Privacy and GDPR in DevOps

Ana-Maria Olteniceanu Ley-Olivia Avila Rojas

April 2023

1 Code of conduct

We certify that generative AI, incl. ChatGPT, has not been used to write this essay. Using generative AI without permission is considered academic misconduct.

2 Introduction

DevOps is a method that combines different practices, tools and cultural philosophies to aid the process of creating and delivering services at a higher speed. It is used by developers and IT teams to help them automate and integrate their work when developing software solutions [1]. Since this method is now broadly used, there are some security issues emerging. The following are, according to the writers of this essay, some of the most important ones:

1. Cloud Security
   To facilitate the management of processes between teams, DevOps uses cloud infrastructure. The cloud platform enables you to run different processes along with minimizing latency, facilitating scalability and reducing cost when creating and developing an application or service [1] [2] . Although the advantages of using the cloud outweighs the disadvantages, there are a lot of security issues and risks when choosing this delivery. The main security issues within cloud computing are of two sorts, data breach and insider/outsider threats. The first mentioned issue is usually due to misconfigurations in the platform or API, and the second is related to lack of understanding of the architecture, platform and code in the cloud system [3].

2. Data management
   As mentioned in the former security issue, the risk of exposing your data through the use of cloud computing is high; this is, however, not the only reason for leakage of data. The DevOps work-chain consists of using many different software tools, and each of these continuously generates large amounts of artifacts and data. Since we are talking about a substantial amount of information, it is crucial that it is managed and stored properly from each stage to mitigate the exposure to outsiders. The data and artifacts generated in each step are presented in figure 1 below [4].

3. Access controls
   The environments and systems used by the DevOps infrastructure are required to have a strongly implemented access control system to protect the system's information. Weak access control can result in an intruder gaining access to the system and in that way steal its data and disrupt operations [5].

4. Security in DevOps
   When delivering a service using the DevOps approach, security is often seen as a reactive measure meaning that it is not implemented at the start of the development process and is not considered a part of the main development process [6]. This results in a staggered or non-reviewed deployment of the service which could result in critical issues.

5. Collaboration
   Building on the previously presented issue, there is usually another team apart from the

Figure 1: Data processed during DevOps Lifecycle

development team that handles the security for the application. This team works separately from the development team and duplicates the operational effort, which could easily be merged together [6]. The lack of collaboration between teams results in slow delivery of the product.

All of the former mentioned security issues of DevOps result in a leak of sensitive data. This data mainly belongs to the system, but in some cases it can also include private personal data. Thus, in this essay we will take a closer look at privacy and how it is handled currently in the context of DevOps. We will also analyse how data and privacy can be handled in accordance to GDPR.

3 Privacy and GDPR

When using the DevOps approach to create software applications, one relies on a high level of information sharing between teams; generally these are the development and operations team. The increased collaboration between teams can lead to unauthorized people gaining access to unrestricted parts of the system, which could entail the risk of potentially harming the system [7]. When it comes to maintaining the privacy of the system while collaborating, there is a dependency between having a reliable cross-team management process and awareness about security and compliance [8].

The vast information sharing also applies during the whole DevOps timeline, when using multiple third party applications. Limits to information sharing are of high importance but organizations usually lack a cloud security architecture and strategy. They are therefore exposing their processed artifacts and data at each stage to the platforms [3].

Despite the high level of unrestricted information sharing, organizations do not tend to have the correct tools and policies to manage the data in a correct manner.[9] In the same way that it is important for organizations to ensure that their service providers are certified to become compliant to GDPR and other security standards, the same can be said about their internal processes. However

it is not necessarily the case that companies take these measures.[9]

It is important to be aware of the different regulations that need to be applied based on the category of the data that's being handled. There are at least four types of data regulation in the cloud and other applications that organizations should be considering, and for the purposes of this essay we will focus on Personal data and, in the European context, on the General Data Protection Regulation (GDPR). The definitions of these are [3]:

- Personal data: all the information that can be used to identify a person.
- GDPR: Europe's data privacy and security law.

As previously mentioned, GDPR is an important privacy regulation and one of its key principles is transparency. This principle requires system administrators to inform individuals about what data is being collected and for what purpose. However, this principle is often not met mainly because of two reasons. Firstly, privacy policies have a tendency of being too long and complicated for regular individuals to understand easily. Secondly, the information provided about the processing of data is often vague and unclear [10].

4 Possible solutions

As is the case with any security issues, there are of course solutions that have been proposed to help alleviate the privacy concerns mentioned in the previous section. These solutions refer to both non-technical solutions (such as policies), as well as technologies that can be integrated into the DevOps workflow. This section will focus on a wide array of solutions for the issues raised previously.

One of the issues that we have mentioned so far has been the amount of information that is being shared by the different development teams, as well as data produced through the DevOps process. The consensus in literature is that in order to tackle these problems, there need to be clear policies set in place. These policies would inform the way in which data is being shared between teams and address how much PII is extracted from users and how it is being processed.[8] This way, all information flow and information processing that is done in development is ensured to be controlled in a way that respects users' privacy.

Of course, such measures need not only be clear and in accordance with legislation, but they also need to be written and conceptualized in such a way that they can easily be implemented into the DevOps lifecycle without (significantly) slowing down development.[11]

The seamless integration of policies into workflow is tied not only to privacy issues, but should be taken into account for all security measures. This integration is paramount to the success of DevSecOps.[9] As has been mentioned before, there tends to be little direct collaboration between development and security teams. In order for this to be solved, there needs to be an integration of these teams' respective pipelines.[9][6] Thus, it will be easier for security to be seamlessly included in every stage of DevOps. Once security is overall increased, privacy will naturally be increased as well.

Once policies have been developed and the security pipeline integrated into development, the right security tools need to be employed. DevOps has a big focus on automation; thus security must follow its lead and use automated tools.[11] They need to be carefully chosen and implemented, otherwise they would not be able to provide the protection needed.[9] The automation of privacy policies could be achieved using methods such as access management, however there has not been much research on the topic.

Privacy is an important aspect of GDPR, however it is not the only one. Developers need to be transparent in their use of PII.[12] For this purpose, there have been tools developed to allow the automation of transparency implementation. One such tool allows for PII to be tagged in the coding process, then for these tags to be aggregated so that a high-level view of personal data can

be generated.[10]

These are some of the solutions proposed in literature. Applying these will result in a better approach to privacy, both in the way it is being handled by the developers and in how it is presented to the users. Thus, it becomes easier for organizations and teams therein to respect GDPR guidelines.

5 Reflection

The issue of implementing security in DevOps is generally worth talking about, even while not considering this essay's focus on privacy. Something interesting that we noticed while researching however, was that the issues of security in DevOps are accentuated when it comes to the topic of privacy.

The traditional approach to security, which is focused on documentation and manual processes/tools, is fundamentally incompatible with the speed that DevOps not only enables but requires. Many teams leave security as an afterthought precisely because it is inconvenient and, if implemented using this traditional approach, slows down development. In these conditions it is unrealistic to modify the rhythm of development so that security can catch up. This means that there is a need for the usage and development of automatic security tools.

This is in many aspects a viable solution and many automated tools exist; however, privacy is an interesting case. Other aspects of security, such as tests, detection of suspicious behaviour in a system etc. are more easily automated. Privacy, as it is in its essence an ambiguous and subjective concept, cannot be easily quantifiable and put into a framework that is ready for automation. The same issues apply to the concept of privacy by design, as many developers do not necessarily consider privacy and other related concepts in the planning or coding phases. This type of considerations needs to be made on a case-by-case basis.

These are intrinsic issues in DevOps and cannot be tackled unless we develop the right technologies that allow the integration of privacy into the development cycle. Furthermore, the lack of education and concern that developers have in regards to security in general (not just privacy) is yet another hurdle. It contributes to ignorance on the topic, which has a severe impact on development. There seems to be an attitude among many development teams that security is very complex and hard to implement, thus they leave all security matters to the security team at the end of the cycle. This makes security much harder to implement in an effective manner.

The fault does not fall only on the shoulders of developers, however. It can be easy for security specialists to give the advice of creating clear privacy policies, but it is a lot more difficult to implement these in practice, especially when taking in consideration the automation requirement that is imposed by DevOps. Thus, security engineers need to be in communication with the development teams to implement security policies and measures that are not only compatible with the workflow, but that do not affect usability and that are understandable by the developers.

6 Conclusion

In conclusion, privacy in DevOps is a complex topic, and the improvement of privacy (using GDPR as a set of requirements) is a multi-faceted issue. In that sense, there are many areas of DevOps which are of interest when it comes to enhancing privacy.

There are many areas in the DevOps cycle where data (including sensitive and personal data) can be leaked. This can include the flow of information within a large development team as well as between different teams. Furthermore, the overall security of an application must be considered, as poor security increases the possibility of a successful cyber incident. This, in turn, can result in data leaks among many other issues. We would like to specifically point out that not only the robustness

of code that is being developed, but the reliance on 3rd party applications and microservices (cloud, for instance) should be questioned and analysed. Despite of (or maybe due to) the number of issues that surround privacy in DevOps, literature and research surrounding this are quite sparse. This leads to a lack of education and awareness, which directly impacts how dev teams tackle privacy. The lack of automated tools required for the successful integration of privacy can also be attributed to this.

Nevertheless, privacy in this context is a field of research that is slowly gaining popularity. There are already some promising research roadmaps that have been developed, as well as tools that try to alleviate the problem of automation (although what we have found is more focused on transparency) and that seem to have potential going forward. We hope to see more and more researchers and developers taking an interest in privacy within DevOps, as well as developing novel ways of ensuring that privacy is taken into account and that GDPR guidelines are being followed.

References

[1] Atlassian, "What is devops? — atlassian," 2016.

[2] C. N. Wiki, "Cloud devops: 3 ways devops and the cloud work together."

[3] Snyk, "Cloud security challenges in 2022."

[4] A. Capizzi, S. Distefano, and M. Mazzara, "From devops to devdataops: Data management in devops processes," in *Software Engineering Aspects of Continuous Development and New Paradigms of Software Production and Deployment* (J.-M. Bruel, M. Mazzara, and B. Meyer, eds.), (Cham), pp. 52–62, Springer International Publishing, 2020.

[5] Hackerone, "Devops security: Challenges and 6 critical best practices — hackerone."

[6] D. S. Battina, "Best practices for ensuring security in devops: A case study approach," *SSRN Electronic Journal*, vol. 4, pp. 38–45, 11 2017.

[7] A. A. Ur Rahman and L. Williams, "Software security in devops," *Proceedings of the International Workshop on Continuous Software Evolution and Delivery - CSED '16*, 2016.

[8] V. Mohan and L. B. Othmane, "Secdevops: Is it a marketing buzzword? - mapping research on security in devops," in *2016 11th International Conference on Availability, Reliability and Security (ARES)*, pp. 542–547, 2016.

[9] S. Nägele, J.-P. Watzelt, and F. Matthes, "Investigating the current state of security in large-scale agile development," in *Agile Processes in Software Engineering and Extreme Programming* (V. Stray, K.-J. Stol, M. Paasivaara, and P. Kruchten, eds.), (Cham), pp. 203–219, Springer International Publishing, 2022.

[10] E. Grünewald, P. Wille, F. Pallas, M. C. Borges, and M.-R. Ulbricht, "Tira: An openapi extension and toolbox for gdpr transparency in restful architectures," in *2021 IEEE European Symposium on Security and Privacy Workshops (EuroSPW)*, pp. 312–319, 2021.

[11] R. Mao, H. Zhang, Q. Dai, H. Huang, G. Rong, H. Shen, L. Chen, and K. Lu, "Preliminary findings about devsecops from grey literature," in *2020 IEEE 20th International Conference on Software Quality, Reliability and Security (QRS)*, pp. 450–457, 2020.

[12] European Parliament and Council of the European Union, "Regulation (eu) 2016/679 of the european parliament and of the council of 27 april 2016 on the protection of natural persons with regard to the processing of personal data and on the free movement of such data, and repealing directive 95/46/ec (general data protection regulation)." Official Journal of the European Union, 2016.