Shift Security to the Left for Effective GDPR Compliance... and Better Code

ASSEMBLA + hackerone

On May 25, 2018, the biggest change to data protection in 30 years, the General Data Protection Regulation (GDPR), will go into effect. GDPR is intended to protect the data of European Union consumers and as such, applies to any companies handling the data of EU citizens, regardless of where the company is located or how innocuous the data might seem.

The data protection requirements under GDPR are strict and the repercussions of failing to meet them can threaten the livelihood of any business. With the potential for $25 million in fines on the table and only 72 hours to report a breach, companies who collect data that can *"directly or indirectly identify"* EU citizens have a lot riding on their ability to meet the demands of the new regulation. However, a recent SAS study showed that *"only 45 percent of organizations have a structured process in place to comply with GDPR."*

GDPR has implications that affect not only how organizations handle consumer data, but also how organizations respond when consumer data is compromised. To prepare for GDPR, it's necessary to paint an end-to-end picture of your security apparatus. That means doing your best to stop breaches (by securing the code) before they happen, but also having a plan for when they, inevitably, do.

In this guide, we will cover how you can protect your code, the key GDPR articles you should dig into, and how you can formulate a plan for when vulnerabilities are discovered.

What we'll cover:

Secure Code Earlier

According to the Department of Homeland Security, it's estimated that 90% of security incidents are the result of defects within the code. Thus, preventing breaches is a question of code security, but not all code is made equal.

Most organizations run on a mixture of code that has been around and iterated on for years as well as code written in the last few months. The trouble is that security and software are ever-changing, so code written even a few years apart was created in very different security contexts and development environments. In turn, this means most applications contain code born on different version control systems and repository types, as well as a mixture of code built on-premise and in the cloud. The burden is on CIOs to balance a security strategy that can house old and new code, and that's no small feat.

Newer code may have the advantage of being built on systems with more sophisticated security tools, but a key factor is missing in many organizational security strategies today. While security spending is on the rise, so is the prevalence and severity of breaches. That's because security spending is focused almost solely on post-deployment issues, rather than on addressing vulnerabilities before they become breaches.

No piece of code will ever be 100% breach free, but security measures can still be shifted left, or focused on earlier in the development lifecycle to prevent breaches. Preventing a breach is always going to be easier and more affordable than dealing with a breach once it's already taken place.

As soon as code is born, it should be reviewed by as many people and systems as possible without compromising innovation or mobility. Tools like static code analysis help teams identify and prioritize vulnerabilities as early as the code creation stage, thereby preventing breaches. On the other hand, legacy code that was built before the rise of the cloud and DevOps likely hasn't been vetted in modern security checks or even been subject to the same levels of collaboration that cloud-based tools provide.

But that doesn't mean older code can't take advantage of newer security systems. If your organization incorporates code written years ago, take advantage of cloud offerings and run static code analysis or vulnerability testing. Organizations have an opportunity to improve their security strategy by shifting security focus and spending earlier in the software development lifecycle (SDLC), as well as subjecting older code to the benefits of newer security tools.

Publish a Vulnerability Disclosure Policy

Your teams are increasingly under pressure to deliver software faster and more often. GDPR will not only add to that pressure, it may also force you to rush the release of new code necessary for compliance. And while vulnerabilities will always be a fact of life, changes forced by GDPR increase the likelihood that errors, bugs, or omissions make it into production.

Securing your SDLC — moving security to the left — is an effective step towards reducing risk. But it doesn't have to slow you down, because there are options that support both compliance and high availability. The result is faster development based on secure repositories that integrate with your existing services, so it works like you already work.

Being aware of vulnerabilities earlier is critical, since GDPR compels organizations to disclose data breaches in 72 hours. That's an unreasonably short time, especially if the bug facilitating the breach requires significant resources to patch. But if you find a bug on your own, there's no need to disclose it at all. Again, moving security to the left is the key.

The best way to identify bugs before they can be exploited is by getting as many eyes as possible looking for vulnerabilities. Those eyes can be your internal resources, external consultants, hackers or researchers, or just a random person who stumbles across a potential issue. Giving those finders a clear path to alerting you of potential vulnerabilities is where a vulnerability disclosure policy (VDP) comes in. Every organization should already have a VDP in place that's easy to find and easy to follow. Unfortunately, that's far from the reality: research has shown that 94% of large companies don't have obvious VDPs.

VDPs can also help alleviate some of the operational impact of GDPR. For example, GDPR article 32, 1.d states that organizations must implement *"...a process for regularly testing, assessing and evaluating the effectiveness of technical and organisational measures for ensuring the security of the processing."* A proven method for implementing regular (and continuous) security coverage is through a VDP (with a more direct method being a bug bounty program).

In addition to helping with GDPR compliance, VDPs are being promoted — and sometimes mandated — by numerous organizations, including the Department of Defense, Food and Drug Administration, National Highway Traffic Safety Administration, and Federal Trade Commission.

Leaders across business and government are also touting the importance and benefits of VDPs more and more. From the cyber security lead at General Motors to the European Commission's Security Union Commissioner to the current U.S. Deputy Attorney General, you can read what they're saying in this "Voices of Vulnerability Disclosure Policy" presentation.

These organizations and leaders are also making it easy for you to get started developing your own VDP, such as with this template provided by the National Telecommunications and Information Administration. HackerOne also provides this complete guide for crafting an effective VDP.

How to Prepare for GDPR's Action Items

In addition to shifting security to the left and getting in front of vulnerabilities, there are a number of specific action items outlined in GDPR that require careful attention. From data handling to activity logs, these are the action items explicitly listed in the GDPR that need to be addressed, so as you're getting ahead of bugs, start getting ahead of these requirements.

How are you going to let people access their data?
Article 16 states that *"the data subject shall have the right to obtain from the controller without undue delay the rectification of inaccurate personal data concerning him or her."* This means that all organizations must have a process in place for users to quickly update inaccurate information about themselves.

How are you going to protect the data?
Next, Article 25 states that *"the controller shall, both at the time of the determination of the means for processing and at the time of the processing itself, implement appropriate technical and organisational measures, such as pseudonymisation, which are designed* ***to implement data-protection principles, such as data minimisation****, in an effective manner and to integrate the necessary safeguards into the processing in order to meet the requirements of this regulation and protect the rights of data subjects."*

This article directly refers to protecting the identity of users. It goes on to stipulate that these safeguards must be put into place to protect data from all angles: at rest, in transit, and in use. That means ensuring that sensitive data is encrypted and organizational access to that data is as limited as possible. In some cases, this can also mean the need to anonymize users. This should be applied to any data that can identify, even indirectly, a user or customer.

How will you maintain an audit trail?
Article 30 deals with record keeping and states that "*Each controller and, where applicable, the controller's representative, shall* ***maintain a record*** *of processing activities under its responsibility."* For security teams, this means that they must deploy real-time auditing capabilities and capture usage details. Logging activity, auditing, and monitoring all personal data processing across your applications is key to meeting the demands of this article.

How will you manage data moving outside of the EU?
Article 46 states that *"In the absence of a decision pursuant to Article 45(3), a controller or processor may transfer personal data to a third country or an international organisation only if the controller or processor has provided appropriate safeguards, and on condition that enforceable data subject rights and effective legal remedies for data subjects are available."*

This article explicitly addresses the need to protect European citizen data when it is transferred outside the EU. To comply with article 46, security teams must look at security tools that are applied at the data level. All data should be encrypted even when being transferred and partners must be

authenticated and authorized using TLS, SAML2 and SSO strategies.

Time is Running Out

GDPR is coming, and that's not intended to scare you. It's a fact, and you need to be well on your way to not only planning for compliance, but putting the processes and procedures in place to facilitate compliance. That likely means changes to code, personnel, team structures, and more.

Here are a few more resources to check out as we all move towards May 25, 2018:

- The United Kingdom's Information Commissioner's Office provided "12 steps to take now" to get ahead of GDPR.
- The US Department of Homeland Security explains how they run their own Software Assurance program.
- A survey by SAS shows that less than half of organizations don't have a GDPR plan in place.
- HackerOne offers "3 things you should do today" to prepare for GDPR.

www.assembla.com