Quotium

Application Security in the Software Development Lifecycle

Issues, Challenges and Solutions

Table of Contents

Executive Summary

Web-based applications are becoming the biggest source of data breaches in almost all enterprises. Indeed, security breaches in web applications can create good opportunities for attackers to steal valuable data, plant malicious code, penetrate deeper into the organization network to reach internal resources and more.

A recent survey of security breaches at Fortune 500 companies showed that breaches in information security could result in financial losses of up to $24 billion. With that said, 90% of large corporations have found one or more breaches in their computer security during the past 12 months. Even worse, 70% of those detected breaches were considered severe, many resulting in proprietary information theft and financial fraud.

Even enterprises with the most sophisticated Web security systems are prone to application security breaches. In fact, breaches in web applications are not blocked even when companies have implemented the most sophisticated defense systems.

This white paper will discuss in detail why application security throughout the entire software development lifecycle is necessary for businesses of all shapes and sizes to prevent web security breaches and how it helps cut down business cost and increase the level of organizational information security.

Introduction

Today, web applications have grown in popularity to become one of the most essential tools for organizations and businesses alike to interconnect with their customers and prospects. Unfortunately, attackers can make use of security flaws in these web applications to steal customer information, expose sensitive customer records and eventually ruin business reputation.

Although many enterprises may find that firewalls and SSL (Secure Sockets Layer) encryption are useful in protecting their application access, it's not enough.

Recent studies show that three out of four websites are vulnerable to attacks and a majority of these attacks are on applications themselves which firewalls or SSL cannot do anything about.

In fact, any company can face a wide variety of security challenges and the business and technical concerns that security flaws can cause. What can you do about application security and how addressing these challenges can help sustain and grow your business?

All of these aspects will be covered in this white paper.

Impact of Security Breaches to Businesses

Security breaches have reached levels that cost businesses billions of dollars in losses every year. Every business that handles confidential information can experience security breaches at some points in their daily working operation.

Unlike what people often perceive about security breaches which usually happen in single events like website hacking, information theft or something in between, security breaches, in fact, have a wider perspective of an organization's information. These breaches can cover the total information assets of your business. With this being said, there are three letters to remember when talking about security breaches in businesses – CIA which stands for Confidentiality, Integrity and Availability.

- **Confidentiality:** whether your confidential information remains confidential or is it open to be compromised by unauthorized persons?

- **Integrity**: can your information remain unchanged to always guarantee accuracy which you can rely on?

- **Availability**: is your information available all the time for those who need it and when they need it?

Security breaches can affect your business in a number of ways, from reducing revenue, adding more costs, to lowering customer and investor confidence, trust and loyalty.

Website downtime in two days can cause a one-billion-dollar online retailer millions of dollars in revenue loss in addition to other recovery costs as well as costs of liability, damages and loss of information.

Recovery costs may include:

- **Cost for a single employee** who is not a Web security technician - to work solely on the web security failure to help fix the breach, make sure it doesn't happen again and inform customers about their risks of information exposure. The average salary for this person is around $65,000 per year which translates into the cost of $5,400 a month.

- **Cost for a security consultant** is from $10,000 to $20,000 per week for his onsite assistance, not to mention any other potential costs if the breach is big.

- **Cost for breach fixing** can be up to $500,000 if you are in the PCI industry (Payment Card Industry).

Beyond recovery costs and revenue loss, security breaches also result in the loss of customers. Recent studies report that 6% of customers will leave a company after its website has been attacked.

It doesn't stop there, as security breaches can cause far-reaching impacts for future development of businesses.

What Is Application Security?

Application security is a process that begins from the application development lifecycle to ensure the highest security possible of the development process (coding), the system, hardware the application runs on and the network it uses to connect, authenticate and authorize users.

Wikipedia defines application security as:

> Application security encompasses measures taken throughout the application's life-cycle to prevent exceptions in the security policy of an application or the underlying system (vulnerabilities) through flaws in the design, development, deployment, upgrade or maintenance of the application.
>
> *- Wikipedia -*

Application security helps eliminate potential attacks to business websites and applications when running internally or publicly on the Internet. This process reduces unexpected risks, financial loss and builds customer trust as a result.

Why Are Traditional Firewalls and SSL Not Effective Against Application Attacks?

Web applications are hosted on web servers. For users to be able to access applications, an HTTP port is required to be open to accept requests from users. To compromise web applications, attackers also use HTTP requests over the accepted ports to gain access to the internal system without being denied by the firewalls because traditional firewalls only protect lower layers and cannot detect HTTP-carried attacks.

Applicative attacks essentially penetrate the application as HTTP traffic which is allowed by the firewall, and then when interpreted by the application, the malicious payload of the requests perform the malicious operations.

SSL is also not effective in protecting against application attacks. SSL is used to encrypt the traffic between the user and the application. As we have already established, attackers carry out web application attacks in the same manner normal users access the application. It means the malicious request is wrapped in SSL on the end user side, and then the SSL is unencrypted in order for the request to be processed by the application. At this point the attack takes place.

It is therefore evident that neither Firewalls nor SSL can protect users against applicative attacks, since these attacks target vulnerabilities at the application source code level. They abuse the way the application code handles and processes user input. Application security flaws are essentially bugs in the application source code. These bugs are the factor behind security vulnerabilities discovered in web applications.

While web applications put organizations at risk of security threats and data breaches, organizations can still mitigate the risks, fix security holes and reduce any costs involved with proper application security solutions.

Actionable Tips to Reduce Security Vulnerabilities of Applications

Though security vulnerabilities can have a high impact on enterprises in terms of financial and data loss, strategic security measures could be taken to reduce risk and help businesses retain customers and guarantee revenue.

Application Security from the Early Design Phases

Robust software design, carefully planned and executed, leads to robust software which costs less to develop and maintain. Application security is no different. Planning software with application security in mind from the initial design phases leads to software with less bugs related to application security, and less potential for vulnerabilities.

This does not mean that security-minded application design ensures secure software, but it means that less flaws will be identified during the development, testing and production phases, and less flaws that encompass the entire application are likely to be found.

Perform Application Security in the Software Development Lifecycle

Introducing and implementing application security early in the software development lifecycle enables enterprises to meet greater customer demands for more secure products and services.

Just as important are remediation costs. As application security flaws are bugs in the code, the same rules apply to security bugs as other bugs – early detection leads to less expensive costs. Later detection in post-production phases can cost as much as ten times to fix than bugs detected earlier.

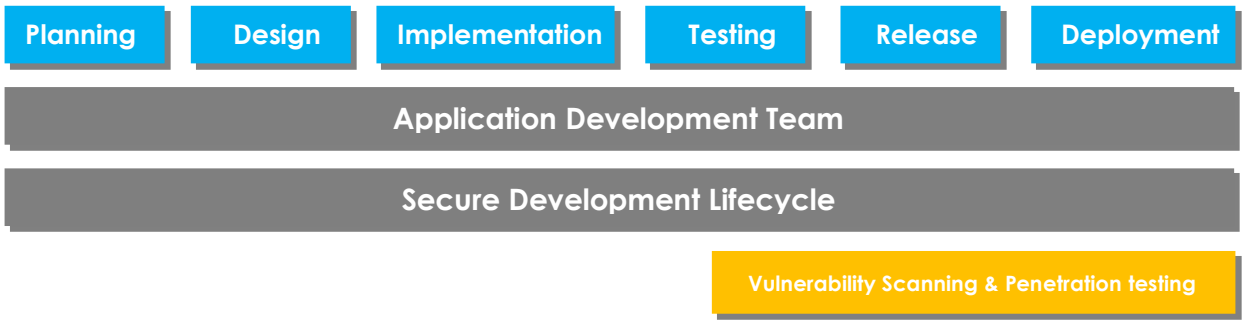

Figure 1: Integrating Security Earlier into Development Lifecycle.

Application Security is often mapped into key phases during software development lifecycle. These phases include:

- **Training** software development teams on application security, organizational policy and capabilities to make sure the team stays informed of the latest updates in security and privacy.
- **Planning and designing:** consider security and privacy when designing new features of products and integrate security into applications with minimal disruption.
- **Implementation:** prevent coding errors from creating vulnerabilities as well as use sophisticated development tools to build more secure code.
- **Verification and testing phases:** apply appropriate verification to software applications and make sure they produce proper functionality as defined in the initial design.
- **Release and response:** make sure to have the right plans or protocols for response when new threats emerge over time.

Involve Relevant Stakeholders

Application security flaws that are identified during testing or production are not just the problem of security teams to handle. Once bugs in the code that affect application security are identified, everybody is involved to remedy them.

In the same manner, responsibility for application security should be distributed among the different stakeholders of the development organization. Since application security is mostly about writing secure code and testing for application security flaws, development and testing teams should be given responsibility for secure development and testing, with the assistance and supervision of security teams.

Evaluate External Software Components

It is a common misconception that external software components do not need to be tested for security. It is considered the responsibility of the vendor, or seen as something that cannot be internally tested.

However, this approach is wrong due to several reasons. Mostly because attackers do not care who wrote the code. If there are vulnerabilities in it they will be found and exploited.

Once external components are integrated into the application, they become an inherent part of the application. Any security flaws these components or modules may contain directly affect the application and the business data within.

Modifications that are made to third party components are often extensive and contain thousands of lines of code. This code is sometimes overlooked in the testing processes as it is essentially not considered entirely in-house, but it is not external code anymore either. Such modifications should be considered as new code and thoroughly tested.

Security does not End at Deployment

Application security is an ongoing process, and should be considered throughout all phases of the SDLC, including post-deployment phases.

Code is constantly changed, either due to bug fixes or the change or addition of more functionality. This code should also be evaluated for application security flaws. In reality, it is often overlooked due to time restrictions, or misconceptions about the fact that small additions of code are not likely to lead to vulnerabilities.

In addition, new attacks are discovered periodically, and existing applications should be evaluated for new problems.

External Assessments vs. Integrated SDLC Approach

The choice of solutions applied during the development, verification and testing phases of the software development lifecycle plays a crucial role in detecting security flaws, mitigating risks of being attacked and most importantly, securing critical business applications and data.

For this reason, when enterprises turn to third-party vendors for security solutions in verification and testing phases, there are two types of services that they should take into consideration.

- **External security assessment** – be it periodical penetration testing or code reviews.
- **Integration of a solution into the SDLC** - Implementing a solution into the development and testing phases.

Vulnerabilities assessment solutions are used as an application solution integrated into the software development and testing phases while penetration testing and code reviews are used as an external service which take place periodically.

All in all, the difference between integrated security in the SDLC and external assessment can be further explained in the following table.

Features	Integrated Security in the SDLC	External Assessment
Running frequency	- Continuous. - Every time the code is changed.	On fixed milestones
Reports	Vulnerabilities are managed as bugs.	Report of the security status at the time.
Measurements	A list of application vulnerabilities.	A list of application vulnerabilities.
Performers	In-house.	- External project resources - Third-party providers.

Costs	The cost to fix a security vulnerability found in production is 6.5 times higher than one found early in the SDLC.	- Higher cost for external security consultants. - Longer processing time as well as higher management overhead.
Value	- Increased security awareness of team members - Improved quality of software - Reduction of the likelihood and impact of exploited vulnerabilities	Preventative control to mitigate security breaches and exposures.

Table 1 – The Difference between Integrated Security and Security Assessment.

Experts' advice:

Secure Development process and security assessment are powerful tools to monitor and search for application flaws and they should be used together to increase the security level of business applications.

A New Approach to Application Security in Software Development Lifecycle

Development and testing teams are required to deliver working, robust applications under strict deadlines. In these conditions, in order to implement application security into the SDLC the suggested application security solution should be accurate, deliver clear results and be easy to use.

Vulnerability testing as part of the SDLC should be part of a larger vulnerability management program (also known as VMP). A strong VMP consists of many parts such as system discovery, asset classification, vulnerability testing, prioritization, remediation, root cause analysis, and more.

A good vulnerability testing solution not only helps you successfully identify the vulnerabilities which actually pose a threat to the application, it also shows you appropriate remediation and provides you with the means to prioritize remediation. Best of all, it should point you to the code section where these vulnerabilities exist, make remediation recommendations and even allow you to play simulations of exploits against these security flaws. It should also integrate in existing development processes.

That's what Seeker from Quotium delivers.

Seeker uses a unique technology which correlates run-time application code and data flow with simulated attacks. This allows for the highest level of accuracy, delivering only relevant vulnerabilities and eliminating false positives. This is crucial for the process of application security in the SDLC as development time is precious, and cannot be spent on analyzing bugs which are in fact false positives. Seeker correlates the outside-in and the inside-out approaches, providing clear visibility into the application internals as well as the outside view of a simulated external application attacker.

Results provided by Seeker are clear and contain everything required by developers, testers, management and security teams to understand the identified vulnerability, easily analyze and remedy it. In addition, Seeker provides vulnerabilities already prioritized by their potential impact. This is done by carrying out simulated exploits against the application.

To accommodate being part of the application development lifecycle, Seeker is provided with multiple tools to facilitate easy integration into existing processes. Seeker interfaces with all common bug tracking systems

to deliver identified vulnerabilities as application bugs. It works with automatic quality assurance tools (such as HP QTP, Selenium, and more) to map and understand the application and its scenarios. These scenarios are then used to perform application security testing on them.

In addition, Seeker can be activated as part of an existing testing plan, for example to carry out application security testing as part of regression testing, part of continuous integration and more.

Conclusion

As application security becomes an essential requirement in software development, potential for severe brand damage, privacy issues and financial loss can be reduced.

The benefits enterprises experience from application security in the SDLC come from avoiding wasted time and effort of addressing application security flaws close to product launch and preventing the complexities of repeating the test phase later in the development cycle or after the application ships.

For projects with 1 million lines of code, application security can help save around $600K annually from defect prevention together with $280K savings in increased developer efficiency and productivity (reduce time spent to fix a breach).

While it's not always possible to avoid security vulnerabilities completely during software development lifecycle, especially when it's a large-scale project, tools and processes can still assist enterprises in finding and fixing flaws efficiently and effectively. More importantly, the testing results from these tools are presented in an actionable and relevant way which can also be integrated into business workflow. This makes the troubleshooting and fixing process faster and saves more time for other business activities.

About Quotium Technologies.

Quotium Technologies is a specialist in the development of innovative software solutions to guarantee the security and functionality of business-critical applications throughout their lifecycle.

Seeker is the leader of the new generation of application security testing software. Easily integrating with existing software testing processes, Seeker allows developers to efficiently develop secure applications.

For more information www.quotium.com or info@quotium.com