WHITEPAPER

SYNOPSYS®

# Agile Development For Application Security Managers

# TABLE OF CONTENTS

testing caters to these same standards. Application security within the agile methodology must also change just as quickly. Application security that provides this need of ongoing security will provide protection without complicating matters for the designers, leading to huge hurdles or forcing these developers to work outside of agile principles.

This means tools that take a long time to run are not effective in these environments, neither are tools that require manual interpretation of results. This is due to the fact it is very likely that by the time the tool is done testing and a reviewer from the security team is done reviewing, the new code would already be in production for days or even weeks.

## Create security stories

Nearly all teams work with at least some level of delivering requirements as user stories. Presenting application security in the form of user stories keeps the fow familiar and works within the standards of an agile project. Additionally, one of the most important things to remember is that security is part of everyone's job, not something that is pawned off onto one person or team. By formulating security as a user story, it reminds the developers of this important part of their task, and presents security as just another aspect which is part of the development and testing cycle.

## Help create an agile application security workfow

Explain to agile developers what is expected in terms of security. Then work directly with them to create a workfow that fts in with current habits, iterations and deadlines. Questions often asked by development teams are:

- Who should run security testing, should each developer run on their own code, or maybe have one QA member who is responsible for security testing?
- How often should security tests be performed—should they be on every piece of code or after integration?
- Who should the results be delivered to? Development or security?
- Who is responsible for sign off?

The meaning of an agile application security workfow is creating a development process with embedded application security throughout all its phases, while still keeping the agile principles of being lean and quick.

## Provide a training program

Many developers do not have the training necessary to properly understand application security and perform the testing. Even if a training program is in place, developer turnover makes it very difficult to have everybody always up to date. Before you hand off responsibility to a development team, you should provide enough information to make the process easier.

Use application security tools that involve training in the process.

Use application security tools that involve training in the process. This training will pay off not only for the project at hand, but with future projects developed by a similar method. This is one of the benefits of agile development, future projects are even easier.

Remember that training does not have to mean two weeks of training covering a multitude of topics, some relevant some not, followed by an examination and usually resulting in the developer instantly forgetting most of what they learned. Training can, mean for example, that if a developer has a specific vulnerability in his code he will need to do a short training on this vulnerability, or training on specific vulnerabilities related to the application they are working on.

### Don't be afraid to make mistakes and improve as you go

Agile development is all about learning as the project proceeds. Any iteration includes improvements and changes, eventually moving towards the end result. The application security process is completed the same way, with changes and improvements along the way.

## Conclusion

Secure software, developed by any means, comes from properly testing and following proper security measures. The common perception that agile development methods cannot embrace secure coding practices and application security testing is, for the most part, false. With some flexibility, it is possible to integrate application security within your agile development system.

Developers should pay attention to the risks of not incorporating security within agile development. When software is not properly tested for security, there is a risk of developing insecure software which can lead to data loss and programs that are susceptible to hackers. While there is a cost to security testing, the cost that can occur due to improper testing will generally dwarf this cost.

### Application security as a natural part of the SDLC

Synopsys' Interactive Application Security Testing (IAST) tool is the run-time code and data analysis application security testing solution for the software development life cycle. By analyzing application behavior in response to simulated attacks, our IAST tool detects code vulnerabilities that pose a real threat. It assists in vulnerability management by generating exploits that demonstrate the risk to business critical data. Our IAST tool is the perfect application security testing solution for the SDLC; it can be fully automated and works seamlessly in agile and continues integration environments. Our IAST tool includes the following benefits:

- Application security testing that integrates seamlessly into the development process, integration with functional testing, build servers, and any other existing automation via a powerful out-of-the-box interface. You just put this IAST tool where you put all the rest of your automatic testing.
- Vulnerable code is highlighted and remediation provided, resulting in minimal work for developers and testers.
- Risks are explained in simple terms, allowing stakeholders to easily prioritize and build a vulnerability management plan.
- Both the cause and the fix are shown for developers, allowing to recreate the scenario, see the code that is causing the problem, and get an easy remediation solution, all in one place.
- Vulnerabilities are managed as any other bugs.