

Technology Alone cannot Defeat Website Attacks

Understanding Technical vs. Logical Website Vulnerabilities

May 2007
Jeremiah Grossman
Founder and CTO, WhiteHat Security

A large, light blue, abstract graphic element consisting of several overlapping, curved shapes that resemble a stylized hat or a wave, positioned in the lower half of the page.

A WhiteHat Security Whitepaper

Introduction

On November 11th, 2003, the chess-playing machine X3D Fritz tied grandmaster and former world champion Garry Kasparov in a four-game match. In this classic contest of Man vs. Machine, X3D Fritz performed so impressively that the game was heralded as a victory for artificial intelligence. X3D Fritz's powerful play was achieved by calculating millions of moves per second accompanied by gigabytes of stored positions. Each time Kasparov moved a chess piece, X3D Fritz would analyze the board by drawing upon its vast knowledge base to select the best possible move. So what do chess, the world's most dominant computer chess machine, and Garry Kasparov have to do with web application, or rather, website security?



Chess: Man Versus Machine. [Photograph]. Retrieved April 17, 2007, from Encyclopedia Britannica Online: <http://www.britannica.com/eb/art-61084>

Figure 1. Garry Kasparov vs. Machine

For many years, security professionals have thought that there would come a day when technology alone could identify all Web application vulnerabilities and prevent all attacks, eliminating the need for the Kasparovs of the world. What we've come to understand is website security is a fundamentally different game than chess, or even network security. It's highly unlikely that machines will ever replace man completely in the process of assessing website security. What's important to understand is why.

Chess is a straightforward game. The board presents a finite number of legal moves and a limited amount of end-game positions. With chess, it's mathematically possible to calculate every move that may result from a given position and further "n"

moves into the future. Since the game itself is defined and finite, although granted extremely large, the path to victory can be completely automated and followed precisely. Eventually, computers will win at chess every time rather than settling for a tie. Websites are at the opposite end of the spectrum: They maintain an open door policy with regard to user interaction, rarely following Internet standards, and never operate the same way twice. Simple tasks such as shopping online or Web banking are drastically different functionally and architecturally. Website vulnerability scanners operate in a complicated environment where the end result of a process is anything but obvious.

The Difference between Technical and Logical Vulnerabilities

Website vulnerability scanners depend on the relative predictability of websites to identify security issues. Using a loose set of rules, scanners function by simulating Web attacks and analyzing the responses for telltale signs of weakness. From experience, we know how a website will normally react when there is a security issue present. We also know that if sending a website certain meta-characters produces a database ODBC error message, a SQL Injection issue has likely been detected. At WhiteHat Security, we call these "**technical vulnerabilities**," and scanners have become fairly proficient at identifying them. But, as websites become increasingly sophisticated, yesterday's telltale signs are today's false-positives. As such, we're not guaranteed that a specific result necessarily indicates that a security issue is present. This has made the automated process of finding simple vulnerabilities very complex, and finding difficult ones impossible.

Consider the following example. If we visit a website and are presented with the following URL:

http://example/order.asp?item=50&price=300.00

Can we guess what the application order.asp, combined with the parameters “item” and “price” do? Using intelligence unique to humans, we can quickly deduce their purpose with relative certainty – this is a product ordering application. The “item” parameter is the particular product we are interested in. In our case, let’s say a digital camera. The price parameter is the amount we are going to pay for our digital camera

What happens if we changed the price of 300.00 to 100.00? Or, 1.00? Does the website still sell us the digital camera? If so, we can easily understand that the website should not have allowed the price alteration. As humans, we possess a natural ability to assess context, and we aptly refer to these types of issues as “**logical vulnerabilities**,” issues that only humans can identify.

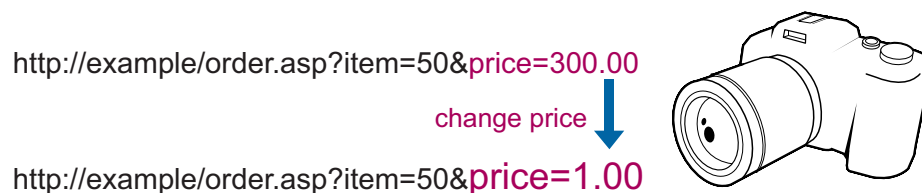


Figure 2. An example of a logic vulnerability – an issue that only humans can identify.

Now, if an automated scanner attempted the very same attack in a generic fashion, how would it decide if a custom website’s response was good or bad? How would it know if the attack worked or was adequately defended? Or, what the item and price parameters were supposed to do in the first place?

The answer is clear: scanners cannot reliably make these assumptions. The numbers in the URL easily could have meant something else entirely when presented in a different context. The rules for what is supposed to happen on a website are not defined as they are in chess. These decisions require contextual knowledge of the system, plus the ability to “logically” understand any number of previously undefined results.

In mathematics, this very large obstacle is commonly referred to as the undecidable problem. An undecidable problem is a problem that cannot be solved for all cases by any algorithm (or computer program). Chess is **NOT** an undecidable problem, since it can be accounted for in all instances at all times. Fully analyzing custom Web application software for vulnerabilities IS an undecidable problem. That’s why the game of chess can be fully automated by a computer and identifying vulnerabilities in custom website cannot.

Conclusion

There are unique aspects of the human mind that computers have yet to duplicate. WhiteHat's statistics, based on aggregate data from thousands of assessments, indicate that only about half of the possible website security issues can be tested for in a completely automated fashion. The remaining tests for logical issues require the involvement of a website security Garry Kasparov.

In a thorough Web application vulnerability assessment, potentially hundreds of thousands of customized tests must be performed. By hand, even the world's best experts would never be able to complete this much work in a feasible amount of time. Similar to the work of X3D Fritz, harnessing the power of a truly enterprise class vulnerability scanner greatly decreases the workload by performing the monotonous tasks that can be automated. Scanners are great at tackling technical vulnerabilities such as Cross-Site Scripting and SQL Injection, and not effective at identifying price list modification, Credential/Session Prediction, and Insufficient Authorization.

By understanding these concepts, the industry has acknowledged that the pairing of experienced security experts with an automated scanner is a best-practice for achieving complete website vulnerability coverage.

While artificially intelligent computers like HAL 9000 may arrive or someone may achieve the mathematical breakthrough of the century, currently technology alone is no match for the human mind.

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.

Continuous 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 continuous – 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.”

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), 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.



WhiteHat Security, Inc. | 3003 Bunker Hill Lane, Suite 220 | Santa Clara, CA 95054-1144 | 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.

05.02.07