

How to integrate Security by Design

02 September 2025, 10:25

Tatiana Galibus

A step-by-step guideline to increase built-in resilience in modern software development

Two generally known facts. One: cyberthreats are increasing in quantity and complexity. We have no choice but to enhance our cybersecurity efforts. Two: it is easier, more powerful and more cost-effective to embed functionality in your software from the start than to add it on when it’s already fully developed.

Combine these two and you immediately understand the importance of Security by Design. It can help you mitigate risks early, reduce costs, and build user trust. Add to this the Cyber Resilience Act and other legislation forcing companies to fully integrate security in their development efforts. Security by Design is thus no longer optional for software developers, it is a necessity.

But how do you evolve from your current development practices to a design and development environment that understands and integrates security throughout the development lifecycle? The step-by-step approach below may be a handy guideline.

What is Security by Design?

Security by Design is the principle of integrating security from the earliest stages of system development, instead of patching it on later. This means considering potential threats and mitigation strategies from the first design decisions, and throughout the software development lifecycle (SDLC).

It helps you build digital environments that are resilient, compliant, and trustworthy, whether you're building a startup SaaS product or managing a large-scale infrastructure.

Meet Acme, our lifelike example

To make the typical security by design workflow more insightful, we describe the trajectory of an imaginary small software company, Acme, that works on building a collaborative to-do list web app for distributed teams called AcmeTasks. They want to ensure their software product is secure, scalable, and compliant with upcoming EU regulations such as NIS2 and the CRA.

Understand the context

Before embarking on the development journey, Acme needs to clearly understand their business reality and what they intend to achieve. They formulate a clear answer to the questions below:

- **What?** They want to build a 3-tier web app (frontend, backend, database) for managing to-do lists.
- **For whom?** This web app's end users include remote workers, small teams, and freelancers.
- **Using which data?** Email addresses, user tasks, attachments, optional billing info.
- **With which constraints?** The web app needs to be CRA-compliant by 2027; and it needs to be made by a limited team of 6 developers.

Define security & business objectives

Acme should understand that the business and security should be completely aligned. In other words: security must support the business goals, not hinder them.

The business objectives: Acme has defined include the launch of an MVP (minimum viable product) within 6 months. The product must be GDPR and CRA compliant for the product to be allowed on the European market. And it should be resilient enough to build customer trust and thus to attract B2B clients.

This translates into the following **security objectives**: Acme needs to protect user data from leaks or tampering and prevent unauthorized access to task lists. But they also need to ensure that the service remains available, even with a heavy workload and unusually high web traffic.

Perform threat modeling (with STRIDE)

If you want to adequately protect your digital assets, you need to understand which threats you must protect them from. This can be done using threat modeling, a method that helps you identify how an attacker might target your system. Acme has decided to use STRIDE, one of the most accessible methods, which focuses on 6 attack types.

The STRIDE analysis of their AcmeTasks application led to the identification of the following threats:

- **S - Spoofing:** an attacker gains access by impersonating a user via an insecure login
- **T - Tampering:** one user maliciously edits another user's tasks
- **R - Repudiation:** a user deletes tasks and denies deleting them
- **I - Information disclosure:** task data are sent without encryption in plain HTTP and therefore readable by third parties
- **D - Denial of Service:** the app's API gets overloaded with task creation spam
- **E - Elevation of Privilege:** a regular user gains admin access

Define security requirements

Based on the above threat model, the Acme developers can now outline security requirements such as:

- All data transmission must use HTTPS
- Use OAuth 2.0 for authentication
- Apply role-based access control (RBAC)
- Implement input validation and output encoding
- Log sensitive actions for traceability
- Limit the number of allowed request rates (rate limiting) to mitigate DoS
- Encrypt user data at rest with AES-256

When these requirements are met, the various types of threats are severely limited and the resulting product will be significantly more secure.

Integrate into the SDLC

Secure development and design are a good start, but it doesn't end there. Security isn't a one time shot, it is a journey. Acme's secure development and design need to become part of an entire Software Development Life Cycle (SDLC), that consists of the steps below:

- **Design:** where the threat modeling and design of the security patterns (e.g. RBAC) are conceived
- **Development:** using secure coding standards, such as the ones listed in the OWASP Top 10
- **Testing:** including static and dynamic analysis, dependency scanning …
- **Deploying:** when rolling out the product, special attention needs to be paid to a.o. secrets management and CI/CD hardening
- **Maintaining:** last but not least, the entire lifecycle includes years of maintaining a running system. Identified vulnerabilities and breaches should be handled swiftly and firmly, using patch management and incident response and they may lead to an even 'securer by design' product in the future

Train your team

A secure digital environment will always require a thorough security-awareness. Employees, end users and developers/designers alike, need to understand the various threats and how to deal with them. And, as the threat and security landscape continuously evolves, a security culture will always rely on continuous learning.

Acme understands this and has taken a number of initiatives to keep their teams up to speed:

- They conduct onboarding training in secure development
- They run threat modeling workshops and code review exercises
- Their design and development team members subscribe to security bulletins (e.g., CVE feeds) to follow up on the latest news and trends

One more piece of advice: you should use platforms like CyberActive if you aim for EU-aligned cybersecurity training focused on the CRA, NIS2, and secure coding best practices.

Monitor, update, improve

Acme knows that the maintenance stage consists of more than a mere follow-up of any reported security issue. If they want their application to remain trusted, they need to continuously and actively monitor, respond, and iterate. Their proactive security policy includes:

- Set up logging & monitoring (e.g., ELK stack, Sentry)
- Audit logs for suspicious activity
- Regularly scan for vulnerabilities
- Review and revise threat models every quarter
- Stay ahead of CRA deadlines and certification steps

Ready to follow in Acme’s footsteps? Get in touch!

Security by Design is about resilience, trust, and long-term viability. For a company like Acme, investing early in structured practices means they can scale safely, win trust, and comply with the evolving legal landscape.

The same goes for your organization. Whether it involves a one-off development effort or a continuous maintenance of an ICT infrastructure, you should grant security the consideration it deserves. It should be embedded into your daily business operations, from the design culture up to the bug tracker.

And we can help you achieve these objectives.

You can learn how to make your company more compliant and resilient against all forms of threat by following our Lunch & Learn trajectory.

You can reach out to the SecDes (security by design) project for SaaS developers and providers.

Or you can find out more about any of the steps discussed above or about guidelines and tools for CRA compliance and security by design: contact tatiana.galibus@sirris.be and she will gladly

provide you with the needed information.

Standards and frameworks for Security by Design

Security By Design has become a major requirement in the development of software and digital environments. Developers and designers are required to conform to certain standards, but they are also offered many useful frameworks and tools to guide them in their security efforts. Below is an overview of some of the most important standards and frameworks that support the Security by Design approach:

- **CRA - Cyber Resilience Act (EU, 2024):** the newest legislation mandating cybersecurity for digital products throughout their lifecycle
- **NIST SP 800-160:** systems security engineering guidance on integrating security into engineering processes
- **OWASP SSDLC:** secure Software Development Lifecycle practices
- **ISO/IEC 27034:** application security framework
- **Microsoft SDL:** security development lifecycle used internally by Microsoft
- **ENISA guidelines:** EU-level cybersecurity recommendations

Build security in from the first line of code

Do you want to develop secure applications that comply with the latest regulations such as NIS2 and the Cyber Resilience Act? This masterclass provides practical insights and hands-on experience with security by design and threat modeling. Take a confident step towards building robust, future-proof software.

Register for the masterclass

Authors



Tatiana Galibus