International Journal of Research Publication and Reviews

Journal homepage: www.ijrpr.com ISSN 2582-7421

The Impact of DevOps on Security Culture

Jyoti Ajjappanavar[a], Sneha Gangal[b], Mr. Nilesh Anvekar[a, b, *]

Student, MCA, KLS Gogte Institute of Technology College Belgaum-590008, India[a]
Student, MCA, KLS Gogte Institute of Technology College Belgaum-590008, India[b]

ABSTRACT :

This study looks at how DevOps methods affect security culture in organizations. DevOps stresses teamwork, automation, & blending development with operations to speed up software deployment. However, this fast pace could bring in security issues if not handled well. By analyzing existing literature and research, this paper explores the impact of DevOps on security culture, covering attitudes, behaviors, and practices related to cybersecurity. Important findings show the need to have a security-focused approach throughout the DevOps process, seamlessly integrating security into development tasks, and fostering collaboration between security and DevOps teams. The study also points out challenges and opportunities in improving security culture in DevOps settings and offers suggestions for organizations aiming to boost their cybersecurity status in the age of DevOps.

1.INTRODUCTION :

Let's conversation almost DevOps, which may be a cool program handle. It's all around groups working together in computer program advancement. Concurring to CA Advances, a whopping 88% of organization administrators are either using or arranging to utilize DevOps within the following five a long time. That's beautiful noteworthy! Presently, Manikin Labs did a think about in 2015 and found that organizations utilizing DevOps have way less disappointments and can send their manifestations more frequently compared to those not utilizing Dev. That's a few genuine proficiency right there! This report needs to plunge into the history of DevOps and how it overseen to incorporate security in its process. We're talking impacts, benefits, a few real-world cases, what's happening presently within the industry, and where things might go within the future. This report is like a profound jump into the world of DevOps - a writing audit to be correct. Affirm, so here's the bargain - security issues are popping up cleared out and right these days. Organizations are confronting more security episodes than ever some time recently. That's why they're all around beefing up their security hones. And that's where DevOps (Secured DevOps) comes into play. It's all almost making security a need at each arrange of software development

The showcase for DevOps was worth 3.7 billion in 2021! Can you accept that? And it's anticipated to bounce up to 41.7 billion by 2030 at this rate of growth – wild stuff! Alrighty, let's get into how we handle our investigate questions - we looked at tons of stuff on the web like web journal posts, conference introductions, and recordings to get it how individuals see security inside the domain of DevOps. We indeed sent out studies to nine organizations that are as of now onboard with DevOps! In our paper, we make a refinement between 'activity' and 'security practice'. An movement is all around coming to a specific objective with unmistakable comes about whereas a security practice is like a bunch of related exercises beneath one umbrella. For case, 'automation of testing' is an movement whereas 'use of computerization activities' could be a greater concept - a security hone. To entirety it up: We recorded out DevOps exercises that either offer assistance or harmed framework security. Looked at diverse security hones received by organizations utilizing DevOps. Attempted to degree how teams collaborate within organizations – think improvement groups, operations groups, and security groups coming together. It's beautiful curiously that there isn't much out there around DevOps particularly. So it looks like we have a few more work to do on this subject within the future – remain tuned!

2. BACKGROUND AND RELATED WORK :

In this section, we provide background information and prior

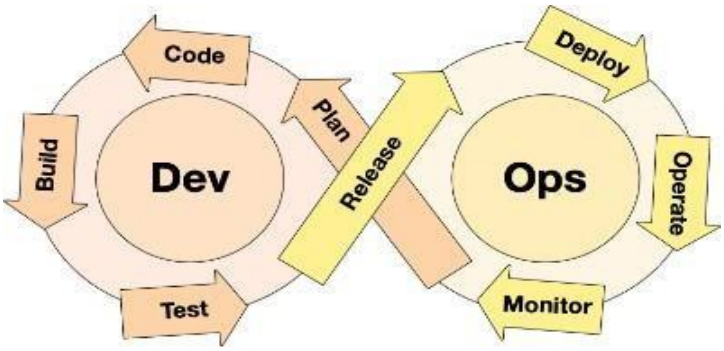
Background definition:

it's a bunch of software activities like continuous planning and continuous deployment. These activities are backed up by cultural things like sharing responsibility and goals. Plus, there are technical bits like automated build processes and automated configuration management. According to Dyck and others, DevOps is all about teamwork among different teams involved in making software. From now on, we'll call companies that use DevOps "DevOps organizations. "Experts in software have been talking a lot about adding security to DevOps. That's why you might have heard of DevOps—it's getting pretty popular these days. Back in April 2012, Turnbull introduced the idea of security teams working closely with everyone else in the organization. In this article, when we say DevOps, we mean blending security principles by getting development, operations, and security teams to work together more closely in a DevOps organization. To help us chat in this article, let's talk about two terms: "activity" and "security practice." An activity in DevOps is all about tackling a small goal with something you can see at the end. On the other hand, a security practice is a bunch of activities grouped together because they're similar. DevOps is all about teamwork among people from different backgrounds. But hey, companies are finding it tough to adopt new ways to improve their apps and get them out to customers faster. Even though DevOps is kind of newish, it's like an extension of agile methods. But there are

some challenges to deal with when organizations try out DevOps. For one thing, changing from old infrastructure to new stuff can be expensive and take up a lot of time. Also, moving into DevOps means you need new tools—and that means training up on them too! There's this whole process in DevOps that goes through planning, building, testing, sending out updates, and watching over things after they're live. The key part here? It's having an automated pipeline that ties all these steps together smoothly—it's what makes sure everything runs without any hiccups!

2.2Related Work

Hey there! Let's dive into the concepts and related work covered in this thesis. Understanding security culture involves grasping key concepts like agile software development, team culture, organizational culture, and guidelines. Prior studies have delved into software security attitudes in organizations involved in software development, implementation, and maintenance. They've also looked at the challenges faced by development teams, individuals, and organizations when it comes to handling security tasks. Agile Software Development follows principles outlined in the Agile Manifesto of 2001. It emphasizes an iterative approach to product development with continuous dialogue between customers and developers. Examples of Agile frameworks include Scrum, Kanban, and Extreme Programming (XP). While focusing on technical excellence and good design strengthens the agile process, it may clash with security practices due to a lack of procedures for security requirements. Early research indicates a suspicion that security work gets overlooked in agile processes. However, Bartsch suggests that an agile methodology's holistic view can heighten developers' sense of responsibility towards security tasks. The DevOps paradigm has gained popularity among development teams by fostering closer collaboration between software development and operations teams. Developers now monitor and manage their software independently, increasing their accountability. Closer integration between development and operations simplifies issue resolution and boosts motivation towards software care-taking rather than just fulfilling a role. Balancing autonomy and guidance is crucial in agile development. When managers intervene to push for security activities within self-organizing teams, developers tend to resist as it restricts their autonomy. Striking a balance where teams meet organizational goals without excessive management interference is the ideal scenario.



4.Team Culture in Software Development

Agile development is all about people and their creativity rather than just descriptions. Xiao et al. discuss security culture as the social norms and habits surrounding security in a team or an organization. The way a group behaves impacts how they do their work Schein says culture is what's clear and important, influencing the direction of things.

The organization's culture is key to effectiveness and feeling safe. Psychological safety means being able to be yourself without worrying about negative consequences related to status, image, or career. Nowadays, with agile methods and autonomous teams, team culture has a big impact on an employee's daily life. This psychological safety also affects how teams discuss security concerns.

Agile development is all about people and their creativity rather than just descriptions. Xiao et al. discuss security culture as the social norms and habits surrounding security in a team or an organization. The way a group behaves impacts how they do their work Schein says culture is what's clear and important, influencing the direction of things.

The organization's culture is key to effectiveness and feeling safe. Psychological safety means being able to be yourself without worrying about negative consequences related to status, image, or career. Nowadays, with agile methods and autonomous teams, team culture has a big impact on an employee's daily life. This psychological safety also affects how teams discuss security concerns.

The organization's culture is key to effectiveness and feeling safe. Psychological safety means being able to be yourself without worrying about negative consequences related to status, image, or career. Nowadays, with agile methods and autonomous teams, team culture has a big impact on an employee's daily life. This psychological safety also affects how teams discuss security concerns.

Research by Bartsch shows that motivation for security knowledge comes from feeling responsible for the product. They also found that good communication boosts this motivation. A study from 2019 revealed that teams with long- term plans focus more on quality and reflection, showing that well-functioning teams plan and learn from past experiences.

5. RESEARCH METHODOLOGY

Hey there! Let's talk about how we went about our study. First off, we took a look at Internet artifacts - stuff like blog posts and presentations. Then, we chatted with nine DevOps organizations to dig deeper into how they view security and the practices they follow.

Analysis of Internet Artifacts

We kicked things off by searching on Google for Internet artifacts using the search term "security in DevOps". We found a bunch of interesting stuff, including mentions of "DevSecOps", "DevOps", "SecOps", and "RuggedOps". Based on this, we used seven different search terms to gather more info:
security in DevOps
DevSecOps
SecDevOps
SecOps
RuggedOps
Security in Continuous Delivery
Security in Continuous Deployment
We looked at the top 50 results for each search term. Any artifacts that didn't talk about the benefits or downsides of DevOps for system security or integrating security practices were left out.
From these artifacts, we learned which DevOps activities are seen as helpful or harmful to system security. We made separate lists for these activities. When an activity was mentioned in multiple artifacts, we only included it once. We also checked out the security practices that software practitioners recommended for integrating security in DevOps. If different terms were used to describe the same practice, we grouped them. The same goes for different activities related to a certain practice.

6. Survey

We asked representatives from nine organizations using DevOps practices about their views on DevOps activities related to system security. They shared insights on security practices too. Remember, an activity our focuses on specific goals with tangible results, while a security practice is a collection of activities grouped by similarities.
Our survey was based on Internet research findings. Participants were tasked with identifying DevOps activities that enhance software security. They chose from a list we compiled from Internet research or added their ideas. Negative DevOps-related activities for system security were minimal compared to positive ones, which we identified through free text responses.
Participants were asked about security practices they use and any additional integration activities. We included questions on collaboration between development and operations, development and security, and security and operations teams using a Likert Scale rating from one to five. The scale helped us assess the level of collaboration: five being the highest and one being the lowest. If a team was missing in an organization, participants could assign a zero rating. Later, we categorized ratings as 'highest', 'high', 'moderate', 'low', or 'lowest'.

7.Result

In this chapter, we're sharing the findings of our conversations with people & their thoughts on security. We noticed common themes & issues about security culture. This chapter is set up like the interview guide. We break down the results into four parts: individual developers, team culture & security activities, organizational factors impacting security work, & customer-related influences. Plus, we'll give specific examples and quotes to explain our findings

Discoveries About Individual Developers

Each person's background, hobbies, and security training affect how they contribute to software security in their team.

Personal Interest

People rated their level of interest in software security from 1 to
Most people (47%) rated themselves as "Sort of interested
(4)" or "Very interested (5)" in software security, showing they care about it. Another 47.1% said they were "Neutral" about it, while one person admitted to being "Kind of uninterested." No one said they were "Very uninterested." Some who were neutral mentioned that they find security interesting if it's explained to them but don't actively seek out information on their own. They might attend conferences but only focus on the security track to stay updated. One individual openly shared that they're not interested in software security despite having experience with it on various projects. They understand its importance but find it boring and time-consuming - something that just has to be done without enjoyment.

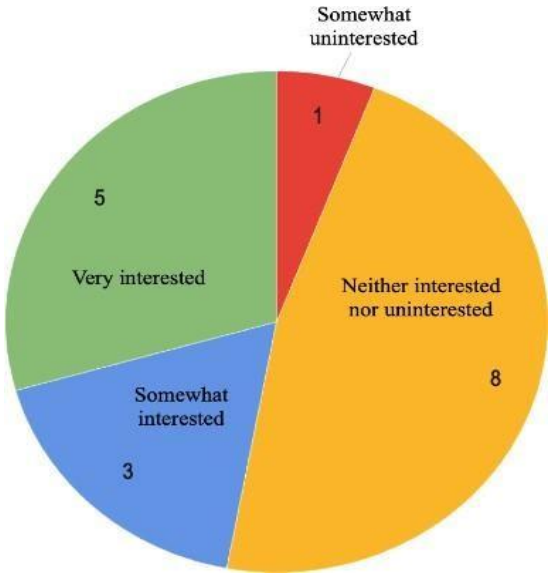


Figure 4.1: Pie chart of self-evaluated personal interest in software security

Awareness and Responsibility

A few older programmers chatted about how younger developers are more aware of software security now. One of them mentioned that not everyone shows interest in security, but most will still spend time on it. "Security takes up a lot of time," noted a newer developer. He mentioned how it can be tough to work with someone who focuses heavily on security. He added that when you focus on making the software work, it's faster, but you might overlook security. Security can sometimes get in the way of practicality because of strict rules around handling data and flows. Some say finding and exploiting software vulnerabilities is like playing Capture The Flag: a game-type approach to security.

Finding a balance between practicality and security is crucial. Many developers see security as something essential for the product to function properly, even though it's not something you see working. Another older developer thought about how important it is to find the right balance between these two aspects as mentioned in quote 1. He stressed that security should never be ignored because it's a significant part of ensuring the product is solid and safe for users.

Knowledge and Experience

One of our senior developers shared that simply being interested in software security isn't sufficient. You also need a basic level of knowledge to make informed decisions. She stressed that this knowledge doesn't just happen; you have to seek it out through education or hands-on projects. A junior developer mentioned that she wants to understand software applications more broadly before delving into specific security measures. Another junior developer noted that her project experiences have heightened her awareness of the importance of software security. She explained that her formal education didn't emphasize this aspect. In general, our interviewees pointed out that they learned about software security mainly from their project experiences and collaborating with other team members.

Findings Regarding Team Culture and Activities

Our material shows that the overall team culture and group dynamics influence the security work. Novice consultants in a team need to understand the culture, including written rules, norms and practices. According to an interviewee, it is easy to notice the culture through daily stand-ups by looking at the code and other activities. He continued saying that the developer then finds out if the culture feels comfortable to them or if they would have to learn something new to fit in.

An interviewee explained a practice where an experienced developer shared work assignments with the novice, such that the novice did not have explicit responsibility. He called the practice shadowing. The novice was assigned one specific senior to ask questions and gain knowledge from throughout the period.This activity was resource-intensive, he expressed, but the experiences the novice gained could make it worth it.

Psychological Safety

Multiple interviewees have talked about the importance of collaboration skills in the team. To promote this, an interviewee emphasized that psychological safety for all team members is desired. He said that constructive communication and feedback is vital for individuals to trust each other. Several interviewees expressed that colleagues had become better at both giving and receiving constructive feedback. Some emphasized the importance of not blaming others for mistakes they have made, as shown in quote 6. An interviewee explained that a mistake never only has one cause. He further said that the importance of psychological safety within a team might be underestimated.

Activities Within the Team

Teams arrange activities to improve the security work. It could be structured activities such as risk assessments and penetration tests and unstructured activities such as discussions during lunch and spontaneous initiatives from an individual. Activities are arranged to improve the team's psychological safety and improve the overall security work through more technical activities. Our interviewees seem to use their peers to learn. Several interviewees highlighted
the availability of their colleagues and their willingness to teach. An interviewee explained a practice where an experienced developer shared work assignments with
the novice, such that the novice did not have explicit responsibility. He called the practice shadowing. The novice was assigned one specific senior to ask questions and gain knowledge from throughout the period. This activity was resource-intensive, he expressed, but the experiences the novice gained could make it worth it.

Competence Development

Different consulting firms have various ways to help their consultants gain knowledge. Some companies even pay their employees to study on their own. This allows consultants to improve their skills in topics they choose during work hours. Other methods include sending employees to conferences and conducting internal lessons. Based on our research,
individual developers have the freedom to pick the topics they are interested in. The consulting firm doesn't prescribe specific subjects but encourages creativity and self-driven projects. One consequence of this freedom, as mentioned by many participants, is that software security is not always a top priority. There are countless topics that consultants want to learn about. For example, a junior employee expressed interest in learning more about software security but admitted it's impossible to learn everything. Another respondent shared that she only starts learning when a task demands it.

Security Roles

Some customers with a high level of maturity assign a security role to one of the developers on the team, which could be called "Security Champion". Several of our interviewees have such a role. They emphasized the value of having an appointed person who aims to raise awareness and quality of the security work. However, they mentioned factors that could enhance the security role. All four interviewees with security roles were particularly interested in software security. They explained that the role consisted of having an extra focus on security, but had no defined tasks. A respondent reflected on his role in quote 10, and said that there should be a note on what the mandate

Findings Regarding Customers' Effect on Security Work

Almost every interviewee mentioned that the focus on security work in teams varies depending on the developed product. An interview explained often easier.to suggest security activities and get prioritization with customers of high maturity level. As consequences of broken software for these industries can be severe, they have experience with prioritizing security. Another interviewee thought that an involved leadership that asks questions is meaningful for the security awareness within a team. He continued by explaining how he notices the customers level of awareness in practice. Positively influencing this are initiatives from the customers such as security training and introducing security roles.

Reputation

Several interviewees mentioned the effect news coverage of broken systems has on customers. Generally, they are afraid of being hacked and then receive bad publicity. Their product
would then get a weakened reputation. An interviewee discussed that this is one of the most important reasons why customers have become more willing to prioritize resources to security activities in recent years.

Trade-Off Between Security and Business

An interviewee highlighted that there is a trade-off between software security and business considerations. He said that he had experienced that the customer did not follow his recommendation, see quote 20. He also expressed that economics and budgets play a large role in prioritizing and doing security activities. Software security is costly, and product owners need to weigh it up against other business concerns.

LIMITATIONS :

In our study, we cannot claim that the set of Internet artifacts is complete as we used seven search strings to collect the necessary Internet artifacts. We do not claim that the identified security practices for integrating security in DevOps is complete. Since the number of surveyed organizations is small, we cannot strongly claim our findings are generalizable. We did not study if there is any relationship between the use of automation activities, and quality of software deployed by the nine DevOps organizations of interest. We also did not discuss whether level of collaboration between different teams had an impact on use of the four security practices, the five automation activities, or the ten security activities. We leave the scope of pursuing these limitations as research guidelines for future work.

CONCLUSION :

Sure, here's a synthesized conclusion based on the analysis of several research papers on the impact of DevOps on security culture: In conclusion, the intersection of DevOps and security culture represents a significant paradigm shift in contemporary software development practices. Through an analysis of multiple research papers, it becomes evident that the adoption of DevOps principles and practices has a transformative effect on organizational security postures.

Firstly, DevOps fosters collaboration and communication

between traditionally siloed development, operations, and security teams. This alignment promotes a shared responsibility for security throughout the software development lifecycle (SDLC), leading to enhanced threat detection and mitigation. Secondly, automation plays a pivotal role in integrating security into the development pipeline. By automating security testing, compliance checks, and configuration management, DevOps practices ensure that security measures are consistently applied and validated, reducing the likelihood of vulnerabilities slipping through the cracks. Furthermore, the cultural shift towards DevOps encourages a proactive approach to security, emphasizing continuous monitoring, feedback, and improvement. This iterative mindset allows organizations to adapt to evolving threats and regulatory requirements more effectively.

However, it's essential to acknowledge that the adoption of DevOps does not guarantee bulletproof security. Challenges such as cultural resistance, tooling complexity, and skill gaps must be addressed to fully realize the benefits of DevOps in enhancing security culture.

In summary, the synthesis of research papers underscores the profound impact of DevOps on security culture within organizations. By promoting collaboration, automation, and a proactive mindset, DevOps empowers teams to build and maintain more resilient software systems in today's ever-evolving threat landscape. Nevertheless, ongoing efforts are necessary to address challenges and ensure that DevOps initiatives align with security best practices effectively.

REFERENCES :

[1] M. Mell and T. Grance. The nist definition of cloud computing. Special Publications (NIST SP)-800-145, (7), 9 2011. NIST Definitions on Cloud Computing.

[2] B. Fitzgerald and K. J. Stol. Continuous software engineering: A roadmap and agenda. Journal of Systems and Software, 123:176 – 189, 2017.

[3]. B. Svensson G. G. Claps and A. Aurum. On the journey to continuous deployment: Technical and social challenges along the way. Information and Software Technology, 57:21 – 31, 2015.

[4] J. Humble and M. Joanne. Why enterprises must adopt devops to enable continuous delivery. The Journal of Information Technology Management, (24), 7 2011.

[5] J. Hernantes C. Ebert, G. Gallardo and N. Serrano. Devops. IEEE Software,33(3):94–100, May 2016.

[6] J. Yankel C. A. Cois and A. Connell. Modern devops: Optimizing software development through effective system interactions. In 2014 IEEE International Profes-sional Communication Conference (IPCC), pages 1–7, Oct 2014.

[7] Shackleford. The devsecops approach to securing your code and your cloud. SANS Institute InfoSec Reading Room, 2 2017. A DevSecOps playbook.

[8] C. Caum. Getting started with policy-driven development and devsecops. https://goo.gl/AevVcX, 2016.

[9] Whitehat Security. Devops invites security to “join the party”. https://goo.gl/spj0wK, 2016.

[10] M. Hornbeek. Devops makes security assurance affordable. https://goo.gl/g0iKfZ, 2015

[11] K. Lindros. How to craft an effective devsecops process with your team. https://goo.gl/ppWtjx, 2016.

[12] C. Romeo. The 3 most crucial security behaviors in devsecops. https://goo.gl/FJKuYQ, 2016.

[13] A. Cureton. Building security into devops: Is devsecops the beginning of the future? https://goo.gl/Npv2Py, 2017.

[14] J. McKay. How to use devsecops to smooth cloud deployment. https://goo.gl/vqoh4L, 2016.

[15] Amazon Web Services. Introduction to devsecops on aws. https://goo.gl/wxl3YM, 2016.

[16] R. Francis. 7 ways devops benefits cisos and their security programs. https://goo.gl/RxieGr, 2015.