Institute of Technology

School of Computing

Department of Software Engineering

Software Engineering Tools and Practices

Name **Id.no**

Kaleb Teshome …….. 1301696

Submitted to: Esmail M

Submitted date: 15-march-2024

Table content
page no:

Introduction

Today, most companies have implemented DevOps practices within their organization. DevOps provides a culture where teams can deliver reliable software and updates faster. This approach presents an opportunity for teams to focus on quality rather than wasting time on operations. However, as a result, security practices are often left to security specialists at the end of the delivery pipeline. A specialized security approach then creates unnecessary overhead within the delivery process as unexpected issues frequently arise at the end of delivery. Consequently, teams lose time fixing the code and starting the same process repeatedly, ultimately making delivery costly and inefficient.

DevSecOps has become increasingly important as most companies have embarked on digital transformation. With these plans, companies are moving to the cloud. This results in moving away from on-premises infrastructures and transitioning to public cloud solutions. Cloud providers offer cost-effective, scalable, highly available, and reliable solutions. However, these advantages come with new security challenges.

Security deserves a higher priority than ever before. With cloud solutions, there is no room for mistakes. By not following security requirements, you might open the door of your network to dozens of security threats. Therefore, security should be considered earlier during the design phase. Developers are more successful with this approach because they first create a secure environment before developing their features. Additionally, developers are more involved and aware of security requirements since it is now part of development.

DevSecOps integrates security into DevOps as an integral component of the SDLC instead of observing security as an afterthought. It also distributes security responsibilities amongst team members. In collaboration with security specialists, teams can implement a “security as code” culture that encourages security to be treated like other software components of the SDLC pipeline.

1. What is DevSecOps:

Security is one of the most significant aspects upon which companies concentrate much of their energies. These efforts are required as hacks, espionage, and malware continue to plague the world, and carefully developed solutions are dealt cruel blows due to these attacks. As development is threatened, companies have resorted to extreme security measures, hampering the development process. This was hardly the answer the companies were looking for, as productivity was not to be constricted in the name of protection. As people searched for answers, DevSecOps emerged as the solution.

But **what is DevSecOps?** We have heard of DevOps, where development and deployment are undertaken with the approach to optimizing the products for automation purposes. At the end of the product development, security is sowed into the product at the final stage. With DevSecOps, security is ingrained at every stage. Let's understand what DevSecOps Services is.

Development, security, and operations, often known as DevSecOps, streamline security integration at each stage of the software development lifecycle (SDLC), from basic design through integration, testing, deployment, and software delivery.

The progression of how development organizations address DevSecOps represents security. Previously, a separate security team would "tack on" security to software at the end, and an independent quality assurance (QA) team would evaluate it.

This was workable when software updates were made available once or twice a year. However, the conventional approach, where the security is bolted, created an unacceptable bottleneck as software engineers adopted Agile and DevOps approaches, hoping to cut software development cycles to weeks or even days.

Agile and DevOps techniques and tools are easily integrated with the application and infrastructure security using **DevSecOps**. When security problems arise, they are more straightforward, quicker, and less expensive to fix (and before they are put into production). DevSecOps services also transforms application and infrastructure security from being the primary duty of a security silo to being a shared responsibility of development, security, and IT operations teams. The DevSecOps process is deemed successful by automating secure software supply without delaying the SDLC.

2. Software Engineering Problems Leading to DevSecOps Initiation:

The initiation of DevSecOps was driven by several software engineering problems, including:

- **Security Vulnerabilities:**
  Traditional software development often neglects security until the later stages, leading to vulnerabilities that are costly to fix. DevSecOps aims to integrate security from the beginning to prevent such issues.
- **Slow Response to Threats:**
  Reactive security measures in traditional development can result in slow responses to emerging threats. DevSecOps promotes a proactive approach to security, enabling faster threat detection and mitigation.
- **Lack of Collaboration:**
  Siloed teams in software development can lead to miscommunication and delays in addressing security concerns. DevSecOps encourages collaboration between development, operations, and security teams to ensure a holistic approach to security.
- **Compliance Challenges:**
  Meeting regulatory requirements and industry standards can be challenging without a structured approach to security. DevSecOps helps in integrating compliance checks and security measures throughout the development lifecycle.
- **Inadequate Testing:**
  Limited security testing in traditional software development can result in undetected vulnerabilities. DevSecOps emphasizes continuous testing and automation to identify and address security issues early on.
- **Complex Infrastructure:**
  Modern software applications often rely on complex infrastructures, making it difficult to secure them effectively. DevSecOps advocates for infrastructure as code and automated security controls to manage and secure complex environments.
- **Third-Party Risks:**
  Integrating third-party components without proper security assessments can introduce vulnerabilities into the software. DevSecOps emphasizes monitoring and managing third-party risks to ensure the overall security of the application.By addressing these software engineering problems through the adoption of DevSecOps practices, organizations can enhance the security posture of their software applications and improve overall development efficiency.

3. Briefly explain DevSecOps life cycle:

DevOps follows a traditional development cycle that involves phases like Plan, Code, Build, Test, Release, Deploy, Operate, and Monitor. Whereas, in DevSecOps, some distinct security steps are integrated into each of the DevOps development phases for thorough security checks, which help organizations build and deliver increasingly secure products at an accelerated rate.

- **Threat Modeling:**
  The first phase of the DevSecOps lifecycle, thread modeling, helps the team assess an application and its surrounding environment to find as much vulnerability as possible before attackers do. By implementing threat modeling within the traditional development process, teams are able to gather a summary of possible attack scenarios, outline the sensitive data workflow, and identify vulnerabilities and potential mitigation options. Like the majority of the processes in DevSecOps, this is also implemented with the help of tools like OWASP Threat Dragon, IriusRisk, Threat Modeler, etc.
- **Scan & Analyze:**
  After the threat modeling phase, the code is analyzed in the scanning phase to ensure it is secure from security vulnerabilities. This phase involves both manual and automated code review, which helps developers to identify security vulnerabilities and bugs earlier in the software development life cycle. This phase involves the use of tools like Static Application Software Testing (SAST) and Dynamic Application Security Testing (DAST).
- **Identity:**
  After code analysis, the team reviews all the data and metrics collected from the previous phases to identify security risks. These risks are then compiled based on their severity and priority. Tools like Klocwork can be used to identify security vulnerabilities within the data and metrics collected.
- **Remediate:**
  Once all the security vulnerabilities are identified and organized in the previous phases, the team moves on to the remediation phase, where steps are taken to rectify issues. This involves the use of various SAST tools that suggest solutions for the identified vulnerabilities, errors, and bugs.This makes it easier for the team to address and rectify the security issues as they arise.
- **Monitor:**
  Though last, this is another critical phase of the DevSecOps lifecycle, where the

team is responsible for tracking all the identified vulnerabilities, the steps taken to mitigate or eliminate those vulnerabilities, and the overall status of the application's security. This allows them to make informed data-driven decisions during the software development lifecycle, which further helps them deliver quality and secure products/features to the users. Apart from tracking the aforementioned aspects, the team can also track and manage the differences between the actual and target metric values, which will allow the organization to experience advancement in operational efficiency across various departments. Though there is no concrete process for implementing DevSecOps, these steps are usually present. Depending on the complexity and size of your project, your development lifecycle might include some other sequential steps.

4. How DevSecOps does works?

DevSecOps is a methodology that integrates security practices within the DevOps process. Here's how it works:

- **Automation:**
  DevSecOps emphasizes automation of security processes throughout the software development lifecycle. Security checks and tests are integrated into the CI/CD pipeline to identify and fix vulnerabilities early in the development process.
- **Collaboration:**
  DevSecOps promotes collaboration between development, operations, and security teams. By breaking down silos and fostering communication, teams can work together to address security concerns effectively.
- **Continuous Monitoring:**
  Continuous monitoring is a key aspect of DevSecOps. Security tools are used to monitor applications and infrastructure in real-time, allowing teams to detect and respond to security threats promptly.
- **Shift Left:**
  DevSecOps encourages a "shift-left" approach to security, where security considerations are integrated into the development process from the beginning. This helps in identifying and addressing security issues early in the development lifecycle.
- **Security as Code:**
  Security policies and configurations are treated as code and stored in version

control systems. This allows for better tracking of changes, easier collaboration, and ensures that security practices are consistent across environments.

- **Compliance and Governance:**
  DevSecOps ensures that security practices align with regulatory requirements and industry standards. By integrating compliance checks into the development process, teams can ensure that applications meet security and regulatory standards.
  Overall, DevSecOps aims to make security an integral part of the software development process, ensuring that applications are secures, reliable, and compliant from the initial stages of development to production deployment.

5. Explain well-known DevSecOps tools:

Certainly! Here are some well-known DevSecOps tools that play a crucial role in integrating security practices into the software development lifecycle:

- **Jenkins:**
  An open-source automation server for continuous integration and continuous delivery (CI/CD).Streamlines build, testing, and deployment stages, ensuring seamless security checks throughout the development lifecycle1.
- **OWASP Dependency-Check:**
  A tool for identifying known vulnerabilities in project dependencies. Helps developers manage and address security issues related to third-party libraries.
- **SonarQube**:
  Provides static code analysis to detect code smells, security vulnerabilities, and maintainability issues. Integrates with CI/CD pipelines to ensure secure code quality.
- **Source Clear:**
  Scans code repositories for vulnerabilities and provides actionable insights. Helps developers prioritize and remediate security issues.
- **Retire.js:**
  Focuses on identifying outdated JavaScript libraries with known vulnerabilities. Essential for securing web applications.
- **Check Marx:**
  A robust static application security testing (SAST) tool. Identifies security flaws in the source code.

- **Snyk**:
  Scans for vulnerabilities in open-source libraries and container images. Integrates seamlessly into the development workflow.
  Remember that these tools contribute to embedding security throughout the development pipeline, making it easier to detect and resolve vulnerabilities early on.

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," describes Shannon Lietz, co-author of the "DevSecOps Manifesto."

- **Rapid, cost-effective software delivery:**
  When software is developed in a non-DevSecOps environment, security problems can lead to huge time delays. Fixing the code and security issues can be time-consuming and expensive. The rapid, secure delivery of DevSecOps saves time and reduces costs by minimizing the need to repeat a process to address security issues after the fact.
  This process becomes more efficient and cost-effective since integrated security cuts out duplicative reviews and unnecessary rebuilds, resulting in more secure code.
- **Improved, proactive security:**
  DevSecOps introduces cybersecurity processes from the beginning of the development cycle. Throughout the development cycle, the code is reviewed, audited, scanned and tested for security issues. These issues are addressed as soon as they are identified. Security problems are fixed before additional dependencies are introduced. Security issues become less expensive to fix when protective technology is identified and implemented early in the cycle. Additionally, better collaboration between development, security and operations teams improves an organization's response to incidences and problems when they occur. DevSecOps practices reduce the time to patch vulnerabilities and free up security teams to focus on higher value work. These practices also ensure and simplify compliance, saving application development projects from having to be retrofitted for security.

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

- **Automation compatible with modern development:**
  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 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 describes Shannon Lietz, co-author of the "DevSecOps Manifesto."
- **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.

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

Certainly! Let's delve into the exciting world of DevSecOps career opportunities and career paths:

- **Career Opportunities:**
  The DevSecOps industry is growing rapidly. In 2020, it was estimated to be worth $2.79 billion, and the growth rate is predicted to be 24.1% between 2021 and 20281.
  As a DevSecOps professional, you'll have the chance to work with cutting-edge technologies and contribute to streamlining development processes.Opportunities exist both locally and internationally. Locally DevSecOps career opportunities in Ethiopia are emerging as the country's tech industry grows.

- ## Career Path:

1.Experience Matters:

Employers highly value experience. Consider roles that prepare you for DevSecOps, such as:
2.**Software Developer:** Gain coding and application development experience.
Operations or Security Roles: Learn about business tools, systems, and processes used to manage and secure software applications.
3. **Certifications**:
Boost your resume with DevSecOps certifications. These validate your skills and demonstrate your commitment to security.
4. **College Degrees:** If pursuing a degree, research majors that align with your career goals.

- ## International Outlook:
  DevSecOps professionals are in demand globally. Companies worldwide seek skilled engineers to enhance their security practices.
  Explore job opportunities in various countries and regions.
  Remember, DevSecOps is not just a job—it's a mindset that integrates security into every aspect of software development.

Conclusion of DevSecOps Services

A new approach called DevSecOps integrates security into the early phases of software development. It ensures complete operation, lessens cyber dangers, and quick software product launches. Software solutions can be produced fast by implementing security at every level of the SDLC. Those who work in the automobile, healthcare, financial, or retail sectors can use these security solutions. It is a management strategy incorporating a continuous delivery cycle with security, operations, application development, and IaaS. DevSecOps Services aim to integrate security into all phases of the SDLC. Continuous integration, cost-effective compliance, and speedy software delivery are all made possible using security at every level of the SDLC. Making everyone responsible for security is its fundamental goal. For more than ten years, Veritis, the Stevie Awards winner, has been a dependable partner for businesses of all sizes, including those on the Fortune 500. We have considerable experience integrating cutting-edge

technology in a fluid environment and providing solutions for IT projects. Veritis provides a range of technological services for your company at a cost-effective solution. Get in touch with us to embrace productivity with the greatest DevSecOps tools.

REFERENCE:

WWW.ACUNETIX.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