

## Combating Malicious Mobile Code

*New Alternatives for a Growing  
Internet Problem*

Websense, Inc.  
World Headquarters  
10240 Sorrento Valley Road  
San Diego, California 92121  
USA  
Tel: 800.723.1166 or 858.320.8000  
Fax: 858.458.2950  
[www.websense.com](http://www.websense.com)

### *Abstract*

*The need for dynamic content, feature-rich Web sites and for a Windows-based scripting language has led to the adoption of various interpreted, compiled, and scripted languages. These include JavaScript, Active X, Visual Basic Script (VBScript), and Java Applets.*

*Although designed with functionality in mind, the adoption of these technologies within popular Internet applications and operating systems has led to the widespread problem of viruses, Trojan horses and their attacks, worms, and script attacks.*

*Websense, Inc. announces a new Websense category Security PG™: Malicious Web Sites to identify those sites with contents that intentionally modify end user systems without consent and potentially cause harm. The timely addition of the Security PG category adds computer security that seamlessly integrates with the functionality of Websense Enterprise, a recognized industry leader*

---

## Contents

Executive Summary .....	1
Introduction .....	2
The Crisis: Malicious Mobile Code (MMC).....	2
The Current Detection Technologies: Strengths and Weaknesses .....	3
The Websense Solution.....	5
Conclusion.....	6
About Websense, Inc. ....	6
Glossary.....	6

---

## Executive Summary

In 2001, CERT/CC received more than twice the number of security incident reports than in 2000. Experts anticipate continuing attacks, with the number of incidents doubling again in 2002.<sup>1</sup>

In *An Overview of Incident and Vulnerability Trends*<sup>2</sup>, CERT notes an increase in incidents involving Trojan horses and Malicious Mobile Code (MMC). The best known of these are the Nimda Worm and VBS/Happytime VBS Worm, which attacked systems around the world in 2001. At the recent CanSecWest Security conference, reports indicate that more than 18,000 systems are still infected with the Code Red Worm<sup>3</sup>. In 2001, worms and other malignant code cost companies more than \$13.2 billion<sup>4</sup>.

The harmful intent of those creating MMC code, combined with design flaws in security systems, operating systems, and infrastructures, is forcing organizations to re-examine their position on employee access to and use of the Internet and Web. The industry is struggling to stay ahead of MMC designers, and numerous security products are available to keep malignant code from executing on computers and computer networks.

---

<sup>1</sup> [http://www.cert.org/stats/cert\\_stats.html#vulnerabilities](http://www.cert.org/stats/cert_stats.html#vulnerabilities)

<sup>2</sup> <http://www.cert.org/present/cert-overview-trends/module-1.pdf>

<sup>3</sup> <http://news.com.com/2100-1001-899245.html>

<sup>4</sup> <http://www.computereconomics.com/article.cfm?id=133>

---

## Introduction

In 2001, CERT/CC received more than twice the number of security incident reports than in 2000. Experts anticipate continuing attacks, with the number of incidents doubling again in 2002.<sup>5</sup>

In *An Overview of Incident and Vulnerability Trends*<sup>6</sup>, CERT notes an increase in incidents involving Trojan horses and Malicious Mobile Code (MMC). The best known of these are the Nimda Worm and VBS/Happytime VBS Worm, which attacked systems around the world in 2001. At the recent CanSecWest Security conference, reports indicate that more than 18,000 systems are still infected with the Code Red Worm<sup>7</sup>. In 2001, worms and other malignant code cost companies more than \$13.2 billion<sup>8</sup>.

The harmful intent of those creating MMC code, combined with design flaws in security systems, operating systems, and infrastructures, is forcing organizations to re-examine their position on employee access to and use of the Internet and Web. The industry is struggling to stay ahead of MMC designers, and numerous security products are available to keep malignant code from executing on computers and computer networks.

## The Crisis: Malicious Mobile Code (MMC)

A variety of security issues center around Visual Basic Script (VBScript), JavaScript, Java Applets, and Active X technologies. Problems include the ability inherent within these technologies to read from and write to files and folders on the client hard drive, to run and attach programs on the client machine, and the potential for distributing malicious mobile code. This poses serious problems for computer security groups, especially in light of the numerous holes reported for products from many companies including industry leaders.

While these software vendors continually provide security fixes and patches to address these known problems, users are often far behind the curve. Customers may not be aware of service pack upgrades and patches and, as a result, they fail to download and implement them. They may also use default security settings on their browsers that do not prevent automatic execution of many scripting technologies such as Java and Active X.

Other critical factors continue to plague security conscious organizations as increased employee access to the Internet and the Web becomes the norm. Browsers<sup>9</sup> are increasingly common carriers of viruses, worms, Trojan horses, and other malicious code.

Perhaps the biggest risks for companies are within their own walls. Many organizations do not have Internet or Web policies in place and, as a result, employees and management alike are uneducated about the risks inherent with external communications.

Designers of MMC are aware of these lacks, and often rely on social engineering techniques<sup>10</sup> to spread their malicious products. The developer builds MMC using psychological and marketing techniques that appeal to the end user's self interest. Malicious mobile code may masquerade as screensavers, games, help files, music

---

<sup>5</sup> [http://www.cert.org/stats/cert\\_stats.html#vulnerabilities](http://www.cert.org/stats/cert_stats.html#vulnerabilities)

<sup>6</sup> <http://www.cert.org/present/cert-overview-trends/module-1.pdf>

<sup>7</sup> <http://news.com.com/2100-1001-899245.html>

<sup>8</sup> <http://www.computereconomics.com/article.cfm?id=133>

<sup>9</sup> <http://www.washingtonpost.com/wp-dyn/articles/A63025-2002Jun5.html>

<sup>10</sup> [http://www.cert.org/incident\\_notes/IN-2002-03.html](http://www.cert.org/incident_notes/IN-2002-03.html)

---

downloads, pornography, and even anti-virus applications. Once the end user visits a malicious site, the MMC script or code may become active.

Another risk facing companies, reports CERT/CC in *Overview Incident and Vulnerability Trends*, is that it is increasingly common to find that employees acting as system administrators have no formal training in the computer field. This lack of training will become ever more problematic as Internet and Web technologies and MMC development continue to cross new technical boundaries and grow in complexity.

## The Current Detection Technologies: Strengths and Weaknesses

Internet and Web security issues have forced the computer industry to develop complex tools to handle outbreaks of MMC. Today, security product vendors rely on one of three main technologies to detect malicious mobile code.

- *Signature-based detection* identifies viruses using a unique string of bits that cannot be forged. Security systems search for this binary pattern and halt file execution when it encounters a virus signature. There are two subsets within signature-based detection.
  - Signature detection for client-based applications. A desktop application uses signatures to detect problems.
  - Signature detection for server- or gateway-based applications. A server application works in tandem with firewalls or proxy servers to detect problems.
- *Sandbox technology* works by creating a restricted space within a computer that limits code execution to that space until the system determines whether behavior is malicious or not. Early sandbox models assumed that local code was acceptable, however, recent MMC activity has forced new sandbox models to assume that all code is unacceptable until proven otherwise.
- *Hashing* applies complex algorithms to a text string and produces a unique identifying number that is significantly smaller than the original file. This unique hash is placed in a hash look-up table, against which security programs check incoming hashes. Whenever a security product encounters an application whose hash ID differs from previously stored data, it flags the file as questionable and refuses to execute it.

Each of these technologies has strengths and weaknesses, which are identified in the following table.

Technology	Description	Manageability / Scalability	Effectiveness
Signatures: Client-based applications	The desktop application uses signatures of known "bad" applications to detect them.	Difficult to manage and maintain updates of signatures on all clients.	Security depends entirely on signature updates. If the database does not contain a specific signature, it is possible that viruses may slip through. Differences between update times and/or missed signature downloads may also impede accuracy.
Signatures: Server- or gateway-based applications	The server application redirects traffic through a firewall or proxy server for inspection. The system compares the signature against a database of known malicious signatures.	Increased levels of packet inspection may result in poor gateway performance.	Same as client-installed signature detection. It is possible with server- or gateway-based applications that missing downloads of new signatures results in all client applications missing malicious code.
Sandbox technology	The application runs on the gateway and/or is redirected from the firewall or proxy server to a simulated computer space that acts as a firewall. The computer allows the questionable application to run behind this wall and monitors its behavior there.	All programs are monitored. This may result in decreased scalability and gateway performance. Complexity of implementation and programming errors in sandbox codes may allow MMC to escape <sup>11</sup> .	Effectiveness depends on the accuracy of the behavioral analysis. False positives are not uncommon. Customers may need to choose between functionality and security.
Hashing malicious code	The application runs a hash on all applicable files then compares the new hash against a database of hashes for known malicious applications.	Comparisons occur on all programs, which may result in decreased performance.	Applications that use hashing generally support only compiled code and cannot detect scripts that run through interpreters such as JavaScript and VBScript.

The truly alarming fact is that most security products available today treat the symptoms: they do not treat the disease. Using the above processes, they may stop the MMC from executing, however, they do not stop it from getting into the system in the first place. Even if an MMC is dormant, it continues to be a threat as long as it remains anywhere within your system.

<sup>11</sup> <http://www.securitymanagement.com/library/000599.html>

---

## The Websense Solution

Websense, Inc. is scheduled to release a powerful new add-on for Websense Enterprise that meets the challenges posed by the rapid proliferation of malignant mobile code. The new Premium category, **Security PG: Malicious Web Sites**, offers a significantly more effective approach by stopping MMC *before* it ever gets into your network and onto your computers.

PGIII harnesses the technologies originally developed for *Project Redshift: Measuring the Web Universe*<sup>12</sup>, and uses combinations of the following processes to identify and isolate Web sites infected with MMC.

- **MMC fingerprinting.** An internally developed tool uses fingerprinting to help identify MMC. If fingerprints match those of known malicious code, Websense blocks the site and adds it to the PGIII category.
- **Data-mining.** An internally developed tool mines the current Websense Master Database, the Web, and sites found by WebCatcher for predefined script tags, function calls, statements, object models, Active X controls, DLLs, and Java Applet classes that could be used for malicious intent. The tool compares data and looks for hidden patterns to help identify code. If the data-mining process results in a match with known malicious mobile code, Websense blocks the site and adds it to the PGIII category.
- **Honey pot processes.** Websense uses a distributed network of machines, called honey pots, to capture and test code that may carry MMC. A honey pot acts as digital network bait, and through deception, attracts intruders including worms, viruses, and Trojan horses. Any new data is added to the Websense Master Database, and Websense blocks the site and adds it to the PGIII category.
- **Websense heuristic processes.** Websense, Inc. uses heuristic-based processes to identify files and applications. Heuristic processes are similar to human learning patterns: they test and analyze files and applications, then handle them based on previously identified rules of thumb and internal guidelines, and past decisions. These processes check security models of each technology and apply complex pattern-matching techniques. Websense blocks any site containing MMC and adds it to the PGIII category.

Using this powerful combination of processes, Websense continues to block malignant sources as long as they remain a threat. Once the threat is over, Websense removes the URL from the SECURITY PG category.

Data for the Security PG category is added to the Websense Master Database data and downloads are available daily. This process is fully automated, and requires no customer intervention.

---

<sup>12</sup> <http://www.websense.com/products/resources/wp/projectredshift.pdf>

## Conclusion

With the rise in security breaches caused by viruses, Trojans horses, worms, other malicious code, and the attendant expenses, companies are increasingly concerned about becoming victims of these attacks. The complexity of system design and the need for continual maintenance of security systems is changing how the industry views employee Internet and Web access.

Websense, Inc. has developed a system that searches for malicious code using a variety of detection methods. Using Websense Enterprise as a platform, the Security PG category of Malicious Web Sites will enable Websense customers to block these sites to minimize the risk of malicious mobile code attacks. Fully automated downloads are available daily to ensure continued security for Websense Enterprise Security PG customers. The Security PG category gives organizations the protection they need and the ease of maintenance they want.

For more information and to download a free, fully functional 30-day trial, visit [www.websense.com/downloads](http://www.websense.com/downloads).

## About Websense, Inc.

Websense, Inc. (NASDAQ: [WBSN](#)), the world's leading provider of employee Internet management solutions, enables organizations to optimize employee use of computing resources and mitigate new threats related to Internet use including instant messaging, peer-to-peer, and spyware. By providing usage policy enforcement at the Internet gateway, on the network and at the desktop, Websense Enterprise enhances productivity and security, optimizes the use of IT resources and mitigates legal liability for our customers. Websense serves more than 20,600 customers worldwide, representing 16.4 million seats. For more information, visit [www.websense.com](http://www.websense.com).

© 2004, Websense, Inc. All rights reserved. Websense and Websense Enterprise a registered trademarks of Websense, Inc. in the United States and in certain international markets. Websense has numerous other trademarks nationally and internationally. All other trademarks are the property of their respective owners.

## Glossary

Term	Definition
<b>CERT/CC</b>	The CERT® Coordination Center (CERT/CC) is a center of Internet security expertise, at the <a href="#">Software Engineering Institute</a> , a federally funded research and development center operated by <a href="#">Carnegie Mellon University</a> .
<b>Data mining</b>	The actions performed by database applications that look for hidden patterns in data. Such data mining software doesn't just change the presentation, but actually discovers previously unknown relationships among the data.
<b>MMC fingerprinting</b>	A digital code that provides source details for files and email messages indicating where they are from.
<b>Hash</b>	A number generated from a string of text. The hash is substantially smaller than the text itself, and is generated by a formula in such a way that it is extremely unlikely that some other text will produce the same hash value. Hashing is also a common method of accessing data records, where hash values are in a hash table for look-up.
<b>Heuristic processes</b>	Computer analysis techniques patterned after human learning behaviors. Heuristic processes are able to "learn," based on previously defined guidelines and decisions. As time passes, these heuristic processes become "smarter," as they have more information on which to base their decisions.
<b>Honey pot</b>	A computer system on the Internet that is expressly set up to attract and "trap" code that attempts to penetrate computer systems without permission.
<b>Malicious Mobile Code (MCC)</b>	Any software program designed to move from computer to computer and network to network, in order to intentionally modify computer systems without the consent of the owner/operator.
<b>Sandbox</b>	A protected, limited environment within a computer hard drive where applications are allowed to "play" without risking damage to the rest of the system.
<b>Signature</b>	A digital code that can be attached to an electronically transmitted message that uniquely identifies the sender. Like a written signature, the purpose of a digital signature is to guarantee that the individual sending the message really is who he or she claims to be. Digital signatures are especially important for electronic commerce and are a key component of most authentication schemes.
<b>Social engineering</b>	Used to describe how designers of malicious mobile code snare and trick people to download and open their products. Often the malicious code is disguised as music files, security applications, or pornography, which may be sent using email, instant messaging, and Chat rooms.
<b>Trojan horse</b>	A virus in which malicious or harmful code is contained inside apparently harmless programming or data. Trojan horses have been found on Web sites, bulletin boards, and in email attachments.
<b>Virus</b>	A piece of programming code that runs by itself and is able to replicate itself again and again. Some viruses are designed to transmit themselves across networks and bypass security systems.
<b>Worm</b>	A self-replicating virus that does not alter files but resides in active memory and duplicates itself.