Web application security: too costly to ignore

White paper

Table of contents

Web application security: too costly to ignore

This paper details why application security addressed throughout the entire software development life cycle will increase the security of your applications, improve regulatory compliance, while also cutting development costs.

The bad news keeps rolling in nearly every day. A major retailer's website is hacked, and thousands of customer records, including credit card numbers, are stolen; a single flaw on the web page of a federal agency has leaked Social Security numbers onto the Internet. It is clear that attacks targeting web applications are on the rise, as stories like these are all too commonplace.

Not only are application attacks growing more prevalent, they are also costly. The research firm Gartner estimates that within the next year, 80 percent of all companies will have suffered through an application security incident. The cost of these incidents ranges from $90 to $305 per compromised record, depending on the nature of the breach and the company hacked. These costs include system cleanup and forensic analysis, regulatory and legal costs, consumer breach notification, and credit monitoring services. When considering those expenses, it is no surprise that the total expense of a single breach can range from several million to well into the billions.

These web application flaws also place organizations at significant risk for non-compliance with government and industry regulations such as Federal Information Security Management Act (FISMA), Health Insurance Portability and Accountability Act (HIPAA), Sarbanes-Oxley (SOX), the Gramm-Leach-Bliley Act (GLBA), and the more recent Payment Card Industry Data Security Standard (PCI DSS).

For attackers, web applications are both easy and worthy targets. Common flaws such as SQL injection, cross-site scripting, poor input validation and broken authentication conditions make it possible for attackers to easily infiltrate these applications to disrupt application availability and destroy or steal sensitive and private information like credit card data. Also, vulnerable web applications not only allow these miscreants to steal and manipulate information within that application, but also to use it as an entry point to the corporate network and back-end applications.

Set aside those security and regulatory compliance concerns for a moment, and consider this: Security vulnerabilities are like any other software defect. And, like any software defect, it is actually more cost effective to find and fix security software defects during development. It is widely accepted by analysts and software development experts that while it may cost a couple hundred dollars to find and fix defects early in the development cycle, those costs can soar to well above $10,000 to fix those defects after the application has been sent to production.

Considering that web application security is so crucial to mitigating the risks of attack and attaining regulatory compliance, the number of web attacks is on the rise, and the fact that it is exponentially more cost effective to remedy those flaws early in the development process, one must question why there is such a chasm between where application security should be and the sad shape of application security today.

The answer: Most organizations approach application security backwards. They bolt security on to the very end of the development process instead of building it in throughout. Typically, security teams will conduct an application security assessment just prior to deployment or worse yet—after the application is deployed, and they will uncover reams of security flaws—unvalidated inputs, logic errors and other defects created in the code during the entire development process. Those security managers then will be forced to go back to the developers and demand the code be fixed before the software can be released into production. This expensive and inefficient approach forces deployment date delays and severely limits the number of security defects the development and quality assurance (QA) teams will be able to resolve and re-test. This bodes badly for the prospect of building secure applications.

As you will see, by simply adding security to existing development checkpoints, such as when current functionality and performance tests are completed, organizations can slash security-related maintenance costs, while also producing dramatically more secure and regulatory-compliant applications.

Web application security: solving a complex challenge

There are many reasons why security flaws work their way into web applications. First, security is rarely considered during the functional requirements phase; that is to say that application owners do not demand security from the onset so developers do not build security into their applications.

Second, even when developers do consider security, they are covering only the basics: authentication, authorization, access control and encryption. They often do not provide comprehensive input validation to prevent SQL-injection and cross-site scripting defects. As a result, developers leave scads of security defects in their source code. This is not because they do not want to develop the most secure software possible. Rather, the primary focus of developers is to build highly functional and available applications. Their attention is on performance and ease-of-use. They tend to think about how the normal user will approach the software, not how attackers will try to bend an application's capabilities and coding weakness for malicious purposes. Even developers who recognize the importance of web application security usually see it as the responsibility of IT security or as part of the QA process.

Also, popular development tools, such as Microsoft® Visual Studio, IBM Rational Application Developer and Eclipse are very powerful at what they do—provide a robust development environment. However, they are not designed to aid in the production of secure software.

The result: Many web applications are rich in functionality but vulnerable to unwanted exploits and intrusions. These vulnerable web applications get deployed—and eventually attacked.

To be effective, software security must be treated just as any other software defect that would affect performance, or functionality. In that way, the security review just before the application is sent to production is more of a checkup than a fire drill. That is how savvy organizations cost-effectively develop secure code while making sure they meet development deadlines.

Toward continuous and secure application development
Remedying security problems during the development process is not something that can be achieved immediately. It takes time to integrate security into the various stages of software development. But any organization that has undertaken other initiatives, such as implementing a configuration management database (CMDB), capability maturity model (CMM) or even undergoing a Six Sigma program, knows the effort is worth it because systematized processes provide better results, more efficiency and cost savings over time.

And just as standardization on development processes—such as whether it's RAD (rapid application development), waterfall or agile—brings development efficiencies, saves time and improves quality, it's clear that strengthening the software development life cycle by possessing the right security testing tools and placing software security higher in the priority list is an excellent and invaluable long-term business investment. The important takeaway is that no matter how large or small your development efforts, all stakeholders—business and application owners, security, regulatory compliance, audit and quality assurance teams—must have a say from the beginning and benchmarks must be set for quality testing.

The section below gives you an idea how the security and regulatory compliance efficiency of web applications can be improved substantially simply by adding a few additional checks throughout the development process.

Strategy and planning phases

Executive-level sponsorship—This is the first stage, and perhaps the most crucial. Without executive-level sponsorship for secure software development and compliance, it is difficult, if not impossible, to get the organizational changes required for success. Organizations with strong executive-level sponsorship know that by implementing a comprehensive web application security program, they are meeting their compliance requirements, preventing security breaches, and saving time and money otherwise spent re-working security defects. And it is management's responsibility to mandate that applications be built and managed securely, and that software vulnerabilities be treated like any other software defect, such as those that cause slow application performance. Savvy development organizations know that it makes financial and organizational sense to do it right, from the beginning. And the only way to begin making these changes is for them to be demanded from top management.

Involvement of all application stakeholders—Organizations need to apply a structured process toward secure software development. This means security teams, analysts, design, development, QA and audit should be evaluating security during requirements, design, development and throughout production. In this way, security issues can be addressed as they arise throughout the application life cycle—from business requirements assessment to development and deployment. For example, by bringing security expertise into the design process, certain vulnerabilities such as application dependencies can be reduced right away, ensuring a design with fewest privileges. The more that potential attack vectors can be mitigated early on, the safer the final product will be.

Requirements phase

During these early phases, it is also useful to identify legal, security policy and regulatory compliance demands. Will the application hold government or industry-regulated information, such as data that may apply to HIPAA, PCI DSS or Sarbanes-Oxley? Will the application have access to, or reside on, the same network or servers as highly confidential information? If so, special attention needs to be paid to its security. Such applications need to have their design and functional requirements reviewed and approved by compliance and security executives.

Design phase

Misuse cases and threat models—During the technical design phase, the security team should develop misuse cases and threat models. While use cases are used to nail down application requirements, misuse cases aim to identify how an attacker may try to misuse the application for profit, or to access the network. Threat modeling is where your teams examine how the application itself could create threats and vulnerabilities. For instance, are certain areas of the application susceptible to denial-of-service attacks, and would a successful attack impact the availability of other applications? Does the application connect to a classified database? If so, would stronger authentication be necessary?

Build phase

Enforce secure coding practices—Throughout the development process, developers need to employ secure coding practices. They need to validate inputs, adhere to least privilege of processes, and generally hold fast to the best practices coding standards for the language and platform. This perhaps is one of the more difficult areas of your secure development initiative. The key is to provide developers with consistent training on secure application development trends and practices. If your organization is unsure of these practices, the Open Web Application Security Project (OWASP) is a great place to start: **www.owasp.org**.

Secure code reviews—In addition to quality and functional code reviews, security defect reviews need to be incorporated throughout development. Here is where software inspection software can help to automatically find and fix security-related defects. These defects include SQL injection, cross-site scripting, unvalidated inputs and other vulnerabilities missed during development. As the application edges closer to completion, it is crucial to also conduct integration tests. For instance, many software security controls operate as individual components and should be tested as such; other vulnerabilities become evident only as the application is pieced together. During this phase, it is crucial that the application be evaluated from several perspectives. Some organizations will perform a "black box" assessment, which is to have a skilled security assessor appraise the application with little to no knowledge about how the application was developed. They will also have another assessor, who does have knowledge of the software's inner workings, perform a "white box" evaluation of the application. Today, software is available that provides both black-box white-box testing.

Test phases

The key to success is to integrate security as the third pillar in application testing: functionality, performance and security; as the application hits normal QA benchmarks, the QA teams also test for security defects. Selecting tools that integrate tightly within your testing environment will go a long way to help in this regard.

Application assessment—For these application security assessments, it is important to select a highly accurate and comprehensive automated assessment tool: a web application vulnerability assessment platform that can assess mature web applications as well as those built using modern web services and Web 2.0 technologies. Choose an automated scanner that integrates into your development environment and delivers fast scanning capabilities, broad security assessment coverage and accurate results derived from combined black-box, white-box analysis.

Deployment phase

Secure application rollout—Ensure that all best practices for secure deployment are adhered to. Secure deployment means that the software is installed with all secure defaults enabled, meaning all file permissions are set appropriately and the secure settings of the application's configuration are used. After the software has been deployed, its security needs to be maintained throughout its lifespan. An all-encompassing software patch management process needs to be in place. Emerging threats need to be evaluated, and vulnerabilities need to be prioritized and managed.

Production

Ongoing assessments—Changes to web applications create risk, and what once was secure can become vulnerable. If security is a one-time activity, a vulnerability that enters the system after the audit can go undetected. Instead, you need to view application security as a process, included throughout the development life cycle in order to create secure web applications. Add security into the practices of every team member associated with developing and running your web applications.

Automate security into the application development life cycle

It's one thing to develop and manage software manually with security in mind. It's quite another to use tools that instill and enforce secure development throughout entire application life cycle.

Fortunately, application assessment and security tools are available today that will help you to get there—without slowing even the most aggressive project schedules. But in order to strengthen development throughout the application life cycle, it's essential to pick tools that aid developers, testers, security professionals and application owners, and that these toolsets integrate tightly with popular IDEs, such as Eclipse and Microsoft's Visual Studio.NET for developers and HP Quality Center software for testers.

While many application security vendors offer solutions to some pieces of the secure development life cycle, such as application security assessments, only HP Application Security Center software brings all of the pieces together. HP Application Security Center helps your developers, QA teams and security professionals quickly assess and remedy application security risk and vulnerabilities. HP Application Security Center provides common security policy definitions, automated security tests, centralized permissions control and web-based access to security information through four applications:

HP DevInspect software—HP DevInspect simplifies security at the earliest stages, during development, by automatically finding and fixing application vulnerabilities. HP DevInspect fully integrates with many integrated development environments (IDEs), including Microsoft Visual Studio, IBM Rational Application Developer and Eclipse. Developers use HP DevInspect to build secure web applications and services quickly and easily, without affecting schedules or requiring security expertise. For comprehensive web security testing, the Hybrid Analysis approach in HP DevInspect combines source code analysis with black-box testing in a single, cooperative process, helping to reduce false positives and find more application security defects.

HP QAInspect software—At the next stage of the software development life cycle, HP QAInspect enables QA teams to manage and conduct functional testing and website security testing from a single platform—without the need for specialized security knowledge. HP QAInspect features deep and intuitive integration into the most popular testing platforms, helping you to test web applications for security without leaving the QA environment. HP QAInspect finds and then prioritizes web application security vulnerabilities and presents detailed information and remediation advice for each vulnerability.

HP QAInspect is integrated tightly with HP Quality Center, thereby enabling companies to fully automate website security testing into the existing test management process without affecting aggressive product release schedules.

HP WebInspect software—For existing applications, and for applications ready to be put into production, HP WebInspect performs web application security testing and assessment for today's complex web applications, built on emerging Web 2.0 technologies. HP WebInspect delivers fast scanning capabilities, broad security assessment coverage and accurate web application security scanning results. HP WebInspect identifies security vulnerabilities that are undetectable by traditional scanners. With its innovative assessment technology, such as Intelligent Engines and concurrent application scanning, you get fast and accurate automated web application security testing and web services security testing.

HP Assessment Management Platform software—A standard for advanced, global security programs, HP Assessment Management Platform is a distributed, scalable, web application security testing platform that helps you address the complexities of today's web application security testing and scanning programs. It lets all constituents get information about application security vulnerabilities and participate in the assessment and remediation process without losing centralized control.

With HP Assessment Management Platform, organizations can perform unlimited, automated web application security testing and assessments while consolidating information into a real-time, high-level dashboard view of the enterprise's current risk posture and regulatory compliance. This consolidates and summarizes the organization's application security status so that you easily can assess and remedy security vulnerabilities in your applications.

Benefits of HP Application Security Center

- Lower risks by detecting security defects early in the application software development life cycle.
- Reduce time and budget for a security risk assessment through consolidated, automated testing.
- Facilitate a coordinated application security testing program across different departments in different locations.
- Provide visibility into the enterprise-wide application security status through pre-configured reports.
- Help management measure the effectiveness of your security risk assessment program.
- Meet legal and regulatory compliance requirements.
- Support complicated sites, including those using JavaScript™, flash, web services, SOAP or Ajax.

Conclusion

HP Application Security Center offers an effective end-to-end application security solution to enable your organization to stay protected from costly security breaches, remain compliant with government and industry regulations, and even reduce the long-term costs associated with application maintenance. It's crucial that application security be addressed throughout the entire life cycle, and only HP and the HP Application Security Center have the expertise and tools—WebInspect, QAInspect, DevInspect—to get you there.

Contact the HP Application Security Center today by phone (866) 774-2700 x1,
e-mail: **qmsecuritysales@hp.com** or
web at **www.hp.com/go/securitysoftware**.

To learn more, visit www.hp.com/software

4AA1-8057ENW, May 2008

Technology for better business outcomes