Security by Design

Elena-Denisa STROE
Faculty of Electronics, Telecommunications and Information Technology,
University POLITEHNICA of Bucharest, Romania
elena_denisa.stroe@stud.etti.upb.ro

Abstract

The security should be an area that can cover multiple technical disciplines that needs to be focused on customers and to try protecting against different threats. There can be multiple disciplines that can be part of the security and those can be: assurance, anti-tamper and information assurance and cybersecurity. Security must be taken into consideration throughout the entire product lifecycle in order to maximize the protection of a system. The purpose of this article is to highlight design security flaws which should always be considered as part of the design flow for an application or a product. The recommendations can be applied in combination with different methodologies, depending on what the company chooses to use, wheatear it is Agile or Waterfall. Principle of security by design will be tackled within the article.

Index terms: design, OWASP, security, web application

1. Introduction

In software engineering, security by design represents a huge impact on the projects of an organization, being incorporated into the product from the beginning, the purpose remaining to create products functionally secure. Security by design is increasingly becoming the most desired development approach to ensure security and privacy in software systems. Companies or any parties which are involved in developing different projects consider and build security into the system at every layer using a robust architectural design. When the security design is taken into account, there is a full architectural design made for this in order to take the best decisions on well-known security strategies, tactics and patterns, defined as reusable techniques for achieving specific quality level.

Companies often expose themselves to risks while experimenting new or advanced technologies. Software development is touching new heights every day, hackers also develop cutting-edge methods to breach cyberspace defenses. Thus, traditional approaches like Vulnerability Assessment and Penetration Testing are insufficient to address the security of the cyber system. It is essential to use ground-breaking methods like security by design, which provides to developers the knowledge to manage the delivery operations and the development testing at any moment for potential flaws. In order to make a system robust when it comes to safety, the security by design is an approach to software and hardware development that seeks to make systems with no critical vulnerabilities and less prone to attacks, through measures such as continuous testing, authentication and best programming practices.

2. Security by Design (SbD)

Security by design is a methodology to strengthen the cybersecurity of an organization by automating its data security controls and developing a robust IT infrastructure. This approach focuses

on implementing the security protocols from the basic building blocks of the entire IT infrastructure design [1].

Although it's not a new concept, the expansion of public cloud usage has made security by design far simpler to be applied. In practice, security by design is about standardized coding, reusable, automated architectures so that your security and audit standards remain consistent across multiple environments.

When a software project is built, the focus has to be on the absence of vulnerabilities, otherwise the production deadlines might not be met or customers might be affected. With contextual knowledge, it is easier to choose the right components, for example, that can reduce risks and mitigate cyberthreats. Below are some objectives that should be taken into consideration when using SbD:

Table 1. Data & Purpose of the system

Objectives	Data System	Purpose of the System
Information to be considered	• What data are you're trying to use? • Where is that data going to go? • Who will use the data or interact with it?	• How are different components connected? • What are the requirements and implications if something goes wrong

3. Principles of Security by Design

- Minimizing Attack Surface Area: this means that in order to minimize attack surfaces, developers should ask what their system is supposed to do, as well as what it should not. It's also important to anticipate the attack vector or any other vulnerability that might potentially compromise your system.
- Least Privilege: users context is only providing the necessary information for their level of access, by establishing certain permissions, in order to prevent unauthorized access to sensitive data.
- Least Common Mechanism: advises against sharing system mechanisms among users or programs that do not require them, in order to function according to initial specifications.
- Separation of Duties: to prevent conflict of interest, wrongful acts, fraud, abuse or errors.
- Defense in Depth: any developed security system is prone to failure and therefore the best approach is to layer the security measures and deliberately overlapping their coverage in order to keep the system safe even when one security measure has failed. Additionally, a notification system should be in place in order be informed when a mechanism was deceived and the system is at risk.
- Failing Securely: depends on the eventuality that the system will fail, hence the need to design an architecture that allows failing without leaving any exposure [4].
- Open Design: The principle of open design states that the security of a mechanism should not depend on the secrecy of its design or implementation. A system that relies on a novel language or method so unusual that no one can currently understand it can and may still be open, in which case the attacker has immediate and full access to the system.

4. Example of security by design

4.1. Failing Securely

- Wrong email log in: In this case the email introduced is incorrect so an error is thrown. The email can be tried for 5 times.

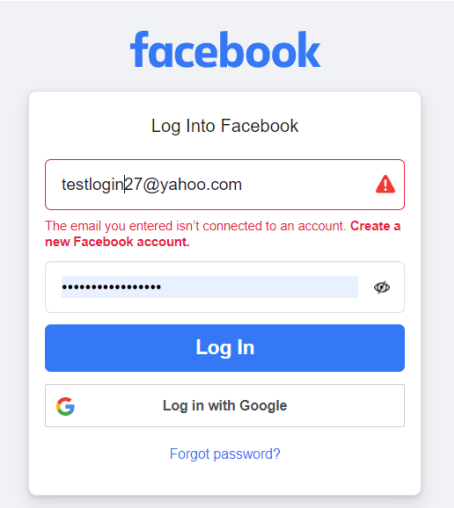

Fig. 1. Wrong email

- Wrong password log in: In this case the password introduced is incorrect so an error is thrown. The password can be tried for 4 times.

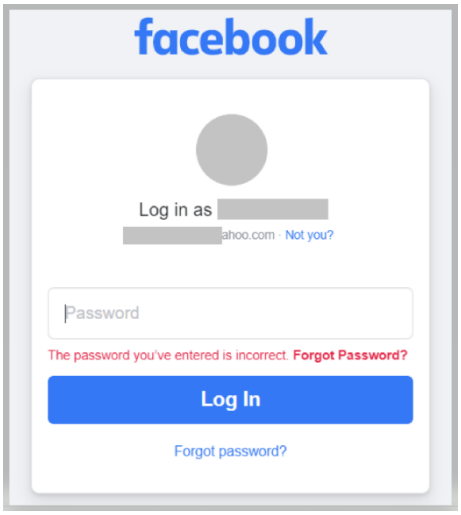

Fig. 2. Wrong password

In this case it is considered the log in page for an application. Sometimes it happens to introduce the wrong data: the email or the password. In case a wrong email will be introduced, meaning that the email doesn't exist at all, then the applications will return an error to highlight that the email is not in the database. Attackers might attempt to guess the email 5 times, so at the 6th attempt the page will be blocked and in that session used, cannot connect to any other email address.

In case attackers fill in an existing email signed up for Facebook, then the next step is to introduce the password. It can be guessed 4 times and at 5th try an error message will be displayed for wrong password. This kind of behavior is displayed in Figure 2.

Regardless of the reason of failure, sensitive user information and system errors should not be exposed. This principle states that only limited information should be shown when errors are encountered by the system. As underlined in Figure 2, when the password is not matching previously inserted user, a dedicated message is being prompted. The problem is that attackers can gather information related to existing accounts by user enumeration due to the fact that if the provided user name does not have an associated account, their messages are different. Attackers might attempt guessing the password 4 times.

4.2. Minimize Third-Party Access

For web applications, making use of the services of third-parties can be convenient for additional functions or data. However, these external parties have different security measures that may or may not be more secure. When an organization agrees to collaborate with other party, then it should take into account that those parties might have different cyber security measures, so it can lead to vulnerability to cybercriminals who might gain access.

4.3. Keep security simple

Contrary to popular belief, keeping the application's security simple is a better option than having complex designs. When organisations use complex systems, then those are very hard to maintain and correct in case of an undesired event. Troubleshooting can be time-consuming which puts the application at further risk.

5. OWASP recommendations

OWASP (Open Web Application Security Project) is an online community that produces free tools, documentation, articles, and technologies to help people secure their websites, web applications, and network resources. That was created for helping developers building highly secure web applications [6].

OWASP suggests that programmers create security controls that are appropriate for managed data value. For example, an application processing financial information must have binding restrictions, compared to a blog or a web forum.

6. Conclusions

As cybersecurity becomes an increasing area of concern for critical infrastructure providers, governments, and private enterprises, it requires greater attention from both management and development team members. Successful implementation will require action from multiple groups and at multiple levels. Security features should be designed into a system so that both human and software vulnerabilities are minimized. In addition, each component of a system should also be secured separately so that if a breach does occur, any damage is going to be limited, and it won't impact and spread through the entire environment.

The principles that guide the security by design approach could differ from one organization to another. But the OWASP listed some principles that programmers should take into account, the one presented in the article. With these in mind, they can design secure products. Below are four security by design principles.

References

[1] https://blog.unguess.io/what-is-security-by-design-the-best-approach-to-cybersecurity
[2] https://www.spiceworks.com/it-security/cyber-risk-management/articles/what-is-security-by-design/
[3] https://www.logicworks.com/blog/2017/01/what-is-security-by-design/
[4] https://www.techslang.com/definition/what-is-security-by-design/
[5] https://learn.microsoft.com/en-us/azure/architecture/guide/security/security-start-here
[6] https://patchstack.com/articles/security-design-principles-owasp/