BEST PRACTICES FOR ORGANIZATIONS TO ACCOMPLISH SECURE SOFTWARE

Grip on Secure Software Development

***Grip on Secure Software Development* is a collection of guidelines regarding cooperation to attain secure software. It is a coproduction by dozens of specialists, clients and suppliers of software under the banner of the CIP: the *Centre for Information security and Privacy*. Rob van der Veer, co-author of Grip on SSD and principal consultant at SIG, explains.**

The guidelines describe the relation between client and supplier, and help a client describe clear and measurable security requirements. The guidelines are publicly available via www.gripopssd.org, and are maintained by representatives from over 20 organizations, including Capgemini, IBM, UWV, PostNL, CIBG, Dictu, Dutch Tax Authority, SVB, Dutch Ministry of the Interior, OWASP, CGI, Sogeti, Ordina, Software Improvement Group, DKTP and Valori. In addition to guidelines, also training material, a testing methodology and template contracts are available for use.

A particular feature of the cooperation between the organizations within CIP is that each has signed a manifest committing the organisation to backing and following the Grip on SSD guidelines. Exceptions are allowed, but only if these are shared with the other participants. Those exceptions are an interesting starting point for extension and improvement of the guidelines. The cooperation therefore results in sharing of knowledge and expertise. One example is the Cloud Security Alliance, working to produce security guidelines specific to cloud environments. The Grip on SSD guidelines are therefore primarily a means of communication, and participation of the various parties ensures continuous improvement. As a result, organizations do not have to reinvent the wheel: guidelines for dealing with security in the context of suppliers, contracts and software projects are readily available. And suppliers appreciate having a standard way of working with several of their clients.

How Grip on SSD started

At SIG, I help organizations with software quality issues, such as “Is this system maintainable?” or “Does it scale?” As a part of that work, I also assess software security, and to my surprise I often find that some very elementary mistakes are still being made. SIG measures software security on a scale from 1 to 5 stars, and we often encounter systems that score 2 stars. Our investigations show that this is typically due to unclear agreements on security and a lack of proper reviews and testing. Many organizations are currently looking for ways to improve security of the software they use.

“It’s good to have a set of rules to attain security right from the start of a development project.”

Dion Kottemon
Former Government CIO

In the beginning of 2013, I started discussing this topic with CIP. Together, we wanted to tackle this problem. Specifically, we saw a need for guidance on how client and supplier can work together to improve software security. We initiated a working group and started writing. The main contributor was Marcel Koers, the CISO of UWV at that time, but many other specialists assisted him. Clearly, we did not need to reinvent the wheel, because a wide array of security knowledge had already been recorded in existing standards and frameworks. The challenge was to summarize the available information in such a way that organizations could immediately start to use it. We accomplished this by grouping the suggested practices under maturity levels – everyone can do it.

The Method

Grip on SSD describes how a client can gain control over developing secure software, either with an internal IT department or with an external supplier. The three pillars of the method are 1) standard security requirements, 2) contact moments, and 3) defining the necessary processes. These processes include risk management, requirements maintenance and increasing the maturity level of the organisation. To define the requirements, a new method for defining audit frameworks was used (SIVA); this way the requirements are meaningful for managers, developers, testers, auditors and security specialists. The requirements are based on the NCSC guidelines for web applications. Each requirement is linked to related guidelines from NCSC or OWASP ASVS (Application Security Verification Standard). In addition, for each requirement the responsible party is indicated: client, supplier or other parties, e.g. a hosting party. In the meantime, IBM has translated the (originally Dutch) requirements to English.

"We assume a supplier knows how to build secure software"

In principle, this is a positive attitude. But how should a supplier know what is sufficiently secure for your organisation? This requires discussion and agreement. Moreover, only specifying functionality and ignoring security will lead to a supplier focusing on the former in case of time pressure. After all, that is the only explicit agreement that the supplier can be held accountable for. Grip on SSD describes what requirements can be agreed

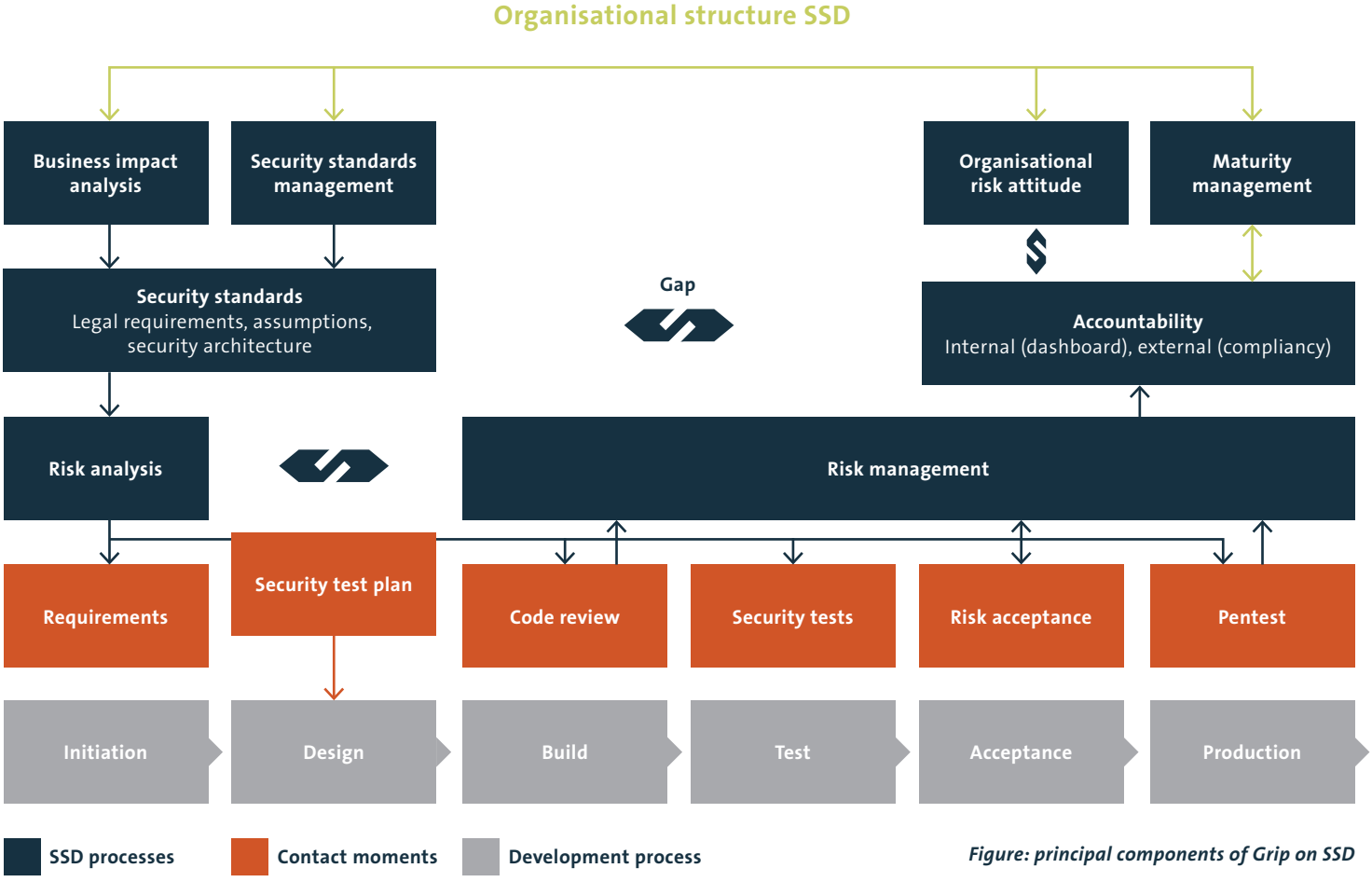

Figure: principal components of Grip on SSD

on up front with a supplier. Furthermore, it explains how to discuss these with a supplier ("comply or explain") and when and how to test these requirements.

"I thought our hosting partner would take care of this"

This is an oft-heard quote. Did we agree on who is responsible for timely rollout of security patches? Grip on SSD makes responsibilities for common security-related tasks explicit.

"Frankly, we did not expect these security requirements to be tested at all"

In practice there is little supervision on software development work. Suppliers are often surprised when an external auditor performs a code review. Although developers are motivated to write secure software, requirements are often either unclear or lacking completely. Moreover, even if any security requirements were specified, these are sometimes not tested at all, or only tested at the last moment using a penetration test. In such a case the supplier patches up the few vulnerabilities found and leaves matters there. Typically, this happens in the most stressful moment of a project, just before release, leaving little time for structural improvements. This is security after the fact rather than security by design. Grip on SSD describes how to clarify requirements to a supplier, and how to test agreements using design and code review.

"Ah, this actually leads to brand damages?"

This is a quote from a developer. In practice, developers often do not sufficiently understand the impact of certain bugs. Grip on SSD stresses that results of risk analysis should be communicated to the development team, so as to increase their awareness.

"It should comply with the OWASP Top 10"

The OWASP Top 10 is instrumental in creating security awareness. However, it is not suitable as a set of requirements, since it is not sufficiently concrete and far from complete. In addition, the OWASP Top 10 is a list of vulnerabilities and threats rather than requirements. One of the OWASP Top 10 vulnerabilities is "Sensitive Data Exposure". Clearly, any organisation wants to prevent this. But how should one determine that a supplier has taken sufficient precautions? The standard security requirements in Grip on SSD give a concrete answer to this question.

We observe that applying Grip on SSD leads to a better understanding of agreements on security by both clients and developers. Hopefully, the abovementioned quotes will soon be remnants of a long-forgotten past.

> **"Taking everything into account, we conclude that this document (because of the use of SIVA) is much more useful than other standards we have seen."**
>
> *TNO*

Grip on Secure Software Development is the result of a broad cooperation between specialists, clients and suppliers under the banner of the Centre for Information security and Privacy (CIP). For more information, also check the product section at www.cip-overheid.nl

About SIG

Software Improvement Group (SIG) helps business and technology leaders drive their organizational objectives through fundamentally improving the health and security of their software applications. SIG combines its proprietary tools and benchmark data with its consultants' expertise to help organizations measure, evaluate and improve code quality - whether they're building, buying or operating software.

As an independent organization, SIG has the largest benchmark in the industry with more than 36 billion lines of code across hundreds of technologies. The expert consultants at SIG use the benchmark to evaluate an organization's IT assets on maintainability, scalability, reliability, complexity, security, privacy and other mission-critical factors. The SIG laboratory is the only one in the world accredited according to ISO/IEC 17025 for software quality analysis.

Founded in 2000 as a spinoff from the University of Amsterdam, the SIG approach remains strongly rooted in academia. The company collaborates continually with universities and research institutes to develop upon its software quality evaluation models and R&D efforts.

SIG is headquartered in Amsterdam and New York with regional offices in Copenhagen, Antwerp and Frankfurt.
Learn more at www.softwareimprovementgroup.com

Fred. Roeskestraat 115
1076 EE Amsterdam
The Netherlands

www.softwareimprovementgroup.com
marketing@softwareimprovementgroup.com