



Doing Business Securely

Did you know that SDLC can save you money? Even up to a factor 200!

Building software that meets non-trivial security requirements is a challenging endeavour. From a technical point of view, software is a complex combination of various components and technologies, and the attacker's toolbox is continuously growing. From a project management perspective, a late discovery of important vulnerabilities and risks might have a significant impact on project delivery time and development cost.

Building security into software requires the right attitude starting from the initial phases in the development lifecycle. It is about thinking in advance rather than thinking after-the-fact. It is equally well about increasing the security quality of your software, which is strongly related to the overall quality of your software product. When properly performed, it will enable you to truly manage the risk and the associated cost of your development projects.

This flyer explains the secure development lifecycle concept, how it can help you in addressing typical problems in secure software, and what Aszure can do for you in this context.

Software security: a reality check

Few organizations succeed in effectively producing secure applications in an efficient manner. One of the main reasons is that software security is commonly thought of as the mere execution of a penetration test at the end of the development lifecycle, in combination with a penetrate-and-patch strategy after deployment. This is a capital mistake, for the following reasons:

- Two sorts of security vulnerabilities exist in software: bugs (implementation errors) and flaws (design problems). Statistics show that software, on average, contains an equal amount of bugs and flaws. Penetration tests are well suited for finding bugs, much less for finding flaws.
- The cost of fixing a vulnerability depends significantly on the time of discovery. Statistics, again, show that the difference in cost between fixing a vulnerability early in the development lifecycle or after deployment can be up to a factor 200! [B. Boehm, 1988]

An appropriate secure software development lifecycle (SDLC) is a critical success factor when it comes to protecting your organization's information in an efficient and effective way. In fact, Gartner states that "Through 2010, organizations with a proper SDLC will experience an 80 percent decrease in critical vulnerabilities".

So, what is SDLC about?

SDLC is a strategic approach for raising assurance in the final security stance of the developed software, and the quality thereof. The basic idea is the augmentation of a traditional software development lifecycle by injecting security-specific techniques and activities in order to construct, evaluate and manage the security properties of the software being developed. If implemented properly, security will no longer pose a risk to a development project, as it is often the case, but become a project- and business enabler.

About Aszure

Aszure is specialized in operational risk management consultancy and staffing. Operational risk management, information security, business continuity and compliance to laws and regulations are major cornerstones of our services.

Aszure assists its customers with the incorporation of operational risk management at all levels and in all areas of their organization, including day-to-day operational decisions. No decision should be taken in any organization without applying proper operational risk management. It can be applied throughout the entire life cycle of any activity or project. In order to minimize risk, it is however preferable to apply operational risk management at the beginning phase of a project or operational activity.

About Aszure Academy

Aszure offers, through the Aszure Academy, an extensive education and awareness program, covering several open classes, events and customized education experiences.

Contact

For further and additional information, please contact us on Tel. +32 (0)9 243 10 20 or info@aszure.com

On all Aszure services our general terms of delivery apply.

Organization-wide benefits

A systematic approach to software security will provide your organization with a number of key benefits.

1. The **security quality** of your software will undeniably increase. In fact, you will gain assurance that security controls are properly designed, implemented and continuously monitored in your software systems.
2. By anticipating potential security vulnerabilities and by addressing them as soon as possible, the effort required to address them will be minimized, which will optimize the **cost efficiency** of the development effort.
3. SDLC will help you achieving **compliance** with laws and regulations. And, in fact, SDLC is a required part of particular security standards (such as PCI DSS).
4. Improved security quality will be appreciated by the customer, safeguarding the **company's reputation**, and SDLC itself can be used as an (informal) quality label.

The SDLC Scene

Over the last years, a number of SDLC's have been published, including Microsoft SDL, OWASP's CLASP and McGraw's TouchPoints SDL's. More recently, some of these SDLC's have evolved towards maturity models such as OWASP's SAMM and McGraw's BSIMM.

We have published an in-depth study of a selection of these models. This study has revealed, among others, that these models should not be applied blindfoldedly: they are best mixed and tuned to the particular organization, in-line with the risk it faces, and tailored to the particular development process and -environment.

Aszure's SDLC approach

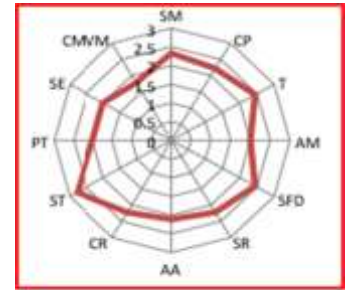
Aszure offers a phase-wise approach to addressing secure software development. Ideally, these phases are all fully executed, sequentially. Depending on your organization's context, it might however make sense to execute only selected parts.

Phase 1: SDLC Assessment

The idea of this first phase is to get an understanding of the current stance of the organization regarding secure software development. Based on interviews with relevant stakeholders and other targeted assessments, the current state-of-practice within the organization is being analyzed and synthesized into an as-is situation. In addition, in order to understand the relative performance of the organization, a maturity modelling is performed based on models such as SAMM or BSIMM (see figure on the right). This will provide insight into the strengths and weaknesses of the organization with regard to secure software development. The outcome of this phase will, among others, provide you with the arguments to justify a more extensive SDLC trajectory within your organization.

Phase 2: Strategic SDLC Improvement

In this phase, the fundamentals of a successful SDLC implementation are laid down. First of all, an awareness program is being started in order to inform the key stakeholders of the upcoming SDLC project and plans. Then, based on the outcome of the previous phase, an SDLC target and roadmap are being defined. The former is an ideal model (the to-be situation) that will be aimed for. Since experience has shown that a gradual improvement strategy is much more effective than a "big-bang" approach, the roadmap assigns priorities to the different target components and defines a timeline for their actual implementation.



The target and roadmap are then further concretized into a (continuous) improvement program, detailing the different goals with KPI's using concrete metrics (which will allow you to monitor the progress of the project), the process used to introduce and manage the different changes, a selection of case studies that will be used as study objects and so forth.

Phase 3: In-depth SDLC Improvement

In this phase, the different improvements defined in the previous phase are being implemented. As this step goes beyond the existing SDLC models, this step requires a lot of expertise. Specific implementation techniques include the definition of an appropriate method, the instalment of an adequate tool (including selection, implementation and tuning), the increase of specific knowledge (e.g., organization-specific security frameworks or coding guidelines) and so forth. It is clear that this phase requires close collaboration with relevant stakeholders. Therefore, this phase is typically executed using a coaching or SPOC model.

SDLC Ecosystem

Finally, it is important to note that SDLC is not just about defining a process for secure software development. In fact, there are four fundamental cornerstones to any successful SDLC trajectory: people, process, tools and knowledge. The combination of the different phases discussed above ensures that these cornerstones are addressed adequately.

Interested in world-class education and training on SDLC: visit the website of the Aszure Academy (www.ascureacademy.eu).

For more information, please contact:

Tel.: +32 (0)9 243 10 20

E-mail: info@ascure.com www.ascure.com

Bijenstraat 16-17, B-9051 Ghent, Belgium