


# WhiteHat Security Web Application Security Risk Report

April 2007 Edition  
Jeremiah Grossman  
Founder and CTO, WhiteHat Security

A large, light blue, abstract graphic consisting of several overlapping, curved, brush-stroke-like shapes that sweep across the lower half of the page.

A WhiteHat Security Whitepaper

## Introduction

The Web application layer is the top target for malicious online attacks. The prevalence of website vulnerabilities undoubtedly contributes to this trend, along with the relative ease with which criminals are able to monetize and exploit their illegal activity. Indeed, many of the largest incidents are a direct result of exploitation of Web application vulnerabilities. Enterprises that want to reduce the risk of financial losses, brand damage, theft of intellectual property, legal liability and fines need to remain informed about how websites are able to be penetrated and how they can best be defended. WhiteHat Security is in a unique position to compile this data and put it to work. Our second release of the Web Application Security Risk Report continues to deliver actionable information and raise awareness about the vulnerabilities in custom Web applications.

Through our flagship service, WhiteHat Sentinel, we perform rigorous and ongoing vulnerability assessments on hundreds of (public-facing) production and development websites each month. Our work gives us a one-of-a-kind perspective into website vulnerability trends across financial, e-commerce, healthcare and high-tech industries. WhiteHat Security can accurately identify which issues are currently the most prevalent and severe. As the only company with access to this depth of cumulative data, we are sharing our findings to provide enterprises with a clearer picture of the vulnerability management issues affecting their websites. This quarter's report represents a more than three-fold sample increase over the last, and is based on data obtained between January 1, 2006 and March 31, 2007.

WhiteHat Sentinel's black-box assessment methodology is efficient, thorough and consistent. We utilize the Web Application Security Consortium (WASC) Threat Classification<sup>i</sup> of 24 Web application vulnerability classes as our standard. This baseline ensures complete coverage of the known types of website vulnerabilities. WhiteHat's process combines unique and authenticated enterprise-class scanning for identification of technical vulnerabilities, verified results to remove false-positives, and custom testing (with multiple-user levels) to uncover business logical flaws. By heavily leveraging our technology platform, this three-phase process is carried out every week on the majority of websites managed under the WhiteHat Sentinel Service.

To be clear, the statistics contained within this report differ from the data collected by MITRE's Common Vulnerabilities and Exposures (CVE)<sup>ii</sup> project. CVE is a dictionary of publicly known vulnerabilities in commercial and open source software products. WhiteHat's data is cumulative and derived from previously unknown vulnerabilities in real-world websites at the custom Web application layer (Figure 1). Also, bear in mind that not all websites have the same overall business value. Some websites are mission critical, while others are static "brochureware." This is important to understand, since the websites managed under WhiteHat Sentinel are more likely to represent the most "important" and "secure" websites found on the Web, conducting high-volume transactions or managing sensitive information. This context is helpful when estimating the current global state of website security.

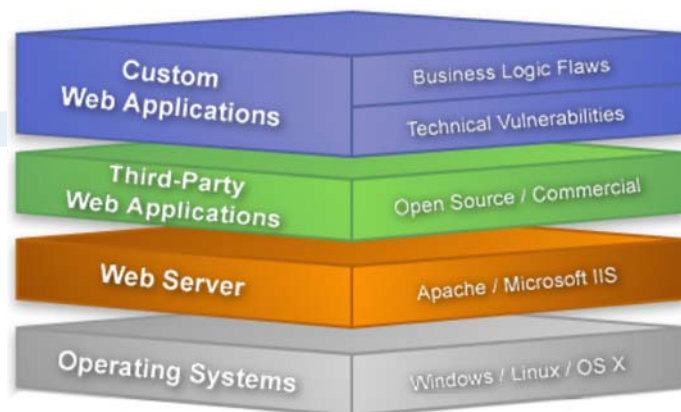


Figure 1. Software Vulnerability Stack

## Top Vulnerability Classes

The number of instances of an individual vulnerability class varies greatly across production websites. For example, one website may possess one hundred unique issues of a specific class, while another website may not contain any. As a result, “top” lists based solely on total vulnerabilities are not necessarily the most helpful to an enterprise seeking to make informed decisions about the security of its websites. WhiteHat’s statistics calculate the percentage likelihood of vulnerability classes to occur within websites (Figure 2), as well their prevalence in the overall population (Figure 3). This two-pronged approach minimizes skewing the data with findings from highly secure and extremely risk-prone website edge cases. Presenting the data in this way helps direct attention toward areas returning the most value.

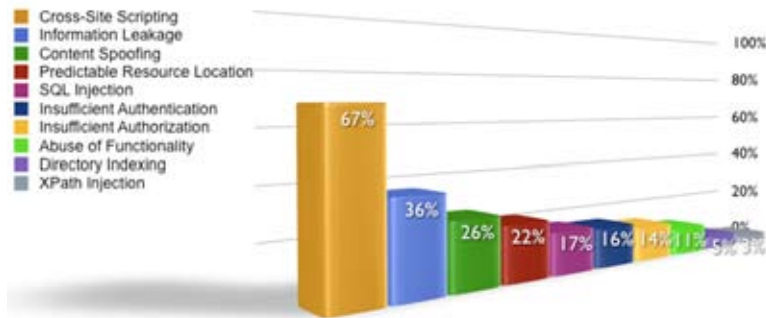


Figure 2. Top 10 Vulnerability Classes by Percentage Likelihood

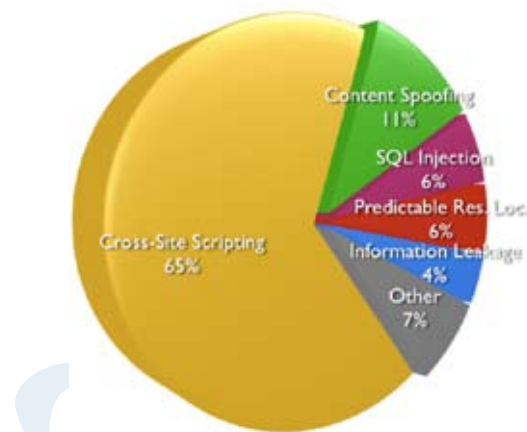


Figure 3. Top 5 Vulnerability Classes in the Overall Population

When comparing Figure 2 against our last report, there is a slight drop in technical vulnerabilities, including Cross-Site Scripting (XSS), SQL Injection, Content-Spoofing and Path Traversal. This trend may be partially attributed to increased developer awareness and vigilance in software security due to increased media coverage of these issues. WhiteHat's data set may also indicate that modern development frameworks such as Microsoft's ASP.NET are almost certainly making an impact. These frameworks have default security safeguards that protect what would otherwise be considered flawed code. However, this improvement may be short lived: A .NET vulnerability<sup>iii</sup> has surfaced which allows XSS attacks to bypass the request filtering security features.

One last thing to note is that XPath Injection has replaced Path Traversal at the end of the range. One conclusion may be the increased use of XML Web Services made publicly available by Web-enabled front-ends has led to the increased appearance of this vulnerability class.

## The Top Ten Vulnerabilities Defined

### 1. Cross-Site Scripting (7 out of 10 websites)

Most industry experts and researchers agree that Cross-site Scripting (XSS) continues to be the most prevalent website vulnerability. Depending on the website, XSS can be especially hazardous to businesses and consumers. New attack vectors employed are responsible for highly effective phishing scams and Web worms that are resistant to commonly accepted safeguards. The evolution of cutting-edge JavaScript malware as a payload has made finding and fixing this vulnerability more vital than ever.

### 2. Information Leakage (1 in 3 websites)

Information Leakage occurs when a website mistakenly reveals or is manipulated to reveal sensitive information such as developer comments, user information, internal IP addresses, source code, revision numbers, error messages/codes, etc., which may all aid an attacker.

### 3. Content Spoofing (1 in 4 websites)

Content spoofing is used in phishing scams as a method of forcing a legitimate website to deliver or redirect users to bogus content. For example, users often receive a suspicious link that instructs them to confirm their user name and password information. Typically, phishing websites are hosted on look-alike domain names mimicking the content of the real site. In the case of Content spoofing phishing scams, fake content is injected into the real website, making it very difficult, if not impossible, for users to detect the difference and therefore protect themselves.

### 4. Predictable Resource Location (PRL)(1 in 5 websites)

Over time, many pages on a website become unlinked, orphaned, and forgotten. These Web pages often contain payment logs, software backups, future press releases, debug messages, source code – nothing, or everything. Normally, the only mechanism protecting the sensitive information within is the predictability of the URL. Automated scanners have become adept at uncovering these files by generating thousands of guesses. Also, through a process called “Google Hacking,” attackers use search engines to discover sensitive information via forgotten links on a website.

### 5. SQL Injection (1 in 6 websites)

SQL Injection has been at the center of some of the largest credit card and identity theft incidents. Today's backend website databases store highly sensitive information, making them a natural, attractive target for malicious hackers. Names, addresses, phone numbers, passwords, birth dates, intellectual property, trade secrets, encryption keys and often much more could be vulnerable to theft. With a few well-placed quotes, semi-colons and commands entire databases could fall into the wrong hands.

#### **6. Insufficient Authentication (1 in 6 websites)**

Insufficient Authentication flaws are typically found within the business logic of an application. Successful exploitation leads to an attacker gaining unauthorized access to protected sections of a website. For example, while logged-in as a normal user, an attacker could impersonate another user on the system.

#### **7. Insufficient Authorization (1 in 6 websites)**

Insufficient Authentication flaws are also typically found within the business logic of an application. Successful exploitation leads to an attacker being able to escalate his or her privileges or exercise unauthorized access. For example, while logged-in as a normal user, an attacker could gain access to another user's data while still being logged-in under their current account.

#### **8. Abuse of Functionality (1 in 7 websites)**

As stated by the WASC Threat Classification, "Abuse of Functionality is an attack technique that uses a website's own features and functionality to consume, defraud, or circumvent access controls mechanisms. Some functionality of a website, possibly even security features, may be abused to cause unexpected behavior. When a piece of functionality is open to abuse, an attacker could potentially annoy other users or perhaps defraud the system entirely."

#### **9. Directory Indexing (1 in 20 websites)**

As a feature of most popular Web servers, Directory Indexing lists the contents of a directory if no specific file name is given and no index file is present (example: index.html). Directory listings provided in this way could reveal sensitive information that was not intended for public viewing, such as pre-released Web pages, log files, temporary files, backup files, etc.

#### **10. XPath Injection (1 in 30 websites)**

XPath Injection is an attack technique, similar to SQL Injection, used to exploit websites that construct XPath queries from user-supplied input. When an attacker is able to modify an XPath query, they may be able to obtain sensitive information from an XML document that would otherwise be out of reach.

The last thing to note about the figures above is the lack of Cross-Site Request Forgery (CSRF) representation. CSRF has had increased media coverage recently and deserves a mention here. CSRF occurs when an attacker forces a user to send a Web request he did not intend to make – a fraudulent wire transfer, password reset, spam relay, or downloading illegal content, for example. But, it certainly does not end there. The challenging part about defending against this attack is that it's a valid request from the authenticated user. There is no "hack," so to speak. Most experts agree that the majority of features on the average website are not protected against this attack and that current scanning capability is extremely limited at CSRF detection. WhiteHat's Security Operations Team is leading the research to develop an efficient process to identify CSRF and make it an integral part of our complete website vulnerability assessment and management process.

## Top Vulnerability Classes by Severity Rating

Using the Payment Card Industry Data Security Standard<sup>1</sup> (PCI-DSS) severity system (urgent, critical, high, medium, low) as a baseline, WhiteHat Security ranks vulnerability severity by the potential business impact if the issue were to be exploited.

The vast majority of websites have at least one high and critical severity vulnerability, and nearly 30% have one or more critical vulnerability. Since the last Web Application Security Risk Report, there has been a big shift from “medium” severity issues to “high” and “critical” due to the transition from a three severity rating system to a five severity system. One primary factor for the shift is not necessarily new vulnerabilities being discovered, but instead a reclassification of XSS as we said would occur when we released our last report. The potential impact of XSS increased over the last year and our reporting is a reflection of the trend.

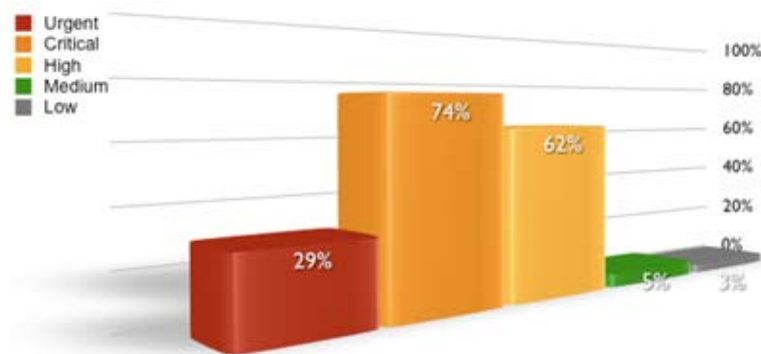


Figure 4. Likelihood of websites having vulnerabilities by severity rating

### Urgent Severity Vulnerabilities

SQL Injection continues to top the lists as the most common and highest severity vulnerability because it enables direct backend database access. As mentioned earlier, malicious attackers have used this method of attack to compromise millions of personally identifiable records. Insufficient Authorization vulnerabilities, typically only identified by manual assessment, are used to gain access to restricted areas of a website, yielding credit card data, addresses, or other customers' order information. Directory Traversal has been creeping up the list as a way to obtain access to files and directories on websites that should not be served out via the Web.

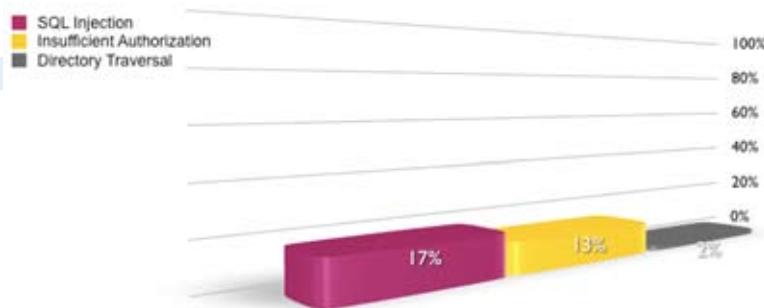


Figure 5. Top 3 Urgent Severity Vulnerability Classes

### Critical Severity Vulnerabilities

XSS is by far the most identified critical-severity vulnerability, appearing in roughly two-thirds of all websites. This is unsurprising, since most are non-persistent types and are by default assigned a critical severity rating. New attack vectors such as XSS-Phishing, Intranet Hacking and Web worms, may cause enterprises to re-evaluate XSS vulnerabilities on a case-by-case basis. Insufficient Authentication is prevalent because many websites serve content or execute functionality without first authenticating a user. Typically, an attacker need only type in the proper URL. Abuse of Functionality, which is often identified in combination with other forms of attack, occupies the third spot here. An attacker uses the existing functionality of a website, such as search boxes, chat rooms, or message boards, for his own purposes which may negatively impact the website or its users.

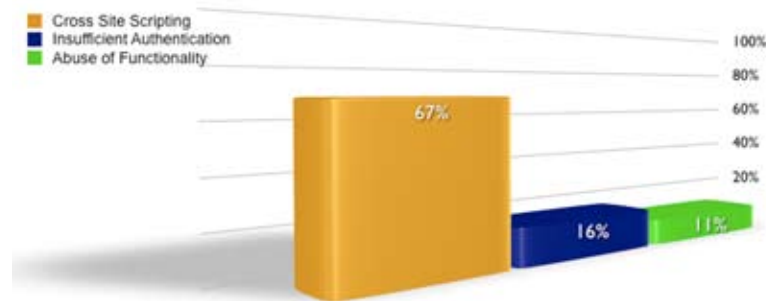


Figure 6. Top 3 Critical Severity Vulnerability Classes

### High Severity Vulnerabilities

Whether mistakenly left in Web page source code or coaxed by an attack, over a third of websites leak sensitive information including internal IP addresses, database names and passwords, software distributions and versions, etc. This type of information is extremely helpful to an attacker attempting to penetrate a system. Next, increasingly sophisticated phishing scams are starting to take advantage of Content Spoofing, found on 1 in 4 websites, as a mechanism to force the real website to host or direct users to bogus content. And, rounding out the list, 1 in 5 websites have files on their Web server that will disclose sensitive information just by requesting the information with the proper URL.

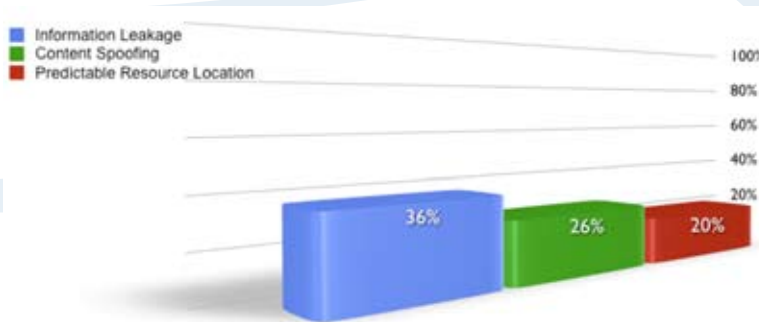


Figure 7. Top 3 High Severity Vulnerability Classes

## Conclusion

WhiteHat Security is dedicated to improving website security and website vulnerability management for its customers and the industry at-large. With 8 out of 10 websites vulnerable to attack, the first step toward stemming the onslaught of attacks is a thorough understanding of the nature of the problem. To make informed security decisions, enterprises require information about the vulnerabilities that exist, their impact, and how to prevent them from occurring. Through this type of industry awareness we expect to see the number and severity of vulnerabilities decrease across the board. This is especially true among enterprises that take a proactive approach to the problem. Organizations are encouraged to do the following:

- *Find and prioritize all website properties by designating their importance to the business and a party responsible for their security.*
- *Find and fix website vulnerabilities before the bad guys exploit them by assessing them for weaknesses with each code change.*
- *Timely remediate vulnerabilities based on severity.*
- *Implement a secure software development process utilizing an organizational standard development framework.*
- *Utilize a defense-in-depth website vulnerability management strategy*

Following these best practices enables organizations to conduct online business with confidence. No company can be expected to write flawless code, or have staff available around-the-clock to address all its Web application vulnerability issues. That's why WhiteHat created WhiteHat Sentinel, a website vulnerability management service that's customer controlled and expert managed. WhiteHat Sentinel is available 24/7, enabling companies to identify, prioritize and ultimately remediate the vulnerabilities that leave websites open to attack.

### References:

- <sup>i</sup> Web Security Threat Classification  
Web Application Security Consortium – <http://www.webappsec.org/projects/threat/>
- <sup>ii</sup> Vulnerability Type Distribution in CVE – <http://www.attrition.org/pipermail/vim/2006-September/001032.html>
- <sup>iii</sup> Microsoft ASP.NET request filtering can be bypassed allowing XSS and HTML injection attacks – [http://www.procheckup.com/Vulner\\_PR0703.php](http://www.procheckup.com/Vulner_PR0703.php)
- <sup>iv</sup> PCI Data Security Standard – <https://www.pcisecuritystandards.org/tech/index.htm>

## The WhiteHat Sentinel Service – Complete Website Vulnerability Management

**Find Everything, Protect Everything** – The WhiteHat Sentinel Service is a unique combination of expert analysis and proprietary automated scanning technology that delivers the most comprehensive website vulnerability coverage available. Worried about the OWASP Top Ten vulnerabilities or the WASC Threat Classification? Scanners alone cannot identify all the vulnerabilities defined by these standards. WhiteHat Sentinel can. Many of the most dangerous vulnerabilities reside in the business logic of an application and are only uncovered through expert human analysis.

**No False Alarms** – No busy security team has time to deal with false positives. That's why the WhiteHat Sentinel Security Operations Team verifies the results of all scans. Customers see only real, actionable vulnerabilities, saving time and money.

**Total Control** – WhiteHat Sentinel runs on the customer's schedule, not ours. Scans can be manually or automatically scheduled to run daily, weekly, and as often as websites change. Whenever required, WhiteHat Sentinel provides a comprehensive assessment, plus custom prioritization recommendations based on threat and severity levels, to better arm security professionals with the knowledge needed to secure them.

**Unlimited Assessments, Anytime Websites Change** – With WhiteHat Sentinel, customers pay a single annual fee, with unlimited assessments per year. And, the more applications under management with WhiteHat Sentinel, the lower the annual cost per application. High volume e-commerce sites may have weekly code changes, while others change monthly. WhiteHat Sentinel offers the flexibility to assess sites as frequent as necessary.

**Simplified Management** – There is no cumbersome software installation and configuration. Initial vulnerability assessments can often be up-and-running in a matter of hours. With WhiteHat Sentinel's Web interface, vulnerability data can be easily accessed, scans or print reports can be scheduled at any time from any location. No outlays for software, hardware or an engineer to run the scanner and interpret results. With the WhiteHat Sentinel Service, website vulnerability management is simplified and under control.

### About the Author

Jeremiah Grossman is the Founder and Chief Technology Officer of WhiteHat Security ([www.whitehatsec.com](http://www.whitehatsec.com)), where he is responsible for web application security R&D and industry evangelism. As an industry veteran and well-known security expert, Mr. Grossman is a frequent international conference speaker at the BlackHat Briefings, ISSA, ISACA, NASA, and many other industry events. Mr. Grossman's research, writings, and discoveries have been featured in USA Today, VAR Business, NBC, ABC News (AU), ZDNet, eWeek, BetaNews, etc. Mr. Grossman is also a founder of the Web Application Security Consortium (WASC), as well as a contributing member of the Center for Internet Security Apache Benchmark Group. Prior to WhiteHat, Mr. Grossman was an information security officer at Yahoo!, responsible for performing security reviews on the company's hundreds of websites.

### About WhiteHat Security, Inc.

Headquartered in Santa Clara, California, WhiteHat Security is a leading provider of website vulnerability management services. WhiteHat delivers turnkey solutions that enable companies to secure valuable customer data, comply with industry standards and maintain brand integrity. WhiteHat Sentinel, the company's flagship service, is the only solution that incorporates expert analysis and industry-leading technology to provide unparalleled coverage to protect critical data from attacks. For more information about WhiteHat Security, please visit [www.whitehatsec.com](http://www.whitehatsec.com).



WhiteHat Security, Inc. | 3003 Bunker Hill Lane, Suite 220 | Santa Clara, CA 95054-1144 | [www.whitehatsec.com](http://www.whitehatsec.com)

Copyright © 2007 WhiteHat Security, Inc. | Product names or brands used in this publication are for identification purposes only and may be trademarks or brands of their respective companies.

04.19.07