Top Vulnerabilities in Web Applications From Open Web Application Security Project's (OWASP) top ten list		Detected by AppScan 4.0?	Protected by AppShield 4.0?
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.	**Y**	**Y**
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**	**Y**
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**	**Y**
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.	**Y**	**Y**
Buffer Overflows	Web application components in some languages that 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.	**Y**	**Y**
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.	**Y**	**Y**
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**	**Y**
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♦**	**Y**
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**	**Y**
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**	**Y**

♦ AppScan checks for defects in some but not all cryptographic functions. Contact Sanctum with questions about specific encryption methods.