Institute of Technology

School of Computing

Department of Software Engineering

Software Engineering Tools and Practices
Individual Assignment

Name: YOSEF TEFERA

ID: 147613

Table of Contents

introduction

DevSecOps is short for development, security and operations. It is a software development model in which these three teams work in close collaboration and in a synchronized fashion. One may say that a DevSecOps team is an agile, cross-functional DevOps team that embeds security practices into their own processes to deliver secure software products and digital services. In other words, DevSecOps is DevOps done securely.

The intent of DevSecOps is to make everyone accountable for security while still operating at the same speed and scale as DevOps development CI/CD pipelines. Adding application security to DevOps is a major challenge because security practices are becoming a "bottleneck" for software development assembly line. However, as cyber threats continue to grow, secure software development processes have never been more important.

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

Software engineering problems that caused the initiation of DevSecOps:

- Lack of collaboration between development and security teams:_This led to security vulnerabilities being introduced into software products due to a lack of communication and coordination between the two teams.
- Manual and time-consuming security testing: Traditional security testing methods were often manual and time-consuming, which slowed down the software development process and made it difficult to keep up with the pace of development.
- Lack of automation in security processes: Security processes were often not automated, which made it difficult to ensure consistency and quality in security testing and remediation.
- Lack of visibility into security risks:_Development teams often lacked visibility into the security risks associated with their code, which made it difficult to prioritize and address security vulnerabilities.
- Lack of accountability for security: There was often a lack of clear accountability for security within software development teams, which made it difficult to identify and address security issues.

DevSecOps was initiated to address these problems by bringing together development and security teams into a collaborative environment, automating security processes, and providing development teams with visibility into security risks. By doing this, DevSecOps helps to improve the security of software products and reduce the time and effort required to develop and deploy secure software.

2 .What is DevSecOps?

DevSecOps is a software development approach that combines development (Dev), security (Sec), and operations (Ops) into a single, collaborative team. The goal of DevSecOps is to improve the security and quality of software products while also increasing the speed and efficiency of the software development process.

DevSecOps principles:

- Collaboration: Development, security, and operations teams work together throughout the software development lifecycle, from planning and design to deployment and maintenance.
- Automation: Security processes and tools are automated as much as possible to reduce the time and effort required to secure software products.
- Continuous feedback: Security feedback is provided to development teams on a continuous basis, so that security issues can be identified and addressed early in the development process.

By following these principles, DevSecOps helps to improve the security of software products, reduce the time and effort required to develop and deploy secure software, and increase the overall efficiency of the software development process.

Key practices of DevSecOps:

- Security testing: Automated security testing tools are used to identify security vulnerabilities in software products.
- Code review: Security experts review code to identify potential security vulnerabilities.
- Threat modeling: Security experts identify and assess potential threats to software products.
- Security training: Development teams are trained on secure coding practices and security best practices.
- Continuous monitoring: Software products are continuously monitored for security vulnerabilities and threats.

DevSecOps is a powerful approach to software development that can help organizations to improve the security and quality of their software products while also increasing the speed and efficiency of the software development process.

3.Briefly explain DevSecOps lifecycle?

The DevSecOps lifecycle is a continuous process that integrates security into every phase of the software development lifecycle (SDLC). It consists of the following steps:

1. Planning and design:

• Security requirements are defined and integrated into the software design.

• Security controls are identified and implemented.

• Threat modeling is performed to identify potential security risks.

2. Development:

• Secure coding practices are followed.

• Automated security testing tools are used to identify security vulnerabilities.

• Code is reviewed for security vulnerabilities.

3. Integration and testing:

• Security testing is performed on integrated code.

• System and integration tests are performed to ensure that the software meets security requirements.

4. Deployment:

• Software is deployed in a secure environment.

• Security configurations are applied to the deployment environment.

• Security monitoring is implemented to detect and respond to security threats.

5. Operations and maintenance:

• Software is continuously monitored for security vulnerabilities and threats.

• Security patches and updates are applied as needed.

• Security incident response procedures are followed in the event of a security breach.

The DevSecOps lifecycle is a continuous process that should be repeated with each iteration of the SDLC. By following the DevSecOps lifecycle, organizations can improve the security of their software products and reduce the risk of security breaches.

4. How dose DevSecOps works?

DevSecOps works by integrating security into every phase of the software development lifecycle (SDLC), from planning and design to deployment and maintenance. This is achieved through a combination of cultural changes, process changes, and technology.

Cultural changes:

- Collaboration: Development, security, and operations teams work together throughout the SDLC.
- Communication: Teams communicate openly and frequently about security risks and mitigation strategies.
- Trust: Teams trust each other to do their part to secure the software product.

Process changes:

- Security requirements are defined and integrated into the software design.
- Security testing is performed throughout the SDLC.
- Code is reviewed for security vulnerabilities.
- Security configurations are applied to the deployment environment.
- Security monitoring is implemented to detect and respond to security threats.

Technology:

- Automated security testing tools: These tools help to identify security vulnerabilities in software products.
- Code review tools: These tools help to identify potential security vulnerabilities in code.
- Threat modeling tools: These tools help to identify and assess potential threats to software products.
- Security monitoring tools: These tools help to detect and respond to security vulnerabilities and threats.

By combining these cultural, process, and technology changes, DevSecOps helps organizations to improve the security of their software products while also increasing the speed and efficiency of the software development process.

Here is a simplified example of how DevSecOps works in practice:

1. A development team is working on a new software product.
2. The security team reviews the software design and identifies potential security risks.
3. The development team implements security controls to mitigate the risks identified by the security team.
4. Automated security testing tools are used to identify security vulnerabilities in the code.
5. Code is reviewed for security vulnerabilities by both the development team and the security team.
6. The software is deployed in a secure environment and security configurations are applied.
7. The software is continuously monitored for security vulnerabilities and threats.
8. In the event of a security breach, the incident response team is notified and takes steps to mitigate the breach and prevent further damage.

By following the DevSecOps lifecycle and using the appropriate cultural, process, and technology changes, organizations can improve the security of their software products and reduce the risk of security breaches.

5. Explain well known DevSecOps tools.

Automated security testing tools:

• OWASP ZAP: A free and open-source web application security scanner.

• Nessus: A commercial vulnerability scanner.

• Burp Suite: A commercial web application security testing suite.

• Fortify: A commercial static application security testing (SAST) tool.

Code review tools:

• SonarQube: A free and open-source code quality and security analysis tool.

CodeClimate: A commercial code review tool.

• Review Board: A free and open-source code review tool.

• Veracode: A commercial SAST tool that includes code review capabilities.

• Coverity: A commercial SAST tool that includes code review capabilities.

Threat modeling tools:

• Microsoft Threat Modeling Tool: A free and open-source threat modeling tool.

• OWASP Threat Dragon: A free and open-source threat modeling tool.

• ThreatModeler: A commercial threat modeling tool.

• iThreat: A commercial threat modeling tool.

• SecurITree: A commercial threat modeling tool.

Security monitoring tools:

• Splunk: A commercial security information and event management (SIEM) tool.

• Elasticsearch: A free and open-source SIEM tool.

• Logstash: A free and open-source log aggregation tool.

• Kibana: A free and open-source data visualization tool that can be used with Elasticsearch and Logstash.

• Grafana: A free and open-source data visualization tool that can be used with a variety of data sources, including security monitoring tools.

Other DevSecOps tools:

• Jenkins: A free and open-source continuous integration (CI) tool.

• Kubernetes: A free and open-source container orchestration tool.

• Docker: A free and open-source containerization platform.

• Ansible: A free and open-source IT automation tool.

Terraform: A commercial infrastructure-as-code (IaC) tool.

These are just a few of the many DevSecOps tools available. The specific tools that an organization chooses will depend on its specific needs and requirements.

6. What are the benefits of DevSecOps?

Benefits of DevSecOps:

- Improved security: By integrating security into every phase of the software development lifecycle (SDLC), DevSecOps helps organizations to identify and address security vulnerabilities early in the development process, before they can be exploited by attackers.

- Reduced time and effort: DevSecOps reduces the time and effort required to develop and deploy secure software by automating security processes and providing development teams with continuous feedback on security issues. This helps to speed up the software development process and reduce the cost of developing secure software.

- Increased efficiency: DevSecOps increases the overall efficiency of the software development process by bringing together development, security, and operations teams into a single, collaborative team. This helps to eliminate silos and improve communication between teams, which leads to a more efficient and effective software development process.

- Improved compliance: DevSecOps helps organizations to comply with security regulations and standards by providing a framework for implementing and maintaining security controls throughout the SDLC.

- Increased customer satisfaction: By delivering more secure software products, DevSecOps helps organizations to increase customer satisfaction and loyalty.

In addition to these benefits, DevSecOps can also help organizations to:

- Reduce the risk of security breaches: By identifying and addressing security vulnerabilities early in the development process, DevSecOps helps organizations to reduce the risk of security breaches.

Improve the quality of software products: By integrating security into every phase of the SDLC, DevSecOps helps organizations to improve the quality of their software products.

- Increase innovation: By reducing the time and effort required to develop and deploy secure software, DevSecOps helps organizations to increase innovation.

Overall, DevSecOps is a powerful approach to software development that can help organizations to improve the security, quality, and efficiency of their software development process.

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

Local and international DevSecOps career opportunities

DevSecOps is a rapidly growing field, with a high demand for skilled professionals. There are many DevSecOps career opportunities available, both locally and internationally.

Local DevSecOps career opportunities

In many countries, there is a growing demand for DevSecOps professionals. This is due to the increasing adoption of DevSecOps practices by organizations of all sizes. Local DevSecOps career opportunities can be found in a variety of industries, including:

• Technology

• Finance

• Healthcare

• Government

• Retail

International DevSecOps career opportunities

There is also a strong demand for DevSecOps professionals internationally. This is due to the global nature of the software development industry. International DevSecOps career opportunities can be found in a variety of countries, including:

- United states
- United Kingdom
- Canada

- Australia
- New Zealand
- Germany
- France
- Japan
- China

DevSecOps career path

The DevSecOps career path is typically divided into three levels:

- Junior DevSecOps Engineer: Junior DevSecOps engineers are responsible for implementing and maintaining security controls throughout the SDLC. They work closely with development and operations teams to ensure that security is integrated into every phase of the software development process.
- DevSecOps Engineer: DevSecOps engineers are responsible for leading and managing DevSecOps initiatives within their organizations. They work with senior management to define and implement DevSecOps strategies. They also work with development and operations teams to implement and maintain security controls throughout the SDLC.
- Senior DevSecOps Engineer: Senior DevSecOps engineers are responsible for providing strategic guidance on DevSecOps initiatives within their organizations. They work with senior management to develop and implement DevSecOps strategies. They also work with development and operations teams to implement and maintain security controls throughout the SDLC.

How to get started in a DevSecOps career

There are a few things you can do to get started in a DevSecOps career:

Get certified: There are a number of DevSecOps certifications available, such as the Certified DevSecOps Engineer (CDSE) certification from the DevOps Institute. Getting certified can help you to demonstrate your skills and knowledge to potential employers.

- Build a portfolio: Showcase your DevSecOps skills and experience by building a portfolio of projects. This could include writing blog posts, giving presentations, or contributing to open source projects.

- Network with other DevSecOps professionals: Attend industry events and meetups to network with other DevSecOps professionals. This can help you to learn about new trends and technologies, and to find potential job opportunities.

Conclusion

DevSecOps is a software development approach that integrates security practices into the DevOps process. It aims to ensure that security is prioritized throughout the software development lifecycle, rather than being an afterthought. By integrating security early on, DevSecOps helps organizations to build secure and resilient applications.

In conclusion, DevSecOps is critical for modern software development. It promotes collaboration between development, operations, and security teams, enabling faster and more secure delivery of software. By incorporating security practices from the beginning, organizations can proactively identify and address vulnerabilities and reduce the risk of security breaches. DevSecOps also emphasizes automation and continuous monitoring, allowing for quick detection and response to potential security issues.

Ultimately, DevSecOps fosters a culture of shared responsibility, where security becomes everyone's concern. It helps organizations to align security objectives with business goals, leading to improved efficiency, reduced costs, and increased customer trust. Embracing DevSecOps is crucial in today's rapidly evolving threat landscape to ensure the development and deployment of secure and reliable software systems.

References

1.https://www.redhat.com/en/topics/devops/what-is-devsecops

2.https://www.gitguardian.com/glossary/devsecops-benefits

3.https://www.practical-devsecops.com/devsecops-life-cycle/