Dangers of Change Approval Processes

Change approval processes are utilized by IT organizations to mitigate risks to the organization and maintain control over their services. These processes can be useful tools to preventing any serious and costly risks to the organization (DORA). However, with regulation often comes administrative issues. This essay will explore the dangers of change approval processes, which includes delays, misallocation of time, and superficial reviews.

By requiring approvals for every deployment within a change approval process, developers are not able to release deployments as often as they would like to (Plat4mation). The approval process creates what developers may see as extra and unnecessary time spent doing work that could otherwise be used for creating actual code development. This lack of alignment between the change approvers, which may be a change approval board, and the developers can result in lackluster change approval requests, ultimately defeating the purpose of them in the first place (Plat4mation).

In addition to causing delays, the extended time required for approvals can create a backlog of pending change requests, further slowing down the development process. This can potentially result in earlier change requests to be unnecessary as the code develops, which equates to wasted time on behalf of both the approving board and the developers (Atlassian). The amount of time required to approve changes, which can be months, can often be unwarranted if the change is small or relatively insignificant. This once again points to a lack of understanding and trust between the approval board and the developers.

Having a board of people to review changes before they are implemented sounds great on paper because it implements a segregation of duties which can prevent tunnel vision, but the execution of this requires some nuance to be effective. A common issue in change approval processes is that the individuals with the power to approve or reject changes are too far removed from the development process to understand the implications of these potential changes (DORA). This can lead to an unnecessary focus on formalities of the change versus its real-world complications. An external review from a change advisory board lacks the detailed knowledge of the code, its history, and its dependencies that the developers have (Split.io).

There are several ways that the issues associated with change approval processes can be mitigated to inspire a more effective process. One suggestion may be to put aside significant restrictions, such as a change approval board, in exchange for peer reviews. Peer reviews help still accomplish the goal of segregation of duties while also maintaining that those performing the reviews are well-versed in the development process. Continuous testing and integration is another way to help prevent bad changes because it can result in earlier detection than what might be seen with a traditional approval board. Implementing more security checks at the development stage, rather than the review stage, helps shed unnecessary weight from the deployment process (DORA).

Overall, change approval processes offer important security measures when performed in an efficient and timely manner. By implementing a less-is-more approach, change approval processes can involve less bureaucracy and encourage productivity.

Sources

Split.io. (2020, May 11). Do you really need that change advisory board? Split.io. <https://www.split.io/blog/change-advisory-board-really-needed/>

Atlassian. (2021, October 12). Best practices for change management in the age of DevOps. Atlassian. <https://www.atlassian.com/blog/atlassian-engineering/best-practices-for-change-management-in-the-age-of-devops>

Plat4mation. (n.d.). How to close the gap between DevOps and change management. Plat4mation. <https://plat4mation.com/blog/how-to-close-the-gap-between-devops-and-change-management/>

DORA. (n.d.). Streamlining change approval. DORA. <https://dora.dev/capabilities/streamlining-change-approval/>