

Effective Trends and  
Techniques for  
Application Security

## Copyright

The copyright in this work is vested in ITWorx, and the document is issued in confidence for the purpose only for which it is supplied. It must not be reproduced in whole or in part or used for tendering or manufacturing purposes except under an agreement or with the consent in writing of ITWorx and then only on the condition that this notice is included in any such reproduction. No information as to the contents or subject matter of this document, or any part thereof, arising directly or indirectly there from will be given orally or in writing or communicated in any manner whatsoever to any third party, whether an individual, firm or company, or any employee thereof, without the prior consent in writing of ITWorx.

© Copyright ITWorx (2006)

## Table of Contents

Copyright.....	1
Table of Contents.....	3
Executive Summary .....	4
The Importance of Secure Applications.....	5
The Rise of Application Security Threats .....	6
<i>Don't Overlook Application Security</i> .....	6
<i>Sometimes It's Just Too Easy</i> .....	7
<i>Not Just Easy – Greater Ramifications, Too</i> .....	7
<i>What a Vulnerable Web!</i> .....	8
<i>Cyber Crime on the Rise</i> .....	8
Towards Delivering Secure Applications.....	9
<i>The Current Model Is Reactive</i> .....	9
<i>Redefining the SDLC for Maximum Security</i> .....	9
Threat Modeling .....	10
Secure Development Practices.....	11
Penetration Testing.....	11
Infrastructure Hardening .....	12
ITWorx Security Solutions .....	13
<i>The SDLC, the ITWorx Way</i> .....	13
<i>STRIDE for Better Threat Identification</i> .....	14
<i>Heavy on the Training, Heavy on the Security</i> .....	14
Conclusion .....	15
About ITWorx .....	16
List of works cited .....	17

## Executive Summary

Rapid advancement in information technology often means increased risk of security breaches. Within just 1-2 years security schemas can become outdated; no longer adequate to meet current regulatory mandates and security best practices. A security report<sup>1</sup> shows that in 2005 alone about 100 significant security breaches were committed affecting more than 56 million users. The combination of increased regulatory compliance requirements and the heightened threats demands a new approach to security.

While enterprises have focused mainly on perimeter based network security, 75 percent of attacks have been shown to occur at the application layer (“Network World”, 2004). With the advent of these new threats, application security has become a much greater concern. The ease of committing such attacks, combined with the greater financial loss often associated, has prompted many companies to invest heavily in the security of their applications. Although browser-based systems have increased in popularity, these systems are often the weakest link in the security chain. The design limitations of HTTP, the multiple points of attack, and the lack of awareness of Web security issues all play a role in this vulnerability.

In order to reduce the security risk associated with these browser based applications, security activities must span all phases of the software development life cycle (SDLC), beginning with the requirements phase. Software developers can ensure the building of secure systems by making security a key system attribute, leveraging reusable secure components, applying defense-in-depth strategies, and continuously training software engineers on new attacks and security techniques.

This paper discusses the latest industry best practices for developing security into applications throughout the SDLC, with emphasis on the methods employed at ITWorx to develop secure solutions.

---

<sup>1</sup> Utimaco Safeware

## The Importance of Secure Applications

“We continuously see the amount of application-based security attacks increasing, largely due to the fact that these attacks are far more business impacting than network attacks. There is no denying that application security has become an absolute must to thwart these attacks. To truly stem the tide of application-level attacks, the industry must embrace a Software Development Life Cycle that begins by focusing on security as the central system attribute and maintains a focus on security at every stage until the software is released.”

Ahmed Hossam, Senior Security Consultant, Microsoft Egypt

“The potential for devastating financial loss and the sheer amount of application-level threats highlights the fact that application security is absolutely integral and indispensable,” said Dr. Tarek Nabhan, Products Division Manager, ITWorx. “The escalating number of application security attacks has called for new methods and techniques to combat these violations. For a truly effective security schema, it is necessary to identify the vulnerabilities within the system and safeguard against breaches before they happen. The key is to be proactive rather than be reactive.”

## The Rise of Application Security Threats

The need to secure information assets has existed for as long as information itself. Codes and ciphers have been developed over centuries to protect information against unauthorized inspection. This practice has become especially critical with the introduction of computer technology. The integration of computers with information technology introduced abundant advantages, but has subsequently increased the vulnerabilities (Gelbstein and Kamal 3). With digitized information also comes digital susceptibility of that information. Moreover just as technology advances, so do the methods used to compromise that technology.

Many advancements made in computer technology, that lead to legitimate opportunities in the field, unfortunately often backfire when those advancements are then used to perpetrate computer crimes. Computer security breaches can be divided into six categories, according to the Computer Security Institute (CSI). The highest rating breach in terms of dollar losses is unauthorized access, followed by denial of service, insider net abuse, misuse of public web application, system penetration, abuse of wireless network, and finally Web site defacement. The data also suggests that security breaches can happen from both outsiders and insiders, nonetheless 75 percent of attacks are shown to occur through the intranet and insiders. It is thus important for organizations to anticipate attacks from all quarters, including their internal users (Gordon et al.).

### Don't Overlook Application Security

It is imperative to differentiate between network and application security as both are vital, but separate, concerns. While a lot of effort and money has been directed at protecting network perimeters recently, application security has long been overlooked. Statistics show that the number of application security breaches are increasing at an alarming rate. Network World states that 75% of current attacks are happening at the application layer; "The battle between hackers and security professionals has moved from the network layer to the Web applications themselves." ("Network World", 2004). According to Gartner research, since 2002 70% of all successful attacks have exploited application vulnerabilities. One of the

main reasons application attacks have become more common is the fact that they are simpler to mount and harder to defend against than network attacks.

### Sometimes It's Just Too Easy

Application attacks are often facilitated by the use of a variety of tools and techniques sometimes as simple as using the Google search engine, through which even a less sophisticated user can gather information about known application vulnerabilities. This technique is often referred to as Google hacking ("Google Hacking Mini-Guide", 2004). When all that is required to uncover application vulnerabilities is the ability to use a Web browser, the potential audience of hackers grows exponentially. With a larger percentage of the population online than ever before, the number of people who become involved in this rudimentary form of hacking is bound to increase, leading to further exposure of application vulnerabilities.

### Not Just Easy – Greater Ramifications, Too

With application attacks, the financial losses associated with such breaches are often far greater than with network attacks. While network attacks might cause a drop in network activity and downtime in business processes, application attacks can expose confidential information to competition, personal and financial records to those who can exploit the information, classified governmental data to terrorist groups, and more. Access to such information costs companies and governments millions of dollars to remediate, and can literally be the downfall of organizations. A recent warning by the head of the Internet Security Unit at the American Ministry of Internal Security states that such breaches can even cause costly physical damage if the applications breached are responsible for controlling production plants ("Done by teenagers and adults and leads to victims, virtual attacks threatens American companies", 2006).

## What a Vulnerable Web!

Browser based systems are the weakest link in the application security chain, due to the inherent design limitations of the HTTP protocol and platform vulnerabilities present in the operating system, application builder and database engine. This problem is further magnified by the general public's lack of awareness about best practices for Web security. Though most businesses think that running their site only on the Intranet improves security, statistics do show that 75 percent of all attacks occur via the Intranet. And while many people think that their Web sites are unknown, a study conducted by PSINet Europe contradicts this belief. PSINet Europe set up an anonymous "dummy server" with no protection, and results showed that the server was hacked 467 times within the first 24 hour period of being connected to the Internet ("Exposed server – magnet for hack attacks", 2003). Another common misconception that leads to application attacks is the view that the HTTPS protocol is a secure protocol. Though HTTPS employs a different default port and adds an additional encryption/authentication layer between HTTP and TCP, it provides very little protection against application attacks, but to the contrary, often acts as a convenient shield to malicious traffic, confounding intrusion detection mechanisms that maybe in its path on the network.

## Cyber Crime on the Rise

With cyber crime incidents on the rise, organizations have become keen to implementing tools and technologies to safeguard against such attacks. A CSI/FBI survey showed that 87% of the sample conducted security audits; a five percent increase from the year before (Gordon et al. 17). Though this may seem like a large number, it comes nowhere near to matching the increasing number of vulnerabilities discovered in applications. The CERT Coordination Center, a security watch group, estimates that 99 percent of all security intrusions result from the exploitation of system configuration errors and known vulnerabilities within software applications. With this increase of application layer attacks, application security can no longer be seen as "optional", but as a mandatory requirement.

## Towards Delivering Secure Applications

The escalating number of application security attacks has called for new methods and techniques to respond to these threats. For a truly effective security framework, it is necessary to identify the vulnerabilities within the system and safeguard against breaches before they happen. The key is to be proactive rather than be reactive.

### The Current Model Is Reactive

Over the past years several techniques have been developed to help protect organizations' confidential information. These methods include, but are not limited to, firewalls, anti-virus software, intrusion detection systems, data encryption, and intrusion detection systems. These tools help prevent or detect malicious attacks, however, the reactive nature of these solutions fails to provide the optimal solution for ensuring a fully secure system.

### Redefining the SDLC for Maximum Security

Effective application security requires more than just preventive tools and detection systems. Until recently, application security was often an afterthought for developers; something they implemented at the end of the SDLC after all the desired features and functionalities of the software had been accomplished.

To truly secure an application requires the redefinition of the creation processes; integrating security standards at all the stages of the SDLC, beginning with the requirements gathering process. By making security a key system attribute from the requirements stage, leveraging reusable secure components, applying defense-in-depth strategies, and continuously training developers on new security techniques, developers can ensure secure applications through and through. In addition, the hardware infrastructure (Operating Systems, web servers, and database engine) must also to be hardened.

Several techniques have been developed – along with accompanying tools– to assist in applying security guidelines. The techniques discussed in this white paper are threat modeling, secure coding, penetration testing, and infrastructure hardening.

### **Threat Modeling**

Threat modeling is an approach that has gained a lot of traction in recent years. Its main objective is to identify and model the threats that could place any system at risk at any given time. Threat modeling should be performed at the design phase of the software, so that as the software is conceptualized, so are the possible threats. This will help software designers identify threats and proper mitigation actions early in the SDLC. MSDN magazine defines the process of threat modeling as “an iterative approach to assessing the vulnerabilities in your application to find those that are the most dangerous because they expose the most sensitive data.” (“Threat Model Your Security Risks”, 2006).

The process consists of three main steps. First, developers must view the system as an adversary would view the system. Adversaries look to find exposed services, and once they have achieved that, they formulate goals to attack the system. In order to do this, they must identify the entry and exit points; the places where the data enters or exits the application. They must then identify the assets that need to be protected from unauthorized users, as well as the trust levels for the system. Trust levels are assigned to entry/exit points to define the privileges an external entity has to access and affect the system (Burns, 2005).

After information about the entry and exit points is collected, the system needs to be categorized by collecting background information about the system to better uncover threats. This involves collecting information about usage scenarios which define how the system will be used in terms of configuration and security goals. Information about the system’s dependence on outside resources and implementation assumptions are also collected.

Once the information about the system is collected, diagrams are used to model the system. Data flow diagrams (DFDs) are one of the most common and useful diagrams used. This model helps identify the key processes, the threats to those

processes, and the possible mitigation actions that can be taken to eliminate those vulnerabilities.

### **Secure Development Practices**

Fundamental to securing a system is the use of secure development practices in the underlying applications. It is far easier and cheaper to keep the security flaws out from the beginning than to fix them after the application is up and running.

Keeping security flaws out at the design stage and keeping them out through the remainder of the SDLC requires vigilance at all levels of the software development organization. To ensure secure development practices, changes in the SDLC must be made to accommodate the security requirements.

First of all, strict and enforceable security standards must be developed. Developers should be trained on an ongoing basis on new best practices for secure coding and the business risks related to software security. The training resources should be made easily available to developers, along with tools in the areas of software security analysis and remediation. The library should also include quick references to various software security issues, and should be updated frequently.

It is important to familiarize the technical team with the security standards and collaborate with them to create formal software security standards and policies. In addition, a set of metrics must be developed in order to ensure that these standards are properly implemented and maintained ("Software security starts with writing secure code", 2006).

### **Penetration Testing**

Penetration testing is a process that aims to expose the vulnerabilities present in the system, measure the level of the harm done if such vulnerabilities are exploited and provide solutions for mitigating and solving these vulnerabilities. The process of penetration testing involves an analysis of the system for any weaknesses, technical flaws or vulnerabilities. This analysis is carried out from the position of a potential

hacker, and can involve active exploitation of security vulnerabilities. Any security issues that are found will be presented to the system owner together with an assessment of their impact and should include a proposal for mitigation or a technical solution.

### **Infrastructure Hardening**

Application security breaches can occur through vulnerabilities present in locations other than the application itself. The hacker may utilize vulnerabilities in the associated infrastructure to attack the application or the data associated with it. It is therefore imperative to secure the entire infrastructure that the application resides upon.

Infrastructure hardening involves identifying open doors and vulnerabilities, or additional features that may be present in the platform or server which might not be needed by the application to run but pose a possible threat. The infrastructure consists of, but is not limited to, the network, platform, Web server, and database. Once the vulnerabilities are identified, actions should be taken to harden or secure this infrastructure.

Hardening the platform involves activities such as installing the latest patches and disabling services and/or features that are not required for the application to run properly.

## ITWorx Security Solutions

ITWorx has developed the know-how, proficiency and expertise to ensure the delivery of secure solutions. ITWorx builds secure software by employing industry best practices as well as ITWorx-specific security measures in all of its software. In addition ITWorx provides three different types of security services: security consulting services to businesses, auditing services for deployed software and security training services for developers.

Consulting services offered by ITWorx include the assessment of the security and threat levels of all existing systems. Once this assessment is performed, ITWorx provides a detailed analysis of the vulnerabilities and their potential impact on the business.

### The SDLC, the ITWorx Way

In order to build secure systems, ITWorx has redefined the SDLC to incorporate in-depth security measures at every stage of application development. By initiating each product with a security analysis of possible vulnerabilities, ITWorx defines an application security plan to follow to mitigate those risks throughout the development cycle. Security is always incorporated as a key, fundamental attribute of every solution.

All ITWorx systems consist of the applications, the inter-application communication, and the servers these applications and their associated data reside up on. Applying the least privilege mode and defense-in-depth strategies is essential to this plan for all applications. This is complemented by the use of secure tested proven reusable components throughout the applications.

Finally in order to keep up with the continuously changing security landscape, our teams' security knowledge and skills are continuously strengthened through ongoing trainings. Instead of following the traditional practice of building a small, concentrated, security team, at ITWorx the entire organization receives role based education on the most up-to-date security best practices. This ensures that the

security is integrated within all development and testing processes and not limited to an afterthought.

## STRIDE for Better Threat Identification

As part of the threat analysis and modeling incorporated in ITWorx SDLC, the company employs the STRIDE method. The STRIDE method helps identify five key threats: Spoofing, Tampering, Repudiation, Information Disclosure, Denial of Service, and Elevation of privilege. Each step in ITWorx SDLC is further enhanced to include different security activities, including new guidelines in the design, coding, testing and deployment phases of the system. To ensure optimal benefit from these guidelines, reviews are carried out to ensure that the guidelines are applied.

Some examples of general design guidelines are removing sensitive information from session cookies, usage of non-predictable session IDs, encryption of sensitive data present in databases and configuration files, and storing uploaded or downloaded files outside the web space, input and parameter validation and exception handling.

Numerous measures are taken during the testing and verification phases, such as developing security test plans based on the threat model and executing security test cases, ensuring that sensitive data is not stored in a readable format, removing unnecessary files or folders from the Web space, performing tests on hardened servers and much more.

## Heavy on the Training, Heavy on the Security

ITWorx goes to great lengths to enhance the knowledge base of every member of its engineering team to ensure that security is a principle shared by all. Software engineers across our development centers receive regular trainings on different aspects of security, including emerging threats, tools and technologies. These presentations focus on how to integrate security guidelines into the development life cycle of the software and include topics such as threat modeling, penetration testing, and infrastructure hardening.

## Conclusion

With all the advancements in the software industry, applications are becoming more vulnerable to attacks from inside and outside the organization. It is thus a necessity for software development companies to integrate security into their processes – from the beginning of the requirements process through to the completion.

Application security can not be painted onto an existing application but rather requires redefining the processes at every stage of the SDLC, and integrating methodologies that ensure explicit attention to system security. This approach redefines security as a core system attribute. ITWorx has introduced new security service offerings that include security consulting and security training to complement its core building secure systems.

Developing secure software is a deliberate and sometimes lengthy and expensive process, but based on the rising number of application layer attacks it would be negligent to not build such security safeguards into newly developed systems and reengineer existing ones. The commitment required to build secure systems is substantial, but ITWorx is convinced that the quality and security of the resulting systems is well worth that investment. By educating engineering teams to avoid common security pitfalls, performing thorough security code reviews, testing applications for security vulnerabilities, and integrating security at every stage of the SDLC, ITWorx provides its customers with systems that not only include security as a feature, but systems that are truly secure.

## About ITWorx

ITWorx is the largest software professional services firm in Egypt. The company offers Portals, Business Intelligence, SOA and Product Development services to Global 2000 companies with a focus on Financial, Telecommunication, Government, and Educational institutions, in addition to a number of Independent Software Vendors (ISVs) across North America, Europe and the Middle East.

By partnering with Magic Quadrant technology vendors Microsoft, BEA Systems, Vignette, Business Objects, Sun, IBM, and Oracle, ITWorx leverages its global delivery capability, CMMi Level 3 certified processes, and model driven development tools to seamlessly extend its customers' IT organization augmenting it with agile, high quality productive capabilities, technology competences and vertical industry know-how

ITWorx integrates international security development principles in its SDLC. All software developed by ITWorx abides by strict security guidelines. ITWorx has developed several solutions for secure online payment and secure banking portals. ITWorx offers includes: secure solutions development, security consultancy, and training services in various security topics. For more information please contact [sales@itworx.com](mailto:sales@itworx.com).

## List of works cited

Gordon, Lawrence A., Loeb, Martin P., Lucyshn, William, and Robert Richardson, 2005 CSI/FBI Computer Crime and Security Survey, <http://www.gocsi.com>

Steven F Burns, Threat Modeling: A Process To Ensure Application Security, January 5, 2005, Sans Institute 2005, [http://www.giac.org/certified\\_professionals/practicals/gsec/4718.php](http://www.giac.org/certified_professionals/practicals/gsec/4718.php)

"Threat model Your Security Risks", MSDN Magazine. April 30, 2006, <http://msdn.microsoft.com/security/securecode/threatmodeling/default.aspx?pull=/msdnmag/issues/03/11/resourcefile/default.aspx>

Gelbstein, Eduardo, and Ahmad Kamal, Information Insecurity, November 2002, United Nations ICT taskforce and the United Nations Institute for Training and Research, New York. [http://www.utimaco.us/secnews/data\\_breaches\\_2005.html](http://www.utimaco.us/secnews/data_breaches_2005.html)

Reuters, "Done by teenagers and adults and leads to victims, virtual attacks threatens American companies", May 16, 2006, Alhayat, p1.

Hayday ,Graham, "Exposed server-magnet for hack attacks", ZDNET, January 29, 2003, [http://news.zdnet.com/2100-1009\\_22-982554.html](http://news.zdnet.com/2100-1009_22-982554.html)

Sima, Caleb, "Software security starts with writing secure code" SecurityPark, January 6, 2006, <http://www.securitypark.co.uk/article.asp?articleid=25401&CategoryID=1>

Long, Johnny, "Google Hacking Mini-Guide", Informat.com, May 7, 2004, <http://www.informat.com/articles/article.asp?p=170880&seqNum=2&rl=1>