



Version 3.10

OWASP’s Top Vulnerabilities in Web Applications

The Open Web Application Security Project (OWASP) is an [Open Source](http://www.owasp.org/) community project and is dedicated to helping organizations understand and improve the security of their web applications and web services. (<http://www.owasp.org/>)

	Vulnerability	Description	Y/N	If “Y,” How BRICKServer™ Protects Against This Threat
A1	Unvalidated Parameters	Information from web requests is not validated before being used by a web application. Attackers can use these flaws to attack backside components through a web application.	No	BRICKServer cannot prevent web designers from building flawed code (see A4). Web applications should <i>always</i> validate input, especially parameters for external database queries.
A2	Broken Access Control	Restrictions on what authenticated users are allowed to do are not properly enforced. Attackers can exploit these flaws to access other users’ accounts, view sensitive files, or use unauthorized functions.	Y	BRICKServer does not rely on user-based access controls (DACs), but instead employs Process-Based Security (PBS) controls that prevent all unauthorized activities. PBS prevents these attacks.
A3	Broken Account and Session Management	Account credentials and session tokens are not properly protected. Attackers that can compromise passwords, keys, session cookies, or other tokens can defeat authentication restrictions and assume other users’ identities.	Y	BRICKServer’s PBS model prevents any form of privilege-based attack, no matter what the attack injection method. Note that authentication spoofing at the application level is NOT prevented by BRICKServer, nor is spoofing by social engineering prevented.
A4	Cross-Site Scripting (XSS) Flaws	The web application can be used as a mechanism to transport an attack to an end user’s browser. A successful attack can disclose the end user’s session token, attack the local machine, or spoof content to fool the user.	No	BRICKServer cannot prevent web designers from building flawed code. If a web application does not properly check inputs or filter meta-characters, users could be exposed to XSS attacks. Prevention: users should set browsers to “high” security mode.
A5	Buffer Overflows	Web application components in some languages that	Y	Buffer overflows are the most common form of privilege-

		do not properly validate input can be crashed and, in some cases, used to take control of a process. These components can include CGI, libraries, drivers, and web application server components.		based attacks. BRICKServer's PBS stops these attacks completely by disallowing execution of any code that does not have a permissions table. The table is hard-coded in BRICKServer and cannot be changed in the field.
A6	Command Injection Flaws	Web applications pass parameters when they access external systems or the local operating system. If an attacker can embed malicious commands in these parameters, the external system may execute those commands on behalf of the web application.	No	BRICKServer will not prevent a malicious parameter from being passed to an external system, and the malicious code could run (see A1 & A4). However, any malicious code attempting to run in the BRICKServer itself would be stopped cold by PBS.
A7	Error Handling Problems	Error conditions that occur during normal operation are not handled properly. If an attacker can cause errors to occur that the web application does not handle, they can gain detailed system information, deny service, cause security mechanisms to fail, or crash the server.	Y	PBS protects here. Error conditions are treated differently in BRICKServer, i.e., control is not passed to the o/s to run at the existing privilege level. Instead, PBS checks its Process Control List (an ACL) to see if a given application has rights to specific system resources at all times. If the privilege doesn't exist, it can't execute.
A8	Insecure Use of Cryptography	Web applications frequently use cryptographic functions to protect information and credentials. These functions and the code to integrate them have proven difficult to code properly, frequently resulting in weak protection.	Y	BRICKServer keeps all server data in the clear (except hashed passwords), so there is no need for cryptographic techniques like key management. The underlying security policy of PBS protects all data in the server by preventing unauthorized access.
A9	Remote Administration Flaws	Many web applications allow administrators to access the site using a web interface. If these administrative functions are not very carefully protected, an attacker can gain full access to all aspects of a site.	Y	Remote administration is effected via a secure tunnel using a proprietary client software application and corresponding server side software. Tunnel encryption uses 128 bit symmetric encryption as standard. No web interface is used for administration; a different port is used for remote admin.
A10	Web and Application Server Misconfiguration	Having a strong server configuration standard is critical to a secure web application. These servers have many configuration options that affect security and are not secure out of the box.	Y	Because BRICKServer's underlying administration and security policies are built-in, mis-configuration of the server is impossible. This not only stops hackers, but administrative accidents as well.

As of Mar 9, 2003