INSTITUTE OF TECHNOLOGY

School of Computing

Department of Software Engineering

Software Engineering Tools and Practice

Course code:-SEng3051

Assignment1 DevSecOps

Name **Id.no**

Endale Gebeyehu …….. 1301039

Submitted to : Mr. Esmael Mohammed

Submitted date : 15- march-2024

TABLE OF CONTENT

Introduction

The emergence of DevSecOps was catalyzed by critical software engineering challenges, necessitating a paradigm shift in software development practices to address security concerns effectively. This integrated approach acknowledges the need for security measures to be seamlessly integrated into the development lifecycle, thus mitigating vulnerabilities and enhancing overall software resilience. evolution of DevSecOps was catalyzed by a series of software engineering challenges that demanded a paradigm shift in approach. These challenges included security vulnerabilities, fragmented security practices, slow and reactive security measures, compliance burdens, and the need for seamless integration within CI/CD pipelines. DevSecOps emerged as a holistic solution to address these issues, emphasizing the integration of security practices into every stage of the software development lifecycle.

1. Software Engineering problems which was cause for initiation of DevSecOps:

Software Engineering Problems Leading to DevSecOps The rise in security breaches, fragmented security practices, slow and reactive security measures, regulatory compliance challenges, and the need for continuous integration and delivery prompted the initiation of DevSecOps.
The initiation of DevSecOps was triggered by various software engineering problems, including:

- **Security Vulnerabilities:** Traditional development practices often prioritize speed and functionality over security, leading to the proliferation of vulnerabilities in software applications.
- **Silos between Development, Security, and Operations:** Historically, development, security, and operations teams operated in isolation, hindering effective collaboration and communication.
- **Manual Security Processes:** Manual security processes were slow, error-prone, and reactive, leading to delays in identifying and mitigating security issues.
- **Compliance Challenges:** Regulatory requirements and industry standards mandated robust security measures, but compliance was often challenging to achieve with traditional approaches.
- **Need for Continuous Integration/Continuous Delivery (CI/CD):** The adoption of CI/CD practices accelerated software development cycles, necessitating security measures that could keep pace with rapid releases.

Some of the software engineering problems that led to the initiation of DevSecOps include:

1. Lack of security awareness: Traditional software development practices often prioritize speed and functionality over security, leading to vulnerabilities in the code that can be exploited by attackers.

2. Siloed teams: In many organizations, security, development, and operations teams work in isolation from each other, leading to miscommunication and a lack of collaboration on security issues.

3. Slow security testing: Traditional security testing processes are often manual and time-consuming, leading to delays in identifying and addressing security vulnerabilities.

4. Compliance challenges: Meeting regulatory requirements and industry standards for security can be difficult without a unified approach to security across development, operations, and security teams.

5. Limited visibility: Without proper monitoring and logging in place, it can be challenging to detect and respond to security incidents in a timely manner.

6. Lack of automation: Manual security processes are error-prone and inefficient, making it difficult to keep up with the pace of modern software development.

DevSecOps was initiated to address these challenges by integrating security practices into the software development lifecycle from the beginning, fostering collaboration between development, operations, and security teams, and automating security testing and monitoring processes development lifecycle. It aims to foster a culture of collaboration, automation, and continuous security testing to ensure that security is not an afterthought but an inherent aspect of development and operations.

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.

DevSecOps is an evolution of the DevOps methodology, emphasizing the integration of security practices into every stage of the software development lifecycle. It aims to foster a culture of collaboration, automation, and continuous security testing to ensure

that security is not an afterthought but an inherent aspect of development and operations.

What exactly is DevSecOps?

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.
It's an approach to culture, automation, and platform design that integrates security as a shared responsibility throughout the entire IT lifecycle

what does DevSecOps stand for?

DevSecOps stands for development, security, and operations. It is an extension of the DevOps practice. Each term defines different roles and responsibilities of software teams when they are building software applications.

Development

Development is the process of planning, coding, building, and testing the application.

Security

Security means introducing security earlier in the software development cycle. For example, programmers ensure that the code is free of security vulnerabilities, and security practitioners test the software further before the company releases it.

Operations

The operations team releases, monitors, and fixes any issues that arise from the sotware.

3. DevSecOps lifecycle

The software development lifecycle (SDLC) is a structured process that guides software teams to produce high-quality applications. Software teams use the SDLC to reduce costs, minimize mistakes, and ensure the software aligns with the project's objectives at all times. The software development life cycle takes software teams through these stages:

- Requirement analysis
- Planning
- Architectural design
- Software development
- Testing
- Deployment

DevSecOps in the SDLC

In conventional software development methods, security testing was a separate process from the SDLC. The security team discovered security flaws only after they built the software. The DevSecOps framework improves the SDLC by detecting vulnerabilities throughout the software development and delivery process.

The DevSecOps lifecycle serves as the backbone of security enhancement within the software development continuum. It embodies a structured flow of stages that enables organizations to embed security practices from inception to deployment, fostering a security-centric culture across teams.

1. **Security by Design**: Embedding security principles at the core of software development processes.
2. **Collaborative Synergy**: Nurturing collaboration among diverse teams for enhanced security posture.
3. **Risk Mitigation Strategies**: Proactively identifying and mitigating security risks during the lifecycle.
4. **Iterative Security Enhancements**: Instilling a cycle of continuous improvement for bolstering security resilience.

Key Phases in the DevSecOps Lifecycle

Embark on a journey through the pivotal phases that define the DevSecOps lifecycle:

1. Planning and Security Integration

- **Define Security Requirements:** Lay down the foundational security requirements and objectives.
- **Integrate Security Controls:** Incorporate security controls early in the planning stage to align with overarching security goals.

2. Continuous Integration and Security Testing

- **Automated Security Testing:** Integrate robust security testing tools into the development pipeline for automated assessments.
- **Vulnerability Identification:** Conduct frequent security assessments to pinpoint vulnerabilities and address them promptly.

3. Deployment and Configuration Security

- **Secure Deployment Practices:** Implement secure deployment protocols and robust configuration management practices.
- **Infrastructure as Code (IaC):** Employ IaC principles for consistent, secure deployments across environments.

4. Monitoring and Incident Response

- **Real-time Monitoring:** Establish vigilant monitoring mechanisms to detect security events and anomalies promptly.
- **Incident Response Protocols:** Define clear incident response procedures and conduct post-incident analyses for continual enhancement.

Real-World Scenario: DevSecOps Lifecycle in Action

Imagine a scenario where a tech-savvy software development team embraces the DevSecOps lifecycle for a new application launch. By weaving security controls into the planning phase, rigorously testing for vulnerabilities during development, and orchestrating robust monitoring post-deployment, the team successfully fortifies the application against potential threats, ensuring a robust security posture throughout the lifecycle.

Conclusion: Embracing the Essence of DevSecOps Lifecycle

In embracing the DevSecOps lifecycle, organizations open the gateway to enhanced security resilience and optimized software development practices. By championing collaboration, automation, and persistent improvement imbibed in the DevSecOps lifecycle, organizations cultivate a security-first mindset that shields their digital assets against evolving threats. Witness the transformative power of DevSecOps as it reshapes security paradigms and propels organizations toward a secure digital future.

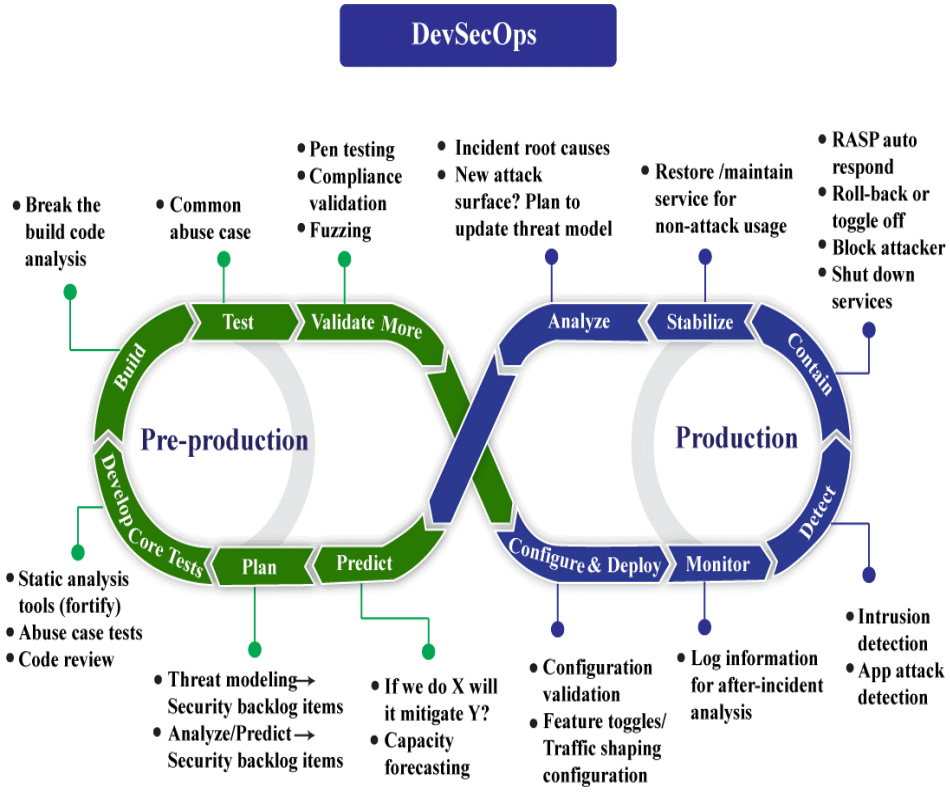

figure:-DevSecOps lifecycle

4. How Dose DevSecOps Works?

DevSecOps operates by embedding security practices into the entire development pipeline, leveraging automation and collaboration to streamline security processes. It involves implementing security controls, integrating security testing tools into CI/CD pipelines, and fostering a shared responsibility for security among development, security, and operations teams.

DevSecOps works by embedding security into every aspect of the software development process, from design to deployment. It involves implementing security controls and automation tools, fostering a culture of shared responsibility, and integrating security testing into the CI/CD pipeline.

DevSecOps works by integrating security practices and principles into the entire software development lifecycle, from planning and coding to testing, deployment, and operations. Here is a detailed explanation of how DevSecOps functions:

1. Shift Left Approach: DevSecOps emphasizes the "shift left" approach, which means addressing security concerns early in the software development process. By incorporating security practices at the beginning of the development cycle, teams can identify and mitigate security vulnerabilities before they become more costly and time-consuming to fix later in the process.

2. Automation: Automation plays a crucial role in DevSecOps by streamlining security processes and ensuring consistency across development, testing, and deployment stages. Automated security tools can help scan code for vulnerabilities, enforce security policies, monitor infrastructure for potential threats, and facilitate rapid response to security incidents.

3. Collaboration and Communication: DevSecOps promotes collaboration and communication among cross-functional teams, including developers, security professionals, operations engineers, and other stakeholders. By fostering a culture of shared responsibility and transparency, teams can work together to address security issues effectively and proactively.

4. Continuous Integration/Continuous Deployment (CI/CD): DevSecOps leverages CI/CD pipelines to automate the building, testing, and deployment of software applications. Security checks and controls are integrated into these pipelines to ensure that code changes are thoroughly vetted for security vulnerabilities before being deployed to production environments.

5. Security as Code: DevSecOps advocates for treating security configurations, policies, and controls as code that can be version-controlled, tested, and deployed alongside application code. This approach enables teams to manage security configurations programmatically and ensure consistency across environments.

6. Monitoring and Incident Response: DevSecOps emphasizes continuous monitoring of applications and infrastructure for security threats and anomalies. Teams use monitoring tools to detect suspicious activity, respond to security incidents promptly, and implement remediation measures to mitigate risks.

Overall, DevSecOps aims to create a culture of security awareness, collaboration, and automation within organizations to build secure, resilient software applications. By integrating security practices into the DevOps workflow, teams can deliver high-quality software that meets stringent security requirements while maintaining agility and efficiency in the development process.

5. well-known DevSecOps tools

There are several well-known DevSecOps tools that organizations can leverage to integrate security practices into their DevOps pipeline. DevSecOps tools that organizations can use to enhance security practices within their DevOps pipeline. By leveraging these tools, teams can automate security testing, identify vulnerabilities early in the development process, and build more secure and resilient applications.
Some popular DevSecOps tools include:

1. SonarQube: SonarQube is a static code analysis tool that helps identify code quality and security vulnerabilities in the early stages of development. It provides developers with feedback on code issues, security vulnerabilities, and compliance violations, enabling them to address these issues before code is deployed.

2. OWASP ZAP (Zed Attack Proxy): OWASP ZAP is an open-source web application security testing tool that helps developers identify and fix security vulnerabilities in web applications. It can be integrated into CI/CD pipelines to automate security testing and ensure that applications are free from common security flaws.

3. Docker Security Scanning: Docker Security Scanning is a tool that scans Docker container images for known vulnerabilities and security issues. It helps organizations identify and remediate security risks in containerized applications before they are deployed in production environments.

4. HashiCorp Vault: HashiCorp Vault is a secrets management tool that helps organizations securely store, manage, and distribute sensitive information such as passwords, API keys, and encryption keys. By integrating Vault into the DevOps pipeline, teams can ensure that sensitive data is protected and accessed securely.

5. GitLab Secure: GitLab Secure is a suite of security tools integrated into the GitLab CI/CD platform, including static application security testing (SAST), dynamic application security testing (DAST), dependency scanning, and container scanning. It

helps organizations automate security testing and ensure that applications are secure throughout the development lifecycle.

6. Veracode: Veracode is a cloud-based application security testing platform that provides static, dynamic, and software composition analysis to help organizations identify and remediate security vulnerabilities in their applications. It integrates with CI/CD pipelines to automate security testing and ensure that applications meet security standards.

6. Bebefits of DevSecOps

- DevSecOps offers several benefits, including improved software quality, faster time-to-market, reduced security risks, enhanced collaboration between teams, and increased overall efficiency and productivity. By integrating security into the development process, organizations can proactively identify and mitigate security threats, thus safeguarding their systems and data

➢ Proactively find and fix vulnerabilities

Unlike traditional approaches where security is often left to the end, DevSecOps shifts security to earlier in the software development lifecycle. By reviewing, scanning, and testing code for security issues throughout the development process, teams can identify security concerns proactively and address them immediately, before additional dependencies are introduced or code is released to customers.

➢ Release more secure software, faster

If security vulnerabilities aren't detected until the end of a project, the result can be major delays as development teams scramble to address the issues at the last minute. But with a DevSecOps approach, developers can remediate vulnerabilities while they're coding, which teaches secure code writing and reduces back and forth during security reviews. Not only does this help organizations release software faster, it ensures that their software is more secure and cost efficient.

➢ Keep pace with modern development methods

Customers and business stakeholders demand software that is fast, reliable, and secure. To keep up, development teams need to leverage the latest in collaborative and security technology, including automated security testing, continuous integration and continuous delivery (CI/CD), and vulnerability patching. DevSecOps is all about improving collaboration between development, security, and operations teams to improve organizational efficiency and free up teams to focus on work that drives value for the business.

- **Improved Software Security:** By integrating security practices into every stage of the development lifecycle, DevSecOps helps identify and mitigate security vulnerabilities early in the process. This proactive approach reduces the likelihood of security breaches and ensures that software is built with security in mind from the outset.

- **Faster Time-to-Market:** DevSecOps streamlines development and deployment processes through automation and collaboration, resulting in shorter development cycles and faster delivery of software updates. By removing bottlenecks and reducing manual intervention, organizations can accelerate their time-to-market without sacrificing security.

- **Reduced Security Risks:** Continuous security testing and monitoring in DevSecOps enable organizations to detect and remediate security threats in real-time. This proactive stance minimizes exposure to security risks and helps organizations stay ahead of evolving threats, enhancing overall security posture.

- **Enhanced Collaboration:** DevSecOps promotes collaboration between development, security, and operations teams by breaking down silos and fostering a shared responsibility for security. Collaboration ensures that security considerations are integrated into every aspect of the development process, leading to more robust and secure software.
- **Cost Savings:** By identifying and addressing security vulnerabilities early in the development lifecycle, DevSecOps helps organizations avoid the costly repercussions of security breaches, such as data loss, downtime, and regulatory

fines. Additionally, automation reduces manual effort and increases efficiency, resulting in cost savings over time.

- **Compliance Assurance:** DevSecOps facilitates compliance with regulatory requirements and industry standards by incorporating security controls and best practices into the development process. By ensuring that security measures are consistently applied and documented, organizations can demonstrate compliance more effectively.
- **Continuous Improvement:** DevSecOps encourages a culture of continuous improvement by promoting feedback loops, monitoring, and metrics-driven decision-making. This iterative approach enables organizations to identify areas for optimization and refinement, leading to ongoing enhancements in software quality and security.

7. About local and international DevSecOps career opportunities, career path:

Overall, DevSecOps offers numerous benefits, including improved software security, faster time-to-market, reduced security risks, enhanced collaboration, cost savings, compliance assurance, and continuous improvement. By integrating security into the development process and embracing automation and collaboration, organizations can build and deploy software more securely, efficiently, and effectively.

DevSecOps offers a wealth of career opportunities both locally and internationally, as organizations worldwide recognize the importance of integrating security into the software development lifecycle. Here's an overview of DevSecOps career opportunities and potential career paths:

Local DevSecOps Career Opportunities:

Local DevSecOps career opportunities refer to job openings and positions available within a specific geographic location or region. These roles involve implementing security practices within the software development lifecycle, collaborating with cross-functional teams, and leveraging automation tools to enhance security in local organizations.it includes:-

1. DevSecOps Engineer: DevSecOps engineers are responsible for implementing security practices and tools into the development pipeline. They collaborate with development, security, and operations teams to ensure that security is integrated throughout the software development lifecycle.

2. Security Automation Specialist: Security automation specialists focus on automating security processes and tasks, such as vulnerability scanning,

configuration management, and compliance checks. They leverage automation tools and scripting languages to streamline security operations.

3. Security Architect: Security architects design and implement secure software architectures and infrastructure. They assess security requirements, define security controls, and architect solutions that mitigate risks and comply with security standards and regulations.

4. DevSecOps Consultant: DevSecOps consultants provide advisory services to organizations seeking to implement DevSecOps practices. They assess existing processes, recommend improvements, and assist with the adoption of DevSecOps tools and methodologies.

International DevSecOps Career Opportunities:

International DevSecOps career opportunities encompass job opportunities that extend beyond a specific geographical region and may involve working with global teams, clients, or projects. Professionals in international DevSecOps roles may have the opportunity to work on diverse projects, collaborate with teams from different countries, and gain exposure to a wide range of security challenges and best practices on a global scale.

1. Global Corporations: Multinational corporations with a global presence often have extensive DevSecOps initiatives spanning multiple regions. Opportunities may include positions in corporate headquarters, regional offices, or remote roles with cross-border collaboration.

2. Tech Giants: Leading technology companies prioritize security and often have dedicated DevSecOps teams focused on securing their products and services. Career opportunities may include positions at companies like Google, Amazon, Microsoft, and Facebook.

3. Consulting Firms: International consulting firms provide DevSecOps services to clients across various industries. DevSecOps consultants may work on projects for clients around the world, offering expertise in implementing and optimizing DevSecOps practices.

4. Remote Work Opportunities: The nature of DevSecOps work, which often involves collaboration with distributed teams and reliance on cloud-based tools, lends itself well to remote work opportunities. Remote DevSecOps roles allow professionals to work from anywhere in the world while contributing to global projects.

Career Path in DevSecOps

A career path in DevSecOps typically involves progressing through different roles and responsibilities within the field of software development, security, and operations. Here is a general outline of a potential career path in DevSecOps:

1. Entry-Level Positions: Entry-level roles in DevSecOps may include positions such as junior DevSecOps engineer, security analyst, or security operations center (SOC) analyst. These roles typically involve foundational tasks in security monitoring, incident response, and tool implementation.

2. Mid-Level Positions: Mid-level DevSecOps roles may include positions like DevSecOps engineer, security automation specialist, or security architect. Professionals in these roles are responsible for designing and implementing security solutions, automating security processes, and collaborating with cross-functional teams.

3. Senior-Level Positions: Senior-level DevSecOps roles may include positions such as senior DevSecOps engineer, security architect, or DevSecOps manager/director. Professionals at this level lead strategic initiatives, define security policies and standards, and mentor junior team members.

4. Specialized Roles: In addition to traditional DevSecOps roles, professionals may specialize in specific areas such as cloud security, container security, infrastructure as code (IaC) security, or compliance. Specialized certifications and advanced training can help professionals differentiate themselves in the job market.

5. DevSecOps Engineer:

- Responsibilities: Implementing security controls throughout the software development lifecycle, automating security testing, and collaborating with development and operations teams.
- Skills: Proficiency in security tools and practices, experience with CI/CD pipelines, scripting languages (e.g., Python, Bash), and understanding of cloud platforms..

7. DevSecOps Architect:

- Responsibilities: Developing security strategies, designing secure systems, evaluating new technologies, and providing guidance on security architecture.
- Skills: Extensive experience in DevSecOps practices, knowledge of compliance standards (e.g., GDPR, PCI DSS), expertise in cloud security, and ability to design scalable and secure solutions.

8. DevSecOps Manager/Director:

- Responsibilities: Leading a team of DevSecOps professionals, setting strategic direction for security initiatives, managing budgets and resources, and collaborating with senior leadership.

- Skills: Strong leadership and project management skills, ability to align security goals with business objectives, experience in building and scaling security programs.

9. Chief Information Security Officer (CISO):

- **Responsibilities:** Overseeing the organization's overall security strategy, managing cybersecurity risks, ensuring compliance with regulations, and representing security concerns at the executive level.
- **Skills:** Extensive experience in cyber security, deep understanding of business operations, strong leadership and communication skills, ability to make strategic decisions.

It's important to note that the career path in DevSecOps can vary depending on individual interests, organizational needs, and industry trends. Continuous learning, staying updated on emerging technologies and security threats, and gaining practical experience through hands-on projects are essential for advancing in this field. Additionally, obtaining relevant certifications such as Certified Information Systems Security Professional (CISSP), Certified Ethical Hacker (CEH), or AWS Certified Security Specialty can also enhance career prospects in DevSecOps.

Overall, DevSecOps offers a diverse range of career opportunities and a clear career path for individuals interested in pursuing roles that combine software development, security, and operations expertise. With the increasing demand for DevSecOps professionals globally, there are ample opportunities for career growth and advancement in this dynamic field.

SUMMARY

Software Engineering Problems and the Emergence of DevSecOps: Historically, software security was often an afterthought, addressed at the project level with code scanning, penetration testing, and reactive incident response. However, as software development practices evolved, the need for integrating security into the entire software development lifecycle (SDLC) became evident. DevSecOps emerged as a solution to this challenge, aiming to embed security practices throughout the development process, from inception to deployment. [1].

What Is DevSecOps?: DevSecOps stands for **development, security, and operations**. It's a framework that integrates security into all phases of the SDLC. Unlike the traditional approach of adding security as an afterthought, DevSecOps prioritizes security from the very start. It emphasizes collaboration, automation, and clear processes to ensure that security is everyone's responsibility, rather than a separate silo.

DevSecOps Lifecycle: The DevSecOps lifecycle involves several key phases:

- **Security by Design**: Embedding security principles at the core of software development processes.
- **Collaborative Synergy**: Nurturing collaboration among diverse teams for enhanced security posture.
- **Risk Mitigation Strategies**: Proactively identifying and mitigating security risks during the lifecycle.

How Does DevSecOps Work?: DevSecOps works by automating security integration throughout the SDLC. Key components include:

- **Continuous Integration & Continuous Deployment (CI/CD)**: These solutions automate application build, testing, and deployment, ensuring security checks are seamlessly integrated.
- **Shift Left Security**: Teams discuss security implications during planning and test for security issues early in development environments.

Well-Known DevSecOps Tools: Here are some notable tools across different categories:

- **Jenkins**: An open-source automation server for CI/CD.
- **Veracode**: A cloud-based security tool for comprehensive visibility and vulnerability detection.
- **GitLab**: Provides comprehensive CI/CD in a single application.
- **Edge Delta**: Offers real-time observability and analytics.
- **Datadog**: Monitors full-stack applications.

Benefits of DevSecOps:

- **Rapid, Cost-Effective Software Delivery**: DevSecOps minimizes delays caused by security issues, resulting in efficient and cost-effective software delivery.
- **Improved, Proactive Security**: By integrating security from the beginning, DevSecOps reduces the risk of deploying vulnerable code.

DevSecOps Career Opportunities:

- **Local and International Opportunities**: DevSecOps professionals are in demand globally. Organizations across industries seek experts who can bridge development, security, and operations.
- **Career Path**: Start as a DevSecOps engineer, then specialize in areas like cloud security, automation, or risk management. Progress to roles such as DevSecOps architect, security consultant, or team lead.

Conclusion

In conclusion, the initiation of DevSecOps was driven by various software engineering problems, such as siloed security practices, inadequate collaboration between development and security teams, lack of focus on security in earlier stages of software development, and the need for continuous security integration. DevSecOps, a methodology that integrates security practices into the software development lifecycle, aims to address these challenges by fostering collaboration, automation, and a shared responsibility for security.The DevSecOps lifecycle involves incorporating security practices from planning to operations, emphasizing early security considerations, automation of security processes, collaboration among cross-functional teams, and treating security as code. This approach enhances security resilience, streamlines development processes, and ensures a more robust security posture throughout the software lifecycle .DevSecOps operates by embedding security practices throughout the development pipeline, leveraging automation, collaboration, and continuous improvement strategies. By integrating security controls, utilizing security testing tools in CI/CD pipelines, and promoting shared responsibility for security, organizations can strengthen their security defenses and respond effectively to security incidents.

REFERENCE

- www.Microsoftoffice.com
- WWW.SPRINGBOADRD.COM/BLOG/SOFTWARE-ENGINEERING/WHAT-IS-DEVSECOPS
- WWW.ATLASSIAN.COM/DEVOPS-TOOLS/DEVSECOPS-TOOLS
- WWW.PRACTICAL-DEVSECOPS.COM/DEVSECOPS-LIFE-CYCLE/
- WWW.IBM.COM/TOPICS/DEVSECOPS
- https://about.gitlab.com/topics/devsecops/