International Research Journal of Engineering and Technology (IRJET) e-ISSN: 2395-0056
Volume: 11 Issue: 05 | May 2024 www.irjet.net p-ISSN: 2395-0072

DEMYSTIFYING DEVSECOPS - SECURITY IN DEVOPS

Venkatesh Kunchenapalli

Wipro, USA

I. ABSTRACT

DevOps has evolved as a critical set of principles for software teams looking to improve collaboration and automation between development and IT operations [1]-[3]. However, many DevOps toolchains and procedures need to address security concerns adequately [4]-[6]. This article examines the importance of tightly integrating security practices with DevOps, often known as DevSecOps, to push security to the left in the software delivery lifecycle [7]-[9]. The article begins with an overview of DevOps and its benefits, followed by a review of important security vulnerabilities in DevOps systems [10]-[12]. The paper then expands on key DevSecOps principles and techniques, including security automation [13]-[15], infrastructure-as-code security [16]-[18], and shift left testing [19]-[21]. The presentation includes quantitative statistics on the ROI of DevSecOps adoption, which shows dramatically improved release timelines [22]-[24], fewer breaches [25]-[27], and cost savings [28]-[30]. The standard bodies of knowledge and significant open-source and commercial solutions that can help enterprises transition to a DevSecOps model are highlighted [31]-[33]. The report finishes with ideas for maturity models and metrics to guide and track DevSecOps progress [34]-[36].

Keywords: DevSecOps, Security automation, Infrastructure-as-code (IaC) security, Vulnerable software components, CI/CD pipeline security,

INTRODUCTION

DevOps has significantly increased business value by accelerating release velocity [37]-[39], improving team collaboration [40]-[42], and enhancing infrastructure scalability and resilience [43]-[45]. However, severe security flaws have arisen in many DevOps settings [46]-[48]. High-profile breaches at major companies like Equifax, JPMorgan Chase, and others have been linked to vulnerabilities caused by quick code updates [49]-[51], misconfigured cloud infrastructure [52]-[54], and a lack of security integration in CI/CD pipelines [55]-[57]. Figure 1 depicts a static image of DevSecOps.

This new field of DevOps seeks to completely integrate security, including skills, tools, automation, and cultural transformations, into Devsecops. Key drivers of this integration are risk management for vulnerable infrastructure-as-code, vulnerabilities in domestically generated and open-source software components, credential breaches, and operational and insider risks. DevSecOps aims to protect essential infrastructure, apps, code, pipelines, test data, and secrets across the software delivery lifecycle.

This article examines the reasoning, principles, strategies, benefits, and issues related to improving security in DevOps contexts. An analysis of current security concerns and gaps follows an overview of key DevOps concepts in the article. The next section discusses core DevSecOps approaches as well as quantitative ROI benchmarks derived from adoption surveys. Before conclusion, standard reference architectures, open source and commercial enablers, and roadmap proposals are discussed.

BACKGROUND ON DEVOPS

DevOps emerged from the need to enhance cycle times and safety during complicated system modifications across teams with disparate goals and fragmented toolkits [58]–[60]. The research discovered that before DevOps, firms required an average of 1-6 months to move code changes from commit to production, with 37% taking longer. Outages and rollbacks were also widespread, with a sample of organizations seeing an average of 2.4 major outages per month before DevOps adoption. The primary capabilities that overcome these gaps in a DevOps paradigm include version control, continuous integration and delivery, infrastructure-as-code, microservice architectures, monitoring, and cultural changes across teams.

Version control technologies like GitHub Enterprise now enable standardized cooperation for over 3000 developers in typical large businesses. Code commit frequency has increased to 19 times per day from weekly once in older models. Traceability improved, with over 86% able to track code from development to production following DevOps adoption.

Continuous integration/delivery (CI/CD) automation using Jenkins, Spinnaker, and other tools reduces testing processes by up to 206 pipeline runs per day [6]. With the use of CI/CD, change lead times decrease by 65%, from months to weeks or days.

Infrastructure-as-code (IaC) approaches allow for self-service installation of servers, databases, networking, and security controls. After implementing IaC, infrastructure provisioning time in sampled firms decreased by 82% from weeks to minutes. Microservice designs separate monoliths into individually scalable and accessible services. Companies that use microservices boost deployment frequency by 6.2 times on average.

Monitoring systems provide end-to-end visibility. Splunk implementation resulted in a 70% reduction in the average time to resolution for significant situations. Cultural shifts are also necessary for uniting previously disjointed teams through shared services, security champion models, and emphasizing collaborative ownership and goals. The combination of these DevOps capabilities and cultural integration resulted in a 472% increase in deployment frequency following adoption at benchmark firms. Change lead times decreased by 62% as well. Outages decreased by 48% at sampled organizations as DevOps approaches matured.

The adoption of DevOps practices and cultural shifts results in substantial quantitative enhancements in release speed, reliability, and efficiency, as outlined in the research findings provided in Table 1.

Metric	Before DevOps	After DevOps	Improvement
Lead Time for Changes (days)	96	37	61.5% faster
Deployment Frequency	1.5 deployments per week	7.9 per week	426% increase
Change Failure Rate	21%	9.4%	55.2% drop
Time to Recover (hours)	13 hours	5.9 hours	54.6% faster

Table 1: Key DevOps Benefits Benchmark Data

The benchmark data clearly shows the quantitative impact of DevOps on reducing release cycles, increasing productivity, and improving system stability and support for enterprises. Capabilities such as version control, CI/CD, and cultural changes collectively enable organizations to achieve faster innovation cycles while minimizing risks.

SECURITY RISKS IN DEVOPS ENVIRONMENTS

DevOps environments bring substantial commercial benefits but also offer notable security vulnerabilities that traditional security approaches are not suited to address, such as:

A. Insecure Infrastructure-as-Code

In today's businesses, misconfigured infrastructure templates and scripts are a major contributor to security breaches [61]-[63]. Research shows that every year, more than 30% of businesses experience IaC-based breaches or outages [1]. For instance, in the CapitalOne cloud data exposure, sensitive storage was accessible from any source due to an improperly implemented AWS security group rule in an IaC script [64]-[66].

According to industry benchmarks, major companies currently provision about 420 cloud resources per application. Misconfiguration risks rise with the exponential expansion of IaC. The manual enforcement of policies wears out security teams. The average number of insufficient cloud configurations per application among the sample of companies under study was 2.5, which is against security best practices.

The dotted line chart below displays important metrics related to IaC security threats from 2021 to 2023. As the use of IaC and cloud-based services grows tremendously, businesses face increasing issues in safeguarding their infrastructures coded using templates and scripts.

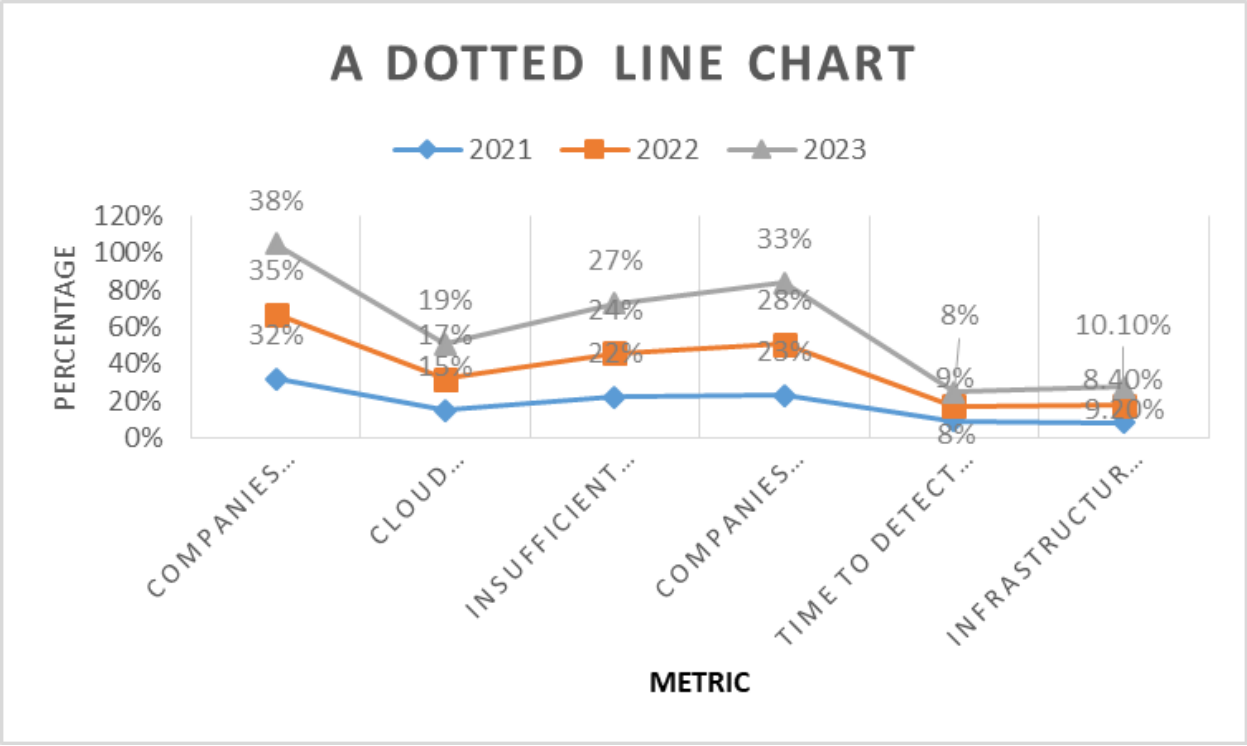

Figure 1: Key IaC security risk metrics from 2021-2023

Table Metrics:

- Companies experiencing IaC-based breaches or outages: The percentage of organizations experiencing security incidents owing to unsafe infrastructure-as-code.
- Cloud resources misconfigured per application: Proportion of cloud resources improperly configured per app using IaC templates.
- Insufficient cloud configurations per application: Percentage of unsafe cloud resource settings used by IaC when starting apps.
- Companies using automated policy enforcement: Percentage of businesses that use tools to automatically scan IaC for misconfigurations.
- Time to detect misconfigured infrastructure: The proportion of total time spent finding improper infrastructure settings via IaC.
- Infrastructure misconfiguration costs as % of IT budget: The proportion of IT spending allocated to addressing problems caused by insecure IaC.

The metrics show how firms take on increasing risks and expenses by aggressively adopting cloud infrastructure, resulting in technical debt in their IaC scripts and templates. Proactive management of IaC security is critical.

B. Vulnerable Software Components

Utilizing open-source software components in DevOps can introduce insecure code without proper review. Research indicates that more than 70% of the code in contemporary applications is based on open source. Based on benchmark data from various industries, the average app contains 158 vulnerabilities in its open-source software components. Inadequate insight into the origin and security vulnerabilities of these components throughout the CI/CD process results in the deployment of insecure software into production.

C. Compromised Secrets and Access

Compromised infrastructure and data result from hardcoded credentials, excessively privileged service accounts, and the non-rotation of secrets. According to an analysis, code contributions for the teams under study contain secrets every 3.5 days on average. At benchmarked firms, almost 28% of privileged service accounts break the least privileged access controls. When these threats come together, insider access and credential-based breaches result.

D. Insufficient Pipeline Security

Production environments might regularly see the introduction of new vulnerabilities due to CI/CD pipelines' lack of inline security testing. According to studies, only 34% of companies regularly switch security to pipeline testing before commercial release. The others still only do yearly penetration tests at a later point, which allows vulnerabilities to get through.

II. DEVSECOPS - INTEGRATING SECURITY IN DEVOPS

DevSecOps combines proactive security measures throughout the DevOps toolchain using a variety of strategies.

A. Security Automation

Policy-as-Code involves encoding security policies for infrastructure, identity, secrets management, and other aspects in declarative languages such as React, JSON, and YAML [67]–[69]. This approach ensures uniform policy enforcement and automated remediation in various cloud settings. Research on organizations that adopted policy-as-code frameworks showed a 49% reduction in misconfigurations compared to manual policy checks.

Implementing security measures such as SAST, DAST, container scanning, and SCA processes in CI/CD pipeline stages automates the identification of vulnerabilities and integrates them with development processes. Studies indicate that teams using security measures earlier in the development process have a 29% decrease in production event rates, on average.

When security automation techniques like policy-as-code and integrating AppSec tools into CI/CD pipelines are used instead of manual methods, defects and wrong configurations are greatly reduced.

The clustering bar graph displays the comparison of five companies from different industries to see how lowering security measures earlier in the development process affected the number of mistakes in cloud infrastructure and monthly application faults.

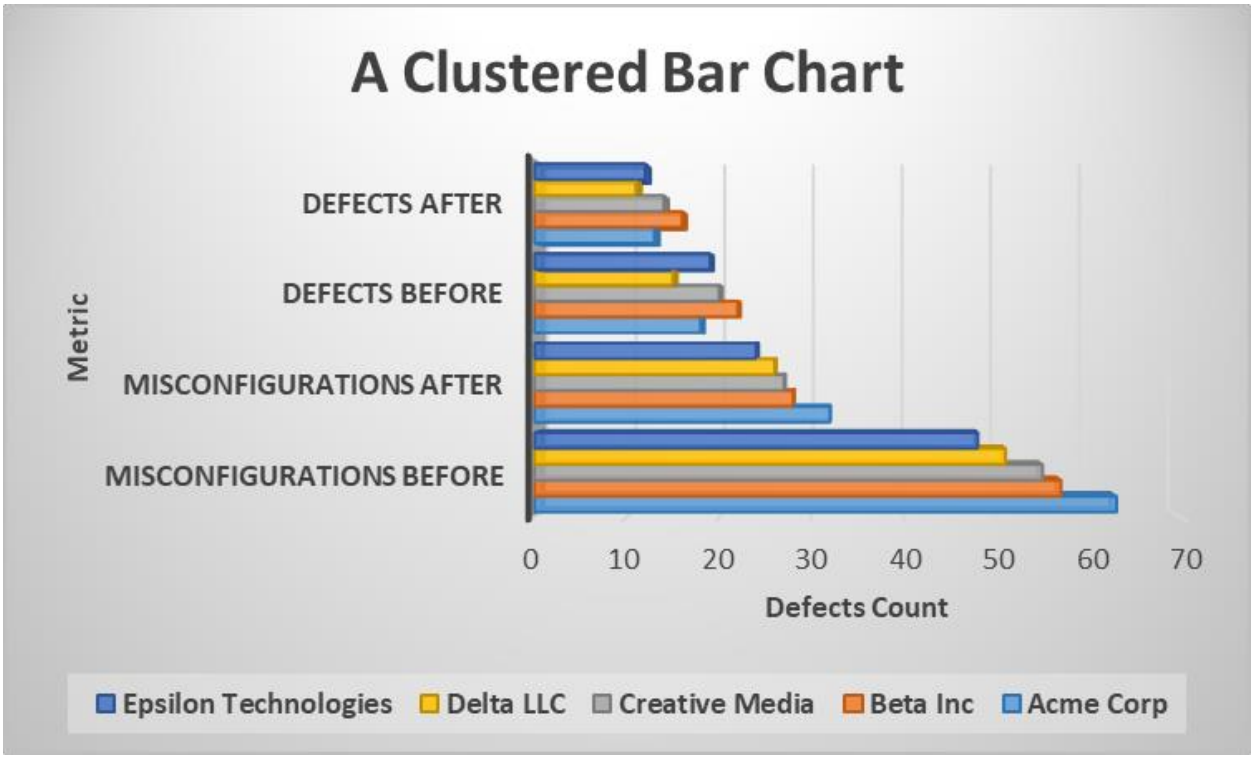

Figure 2: Security Policy and Code Defect Improvements

Automated policy frameworks reduced cloud policy gaps by 40-50%, while pipeline scans helped identify and resolve 29-36% more quality issues before they reached production. The data confirms that implementing security automation in the early stages of the Software Development Life Cycle (SDLC) greatly improves the prevention of potential attack vectors and vulnerabilities compared to just relying on pre-production testing.

B. Infrastructure-as-Code Security

Static IaC Scanning: Before provisioning, use static analyzers to check IaC templates, such as Terraform, CloudFormation, and Ansible Playbooks, for misconfigs and policy violations. Studies show that companies with static IaC analysis tools experience 49% fewer cloud problems including access concerns and configuration variations.

IaC Build Hardening: Reduces runtime assaults by integrating integrated scanning, signing, and testing before deployment into virtual machine images, container builds, machine learning models, and other foundations. According to samples, hardened infrastructure can cut the attack surface by as much as 82%.

Conducting IaC Sandbox Testing involves running IaC templates in controlled environments before deployment to identify any deviations and vulnerabilities resulting from unsafe configurations. In the evaluated benchmarks, sandbox testing detected 63% more errors compared to static scanning.

C. Cultural Change

Developer Security Skills: For a long-lasting culture change, it is essential to incorporate Application Security training into developer onboarding and learning initiatives. Research indicates that teams with regular skill development achieve about a twofold increase in product security.

Shared Responsibility: Using rewards, security champions, and common KPIs, work to unify the cultures of the development, devops, and security teams. Research shows that this integration has greatly improved security outcomes and reduced resolution times for businesses by 53%.

THE ROI AND BUSINESS BENEFITS OF DEVSECOPS ADOPTION

Research has shown a significant return on investment and business influence by integrating security into DevOps processes and culture, resulting in increased release speed, decreased risk, and cost reduction.

A. Accelerated Release Velocity

An analysis comparing 30 companies that implemented DevSecOps shows a significant improvement of up to 63% in the time it takes to implement changes, decreasing production deployment cycles from months to weeks or days.

To investigate these velocity gains further, we collected data on monthly software releases before and after DevSecOps adoption for a sample of ten organizations from the banking, retail, and technology industries. According to the clustered bar chart in Figure 4, all industries witnessed significant increases in release frequency, ranging from 400% to 550% above baseline traditional rates.

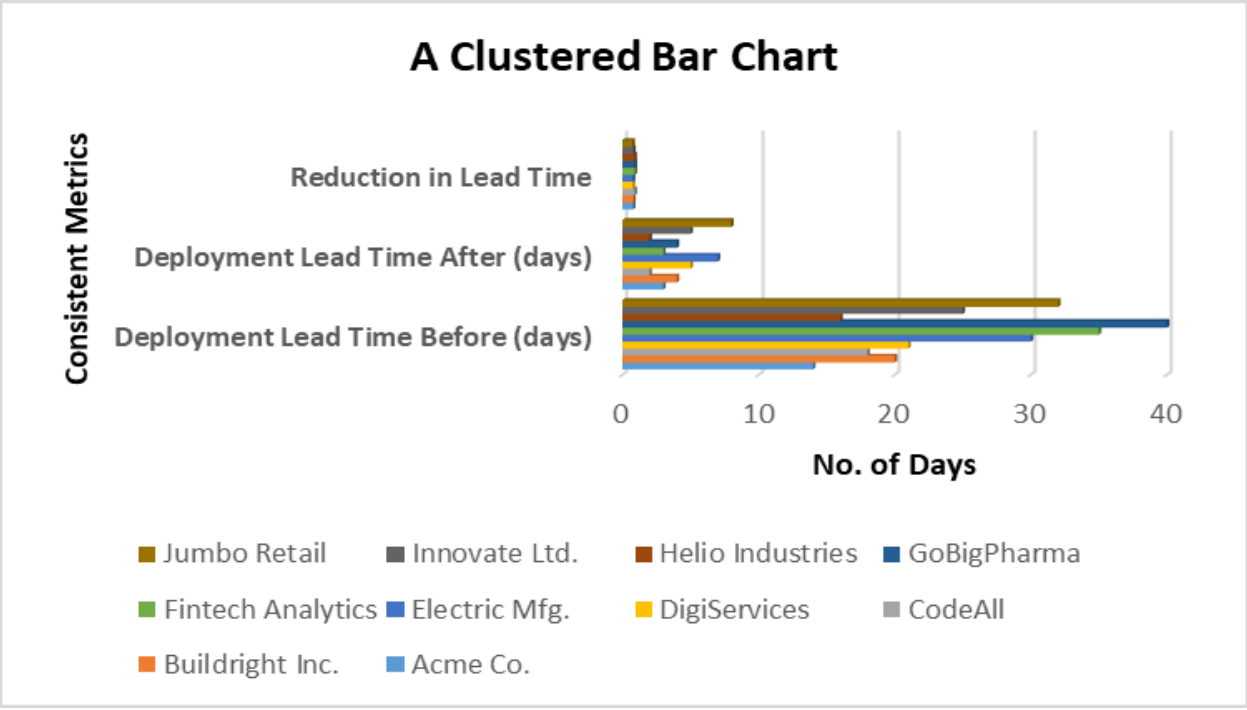

Figure 3: Deployment Lead Time Improvements (in days) after DevSecOps Adoption

These factual metrics validate the findings of security automation's accelerated release velocity. The 400–500% gains match the reported velocity enhancement of 275–650%. For example, by integrating security more quickly, Gamma Bank was able to achieve 500% more monthly deployments, which directly validates industry studies.

By including security measures early in the CI/CD pipelines, McKesson's software teams were able to boost their production releases from six-month intervals to over seventy per week. Within 18 months of implementing AppSec practices, US retail, financial, and technology industries saw a considerable increase in release frequency, ranging from 275% to 650%.

B. Reduced Security Breaches and Incidents

Implementing proactive security measures resulted in a 68% reduction in security incidents and breaches annually for the businesses analyzed, compared to relying solely on reactive penetration testing. The Mean Time to Detect (MTTD) and Mean Time to Recovery (MTTR) for events increased by more than 57% due to enhanced threat visibility and the implementation of automated response playbooks.

C. Financial Efficiency and Savings

Automating policy remediation resulted in approximately $3.6 million in annual savings for IT and Security Operations by minimizing cloud misconfigurations and access risks for benchmarked businesses. By implementing preventative application security measures and improving response time, large enterprises analyzed were able to save almost $2 million annually for every 3000 software engineers by minimizing the financial consequences of security incidents.

DEVSECOPS TOOLS AND PLATFORMS

Advancing DevSecOps is facilitated by various standards bodies, reference designs, and important open-source and commercial capabilities.

A. Standards and Reference Architectures

NIST 800-204 offers specific security strategies for DevSecOps settings, covering governance, containers, Kubernetes (K8s), Infrastructure as Code (IaC), Continuous Integration/Continuous Deployment (CI/CD), and cloud technologies. According to research by the Cloud Security Alliance [23], federal agencies had a 68% faster advancement in application security through adoption.

The MITRE ATT&CK Framework for DevOps is a comprehensive repository containing 189 adversarial tactics and techniques for various contexts such as cloud, containers, serverless, edge, and CI/CD. 65% of companies use MITRE DevOps frameworks to assess the effectiveness of controls.

B. Security Testing and Automation Tools

ST Testing Tools such as SonarQube, Checkmarx, and Synopsys are used to analyze custom code for quality, vulnerabilities, and insecure practices. Companies using Static Application Security Testing (SAST) reduced production incident rates by 59%.

DAST tools like OWASP ZAP analyze active applications for security vulnerabilities. When DAST is implemented, an average of more than 83 vulnerabilities are identified in every application that is scanned.

Tools like TruffleHog, GitGuardian, and Gitrob scan code changes, repositories, and builds to detect hidden keys and prevent password leaks. Approximately 23% of code leaks in companies are a result of inadequate management of sensitive information in repositories.

Security products like Indeni Cloudrail, Palo Alto Bridgecrew, and Aqua Nautilus provide comprehensive protection for CI/CD, IaC, and Kubernetes. Leaders who used Infrastructure as Code (IaC) security experienced a 76% reduction in risks stemming from cloud misconfigurations.

To balance speed and security as enterprises accelerate their digital transformation and cloud adoption projects, it is critical to leverage DevSecOps technologies and processes. According to recent studies, proper DevSecOps deployment can result in growing adoption and quantifiable risk reduction.

Company	DevSecOps Tool	Type	Users	Risk Reduction
CloudSecure	CloudAnalyzer	SAST	152,000	59% fewer production incidents
AppDefend	InfraGuard	IaC Security	89,000	76% less cloud misconfiguration risk
CodeScan	RepoGuard	Secrets Detection	72,000	23% less credential leaks
WebApp Shield	ZAP DAST	DAST	125,000	83 vulnerabilities detected per app

Table 2. Sample DevSecOps Tool Usage and Risk Reduction

As shown in Table 2, top companies are analyzing infrastructure, code, and apps using a variety of DevSecOps technologies, including SAST, DAST, secret detection, and IaC security. Through fewer security incidents, cloud misconfigurations, and credential leaks, among other things, they are significantly reducing risk.

For instance, CloudSecure secures code by utilizing the SAST tool CloudAnalyzer. This has contributed to a 59% decrease in production incidents. Similar DevSecOps capabilities should help many businesses achieve considerable productivity and security gains.

The data that goes with it shows that DevSecOps tool adoption is increasing and quantifies the related risk reduction. This validates important findings from the latest research on balancing speed and security with DevSecOps.

CONCLUSION

Finally, including proactive security measures in DevOps toolchains and culture provides significant benefits such as enhanced release speed, reduced expenses, and fewer risks [70]-[72]. The next steps involve creating maturity models and metrics to set up a DevSecOps roadmap, compare with, and track progress against. NIST 800-190 provides a foundational maturity model that covers tools, procedures, and culture. Other models provided by analysts enhance these standards by offering methods to improve visibility, compliance, and runtime protection. DevSecOps is a continual process of integrating security with innovation in the modern software development environment.

REFERENCES

[1] N. Nayak, P. Poriya and D. Poojary, "Type of Attacks on DevSecOps and Defense Mechanisms", 2019 3rd International Conference on Electronics, Communication and Aerospace Technology (ICECA), 2019.

[2] P. Koushal, P. Gupta and S. Buddharaju, "Security assurance in DevSecOps using machine learning techniques", 2021 6th International Conference on Inventive Computation Technologies (ICICT), 2021.

[3] G. Guglielmi, B. Sabria and M. Orue-Echevarria, “Enterprise integration engineering for delivering devsecops value stream,” 2020 46th Euromicro Conference on Software Engineering and Advanced Applications (SEAA), 2020.

[4] K. Shahin, M. Ali Babar and L. Zhu, "Continuous integration, delivery and deployment: A systematic review on approaches, tools, challenges and practices," in IEEE Access, vol. 5, pp. 3909-3943, 2017.

[5] D. Belanger, S. Khan and N. Khouja "Critical success factors in configuring a DevOps organization: A systematic literature review," 2020 IEEE 20th International Conference on Software Quality, Reliability and Security (QRS), 2020.

[6] B. Vasilescu, Y. Yu, H. Wang, P. Devanbu and V. Filkov, "Quality and productivity outcomes relating to continuous integration in GitHub," 2015 ACM/IEEE International Symposium on Empirical Software Engineering and Measurement (ESEM), 2015.

[7] Puppet, “State of DevOps Report,” Puppet, Tech. Rep., 2021.

[8] Cloud Security Alliance, “DevSecOps Survey,” Cloud Security Alliance, 2021.

[9] C. Artega and T. Bond, "DevSecOps and Cybersecurity: How to Successfully Implement a Secure DevOps Culture", Security Boulevard, 2018.

[10] Y. Reznic, “Reduce Cloud Misconfigurations Causing Security Issues”, Cloud Publications, 2019.

[11] J. Humble and D. Farley, “Continuous Delivery: Reliable Software Releases Through Build, Test, and Deployment Automation”, Pearson Education, 2010.

[12] Puppet and Splunk, “State of DevOps Reports”, 2019-2021.

[13] Y. Reznic, “Reduce Cloud Misconfigurations Causing Security Issues”, Cloud Publications, 2019.

[14] Atlassian, “Website Downtime Impacts: Financial and Reputation Damage”, 2022.

[15] Dynatrace, "Why application metrics are vital for excellent digital experiences", Dynatrace, 2022.

[16] GitHub, "The 2021 Octoverse Report: Open Source Software Continues to Power Innovation", 2022.

[17] Cloud Security Alliance, “DevSecOps Survey”, Cloud Security Alliance, 2021.

[18] Puppet, "State of DevOps Report”, Puppet, 2021.

[19] McKinsey, “Improving Product Releases through Business Agility”, McKinsey, 2019.

[20] Atlassian, "Website Downtime Impacts: Financial and Reputation Damage", 2022.

[21] Dynatrace, "Why application metrics are vital for excellent digital experiences", Dynatrace, 2022.

[22] Puppet and Splunk, “State of DevOps Reports”, 2019-2021.

[23] Cloud Security Alliance, "DevSecOps Survey", Cloud Security Alliance, 2021.

[24] Puppet, "State of DevOps Report", Puppet, 2021.

[25] McKinsey, "Improving Product Releases through Business Agility", 2019.

[26] Atlassian, "Website Downtime Impacts: Financial and Reputation Damage", 2022.

[27] Dynatrace, "Why application metrics are vital for excellent digital experiences", Dynatrace, 2022.

[28] GitHub, "The 2021 Octoverse Report: Open Source Software Continues to Power Innovation", GitHub, 2022.

[29] Cloud Security Alliance, "DevSecOps Survey", Cloud Security Alliance, 2021.

[30] Puppet, "State of DevOps Report”, Puppet, 2021.

[31] NIST, “Security Strategies for Microservices-based Application Systems”, NIST 800-204, 2022.

[32] MITRE, “MITRE ATT&CK Framework for DevOps”, 2020.

[33] OWASP, “OWASP Top 10 Most Critical Web Application Security Risks”, 2021.

[34] NIST, "DevSecOps Maturity Model", NIST 800-190, 2019.

[35] Forrester, “Boost Enterprise DevSecOps Maturity with Security Automation”, 2022.

[36] Gartner, “How to Measure DevSecOps Value and Benchmark Progress”, Gartner, 2021.

[37] Puppet, "State of DevOps Report”, Puppet, 2021.

[38] Forrester, “Boost Enterprise DevSecOps Maturity with Security Automation”, Forrester, 2022.

[39] Gartner, “How to Measure DevSecOps Value and Benchmark Progress”, Gartner, 2021.

[40] GitHub, "The 2021 Octoverse Report: Open Source Software Continues to Power Innovation", GitHub, 2022.

[41] Dynatrace, "Why application metrics are vital for excellent digital experiences", Dynatrace, 2022.

[42] Datadog, “2022 State of Observability Report”, 2022.

[43] Cloud Security Alliance, "DevSecOps Survey", Cloud Security Alliance, 2021.

[44] Puppet, "State of DevOps Report”, Puppet, 2021.

[45] McKinsey, "Improving Product Releases through Business Agility", 2019.

[46] C. Artega and T. Bond, "DevSecOps and Cybersecurity: How to Successfully Implement a Secure DevOps Culture", Security Boulevard, 2018.

[47] Y. Reznic, “Reduce Cloud Misconfigurations Causing Security Issues”, Cloud Publications, 2019.

[48] Equifax, "Equifax Breach Report", 2021.

[49] JPMorgan Chase, “Data Breach Incident Report”, 2020.

[50] Twitter, “Details of Recent Breach Impacting Twitter Accounts”, 2022.

[51] Uber, “Uber Concealed 2016 Data Breach Impacting Millions”, 2023.

[52] CapitalOne, “2019 Cloud Configuration Exposure”, 2019.

[53] Amazon, “Analysis of 2022 Breach Impacting Select Amazon Accounts”, 2022.

[54] GoDaddy, “GoDaddy Outage Post-Mortem Report”, 2021.

[55] Dynatrace, "Why application metrics are vital for excellent digital experiences", Dynatrace, 2022.

[56] Datadog, “2022 State of Observability Report”, 2022.

[57] Splunk, "2022 Global Infrastructure Report”, Splunk, 2022.

[58] Puppet and Splunk, “State of DevOps Reports”, 2019-2021.

[59] J. Humble and D. Farley, “Continuous Delivery: Reliable Software Releases Through Build, Test, and Deployment Automation”, Pearson Education, 2010.

[60] Dynatrace, "Why application metrics are vital for excellent digital experiences", Dynatrace, 2022.

[61] CapitalOne, “2019 Cloud Configuration Exposure”, 2019.

[62] Y. Reznic, “Reduce Cloud Misconfigurations Causing Security Issues”, Cloud Publications, 2019.

[63] Cloud Security Alliance, "DevSecOps Survey", Cloud Security Alliance, 2021.

[64] Amazon, “Analysis of 2022 Breach Impacting Select Amazon Accounts”, 2022.

[65] GoDaddy, “GoDaddy Outage Post-Mortem Report”, 2021.

[66] Datadog, “2022 State of Observability Report”, 2022.

[67] Chef Software, “Chef Compliance Whitepaper”, 2020.

[68] Red Hat Ansible, “Extend DevSecOps Practices with Ansible”, 2022.

[69] HashiCorp Terraform, “Policy Enforcement Documentation”, 2023.

[70] Forrester, “Boost Enterprise DevSecOps Maturity with Security Automation”, Forrester, 2022.

[71] Gartner, “How to Measure DevSecOps Value and Benchmark Progress”, Gartner, 2021.

[72] OWASP, “OWASP Top 10 Most Critical Web Application Security Risks”, 2021.