WhiteHat Website Security Statistic Reports

December 2008, 6th Edition

Web application security has become a priority for organizations. Just a few years ago, the security industry focused on network attacks. However, as more business moved to the Web, the attacks moved as well. Even so, it has taken several high-profile breaches, most recently websites of the U.S. Department of Homeland Security, CBS, Sony Playstation and countless other websites to raise awareness of mass scale attacks. These events have elevated Web application security to the executive level. Once business managers began to understand that Web breaches were affecting their bottom line, the motivation to prevent attacks grew. Like network security before it, Web application security has graduated from hacks for notoriety to serious business.

Phishing and other early Web attack techniques still exist, but the difference is that 79% of the attacks exploit legitimate websites. As the attackers grow smarter, the challenges for organizations become more complex. With more business on the Web, the number of sites that need to be secured has grown. The attacks are coming more frequently and using more sophisticated techniques, testing the knowledge of internal staff. Compliance requirements such as PCI are expanding their coverage of Web application security, which is also driving companies to explore solutions.

For many companies, 2008 was the year that Web application security became a reality. However, it is difficult to know where to start. After all, you cannot fix if you don't know what is broken. That's the race business owners are in with the criminal element.

This report provides a high-level perspective on the leading Web application security issues across industries such as retail, financial services, technology and healthcare, based on real-world sites.

The WhiteHat Website Security Statistics Report provides a one-of-a-kind perspective on the state of website security and the issues that organizations must address to avert attack. WhiteHat has been publishing the report, which highlights the top ten vulnerabilities, vertical market trends and new attack vectors, since 2006.

The WhiteHat report presents a statistical picture of current website vulnerabilities, accompanied by WhiteHat expert analysis and recommendations. WhiteHat's report is the only one in the industry to focus solely on unknown vulnerabilities in custom Web applications, code unique to an organization, within real-world websites.

WhiteHat issues continued installments of the Website Security Statistics Report on a quarterly basis. To ensure the report remains useful and relevant, WhiteHat incorporates feedback and ideas from leading industry thought leaders and influencers. Based on feedback already received, the latest report includes: comparing vulnerability prevalence by severity, top ten vulnerability classes sorted by percentage likelihood and an outline of the types of technology typically encountered during WhiteHat vulnerability assessments mapped with the associated vulnerability percentage breakdown.

Data Collection Process

Web security is a moving target so, enterprises need timely information about the latest attack trends, how they can best defend their websites, and visibility into their vulnerability life-cycle. Through its Software-as-a-Service (SaaS) offering, WhiteHat Sentinel[1], WhiteHat Security is uniquely positioned to deliver the knowledge organizations need to protect their brands, attain PCI compliance and avert costly breaches.

WhiteHat customers use Sentinel, an annual subscription service, to assess and manage their website vulnerability exposure. Each week, WhiteHat Sentinel assesses hundreds of public-facing and pre-production websites for vulnerabilities using a consistent and repeatable three-phase process:

1. *Proprietary scanning technology identifies technical vulnerabilities such as Cross-Site Scripting, SQL Injection, some forms of Cross-Site Request Forgery and many others.*
2. *Experts create customized tests for each website to uncover business logic flaws including Insufficient Authorization, Insufficient Authentication, Abuse of Functionality, etc.*
3. *Results are verified to remove false-positives and assign an appropriate level of severity in order for data to be accurate and actionable.*

This one-of-a-kind perspective gives WhiteHat an unparalleled view into the state of website security across vertical markets and different attack vectors, in companies of all sizes. Whether an organization is currently starting a website security program, or has been assessing their sites for years, this report provides insight into the breadth of attacks, overall industry health, and other issues that can help focus a plan of attack and raise awareness of nascent attack trends.

It is important to note that the websites WhiteHat Sentinel manages likely represent the most “important” and “secure” websites on the Web - those conducting high-volume transactions and regulating sensitive information across the retail, finance, insurance, healthcare, and IT industries. With access to an enormous sampling of vulnerabilities in custom Web applications, we are able to publish the most prevalent issues on an aggregate basis. It is also important to understand the differentiation between the data contained within this document and the statistics presented in reports by Symantec[2], Mitre (CVE)[3], IBM X-Force[4], SANS[5], and others. Those reports track *publicly* disclosed vulnerabilities in commercial and open source software.

WhiteHat Security captured the data contained within this report by focusing solely on previously unknown vulnerabilities in custom Web applications, code unique to an organization, on real-world websites (Figure 1).

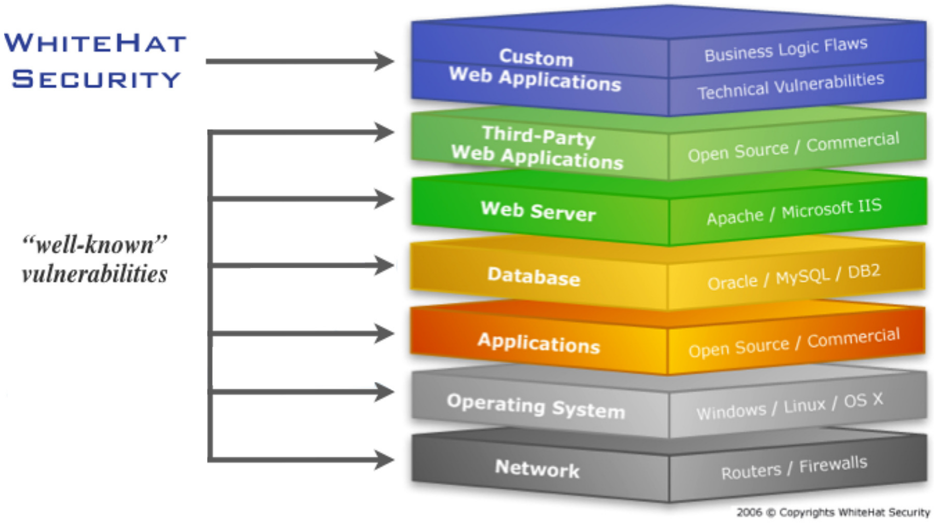

Figure 1. Software vulnerability stack

After more than two years of reporting on the industry, for the first time we see a positive trend--the majority of vulnerabilities discovered have been resolved. This is significant because it demonstrates that a consistent, methodical Web application security program does in fact make organizations more secure. Consistency is important, because as the data will show, new attack techniques are constantly being tested in the wild and only a regular assessment approach will identify these new threats. PCI-DSS is also placing pressure on application security practitioners to intensify their efforts.

Data Overview

- *877 total websites*
- *Vast majority of websites assessed for vulnerabilities weekly*
- *Vulnerabilities classified according to WASC Threat Classification*
- *Vulnerability severity naming convention aligns with PCI-DSS*
- *Obtained between January 1, 2006 and December 1, 2008*

Key Findings

- *Total identified vulnerabilities (open & closed): 14,718*
- *Current open vulnerabilities: 5,283 (64% resolved)*
- *Historically, 82% of assessed websites have had at least one issue of HIGH, CRITICAL, or URGENT severity*
- *63% of assessed websites currently have issues of HIGH, CRITICAL, or URGENT severity*
- *Historically, websites average 17 vulnerabilities identified during the lifetime of the assessment cycle*
- *Websites currently average 6 open vulnerabilities*
- *Cross-Site Request Forgery gained two spots in the Top Ten moving to #8*
- *Vulnerability time-to-fix metrics are not changing, typically requiring weeks to months to achieve resolution*
- *Roughly 50% of the most prevalent Urgent severity issues have been resolved*

When interpreting the results there are several factors that should be considered:

- *The mix of websites ranges from highly complex and interactive sites with a large attack surface to static brochureware sites.*
- *Vulnerabilities are organized and counted based upon a unique Web application and class of attack. For example, if there are five possible parameters in a single Web application (/foo/webapp.cgi), three of which are vulnerable to SQL Injection, it is counted as one vulnerability (not three).*
- *Vulnerabilities do not include "best practice" findings. For example, if a website mixes SSL content with non-SSL on the same Web page, while this may be considered a business policy violation, it must be taken on a case-by-case basis. As an example, the lack of encrypted passwords or data storage on the system are not considered vulnerabilities for the purpose of this report. Only issues that can be directly exploited remotely are included.*
- *Vulnerability assessment processes are incremental and ongoing, the frequency of which is customer-driven and as such should not automatically be considered "complete." However, the vast majority of WhiteHat Sentinel customers do assess their sites on a weekly basis. When interpreting the data it is best to keep in mind new attack vectors are always being researched by attackers, making it best to view the data as a best-case scenario based on the most up-to-date information available.*

Vulnerability Prevalence by Severity

In order for organizations to take appropriate action, each website vulnerability must be independently evaluated for business criticality. For example, not all XSS or SQL Injection vulnerabilities are equal, making it necessary to consider its true "severity" for an individual organization. Using the Payment Card Industry Data Security Standard[6] (PCI-DSS) severity system (Urgent, Critical, High, Medium, Low) as a baseline, WhiteHat rates vulnerability severity by the potential business impact if the issue were to be exploited and does not rely solely on default settings. It should also be noted that according to the PCI-DSS, any websites with URGENT, CRITICAL, or HIGH severity issues cannot be considered compliant.

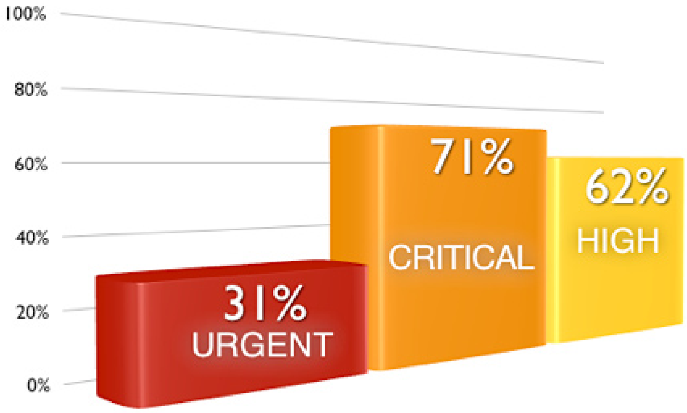

Figure 2. Percentage likelihood of websites having a least one vulnerability (sorted by severity)

Overall vulnerability counts continue to decline, but the likelihood of websites having at least one issue of a specific severity has also remained constant when compared to previous reports. While having dozens of Urgent, Critical, or High severity issues makes it easier for an attacker to achieve a successful data compromise, finding and exploiting a single issue is all that is required. The closer these numbers approach to zero, the better, yet they remain unchanged. They remain unchanged largely because organizations have difficulty remediating 100% of their flaws of a specific type and severity.

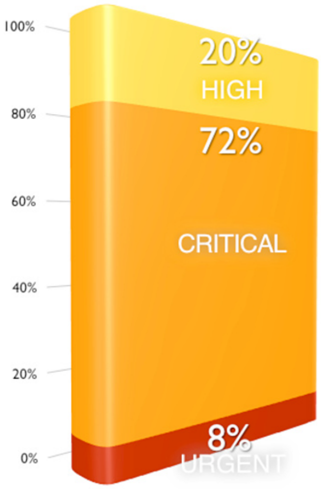

Figure 3. Percentage of vulnerabilities (sorted by severity)

The breakdown of vulnerabilities by severity continues to remain steady. These numbers will track closer to new attack techniques being publicly disclosed rather than the actual security of websites. When issue types of attacks arise they will be identified by WhiteHat Sentinel in existing websites even when their code has remained unchanged. Historically entirely new classes of attack are rare, especially those that affect a large percentage of existing websites. It is more common that existing attack techniques are improved upon, increasing their severity in certain edge cases that must be considered on a website by website basis.

The Top Ten

WhiteHat Security determines the most prevalent issues by calculating the percentage likelihood of a particular vulnerability class occurring within websites (Figure 4). This approach minimizes data skewing in website edge cases that are either highly secure or extremely risk-prone.

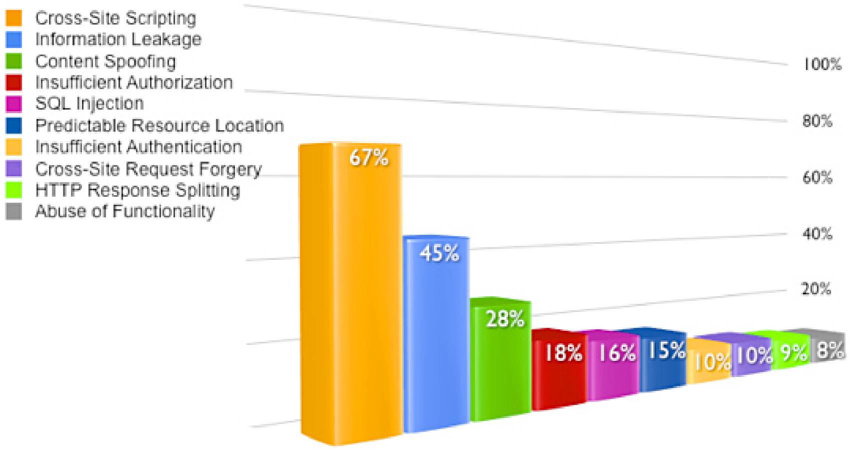

Figure 4. Top 10 vulnerability classes (sorted by percentage likelihood)

Since the last Website Security Statistics Report, the Top Ten graph has seen a few notable changes, while other areas have remained static. This should demonstrate that the data contained within this report is a representative sampling of the security of the Web's more important e-commerce related websites. Adding volumes of new website vulnerability data has had no significant alteration on the statistics.

As predicted, Cross-Site Request Forgery (CSRF) moved up to #8 on the Top Ten. As a reminder, all statistics surrounding CSRF should be considered suspect, including all other reports issued worldwide, because identifying this issue reliably by purely automated means remains elusive. WhiteHat has been making steady gains in automatically detecting CSRF, but most are found though manual custom testing by WhiteHat's Security Operations Team (the same for all researchers and pen-testers globally). WhiteHat estimates that a better judgment of CSRF's prevalence is similar to that of Cross-Site Scripting (XSS).

Business logic flaws have remained steady in the Top Ten, demonstrating that these workflow flaws, which include Insufficient Authorization, Insufficient Authentication, Abuse of Functionality, and Content Spoofing, are still overlooked by many organizations. While not at the top of the list when calculating raw numbers, these flaws are still highly prevalent across websites and can lead directly to business loss through non-sophisticated attacks.

To supplement vulnerability likelihood statistics, the following graph (Figure 5) illustrates prevalence in the overall vulnerability population. Notice how greatly the two graphs differ. The reason is that one website may possess hundreds of unique issues of a specific class, such as Cross-Site Scripting, SQL Injection, or Information Leakage, while another website may not contain any.

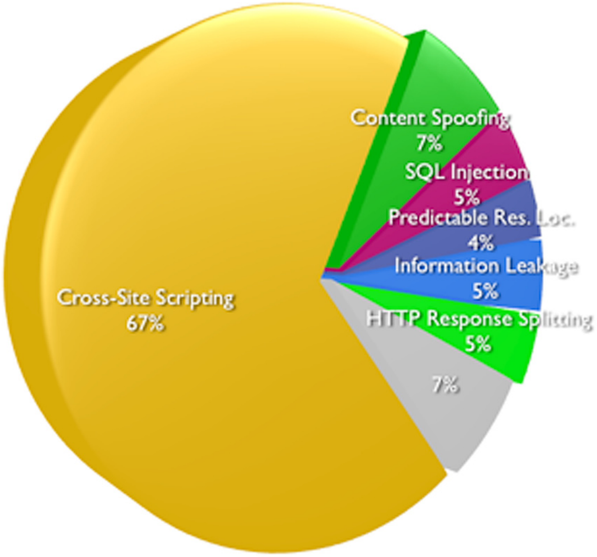

Figure 5. Vulnerability classes (sorted by class population)

Development Technology and Vulnerabilities

Table 1 provides insight into the types of technologies encountered during WhiteHat Sentinel vulnerability assessments and the associated vulnerability percentage breakdown. The statistics are not meant to establish which technology is more secure. For example, the under-representation of PHP likely means that this technology is not being utilized by those in the sample set relative to others. The large set of "unknown" are those without file extension, probably a large supply of Java servlet containers. In future reports, we plan to offer *likelihood of vulnerability* numbers related to specific technologies.

URL Extension	% of websites	% of vulnerabilities
unknown	58%	41%
asp	26%	25%
aspx	22%	9%
jsp	8%	7%
xml	9%	1%
do	7%	3%
php	5%	3%
html	4%	2%
old	3%	1%
dll	3%	1%
cfm	3%	3%

Table 1.

Attack Surface and Number of Vulnerabilities

Application inputs are areas where arbitrary data is received, potentially leaving the software open to attack (attack surface). Application inputs include, but are not limited to, query and POST data parameter names/values, cookies, files paths/names, and so on. WhiteHat is constantly improving its ability to accurately and comprehensively identify application input points via Sentinel technology. These numbers come directly from spidering all of a website's Web pages while maintaining a logged-in state. It is possible, but not yet confirmed through our data, that a correlation exists between the numbers of application inputs and the number of overall vulnerabilities. In the future we plan to measure the correlation between when a website changes, or increases its attack surface, to the number and type of vulnerabilities found.

Average number of inputs per website: 229
Average ratio of vulnerability count / number of inputs: 1.95%

Time to Fix

When website vulnerabilities are identified, there is always a certain amount of time required for the issue to be resolved. Resolution could take the form of a software update, configuration change, Web application firewall rule, etc. Ideally the time to fix should be as short as possible because an open vulnerability represents an opportunity for hackers to exploit the website. But, no remedy is instantaneous. While the issue is being handled, an organization has four options:

1. *Take the website down*
2. *Revert to an older version of the website/code (if it is secure)*
3. *Stay up while exposed*
4. *Virtually patch the issue with a Web Application Firewall (i.e. WhiteHat Sentinel / F5 Application Security Manager integration[7])*

Vulnerabilities can and do exist despite the most regimented software development lifecycle and commitment to defense-in-depth. Normally: option #1 (taking down the website) is employed when an incident has occurred such as when a website is "hacked"; option #2 (rolling back the code) is preferable when a hot fix is not back-ported to development and is later overwritten, likely the result of a high severity high threat issue. Practically speaking, the vast majority of website owners default to option #3 (risk acceptance), essentially assuming the risk rather than halt business for lack of better options. Option #4 is taking hold for organizations who require immediate protection and additional time to resolve issues in the code as time and budget allow.

The remediation challenges most organizations face are the time consuming process of allocating the proper personnel, prioritizing the tasks, QA / regression testing the fix, and finally scheduling a production release. IT security professionals are challenged by a general lack of corporate awareness of the Web security problem and the resulting limited motivation to remediate an issue once it has been identified. At the end of the day, the remediation process takes time, but what is important to understand is how much.

To perform this analysis, we focused on vulnerabilities identified and resolved within the last twelve months, between December 1, 2007 and December 1, 2008. The data was then sorted by the most common URGENT, CRITICAL, and HIGH severity issues. There are two aspects worth noting that may bias the sampling in opposing directions:

Should a vulnerability be resolved it could take up to seven days before it is retested and confirmed closed by WhiteHat Sentinel, depending upon the customer's scan schedule. However, a customer can proactively use the auto-retest function to get real-time confirmation of a fix.

Not all vulnerabilities identified within this period have been resolved, which means the time to fix measurements are likely to grow (See table 2).

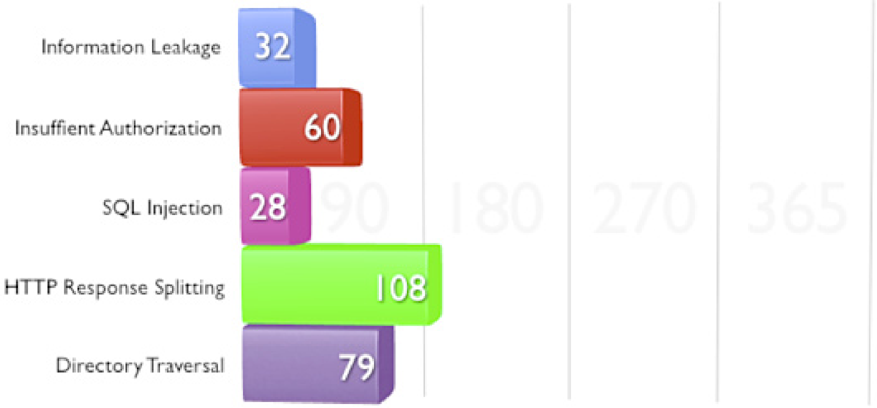

Figure 6. Average number of days for the top five URGENT severity vulnerabilities to be resolved

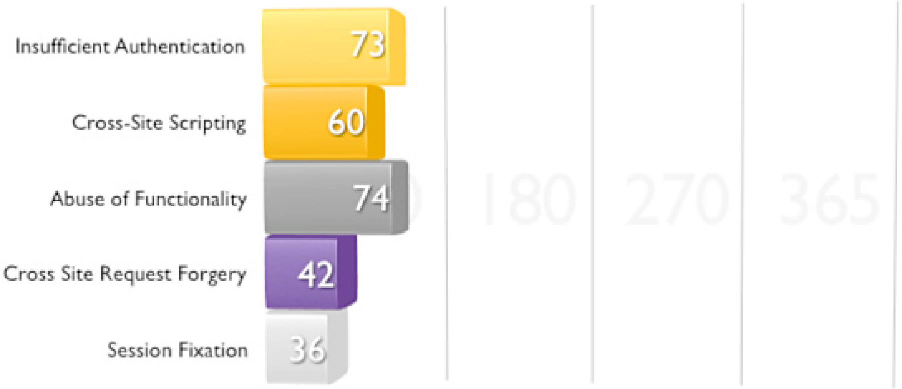

Figure 7. Average number of days for the top five CRITICAL severity vulnerabilities to be resolved

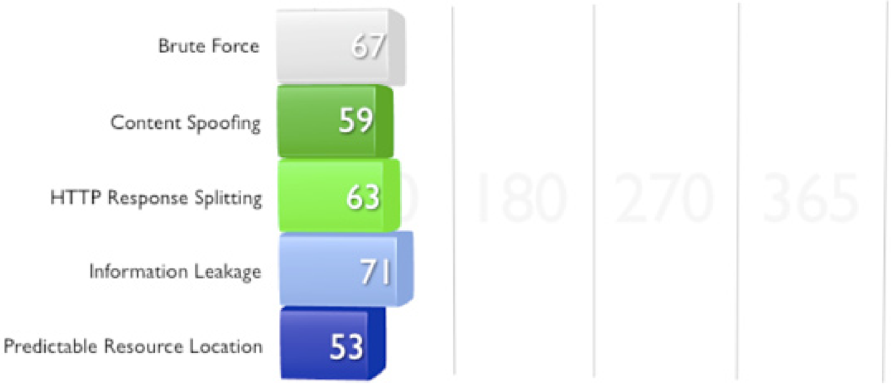

Figure 8. Average number of days for the top five HIGH severity vulnerabilities to be resolved

Class of Attack	% Resolved	Severity
Information Leakage	55%	urgent
Insufficient Authorization	55%	urgent
SQL Injection	42%	urgent
HTTP Response Splitting	17%	urgent
Directory Traversal	25%	urgent
Insufficient Authentication	72%	critical
Cross-Site Scripting	52%	critical
Abuse of Functionality	60%	critical
Cross-Site Request Forgery	49%	critical
Session Fixation	87%	critical
Brute Force	81%	high
Content Spoofing	83%	high
HTTP Response Splitting	71%	high
Information Leakage	75%	high
Predictable Resource Location	86%	high

Table 2. Percentage of vulnerabilities resolved (sorted by class & severity)

Though time-to-fix and percentage fixed measurements are slowly improving, there is still significant room for improvement. Most Urgent severity issues are taking roughly a month or two to fix, while Critical issues average just over two months. However, when you compare these metrics to those of the last report, there is indeed improvement. IT security and development organizations are coordinating new procedures when it comes to dealing with website vulnerabilities and it is working to close the time-to-fix gap. Still, challenges remain to a speedy remediation cycle including:

- *A disconnect between many IT security and software development groups. IT security possesses little control over the security of the website in comparison with that of the network or its hosts.*
- *IT security has a difficult time explaining the details of a vulnerability to an unfamiliar audience and conveying the overall risk. This stems largely from a lack of adequate secure software development training for developers.*
- *The business may not allocate the resources necessary to resolve the issue, instead opting to focus on features rather than vulnerability remediation.*

Comparing Industry Verticals

Figure 9 shows the percentage of websites with at least one Urgent, Critical, or High severity issue sorted by industry vertical. The majority of websites have these types of issues, which also would likely not allow them to be classified as PCI-DSS compliant. Beyond the generally poor state of website security, we did notice that the retail sector continues to outperform other verticals since the last report. We maintain the likely cause is they receive a larger volume of battlefield testing.

The bulk of a retail website's (and other popular important websites) functionality is accessible without the need to login or difficult registration process. This means more external attackers are able to target these websites and spot weaknesses, often exploited, which are then remediated by the organization. This is in contrast to the financial services or insurance sectors where the bulk of functionality is protected behind a login screen and an account is therefore harder to access without doing business directly with the company. So, once an attacker gets an account, it is likely that considerably fewer people have tested these areas of functionality before them. Constant battlefield testing is an important factor in website security.

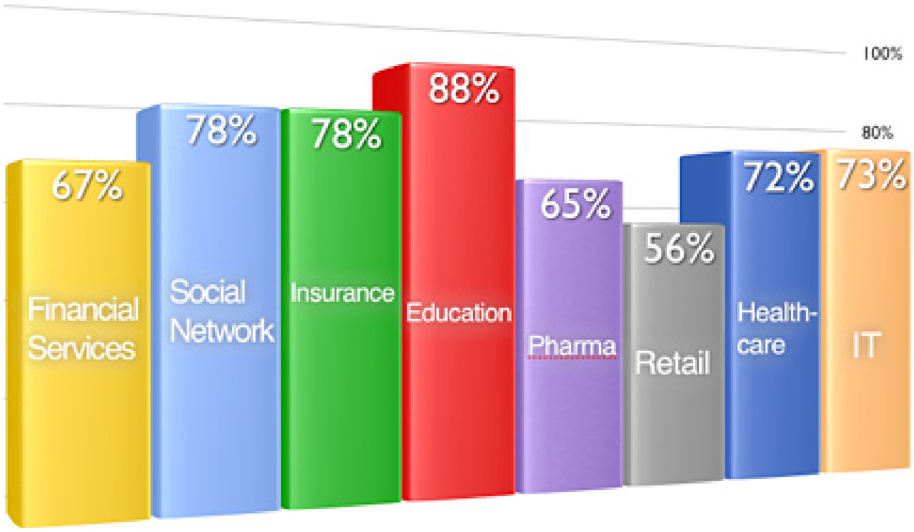

Figure 9. Percentage of websites with an URGENT, CRITICAL or HIGH severity vulnerability sorted by industry vertical

Conclusion

With the right processes in place providing visibility, such as ongoing vulnerability assessments, website security can be measurably improved. Continual assessments allow for newly introduced vulnerabilities to be identified in near real-time, enabling organizations to take immediate action, empower root-cause analysis, and demonstrate effective due diligence. Historically, annual consulting engagements and desktop vulnerability scanners have not been successful in driving this process.

As shown in this report, WhiteHat Sentinel customers have nearly fixed two-thirds of identified issues, do so in a shorter amount of time, and decrease their average number of vulnerabilities. We will enable our customers to continue this trend through technology and process innovation, like our integration with Web application firewalls and other software that makes the remediation process easier and more efficient.

WhiteHat Security is dedicated to improving website security and website vulnerability management for its customers and the industry at-large. With 6 out of 10 websites still currently vulnerable to attack, the first step toward stemming the onslaught of attacks is with 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're determined to help organizations see the number and severity of vulnerabilities decrease across the board. Organizations are encouraged to do the following:

1. *Find and prioritize all websites by designating their importance to the business; identify the party responsible for their security.*
2. *Find and fix website vulnerabilities before the bad guys exploit them by frequently assessing them for weaknesses.*
3. *Base timely remediation of vulnerabilities on severity, difficulty of exploitation, and business value of the website.*
4. *Implement a secure software development process utilizing an organizational standard development framework, scheduled developer security education program, and success incentives.*
5. *Utilize a defense-in-depth website security strategy that includes a Web Application Firewall providing organization with timely and additional security against zero-day threat and difficult to resolve issues.*

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. WhiteHat Sentinel, a website vulnerability management service, was created to enable companies to take control of website security and narrow their window of risk. WhiteHat Sentinel is available 24/7, enabling companies to identify, prioritize and ultimately remediate the vulnerabilities that leave websites open to attack.

Glossary: The Top Ten Defined

1. **Cross-Site Scripting**[8] (XSS) (7 out of 10 websites) is easily the most prevalent website vulnerability. XSS has proven to be extremely hazardous to businesses and consumers in the form of either Web Worms9, "Phishing with Superbait10" scams, Javascript malware-laced defacements, and malicious Web Widgets. The evolution of JavaScript malware, finding its way into more and more attackers toolboxes, has made finding and fixing this vulnerability more vital than ever.

2. **Information Leakage**[11] (2 in 5 websites) occurs when a website knowingly or unknowingly reveals sensitive information such as developer comments, user information, internal IP addresses, source code, software versions numbers, error messages/codes, etc., which may all aid in a targeted attack. While most of the time rated MEDIUM or LOW severity, several Information Leakage issues could be used in combination to compromise a website.

3. **Content Spoofing**[12] (1 in 4 websites) is often used in phishing scams (or intelligence gathering) 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. **Insufficient Authorization**[13] (1 in 6 websites) 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, exercise unauthorized access, and potentially defraud the systems. 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.

5. **SQL Injection**[14] (1 in 6 websites) has been at the center of some of the largest credit card, identity theft incidents, and mass scale website compromises. 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 entered into a standard Web browser entire databases could fall into the wrong hands.

6. **Predictable Resource Location**[15] (PRL) (1 in 6 websites) Over time, many pages on a website become unlinked, orphaned, and forgotten--especially on websites experiencing a high rate of content and/or code updates. These Web pages sometimes contain payment logs, software backups, post dated 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.

7. **Insufficient Authentication**[16] (1 in 10 websites) 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. These types of issues are common in financial, healthcare systems, and general content management systems where there is a high concentration of complex business logic functionality.

8. **Cross-Site Request Forgery** [17] (1 in 10 websites) (aka Session Riding, Web Trojan, Confused Deputy, etc.) allow an attacker to force an unsuspecting user's browser to make a Web request they didn't intend. For example, the attacker could force a user to compromise their own banking, eCommerce or other website accounts invisibly without

their knowledge. Since the forged request is coming the legitimate user, even when they are logged-in, the website will accept it as being the intent of that user.

9. **HTTP Response Splitting**[18] (1 in 11 websites) is an attack technique in which a single request is sent to the website in such a way that the response may appear to look like two. Depending on the network architecture of the website or the behavior of a users Web browser, the "second" HTTP response that's under the control of the attacker can be used to poison cache servers, deface Web pages, perform session fixation, etc.

10. **Abuse of Functionality**[19] (1 in 12 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."

References

[1] WhiteHat Sentinel: http://www.whitehatsec.com/home/services/services.html

[2] Internet Security Threat Report: http://www.symantec.com/business/theme.jsp?themeid=threatreport

[3] Vulnerability Type Distributions in CVE: http://cwe.mitre.org/documents/vuln-trends/index.html

[4] IBM Internet Security Systems X-Force® 2007 Trend Statistics: http://www.iss.net/x-force_report_images/2008/index.html

[5] SANS Top 20: http://www.sans.org/top20/

[6] PCI Data Security Standard: https://www.pcisecuritystandards.org/tech/index.htm

[7] WhiteHat Sentinel and F5 WAF Integration: http://www.whitehatsec.com/home/assets/movies/F5WAFintegration640.html

[8] Cross-Site Scripting: http://www.webappsec.org/projects/threat/classes/cross-site_scripting.shtml

[9] Cross Site Scripting Worms and Viruses: http://www.whitehatsec.com/home/assets/WP5CSS0607.pdf

[10] Phishing with Superbait: http://www.whitehatsec.com/home/assets/presentations/phishing_superbait.pdf

[11] Information Leakage: http://www.webappsec.org/projects/threat/classes/information_leakage.shtml

[12] Content Spoofing: http://www.webappsec.org/projects/threat/classes/content_spoofing.shtml

[13] Insufficient Authorization: http://www.webappsec.org/projects/threat/classes/insufficient_authorization.shtml

[14] SQL Injection: http://www.webappsec.org/projects/threat/classes/sql_injection.shtml

[15] Predictable Resource Location: http://www.webappsec.org/projects/threat/classes/predictable_resource_location.shtml

[16] Insufficient Authentication: http://www.webappsec.org/projects/threat/classes/insufficient_authentication.shtml

[17] Cross-Site Request Forgery: http://en.wikipedia.org/wiki/Cross-site_request_forgery

[18] HTTP Response Splitting: http://www.webappsec.org/projects/threat/classes/http_response_splitting.shtml

[19] Abuse of Functionality: http://www.webappsec.org/projects/threat/classes/abuse_of_functionality.shtml

The WhiteHat Sentinel Service – Total Website Security

Find and Fix Vulnerabilities, Protect Your Website – The WhiteHat Sentinel Service delivers the most accurate and 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.

Virtually Eliminate False Positives – 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.

Virtual Patching is Now a Reality – WhiteHat Sentinel can directly configure policies on a WAF to protect against vulnerability exploits (e.g., cross-site scripting, SQL injection) that were found during the scanning process. Normally, this will be a two step process: (1) identify vulnerabilities using WhiteHat Sentinel and (2) create highly-targeted policies on WAF. This makes the process simpler for the end user – find the problem, then fix the problem with the click of a button.

Dynamic Improvement and Refinement – WhiteHat Sentinel stays one step ahead of the latest website attack vectors with persistent updates and refinements to its service. Updates are dynamic – as often as one day to several weeks, versus up to six months or longer for traditional software tools. And, Sentinel uses its unique "Inspector" technology to apply identified vulnerabilities across every website it evaluates. Ultimately, each site benefits from the protection of others.

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 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 WhiteHat Security, Inc.

Headquartered in Santa Clara, California, WhiteHat Security is the leading provider of website security solutions that protect critical data, ensure compliance and narrow the window of risk. WhiteHat Sentinel, the company's flagship product family, is the most accurate, complete and cost-effective website vulnerability management solution available. It delivers the flexibility, simplicity and manageability that organizations need to take control of website security and prevent Web attacks. Furthermore, WhiteHat Sentinel enables automated mitigation of website vulnerabilities via integration with Web application firewalls. To learn more about WhiteHat Security, please visit our website at www.whitehatsec.com.

WhiteHat SECURITY

WhiteHat Security, Inc. | 3003 Bunker Hill Lane | Santa Clara, CA 95054
408.343.8300 | www.whitehatsec.com