

Web Applications Vulnerability

23/10/2006 11:20 by the6dmin

1.1 Identify Web Applications

The web is currently rich in all sorts of applications implemented by individuals and organizations. Applications range from content management systems (CMS) to discussion forums and fully functional portals. Regardless of the platform used to develop these systems it being PHP, Dot NET or CGI; several security breaches have been identified through coding errors and limitations in the platform used. In PHP, functions used to implement reading local web site files can allow a malicious application to access major system files on any server Windows or UNIX. Another problem with applications is the use of clearly defined SQL statements within the code to read, update or delete from the database. This makes the database open to attacks through knowledge of its structure. With the advancement of web applications, programmers use built in components from other sites. Using shared objects resulted in Cross-Site Scripting (XSS) which enabled attackers to place programs that control the user's browser behaviors through corrupted Java Script code as to execute commands without users' consent.

1.2 Detect Web Application Vulnerabilities

For any running server to detect these vulnerabilities, administrators need to do regular scans for potential security openings and close them as much as possible. In PHP, there is a setup to allow PHP to run in safe mode, in which files are not allowed to run unless within the application folder, also it limits the files that can be opened and the classes that can be executed. This is a two way blade; the web application functionality can also suffer as a result of this security restriction. Companies should also invest in security professionals to identify problems in the server or associated applications.

1.3 Protection against Web Application Vulnerability

It is recommended to run the latest version of the development platform. Software vendors release patches to cover potential hazards identified by security professional, administrators should upgrade and install these patches. Applications should run with the least possible settings. For example, in PHP features like register global and allow file opening can be turned off if not required by the application. To escape XSS; encode output using built in functions and validate information entered by users before placing it within files to avoid remote attacks. Many organizations offer testing tools and guides on securing against XSS, proper server configurations and explanation to information in servers log files.

References " SANS Top-20 Internet Security Attack Targets (2006 Annual Update) ", SANS Institute Resources, 2006 <http://www.sans.org/top20/?ref=1697#c1> > (15 Nov 2006) PHP Manual , Features page , 2006 , <http://au.php.net/features.safe-mode> > (15 Nov 2006) Hardened PHP Project, Why Page, 2006, <http://www.hardened-php.net/suhosin/why.html> > (15 Nov 2006) " Application configuration management testing ," OWASP Articles , 31 Jul 2006 http://www.owasp.org/index.php/Application_configuration_management_testing > (15 Nov 2006)