, conducting security assessments, and developing security strategies. They help organizations identify security gaps, mitigate risks, and comply with industry regulations and standards.

➢ International DevSecOps Career Opportunities:

1. **DevSecOps Architect:** DevSecOps architects design and implement secure software architectures, establish security best practices, and oversee the integration of security controls into the software development process. They play a strategic role in shaping the overall security strategy of organizations.

2. **Security Operations Center (SOC) Analyst:** SOC analysts monitor security alerts, investigate security incidents, and respond to cybersecurity threats in real-time. They work in SOC environments to detect and mitigate security incidents, analyze security logs, and maintain the security posture of organizations.

3. **Chief Information Security Officer (CISO):** CISOs are senior executives responsible for leading the organization's cybersecurity strategy, managing the information security program, and ensuring compliance with regulatory requirements. They oversee the implementation of DevSecOps practices to protect sensitive data and mitigate cyber risks.

DevSecOps Career Path:

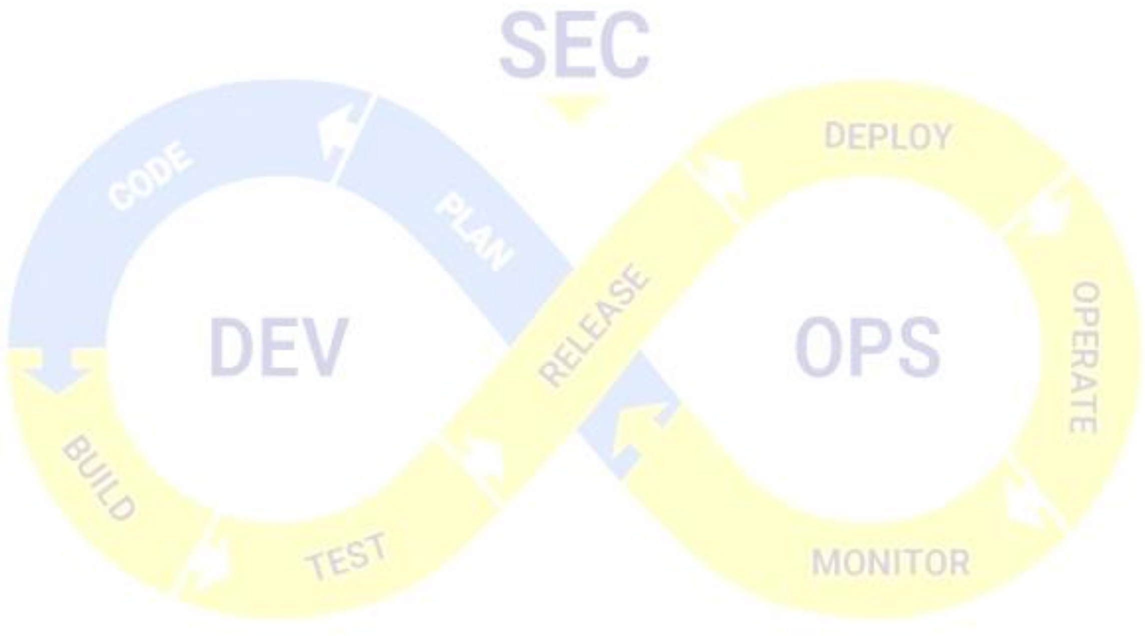
- Entry-Level: Junior Security Analyst, Security Operations Analyst

- Mid-Level: DevSecOps Engineer, Security Engineer, Security Consultant

- Senior-Level: DevSecOps Architect, Chief Information Security Officer (CISO)

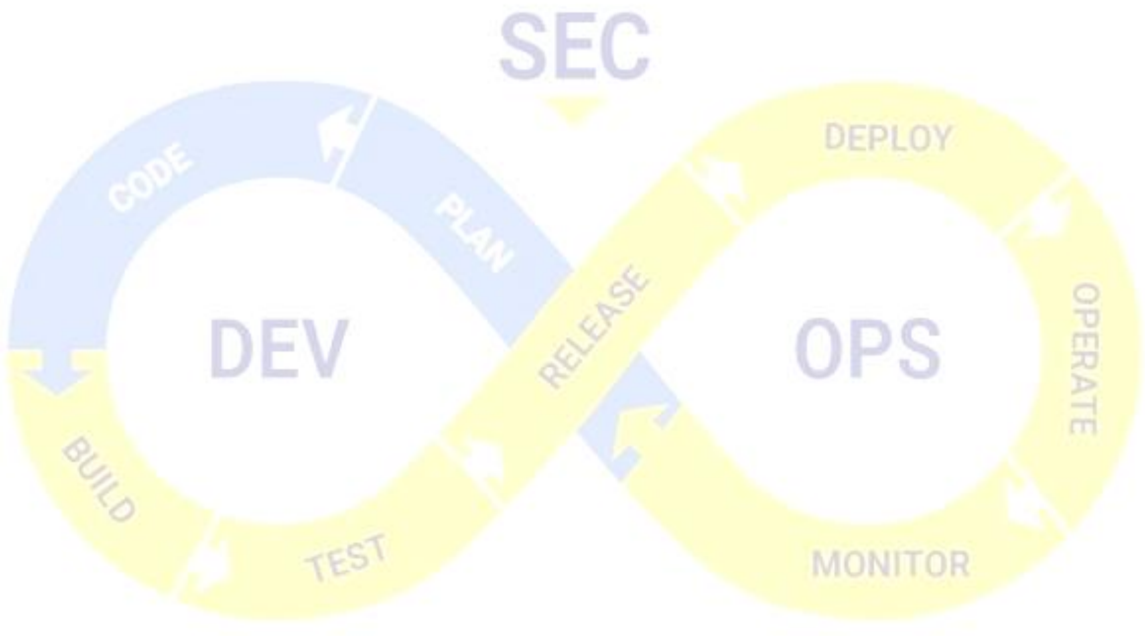
To advance in a DevSecOps career, professionals can pursue certifications such as Certified DevSecOps Professional (CDP), Certified Information Systems Security Professional (CISSP), or Certified Ethical Hacker (CEH). Continuous learning, hands-on experience, and staying updated on industry trends are essential for career growth in DevSecOps.

Overall, DevSecOps offers diverse career opportunities locally and internationally, with roles ranging from entry-level positions to senior leadership roles. By acquiring relevant skills, certifications, and experience, professionals can build successful careers in this dynamic and high-demand field.



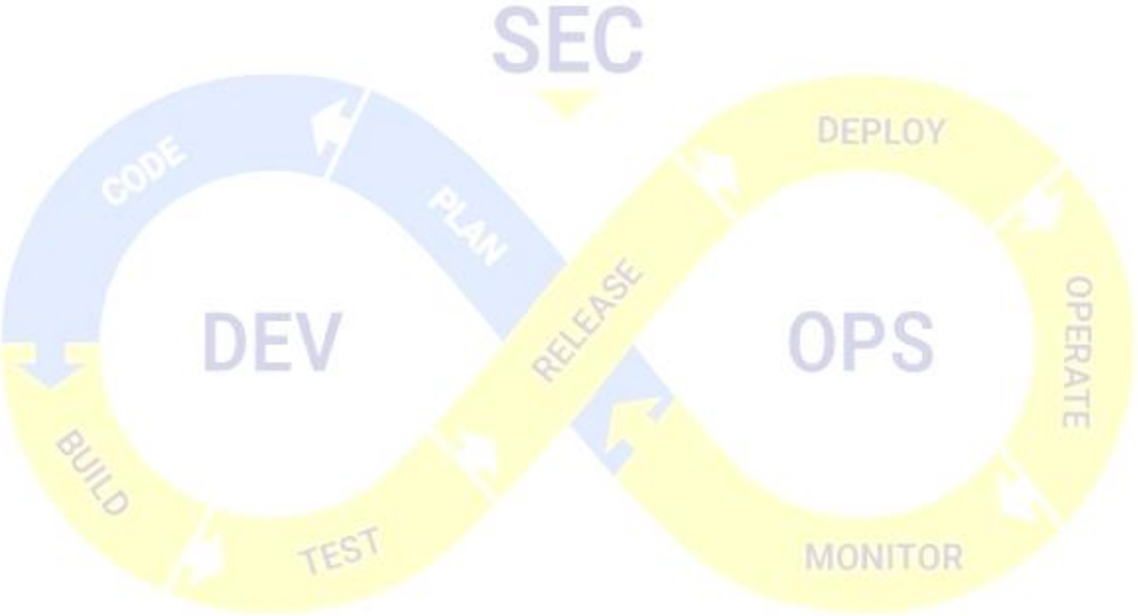
CONCLUSION

There are so many well-known DevSecOps tools that organizations can use to enhance their security practices throughout the software development and operations.These tools help organizations automate security processes, detect vulnerabilities, manage security configurations, and ensure compliance with security standards throughout the software development lifecycle. By integrating these tools into their DevSecOps practices, organizations can strengthen their security posture and build more secure and resilient software systems. Also DevSecOps offers diverse career opportunities locally and internationally.



Reference

- ***https://aws.amazon.com***
- ***https://www.synopsys.com/glossary***
- ***https://www.ibm.com***
- ***https://www.veritis.com***
- ***https://www.browserstack.com***


SEC
CODE
PLAN
DEV
BUILD
TEST
RELEASE
DEPLOY
OPS
OPERATE
MONITOR