Sanctum Technical Advisory

Server-Based Worms

Computer Worms are destructive, self-contained programs that replicate from machine to machine across network connections often clogging networks and information systems as they spread. Their ability to spread quickly across the Internet has made worms the weapon of choice for hackers and vandals. Unlike viruses, worms do not depend on other carrier applications or software hosts to multiply. A ***server-based worm*** is a new type of worm that specifically attacks the Web server and all its Web applications. These applications can include code such as the Operating System, databases, or code sitting on your application server from either 3rd party or in-house developed software.

A server-based worm self-replicates malicious code that can contaminate a large number of servers within minutes and gets to the most vulnerable servers within a few hours. Upon gaining a new "base for operation", a server-based worm will cause a large amount of damage to the infected web site by corrupting/deleting files, exposing sensitive information, and ultimately creating a back door that any malicious hacker can use later to gain complete control of the system. Once a site is penetrated it is used for performing further attacks on other web sites.

In 2001, several major server-based worms were launched and did considerable destruction to web servers and the applications that reside on and behind them.

Code Red

Code Red exploits a known vulnerability in Microsoft IIS servers and attempts to connect to TCP port 80 on a randomly chosen host assuming that a Web server will be found. Upon a successful connection to port 80, the worm uses an HTTP GET attack buffer overflow with special characters to exploit the Index Service DLL vulnerability (published June 18th 2001). The specific DLL exploited is idq.dll which allows for ISAPI extensions to access administrative scripts (.ida file types) and Internet Data Queries (.idq file types). The buffer overflow is targeted at default.ida, which allows the exploiting process to access system space and gain control of the system. This particular worm does have some destructive payload, meaning it can destroy or delete files, in additional to site defacement, denial of service (DoS) and reducing the system's stability by leaving open doors in the system. Infected servers may experience performance degradation as a result of the scanning activity of this worm. This degradation can become quite severe since a worm can infect a machine multiple times simultaneously.

Code Red infected over 1 million servers around the world and at its peak, infected over 2000 servers per minute (*CAIDA*). The total damage caused by the worm exceeded $2.6B for cleanup and lost productivity (*CNET*).

www.SanctumInc.com
ph: 877-888-3970 (US Only)
408-352-2000 (international)
fax: 408-352-2001

Sanctum Inc.
2901 Tasman Drive Ste 205
Santa Clara, CA 95054

Nimda

The NIMDA (ADMIN spelled backwards) worm can be activated in many different ways. When the worm attacks IIS Web servers, it checks to see if the computer was previously compromised by the Code Red II worm, which creates a "back door" that any malicious user can use later to gain control of the system. If the Nimda worm finds such a computer, it simply uses the back door created by Code Red II to infect the system. Also, the worm attempts to exploit the known IIS directory traversal vulnerability. Nimda also propagates by infiltrating unsecured Web sites and attaching itself to an unsuspecting computer user's Web browser. Nimda may show up as a sound or .wav file. When a user opens the underlying file, the program opens the computer's hard drive, allowing the computer to be accessed by third parties via the Internet. Nimda can also be spread via email attachments. The damage caused by the worm ranges from corrupted/deleted files, illegal access to backend systems via the Internet to forcing companies to shutdown all networks and web servers to stop the spread.

Nimda infected over 160,000 servers at its peak and doubled the average Internet response time (CAIDA). What is more alarming is that within 24 hours of NIMDA hitting, 50% of the infected hosts went offline as the only mean of stopping the infection (CAIDA).

Solutions

The move from “human” attacks on web sites to the usage of computerized agents has caused a huge increase in the number of damaged sites. Typically when a worm starts spreading, it will contaminate a large number of servers within minutes and will get to the most vulnerable servers within a few hours. Fortunately, automated security solutions exist that offer proactive fixes for the ongoing defense against destructive server-based worms.

Traditional solution – manual patches and patch latency

Traditionally, manual auditing has been used to identify application vulnerabilities, and then installing patches, if available, that block the vulnerabilities used by the worm. However, patch latency – or the delay in time from the point the vulnerability is found to when the patch is installed– has been a real life problem. When a vulnerability is found – the clock starts ticking:

- A patch is created and tested by the vendor or developer
- The patch is downloaded by the user
- The patch is tested internally in the staging environment
- The patch is installed into the production environment, usually bringing the site down during the installation – the clock stops ticking

The typical patch latency period is a few weeks to many months. During this period the site is completely vulnerable to attack. The patch mechanism also demands a high level of knowledge and awareness from the security administrator together with constant follow-up looking for new vulnerabilities. This is a very reactive solution and enables the user to only address known problems. In today’s time pressured environment, manual patching is no longer a practical solution.

Detecting the holes - Application Vulnerability Assessment

The first step in improving the manual audit and patch process is to automate the method for finding the holes. This can be done with automatic application vulnerability assessment tools. These tools enable the security officer to find the problems in his/her applications and know which patches to download or coding techniques to use to close the hole. This reduces the knowledge needed by the user and significantly reduces the time needed to audit the site, but requires the right assessment pattern to identify the specific vulnerabilities.

Sanctum's AppScan is the leading vulnerability assessment tool available today that uses its policy recognition engine to identify holes, previously unknown, within the application. This identification is crucial in finding the problem before the worm strikes. This, however, is not enough, as it still requires a patch to be installed before the worm attacks. The window of opportunity for the worm to act is extremely small and therefore just knowing the hole exists is not enough to solve the problem.

Detecting the attack – Intrusion Detection

Another approach to blocking the worm is to detect it when it tries to get in the hole by comparing it to a list of known attack patterns. This type of online detection is performed by host-based intrusion detection systems (IDS). These products are equipped with a list of potential attack signatures used to identify the attack and block it *if* the exact attack is identified using the signature. To get coverage for every server with host-based IDS, the administrator needs to load the IDS software on each computer separately. Every time a new worm is discovered, the administrator will need to add new attack signatures to each of the servers in order to detect the worm. A host IDS system will eliminate the need to shut down the server to install the patches, however, it still leaves the problem of patch/attack signature latency: the IDS system still needs to be updated fast enough to compete with the spread of the worm. Even worse, attacks using unknown holes will continue to go undetected and will successfully penetrate the system leaving the site vulnerable to the next worm attack.

Preventing the worm attack – Application Firewalls

Application firewalls block worms by providing comprehensive online *intrusion prevention* stopping worms from ever reaching the server. By dynamically recognizing the correct behavior of the site, the application firewall only allows the site to be accessed in the approved manner. The application firewall blocks all illegal access (including worms) to the site without the need for a signature of the specific attack.

Using application firewalls, you will stop worms without any patches or updates eliminating the patch latency issue. Sanctum's application firewall, AppShield, protects against worms by intelligently and autonomously preventing unauthorized application behavior. The result is AppShield stops any known and unknown worm or human attack. Better yet, since AppShield stops the worm, it also ensures that the worm does not spread to other unsuspecting servers. Once an application firewall is installed, it will protect the site from any future unknown worms. Unlike host-based IDS, application firewalls do not require code to be installed on each web server or attack signatures to be updated. The application firewall is managed centrally and is easy to maintain.

Summary and Recommendation

Server-based worms are malicious programs with the sole purpose of manipulating data and shutting down sites. They can happen any time of the day or night with no warning. Good business practices direct you to install patches as quickly as possible – but given the reality of time pressures and the lack of skilled resources protecting most of today's corporate websites, the only full protection against server-based worms can be achieved by installing automated application firewalls like Sanctum's AppShield. This is the only system that does not demand any patches to perform full identification and prevention of all currently known and unknown attacks. By removing the need for the patches or signatures, an application firewall is able to protect against worm attacks without any latency. Application firewalls stop worms from attacking your site, minimizes the administration, reduces downtime of your servers, and protects against current and future attacks automatically.

Additional information

http://www.cert.org/advisories/ - Advisories regarding the specific worms.
http://www.securityfocus.com/ - In-depth analysis of the specific worms.
http://www.sanctuminc.com/news/alerts/2001/20010801_codered.html - Sanctum's vulnerability alert on the Code Red worm.
http://www.sanctuminc.com/news/alerts/2001/2001901_nimda.html -- Sanctum's vulnerability alert on the Nimda worm.

www.SanctumInc.com
ph: 877-888-3970 (US Only)
408-855-9500 (international)
fax: 408-855-9521

Sanctum Inc.
2901 Tasman Drive Ste 205
Santa Clara, CA 95054